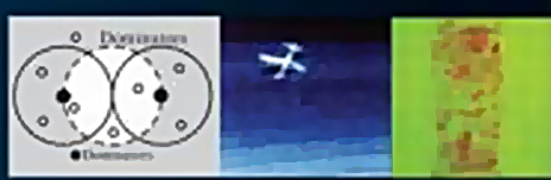
CRC Press
Taylor & Francis Group
A CHAPMAN & HALL BOOK

Handbook of Sensor Networking

Advanced Technologies and Applications

EDITED BY
JOHN R. VACCA



Handbook of Sensor Networking

Handbook of Sensor Networking

Advanced Technologies and Applications

EDITED BY

JOHN R. VACCA

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

MATLAB® is a trademark of The MathWorks, Inc. and is used with permission. The MathWorks does not warrant the accuracy of the text or exercises in this book. This book's use or discussion of MATLAB® software or related products does not constitute endorsement or sponsorship by The MathWorks of a particular pedagogical approach or particular use of the MATLAB® software.

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2015 by Taylor & Francis Group, LLC
CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works
Version Date: 20141124

International Standard Book Number-13: 978-1-4665-6972-0 (eBook - PDF)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (http://www.copyright.com/) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

This book is dedicated to my wife, Bee.

Contents

Foreword xi
Preface xiii
Acknowledgments xvii
Editor xix
Contributors xxi

SECTION I Sensor Networking Software and Architectures

1 Sensor Networking Internet of Things **1**-1
Lauren Collins

2 Sensor Network Platform and Operating Systems **2**-1
Xinheng (Henry) Wang and Shancang Li

3 Mobile Crowd Sensing **3**-1
Manoop Talasila, Reza Curtmola, and Cristian Borcea

SECTION II Wireless Sensor Information Networks

4 In-Network Processing in Wireless Sensor Networks **4**-1
Qiao Xiang and Hongwei Zhang

5 Wireless Sensor Hierarchical Networks **5**-1
Shancang Li and Xinheng (Henry) Wang

6 Efficient Distributed Bayesian Estimation in Wireless Sensor Networks **6**-1
Andrew P. Brown, Ronald A. Iltis, and Hua Lee

SECTION III Sensor Networking Data Manipulation and Advanced Hardware

7 Constructing Load-Balanced Data Aggregation Trees in Probabilistic Wireless Sensor Networks 7-1
Jing (Selena) He

8 Biometric Sensors and How They Work 8-1
Sinjini Mitra

9 Radio Frequency Identification Device Readers and Tags 9-1
C.J. Wiemer

SECTION IV Sensor Networking Signal Processing

10 Multimedia/Audio Sensor Networking Signal Processing 10-1
Juan R. Aguilar

11 Data Mining in Sensor Networks 11-1
Sinjini Mitra and Pramod Pandya

SECTION V Obstacles in Sensor Networks

12 Wireless Sensor Network Security: PHY-Layer Security 12-1
Andrea Bartoli, Juan Hernandez-Serrano, Miquel Soriano, Mischa Dohler, Apostolos Kountouris, and Dominique Barthel

13 Impact of Correlated Failures on Wireless Sensor Network Reliability 13-1
Jorge E. Pezoa and Silvia Elena Restrepo

14 Acoustic Sensors and Algorithms for Urban Security 14-1
Juan R. Aguilar

SECTION VI Sensor Networking Applications and Uses

15 Smart Buildings and Smart Cities 15-1
Rim Boujrad and Thomas M. Chen

16 Imaging Technology 16-1
Hua Lee

17 Reconfigurable MIMO FMCW Imaging Technique and Applications in Acoustical and Microwave Imaging 17-1
Hua Lee and Michael Lee

18 Recent Advances in EO/IR Imaging Detector and Sensor Applications 18-1
Ashok K. Sood, Yash R. Puri, Nibir K. Dhar, and Dennis L. Polla

19 Vehicular Sensor Networks 19-1
Juan Pan and Cristian Borcea

20 Acoustic Sensing System for Underwater Geolocation and Navigation **20**-1
Hua Lee

21 Load-Balanced Virtual Backbones in Wireless Sensor Networks **21**-1
Jing (Selena) He

Appendix A: List of Sensor Networking Standards, Organizations, and Protocols ..... **Appendix A**-1
John R. Vacca

Appendix B: Glossary ..... **Appendix B**-1
John R. Vacca

Index **Index**-1

Foreword

In the mid-1990s, analysts and the media focused on how many people were using the Internet in countries around the world and how rapidly that number was increasing. Great stuff and a fascinating time. Then, there were those of us that looked way into the future (in Internet years) and viewed the Internet, or global area networking, as a place that would not only connect people but a variety of machines and devices that could replace people by replicating and intelligently performing many functions that, at one time, were performed by humans. We are there!

John R. Vacca has been a forward thinker in how omnipotent networks will work and envisioned many types of devices and functions that can be networked. His work in these areas is unsurpassed by other writers and visionaries.

In this book, John has pulled together the knowledge that is needed for managers and developers to keep moving forward, and to do so faster, with sensor networking machines and devices that can perform both the mundane and complex tasks that are required for process management and production that is necessary to propel enterprises and thus drive the global economy.

Sensor networks are rapidly growing in size, scope, and functionality. This book provides a bottom-up analysis of sensor networking technology. It will enable managers and designers to create and deploy this important technological breakthrough.

Michael Erbschloe*
Information Security Consultant

* Michael Erbschloe teaches information security courses at Webster University in St. Louis, Missouri.

Preface

Scope of Coverage

This comprehensive handbook serves as a professional reference as well as a practitioner's guide to today's most complete and concise view of advanced sensor networking technologies. It offers in-depth coverage of sensor networking theory, technology, and practice as they relate to established technologies as well as recent advancements. It explores practical solutions to a wide range of sensor networking issues. The chapters are authored by leading experts in the field and address the immediate and long-term challenges in the authors' respective areas of expertise.

The primary audience for this handbook consists of researchers and practitioners in the industry and academia who share experiences and ideas on innovative developments in any aspect of sensor networks, including hardware of sensor networks, wireless communication protocols, sensor networks software and architectures, wireless information networks, data manipulation, signal processing, localization, and object tracking through sensor networks, obstacles, applications, and uses. This book will also be useful for sensor networking technologists, engineers, scientists, federal and state governments, and security and law enforcement agencies working with other individuals and related professionals who are interested in sensor networks. This comprehensive reference and practitioner's guide will also be of value to students in upper-division undergraduate and graduate-level courses in sensor networking.

Organization of This Book

The book is organized into six sections composed of 21 contributed chapters by leading experts in their fields, as well as two appendices, including an extensive glossary of sensor networking terms and acronyms.

Section I: Sensor Networking Software and Architectures

Section I discusses the sensor networking Internet of Things, interoperability, agent-based simulation decision support, platforms and operating systems, programming and middleware, connectivity and communication, and scheduling, tasking, and control.

Chapter 1, Sensor Networking Internet of Things, sets the stage for the rest of the book by presenting insight into a global network infrastructure, linking physical and virtual objects through the exploitation of data capture and communication capabilities. This infrastructure includes existing and evolving Internet and network developments. The chapter offers specific object identification, sensor networking, and connection capability as the basis for the development of independent cooperative services and applications. These will be characterized by a high degree of autonomous data capture, event transfer, network connectivity, and interoperability.

Chapter 2, Sensor Network Platform and Operating Systems, describes how operating systems for wireless sensor network nodes are typically less complex than general-purpose operating systems. They more

strongly resemble embedded systems for two reasons: First, wireless sensor networks are typically deployed with a particular application in mind rather than as a general platform. Second, the need for low costs and low power leads most wireless sensor nodes to have low-power microcontrollers, ensuring that mechanisms such as virtual memory are either unnecessary or too expensive to implement.

Chapter 3, Mobile Crowd Sensing, presents the concept of mobile crowd sensing and its applications in everyday life. It also presents an overview of McSense, the mobile crowd sensing platform, and its prototype implementation. The chapter also describes the ILR scheme to achieve data reliability in McSense and analyzes ILR's security. The experimental evaluation and simulation results for ILR are also presented. The chapter also discusses a number of lessons learned from McSense's field studies, as well as potential improvements for ILR.

Section II: Wireless Sensor Information Networks

Section II discusses in-network processing protocols in wireless sensor networks, wireless sensor hierarchical networks, and efficient distributed Bayesian estimation in wireless sensor networks.

Chapter 4, In-Network Processing in Wireless Sensor Networks, introduces different INP methods in WSN, including data aggregation, packet packing, and network coding.

Chapter 5, Wireless Sensor Hierarchical Networks, examines how the hierarchical architecture is very popular in designing WSNs, which provides scalability, ad hoc, is fault tolerant, and enables easy integration of applications into WSNs.

Chapter 6, Efficient Distributed Bayesian Estimation in Wireless Sensor Networks, provides a detailed discussion of distributed estimation algorithms, with an example application of sensor node localization in mobile ad hoc networks.

Section III: Sensor Networking Data Manipulation and Advanced Hardware

Section III discusses load-balanced data aggregation trees in wireless sensor networks, biometrics sensors and how they work, and radio frequency identification device (RFID) readers and tags.

Chapter 7, Constructing Load-Balanced Data Aggregation Trees in Probabilistic Wireless Sensor Networks, explores the DAT construction problem under the PNM, by considering the balancing of the traffic load among all the nodes in a DAT.

Chapter 8, Biometric Sensors and How They Work, provides an in-depth presentation of what different types of biometric sensors exist in the world today, how they work, and what some of the principal underlying issues are.

Chapter 9, Radio Frequency Identification Device Readers and Tags, explains how the RFID technology involves the wireless communication between a tag and a reader.

Section IV: Sensor Networking Signal Processing

Section IV discusses multimedia/audio sensor networking signal processing and data mining.

Chapter 10, Multimedia/Audio Sensor Networking Signal Processing, addresses the family of array signal processing that are designed to control in a desired way the directivity pattern of an array of sensors using electronic software means.

Chapter 11, Data Mining in Sensor Networks, is a detailed overview of all the data mining tools applicable to time series sensor streams along with associated issues and challenges.

Section V: Obstacles in Sensor Networks

Section V covers wireless sensor network security, fault tolerance and diagnosis, and acoustic sensors for urban security.

Chapter 12, Wireless Sensor Network Security: PHY-Layer Security, details a recently introduced security scheme at the PHY layer that is shown to yield enormous benefits with regard to attacks and unintended exhaustion messages by making use of an authentication preamble (AP).

Chapter 13, Impact of Correlated Failures on Wireless Sensor Network Reliability**,** covers the probabilistic model for correlated node failures in WSNs.

Chapter 14, Acoustic Sensors and Algorithms for Urban Security, discusses the sensor network architecture consisting of sensor nodes that are distributed inside the area under surveillance.

Section VI: Sensor Networking Applications and Uses

Section VI discusses smart buildings and smart cities, imaging technology, reconfigurable MIMO FMCW imaging technique and applications in acoustical and microwave imaging, recent advances in EO/IR imaging detectors and sensors, vehicular sensor networks, and load-balanced virtual backbone for sensor networks.

Chapter 15, Smart Buildings and Smart Cities, focuses on technologies. But smart buildings are more than technology. Their ultimate purpose is to improve building performance and better serve the needs of their occupants.

Chapter 16, Imaging Technology, provides a comprehensive overview of five imaging systems as examples of the development and optimization of the imaging technology.

Chapter 17, Reconfigurable MIMO FMCW Imaging Technique and Applications in Acoustical and Microwave Imaging, introduces the concept and approach of the alternative operating modality known as step-frequency FMCW imaging.

Chapter 18, Recent Advances in EO/IR Imaging Detector and Sensor Applications, covers recent advances in UV, visible, and infrared detector technology (i.e., materials and devices for a variety of imaging, detection, and sensor applications).

Chapter 19, Vehicular Sensor Networks**,** focuses on leveraging vehicular sensor networks for congestion avoidance.

Chapter 20, Acoustic Sensing System for Underwater Geolocation and Navigation, presents a simple high-performance approach to underwater geolocation and navigation.

Chapter 21, Load-Balanced Virtual Backbones in Wireless Sensor Networks, investigates the LBVB problem under both deterministic and probabilistic WSNs.

John R. Vacca
Managing and Consulting Editor
Techwrite
Pomeroy, Ohio

MATLAB® is a registered trademark of The MathWorks, Inc. For product information, please contact:

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098 USA
Tel: 508 647 7000
Fax: 508-647-7001
E-mail: info@mathworks.com
Web: www.mathworks.com

Acknowledgments

There are many people whose efforts on this book have contributed to its successful completion. I owe each a debt of gratitude and want to take this opportunity to offer my sincere thanks.

A very special thanks to my executive editor, Rick Adams, without whose continued interest and support this book would not have been possible, and to my editorial assistant Sherry Thomas, who provided staunch support and encouragement when it was most needed. Thanks to my project editor, Cynthia Klivecka; project coordinator, Joselyn Banks; and project manager, Catherine Kuruvilla Jacob, whose fine editorial work has been invaluable. Thanks also to my marketing manager, Christopher Manion, whose efforts on this book have been greatly appreciated. Finally, thanks to all of the other people at CRC Press (Taylor & Francis Group), whose many talents and skills are essential to a finished book.

Thanks to my wife, Bee Vacca, for love, help, and understanding of my long work hours. Also, a very special thanks to Michael Erbschloe, for writing the foreword. Finally, I thank all the following authors who contributed chapters that were necessary for the completion of this book: Lauren Collins, Xinheng (Henry) Wang, Shancang Li, Manoop Talasila, Reza Curtmola, Cristian Borcea, Oiao Xiang, Hongwei Zhang, Andrew Brown, Ronald A. Iltis, Hua Lee, Jing (Selena) He, Sinjini Mitra, CJ Wiemer, Juan Aguilar, Pramod Pandya, Andrea Bartoli, Juan Hernandez-Serrano, Miquel Soriano, Mischa Dohler, Apostolos Kountouris, Dominique Barthel, Jorge E. Pezoa, Silvia Elena Restrepo, Rim Boujrad, Tom Chen, Ashok K. Sood, Yesh R. Puri, Nibir K. Dhar, Dennis L. Polla, and Juan Pan.

Editor

John R. Vacca is an information technology consultant, professional writer, editor, reviewer, and internationally known, best-selling author based in Pomeroy, Ohio. Since 1982, John has authored/edited 76 books. Some of his most recent books include

- *Network and System Security*, Second Edition (Syngress [an imprint of Elsevier Inc.], September 23, 2013)
- *Cyber Security and IT Infrastructure Protection* (Syngress [an imprint of Elsevier Inc.], September 23, 2013)
- *Managing Information Security*, Second Edition (Syngress [an imprint of Elsevier Inc.], September 23, 2013)
- *Computer and Information Security Handbook*, Second Edition (Morgan Kaufmann [an imprint of Elsevier Inc.], May 31, 2013)
- *Identity Theft (Cybersafety)* (Chelsea House Pub, April 1, 2012)
- *System Forensics, Investigation, and Response* (Jones & Bartlett Learning, September 24, 2010)
- *Managing Information Security* (Syngress [an imprint of Elsevier Inc.], March 29, 2010)
- *Network and Systems Security* (Syngress [an imprint of Elsevier Inc.], March 29, 2010)
- *Computer and Information Security Handbook*, First Edition (Morgan Kaufmann [an imprint of Elsevier Inc.], June 2, 2009)
- *Biometric Technologies and Verification Systems* (Elsevier Science & Technology Books, March 16, 2007)
- *Practical Internet Security* (Hardcover) (Springer, October 18, 2006)
- *Optical Networking Best Practices Handbook* (Hardcover) (Wiley-Interscience, November 28, 2006)
- *Guide to Wireless Network Security* (Springer, August 19, 2006)
- *Computer Forensics: Computer Crime Scene Investigation* (with CD-ROM), Second Edition (Charles River Media, May 26, 2005)

He has also published more than 600 articles in the areas of advanced storage, computer security, and aerospace technology (copies of articles and books are available upon request).

John was also a configuration management specialist, computer specialist, and the computer security official (CSO) for NASA's space station program (Freedom) and the International Space Station Program, from 1988 until his retirement from NASA in 1995.

In addition, John is also an independent online book reviewer. Finally, John was one of the security consultants for the MGM movie titled *AntiTrust*, which was released on January 12, 2001. A detailed copy of the author bio can be viewed at http://www.johnvacca.com. John can be reached at john2164@windstream.net.

Contributors

Juan R. Aguilar
Laboratorio de Electro Acústica & Procesamiento de Señal E.I.R.L.
Santiago, Chile

Dominique Barthel
Orange Labs
Meylan, France

Andrea Bartoli
Telematic department
Universitat Politecnica de Catalunya
Barcelona, Spain

Cristian Borcea
Department of Computer Science
New Jersey Institute of Technology
Newark, New Jersey

Rim Boujrad
College of Engineering
Swansea University
Swansea, United Kingdom

Andrew P. Brown
Toyon Research Corporation
Goleta, California

Thomas M. Chen
School of Engineering and Math Sciences
City University London
London, United Kingdom

Lauren Collins
Winning Edge Communications
Peotone, Illinois

Reza Curtmola
Department of Computer Science
New Jersey Institute of Technology
Newark, New Jersey

Nibir K. Dhar
Defense Advanced Research Project Agency
Arlington, Virginia

Mischa Dohler
Department of Informatics
King's College London
London, United Kingdom

and

Worldsensing
Barcelona, Spain

Jing (Selena) He
Department of Computer Science
Kennesaw State University
Kennesaw, Georgia

Juan Hernandez-Serrano
Telematic department
Universitat Politecnica de Catalunya
Barcelona, Spain

Ronald A. Iltis
Department of Electrical and Computer Engineering
University of California
Santa Barbara, California

Apostolos Kountouris
Orange Labs
Meylan, France

Hua Lee
Department of Electrical and Computer Engineering
University of California
Santa Barbara, California

Michael Lee
Department of Electrical and Computer Engineering
University of California
Santa Barbara, California

Shancang Li
School of Computing
University of the West of Scotland
Scotland, United Kingdom

and

Faculty of Engineering
University of Bristol
Bristol, United Kingdom

Sinjini Mitra
Information Systems and Decision Sciences Department
Mihaylo College of Business and Economics
California State University
Fullerton, California

Juan Pan
Department of Computer Science
New Jersey Institute of Technology
Newark, New Jersey

Pramod Pandya
Information Systems and Decision Sciences Department
Mihaylo College of Business and Economics
California State University
Fullerton, California

Jorge E. Pezoa
Department of Electrical Engineering
and
Center for Optics and Photonics
Universidad de Concepción
Concepción, Chile

Dennis L. Polla
College of Science and Engineering
University of Minnesota
Minneapolis, Minnesota

Yash R. Puri
Magnolia Optical Technologies Inc.
Woburn, Massachusetts

Silvia Elena Restrepo
Department of Electrical Engineering
and
Center for Optics and Photonics
Universidad de Concepción
Concepción, Chile

Ashok K. Sood
Magnolia Optical Technologies Inc.
Woburn, Massachusetts

Miquel Soriano
Telematic Department
Universitat Politecnica de Catalunya
Barcelona, Spain

Manoop Talasila
Department of Computer Science
New Jersey Institute of Technology
Newark, New Jersey

John R. Vacca
Techwrite
Pomeroy, Ohio

Xinheng (Henry) Wang
School of Computing
University of the West of Scotland
Scotland, United Kingdom

C.J. Wiemer
kCura Corporation
Chicago, Illinois

Qiao Xiang
Department of Computer Science
McGill University
Montréal, Québec, Canada

Hongwei Zhang
Department of Computer Science
Wayne State University
Detroit, Michigan

I

Sensor Networking Software and Architectures

1 **Sensor Networking Internet of Things** *Lauren Collins* **1**-1
Introduction • Fields Using Sensor Networking • Implanted Medical Devices and Physiological Parameters • Sensor Networking in Manufacturing and Production • Location-Based Services • Vision Sensing • Summary • References

2 **Sensor Network Platform and Operating Systems** *Xinheng (Henry) Wang and Shancang Li* **2**-1
Introduction • Architecture of a WSN Node • Reconfigurable WSN Platforms • Sensing • Communications • Power Unit • Operating Systems • Summary • References

3 **Mobile Crowd Sensing** *Manoop Talasila, Reza Curtmola, and Cristian Borcea* **3**-1
Introduction • Mobile Crowd Sensing Applications • Mobile Crowd Sensing Applications and Platforms • Data Reliability Issues in Sensed Data • McSense: A Mobile Crowd Sensing Platform • ILR in Crowd-Sensed Data • Experimental Evaluation: Field Study • Simulations • Field Study Insights and Improving the ILR Scheme • Summary • Acknowledgment • References

1

Sensor Networking Internet of Things

1.1 Introduction **1**-1
1.2 Fields Using Sensor Networking **1**-2
Environmental Sensors • HVAC Systems • Data Center Sensors • Spray Foam Insulation
1.3 Implanted Medical Devices and Physiological Parameters **1**-8
1.4 Sensor Networking in Manufacturing and Production **1**-9
1.5 Location-Based Services **1**-9
1.6 Vision Sensing **1**-10
1.7 Summary **1**-12
References **1**-12

Lauren Collins
Winning Edge Communications

1.1 Introduction

The function of sensor networking integrates computation, communication, and sensing into a single, low-powered mechanism. Although the capabilities of a single mechanism may seem insignificant, the arrangement of hundreds or thousands of these devices tremendously advances technological opportunities. A sensor network comprises sensor nodes, linking physical and virtual objects through the exploitation of data capture and communication capabilities. The networks are able to collect sensory information with other connected nodes, oftentimes globally, and are efficient in performing certain processes either to make decisions or store useful data. The "Internet of Things" (IoT) is a concept where sensor networks allow devices, appliances, and transports to dynamically join the Internet to collaborate, track, and accomplish tasks. This chapter identifies use cases where object-identification sensor networking and connection capability serve as the basis for the development of independent cooperative services and applications.

Figure 1.1 illustrates the connectivity between a controller and the wireless network that serves *n* number of sensor nodes, three in this case. These sensor nodes collectively communicate with a central controller to assemble and configure themselves to perform subsequent operations. Complex deployments may successively extend control to cloud environments. Thus, as the need arises, sensor networks can be seamlessly expanded without interrupting or having to alter the physical layer of an existing infrastructure. Adaptation mechanisms may respond to topology changes or conversely could cause a network to shift between modes of operation.

The economy supports the vision for the use of sensor networking applications across diverse fields, including monitoring environmental conditions, aggregating health care, and deploying ubiquitous technology environments. Today, mobile devices directly connect people to other individuals and to devices. Unlike traditional wireless mechanisms, wireless sensor nodes do not need to communicate

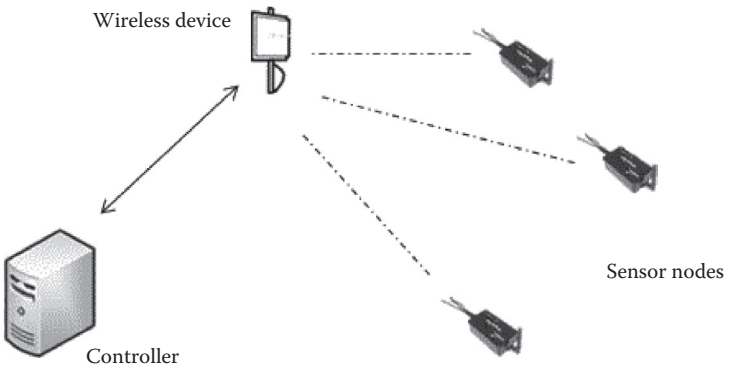


FIGURE 1.1 Sensor nodes communicate locally to a wireless network in order to transmit data back to a central collector.

directly with the nearest tower but simply with their local neighbors. The flexibility of each sensor becoming part of the overall infrastructure provides a mesh network to transmit data between the thousands of tiny embedded devices. These architectures dynamically acclimate and support the new implementation of new nodes or the extension of covering a greater geographic area. Furthermore, the system is capable of adapting to sensor failures, ascertaining a routing topology, and effectively transmitting data back to the collection point.

Automation is currently being deployed with wireless sensors, and the evolution of these disparate systems is on the brink. For example, a data center is easily monitored for water leaks or fire by sensors that, by design, form a wireless interconnection network and instantaneously alert the detection of fire or water. Contrasting from wired systems, deployments of wireless sensors are more cost effective and easier to build to scale. Instead of running thousands of feet of cabling routed through conduit, the installation of these tiny devices is robust and capable of monitoring for aberrations for years on a single set of batteries.

The future Internet, referred to as the "Internet of Things," takes a unique identifier and is capable of repeatedly transferring and relating data over a network without human or computer interaction. Automated decisions with the use of "cloud" applications and infrastructures support the capability of transporting your preferences. As new decisions are made, any environment has the ability to dynamically collaborate and proficiently accomplish tasks to meet your needs without you having to do anything. However, as so many networks comprise the Internet, we must meticulously investigate and analyze the challenges involved with this integration.

There are far too many legacy infrastructures in production, hindering the transition to sensor networking IoT. Security has raised significant concerns where businesses and their systems are still utilizing closed network infrastructures. The transition to an open, highly available IP scheme will allow for streamlined network architecture, capability for data sharing, and profound optimization for mobile integrations. The IoT will associate, share, and distribute data across inanimate objects. Sensor networking IoT presents the ability to collect and transmit data from sensors connected to objects such as animals, plants, infrastructures, and people. Sharing and processing such data permit technology to advance the usability of everything from technology equipment to medical devices to transportation vehicles.

1.2 Fields Using Sensor Networking

Sensor networks bestow a monitoring and control capacity supporting applications across multiple industries. Whether the industry is manufacturing, transportation, environmental, health care, or aerospace and military, sensor technology comprises signal processing, computation, and wireless networking.

The technological advancements made over the years have enabled the production and implementation of far more capable yet inexpensive devices, permitting millions of sophisticated applications, linking the physical world to sensor-driven data networks. Sensor network architecture necessitates the delivery of sensor information to the user at a low bit rate through low-powered transceivers. Dense geometry and integrated network sensors are now distributed at a fraction of the cost of typical sensor and actuator systems. Application instances range from local to global throughout factory automation, environmental monitoring, medicine, and condition-based preservation.

The evolution of sensor networking supports varying uses in the field, validating a comprehensive architecture that is technology independent. Computational resources not only lead to efficient resource utilization but also permit a rich interface between application and protocol processing. Consequently, developers are able to make use of application-specific communication protocols, significantly refining system performance. Vigilant and accurate selection of nodes, accelerators, storage, and wireless communication demonstrates the capabilities through a collection of real-world deployments respective to each field. Additionally, performing an all-inclusive theoretical analysis leads to improved real-time, efficient data collection without sacrificing flexibility.

1.2.1 Environmental Sensors

Environmental sensor networks present a commanding blend of distributed sensing capacity, real-time data visualization and analysis, and integration with adjacent networks and respective remote sensing data streams. The scope of environmental sensor networks ranges from measuring global patterns to those only involved in monitoring a 600 ft^2 data center. Environmental sensors have the ability to analyze and monitor conditions as well as facilitate the early detection of conditional deficiencies. Energy efficiency allows for appropriate measures to be taken to ensure restoration of a system's optimal operating levels.

The science of environmental data collection encompasses numerous sensor readings from a set of points in an environment over n period of time and then has the ability to detect trends and dependencies, subsequently furnishing reports or making decisions based upon the set of data. Consider hundreds or thousands of data points spread throughout the city and collecting data for several months in an effort to analyze trends. For the data to be meaningful, it must be collected at regular intervals with the nodes remaining at predefined established locations. At the network level, environmental data collection is illustrated by having nodes continually sensing and transmitting data back to base stations that collect and store data using an application or conventional methods. In typical usage scenarios, nodes are evenly distributed over an environment. The distance between adjacent nodes may be insignificant, yet the distance containing the entire network is momentous.

1.2.2 HVAC Systems

Environmental controls for heating, ventilation, and air conditioning (HVAC) and electricity can create phenomenal cost savings for both businesses and homeowners. Sensor-enabled HVAC systems now offer integrated weather data and prediction analysis to understand and plan for energy use and even have control of the systems from mobile devices. The abilities of sensor networking provide value as an added benefit to HVAC control, enabling people to embrace the new technology. Current infrastructures support the integration of sensor networks into existing building management roles. Basic integrations utilize the placement of multiple sensors into the control system, providing perception into the behavior of long-term sensor network deployments. HVAC control architecture has progressed over time, making the integration of wireless sensors more feasible.

Cabling used to be run directly from the electrical panel to the sensor inputs and actuator outputs. The functionality of these control systems was fundamental and unyielding, and adds, moves, or changes commanded extensive rerouting of wiring and connections. The control architecture is presented in

a distributed approach rather than a closed, wiring intensive system. Earlier systems encompassed smaller installations, consequently utilizing a centralized scheme with a master sensor having multiple slave sensors. As systems and their complexity increased, multimaster and multislave systems started to flourish. Tiered architectures were on the rise, including gateways, controllers, and repeaters. Single points of failure were introduced, and device failures negatively affected systems. After the deployment of wireless sensor nodes, the nodes discover the topology of the network and determine optimal routing paths.

In environmental monitoring, it is not essential for the nodes to develop the optimal routing paths. The sensors and nodes perform a specific function, where the network devices calculate the ideal routes and communicate vital data out to the nodes. Since the physical topology of these networks is relatively constant, the only variation is in the nature of radio frequency (RF) communication. Communication between nodes may become intermittent, but the overall network consistency remains stable and is able to effectively deliver data. Once the network is configured, the nodes will intermittently poll their sensors and transmit data up to the core. Depending on the deployment and infrastructure, data transmission may take milliseconds, while others may take minutes. The conventional environment parameters being monitored, such as light intensity, humidity, and temperature, do not change rapidly enough to require superior reporting rates.

Real-time sensors placed throughout the office space, or in a single data center room, allow each sensor node to turn the area's HVAC on or off. Businesses and homeowners continue to conserve energy and eliminate costs of operations or electricity. Too many commercial buildings continue to employ a static occupancy schedule as shown in Figure 1.2, which denotes system runtimes to coincide with working hours and staff presence. Implementing sensor network architectures to monitor and control temperatures meets the needs of both the cycling of HVAC systems along with building performance requirements in an effort to satisfy their tenants. In addition to the design and implementation of a low-cost and accurate wireless sensor node system, control architecture is critical to define the thresholds for zones. The sensors all connect back to a base station, which send relevant data to a database collector to analyze the sensor data and make decisions based on that data.

1.2.3 Data Center Sensors

What could be more critical than the happiness and comfort of employees in an office? Equipment. Too often, servers and switches are thrown in a small closet if an office is small or there is not a lot of equipment needed to run an office. These closets usually have poor to no ventilation and usually no cooling or humidity monitoring solutions in place. When our equipment is deployed in the said conditions, that equipment is forced to run at higher processing rates at all times, not just when processing power is needed to perform calculations or procedures. Over time, components of these systems will fail quicker than if the elements were monitored and accounted for to efficiently maintain the equipment and environments.

For all the businesses that are not considered "small businesses" or have more than one closet of equipment, data centers accommodate a phenomenal amount of equipment. It would be impossible to monitor and control the elements of a 24 h data center without wireless sensor network nodes. Energy efficiency continues to be an issue in some data centers; however, once a system is put into place, nodes are able to sense and coordinate data to control elements such as temperature and humidity distribution. Unfortunately, implementing sensor nodes and networking them is not the only component to ensure efficiency in these large data centers. Figure 1.3 exhibits the strategic placement of equipment racks in a hot aisle/cold aisle arrangement.

Even though data center designs have incorporated hot aisle and cold aisle containment for years, energy efficiency continues to present challenges. Sensor networks were able to present data proving more needed to be done in an effort to contain the elements. Figure 1.4 illustrates the two solutions data centers are implementing to further assist sensor nodes in controlling the environment. Plastic curtains

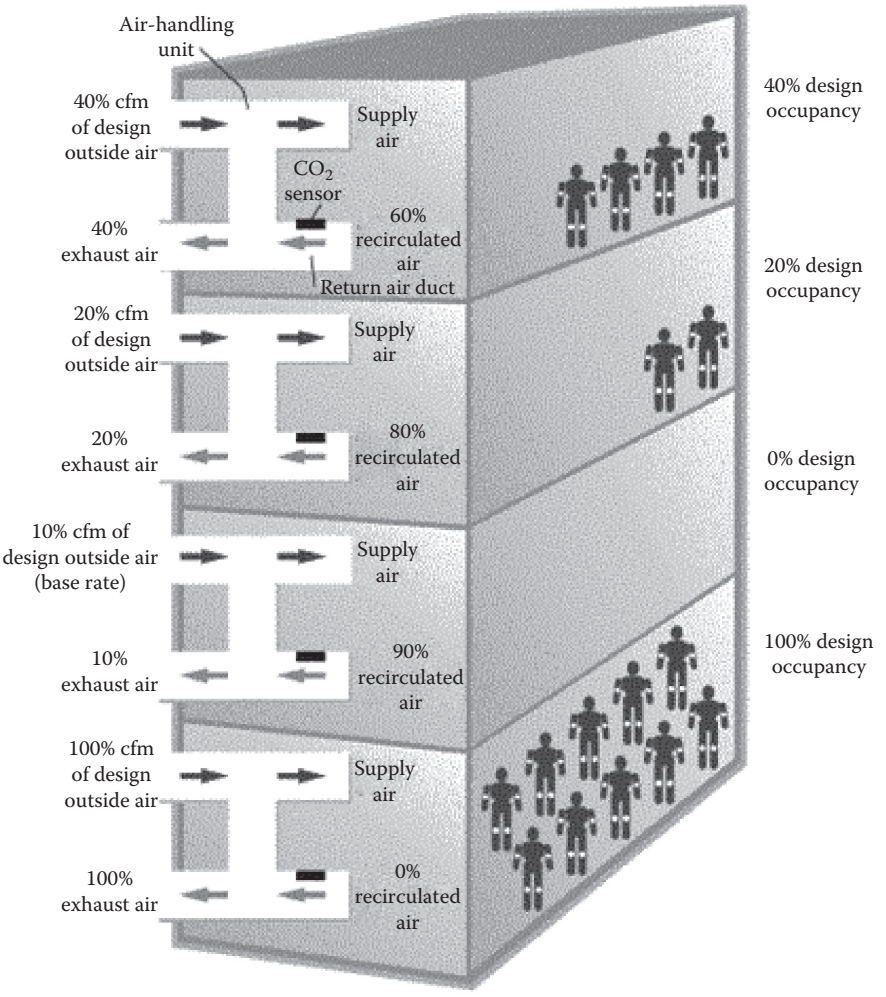


FIGURE 1.2 Buildings have the ability to control the environment in spaces with multiple floors and differing business hours correlated to the amount of employees present.

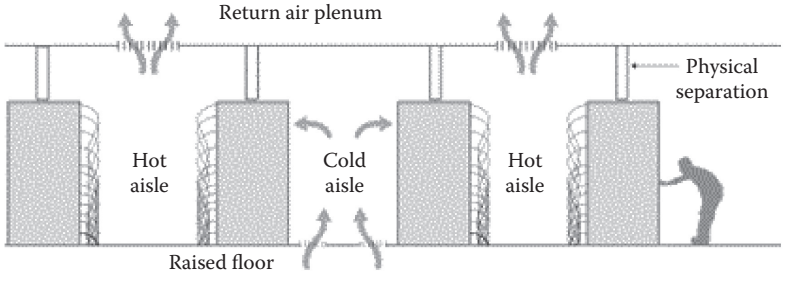


FIGURE 1.3 Hot and cold aisle containment is present in data centers designed to work in tangent with sensor nodes to increase efficiency.

FIGURE 1.4 The picture on the left shows plastic curtains data centers have implemented to contain hot and cold temperatures. The picture on the right is a more viable solution but at an exponentially higher cost.

are able to physically separate the temperatures but introduce other issues, such as making it difficult to freely carry equipment while moving around the curtains and running cables under and around the curtains. Subsequently, multiple manufacturers now offer complete prefabricated containment systems, incorporating the racks, air containment, and cabling trays.

1.2.4 Spray Foam Insulation

Annual heating and cooling energy exemplifies 75%–85% of the total life-cycle environmental impact of a structure. Building on the controls put into place to monitor temperature, humidity, and efficiency is the technological innovation of construction sensor deployments to analyze and monitor moisture in dwellings with wood construction. Moisture monitoring is a perilous component of all wood frame construction but not a factor in systems controlling nuclear reactors or biomedical interstitial space, where a fragment of contaminated moisture could prove to be structurally damaging. Green construction currently promotes the installation of moisture detection systems in new wood-dwelling construction projects. Wireless units, shown in Figure 1.5, are placed spatially for n number of sensors to connect to in order to collect data and then make calculations based on the data. If you have ever built a home, permits and building codes do not easily warrant the ease of placement of these wireless units nor do they support the necessity for hardline power to be run to the wireless units. These units can be mounted in an area that is out of sight, still allowing the technician with easy access if necessary.

Consequently, these environmental monitoring systems must be deployed at the final framing stage of construction, so the leak detection tape or embedded moisture sensors can be run in properly to provide real-time monitoring in the future. The process of applying open or closed spray foam insulation in ceilings or walls has been known to trap moisture in diverse assemblies. With the desire to build big wood frame structures, real-time structural analytics closely monitor the air quality and fabrication integrity to prevent the deterioration of materials. Figure 1.6 exhibits two types of moisture sensing systems. On the left, embedded sensors provide an indirect moisture management solution applicable to concrete, masonry, or supplementary substrates. The sensors are embedded in material to obtain the relative moisture levels of the surrounding area through capillary absorption. On the right of Figure 1.6, a flat adhesive tape is shown comprising copper conductors. The tape is designed with a low profile and works the

FIGURE 1.5 Wireless access units are placed throughout a home or business structure. Depending upon the size of deployment, there may only be one unit necessary for all the sensors to connect back to.

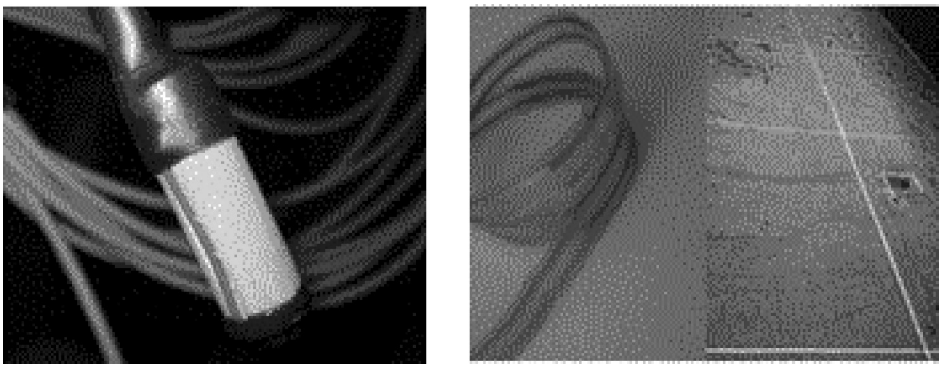


FIGURE 1.6 Pictured left to right, embedded moisture sensor used in concrete or masonry construction and a flat moisture detection sensor tape for use in other applications such as wood construction.

same way as the neighboring picture. Sensor readings are acquired by a data acquisition device and transmitted to the collector, where sensor-specific temperature-compensated calibration curves are applied. Calibration constants are individually applied to each sensor for accurate moisture content calculation.

There are significant challenges for sensor deployments where far more data are generated than can be managed by conventional approaches in field research, positioning data quality and integrity beyond the capacity for individuals to efficiently control. There is a need for automated quality assurance and control, assurance of the integrity of the sensors that may be affected by aging or bad connectivity, and the error checking that should be done. As networks age and environmental conditions deteriorate equipment, expect nodes to fail over time. Periodically, the network converges to account for link or node failures or to redistribute network loads. The most significant features of environmental requirements are precise synchronization, low data rates, sustainable deployments, and static network topologies.

1.2.4.1 Medical Care and Health-Care Innovations

Technological advancements have the potential to augment medical procedures, serving as a means to educate students and to develop the approaches by which doctors perform surgeries. Universities are using a new technique that will improve the experience for the patient and simplify the process for surgeons. Since technology evolves so quickly, it is necessary for us to continue educating ourselves with the medical breakthroughs and advancements that are used today and build upon them. Therefore, continuously making iterations and improvements using image- and robotic-guided computer technology can lead to medical innovations that can help make the world a better place.

Therefore, applications employing sensor networking in the medical field will simplify procedures and make knowledge more readily available to care for patients. Additionally, doctors' care for patients is now tightly integrating the use of computers, technology, and any number of applications that may incorporate the use of IoT. Back in the days when computers were not as significant, doctors would use paper and pen to document and keep track of people's medical needs and appointments. Now when you go to the doctor or hospital, all of your medical history is maintained within a proprietary, secure application. Distinctive parameters enable the physician to monitor a patient and their care.

It is necessary to educate students and doctors, to employ professionals in the work force, and to take note of the progress made by these advancements in medical technology. By doing this, technology compliments education, which is what we need in the world today. If we can get people interested in this field, they will be able to have meaningful relationships and make an impact on lives. Teaching hospitals are valuable places to test theories and use image- and robotic-guided computer technology. Many doctors have been practicing too many years to be able to envision how technology can be used in an operating room. These hospitals have students who bring a mindset of technology into the operating room or to the medical profession. Doctors and other medical professionals are traveling globally to hospitals for a chance to use robotics and experience the applications and use of IoT to perform surgeries. These systems and methods will not replace surgeons but will assist them in planning and performing successful surgeries in the future.

1.3 Implanted Medical Devices and Physiological Parameters

Today, treating patients exists in facilities outside of hospitals or a provider office. The use of analog-to-digital converters (ADCs) [1] utilizes minimal power consumption, therefore reducing the patient's exposure to electromagnetic radiation. The converter exudes sustained battery life and is powered wirelessly. These sensors, which are implanted in the patient's heart or brain, ultimately reduce the intrusiveness and costs associated in providing care, utilizing applications to monitor and restore major functions. The reduction in capacitor charging voltage and the quantity of measurements in the data conversion sequences surges energy efficiency to 19.5 fJ per conversion, impacting overall power savings without sacrificing data resolution and affecting other hardware functionality. The converter is fabricated with 0.18 μm complementary metal–oxide–semiconductor (CMOS) processes, a mature standard processing technology sustaining significant volume manufacturing.

CMOS is a technology responsible for the composition of integrated circuitry. Many technical individuals relate a CMOS battery to the internal battery of a computer. Well, take that one step further and at a simplified scale, and sensor nodes employ the same technology. There are two relevant characteristics of the composition making up CMOS pertinent to medical devices and sensor nodes: low static power consumption and high noise immunity. CMOS logic dispels a considerably lesser amount of power than comparable logic circuits since CMOS only dissipates power when switching. Consider medical technology for patient care; a reduced number of arithmetic steps simplifies the logic, design, and data conversion cycles.

As the world population grows, we must identify the emerging needs to augment technology and define how that technology can be used in the world of medicine. Computer-guided surgeries are taking

place today, and the technology continues to grow related to this field. Not only do these types of technology allow for a longer life, these advancements result in more confidence from both the patient and medical staff. Aside from the physical benefits robotic sensors offer in medicine, neurology patients are also benefitting from tractography-guided surgery. High-definition fiber tractography (HDFT) [2] incorporates a mechanism that traces the course of nerve and fiber connections throughout the body. By observing the flow of water molecules, HDFT has the capacity to reconstruct the unprecedented elements and structures through complex fiber crossings.

In the long run, the advances in computer technology are just going to keep growing and developing. It has become a basic necessity in the modern everyday life. From education to medicine, sensor networking IoT technology serves as an integral benefactor to how the medical world operates. Predicting the future of the Internet is truly impossible; however, trending sensor networking IoT will be at the center of the medical field and the advances to come.

1.4 Sensor Networking in Manufacturing and Production

The growth and adoption of the IoT is a significant component of efficient manufacturing and production. Although logistic companies have employed sensors and automation for multiple decades, the controllers, sensors, actuators, and management systems are severely disconnected from IT systems and engineering teams. High-priced production equipment has been profoundly implemented in a wired, closed network infrastructure (some are even still using token ring cabling). The industrial grade sensors and controllers were much more expensive to purchase and implement, making it extremely difficult to upgrade existing legacy infrastructures in plants and warehouses. As economic growth increases, solutions are continually implemented and benefiting manufacturing companies worldwide.

1.5 Location-Based Services

The implementations of IoT institute data collection from sensors and allow for effective communication between software systems, plant supervisors, floor workers, and all aspects of a manufacturing outfit, including logistics. Everyone has checked the availability of an item online before a purchase. Sensor data can be used in location-based services (LBS), such as asset tracking, supply monitoring, and access management. Product management software is able to provide real-time information regarding available stock. The same technology used in access control is under the umbrella with the radio frequency identification (RFID) interface. Once an item's RFID tag is scanned, the information is transmitted using an Internet gateway and then uploaded into the "cloud," making it accessible to everyone. The data can also be sent to the cloud utilizing Wi-Fi, as shown in Figure 1.7.

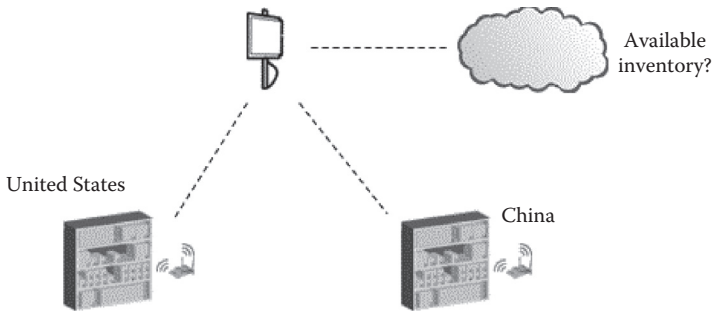


FIGURE 1.7 Sensor networking incorporates LBS to provide users with real-time inventory statistics. In this figure, two global plants are able to communicate wirelessly back to the cloud and dynamically update as stock increases or decreases.

1.6 Vision Sensing

Industrial applications strive to fulfill increasingly elevated obligations regarding extreme precision and superior reliability. The integration of sensor systems and applications supports the capability to fulfill fast-paced, accurate manufacturing conditions. Automobile and parts manufacturers have employed the use of vision sensor technology, permitting the validation of quality of manufactured items. Human inspections formerly assisted sophisticated machines; however, the traditional methods for industrial inspection and control are no longer realistic. The speed of today's manufacturing process has added to the complexity of production lines. Subsequently, plant owners and managers are integrating new solutions to monitor real-time quality and control of the manufacturing process.

Machine vision systems utilize a sensing technique that can perform automated tests that verify the color, shape, integrity, and preciseness. Harsh environmental factors do not inhibit these sensor networking devices. Whether these sensors are placed in dirty areas, alongside high vibrations, or in extreme temperatures, the sensors perform flexibly and flawlessly. Applications utilizing vision sensors can range from classification, inspection, assembly authentication, to process control and monitoring. The same sensor technologies also play an integral part in medicine as applications and operations are using robot guidance during medical procedures.

Implementing a vision sensor into an existing product line is easier than you would think. The major obstacle with traditional product lines is the slow operation and the occasional need to take a machine off-line. Conversely, when implementing vision sensors, identifying manual elements, and configuring automatic visual sensing, the automated analysis of the data considers the entire solution by defining measurements, identifying analysis control, and encompassing product functionality and quality. The following is an example of items a plant manager may identify when specifying initial requirements to implement sensors in automated, rigorously controlled environment. General requirements when implementing a vision sensor are as follows:

- Identify features considered mandatory and those that are optional.
- Define gauging system and measure line speed.
- Define real-time speed of response to analysis and production.
- Implement variable for any derivatives, ability to accommodate future variations.
- Define measurements, geometry, and aperture situations.
- Insert statistical process control (SPC) and quality.
- Implement a closed-loop response to the line for process control.

Once the requirements have been identified, an application can be written to measure all defined variables. The level of complexity will dictate whether one or multiple controllers may be necessary. Figure 1.8 pictures the "master controller" at the top, and the other two pictured below the dotted line are slave controllers. The master controller performs sequencing operations and translates information received into raw data. A controller is composed of an interface and an SPC, which has the capacity to restrain main line function in the event the unit is produced out of specifications or in the event of a major line fault.

The implementation of vision sensing consists of arranging equipment, then connecting and configuring the machine vision to interface with a programmable logic controller (PLC). The PLC offers automation solutions for high-level applications by storing procedures, functions of positioning and sequencing control, and timed counting of the input/output of units. The master controller performs sequencing operations, translating information received via the operator panel (OP). The two slave controllers have extended sensor capabilities via a series of add-on multiplexers (MUX), which undertake the vision process for the production line.

The automation of manual process possesses one primary intention: institute means of one or multiple manual processes in an effort to streamline processes and increase quantity, maintaining quality. The concept behind implementing visual sensors to automate product lines assumes a fault-free

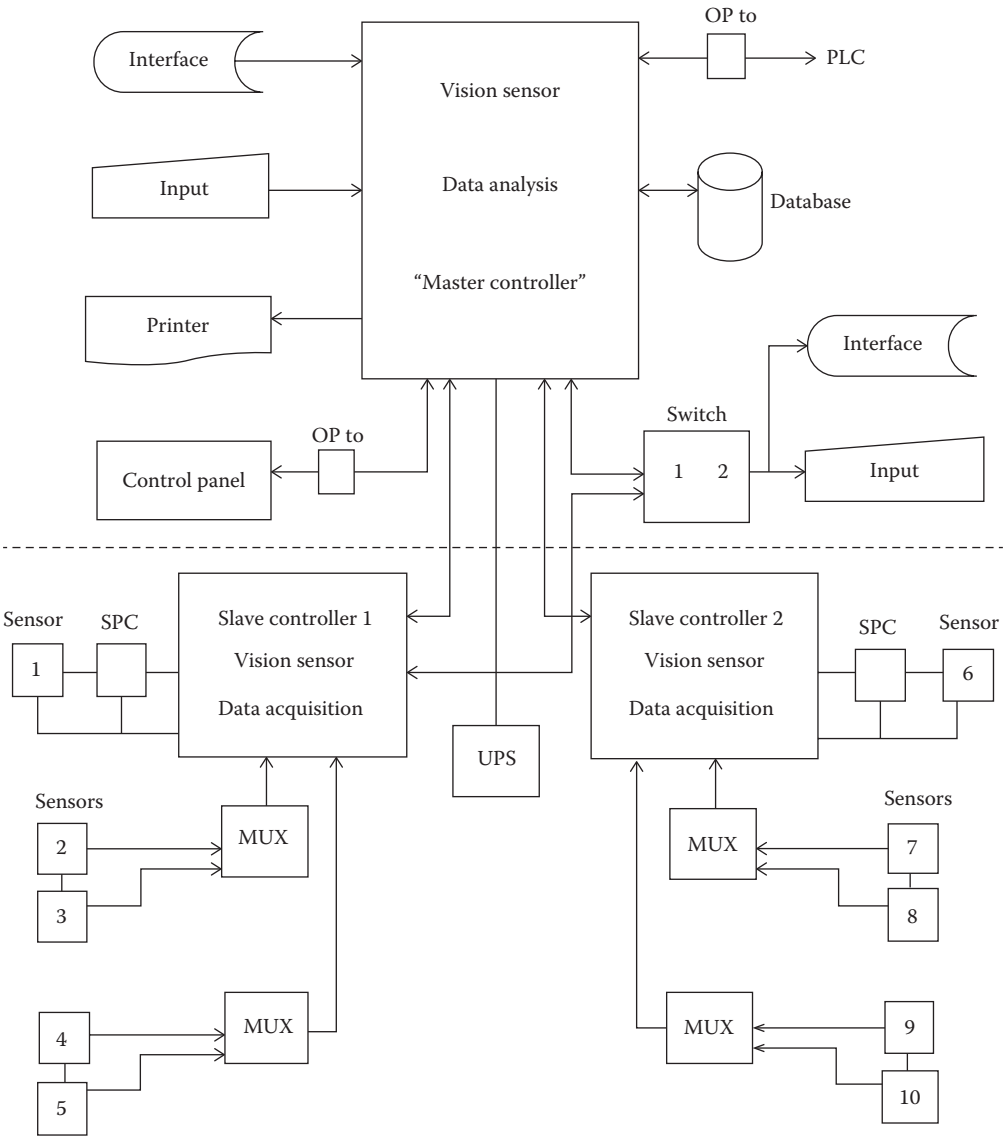


FIGURE 1.8 Sensor hardware architecture and application detail for manufacturing processing.

measurement signal, as identified and defined in the list presented earlier. The sensor transfer function is unit specific as such; several sensors may be placed in an effort to measure multiple aspects of a unit. Once the appropriate communication lines have been established between the controllers, the vision sensors automatically begin data collection. Vision sensors are placed in such a way to capture and collect data pertaining to the list presented earlier. Images and calculations are stored in buffers as operation time is of the essence. In order for meaningful calculations to be done, it is necessary to measure the output of a sensor in response to an accurate known input. This process is known as calibration, and the devices that produce the input are described as calibration standards.

Connected sensors provide an unprecedented level of visibility into operations and workflow in a much broader range than manual processes. The data gathered connect the proper people to relevant information and relevant devices and have the ability to include suppliers, partners, and distribution.

Visualization is improved across the board giving management access to data regarding equipment and line efficiencies, and tools and alerts from any location at much lower costs than former custom, manual systems.

Flexibility and innovation have become two key goals in manufacturing, hence the trend toward automated manufacturing systems. Flexible manufacturing systems are designed to construct a variety of products from universal machinery with a minimum number of workers. The ultimate design of a system incorporates material in the form of bars, plates, powder, and oil to produce any compilation required exclusive of manual intervention. Evidently, this model ensures robotic control with sensors and nodes serving as the foundation for the means of effective and efficient manufacturing today.

1.7 Summary

Sensor networking IoT endeavors to exemplify the networking of systems and enhance the compilation of data by traveling in a less linear fashion. The common thread among the deployments of sensor networks is whether or not demand for the data and applications will support the capacity needed to transmit the data in a useful format and in a timely manner. A challenge continuing to present suppliers and technologists is perfecting the design and implementation of sensor networks utilizing minimal power consumption yet delivering optimal processing technology. Data and networking were formerly transport centric, the concentration being on delivering packets. Considering application innovation coupled with the need to transport data efficiently and cost effectively, the market must evolve to a service-centric mentality.

Sensor network deployments are found in all types of industries, not only the ones named in this chapter. The military, space, and weather sectors heavily rely on data analytics gathered from sensor nodes, and the IoT allows immediate decisions to be made based on collectors' interpretation of calculated results. Substantial variations in business and technology paradigms are set forth in the following chapters, guiding professionals and technologists to competitive strategies and practices relative to the sensor networking IoT.

References

1. Lee, Bernadette. A*STAR IME develops ultra low power analog-to-digital converter for medical devices and wireless sensor nodes. Institute of Microelectronics, Singapore, August 20, 2012. Web. September 07, 2014.
2. P.J. Basser, MRI tractography based transit time determination for nerve and muscle fibers, Department of Health and Human Services, Washington, DC, 2014.

2

Sensor Network Platform and Operating Systems

Xinheng (Henry) Wang
University of the West of Scotland

Shancang Li
University of the West of Scotland and University of Bristol

2.1 Introduction **2**-1
2.2 Architecture of a WSN Node **2**-1
2.3 Reconfigurable WSN Platforms **2**-5
2.4 Sensing **2**-7
2.5 Communications **2**-7
2.6 Power Unit **2**-8
2.7 Operating Systems **2**-10
2.8 Summary **2**-12
References **2**-12

2.1 Introduction

Platform is defined as "a standard for the hardware of a computer system, which determines what kinds of software it can run" by Oxford Dictionary. As its definition implies, a platform defines the hardware and software of a system, also a standardized solution to the problems. The platform of wireless sensor network (WSN) is, therefore, a system of standardized hardware and software that enables sensing and wireless networking. This makes sensing and networking the two most important components in WSNs.

A good platform will bring many benefits to the system designer, manufacturer, and service operator. It will reduce the time of products to market. This is particularly attractive in modern competitive international market. Based on the basic platform design, further components could be added or removed from the platform to meet a specific demand. Thus, its basic hardware components and software could be reused by other applications. In addition, based on a basic platform and through further development, the bugs in the software and hardware faults could be corrected over and over to improve the performance of the platform. Further developed applications based on the basic platform will be more reliable, and a high-quality service could be guaranteed.

WSN platform comprises two subplatforms: one is the subplatform of individual network node; the other is the subplatform of networked nodes. In this chapter, we will first look at the design of individual node, and then we will look at the design and management of the network formed by these nodes.

2.2 Architecture of a WSN Node

For a WSN node to work properly and fulfill its basic functions, it has to have four basic components, namely, sensing, communication, processing, and power, as shown in Figure 2.1.

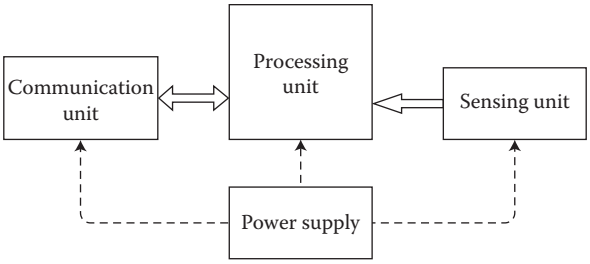


FIGURE 2.1 Basic architecture of a WSN node.

Processing unit plays an extremely important role in the platform. Based on the power and size of the platform, the processing unit could be classified into four levels, from embedded PCs to PDA-class platforms, such as Stargate [1] and CerfCube [2], and from low-power motes to even lower-power system-on-a-chip (SoC) platform.

Embedded PCs and PDA-class platforms roughly form the same class in terms of its hardware architecture. Both are employed with 32-bit CPUs and megabyte memories and secondary storage such as flash memory or disk. However, embedded PCs are no longer used nowadays because of the evolvement of technology and its size limitation that makes them difficult to be deployed. PDA-class platforms are mainly used as gateways to collect, store, and process the data from the sensor network and communicate with external networks. Here, Stargate and CerfCube are used as an example to illustrate the architecture design of this kind of platforms.

Stargate was designed by Intel and then licensed to Crossbow Technology for commercialization. The functioning block of a Stargate is shown in Figure 2.2.

Stargate employs an Intel PXA255 processor, which uses the Intel XScale™ microprocessor. XScale is a 32-bit RISC microarchitecture, which is noted for its efficiency by using less silicon transistors and thus consuming less power, making it smaller and less expensive to manufacture. It is also efficient in processing multimedia data through the support of 16-bit data types and enhanced 16-bit multiply and accumulates operations that accelerate the multimedia CODEC algorithms.

On Stargate board, working with PXA255 processor is the Intel SA-1111 companion chip, which enables direct access to one SDRAM system memory and one 32 MB flash memory and provides buffering for one PCMCIA slot and one compact flash slot for further storage. A MICA2/GPID connector and an optional I2C are also provided to connect the Crossbow's MICA2 sensor nodes and other electronic devices to Stargate smoothly. A watchdog timer, an LED, and a gas gauge can also be found embedded on the board. Stargate's daughter card provides extra interfaces for more functional components to be embedded on the board, including a 10Base-T Ethernet port, a USB port, a JTAG port, and an external ac power supply adapter port.

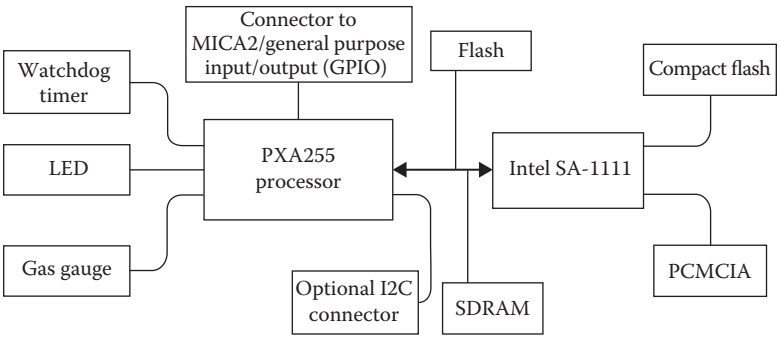


FIGURE 2.2 Stargate block diagram.

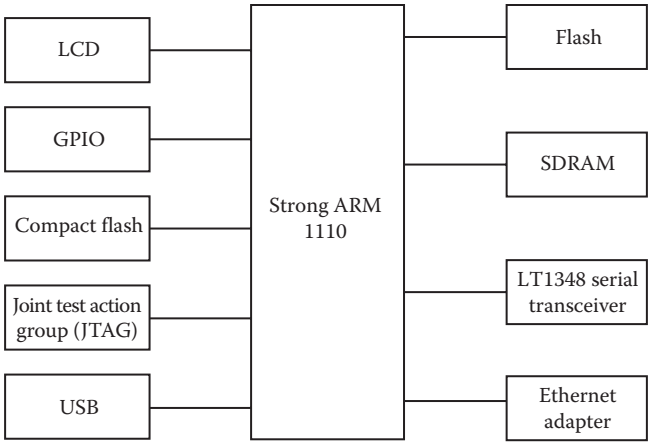


FIGURE 2.3 Block diagram of CerfBoard.

CerfCube represents another main series of platforms by implementing ARM processors, where XScale and ARM processor are two mainstream processors for handheld devices. CerfCube is the product of Intrinsyc Software International, Inc. Inside CerfCube is its board called CerfBoard, whose block diagram is shown in Figure 2.3.

CerfBoard has a few series of boards with different processors. Take CerfBoard V3.0 as an example, it integrates an Intel Strong ARM 1110 Microprocessor, 16 MB flash, 32 MB SDRAM, an Ethernet port, three RS232 serial ports, LCD interface, CompactFlash socket, USB interface, JTAG interface, and 16 digital I/O lines that all lines have programmable interrupt capacity and the first four have LED indicators.

Scaling down from PDA-class platforms integrated with processors like XScale and ARM is another popular class of sensor platforms. Motes are the most famous examples in this class. Motes are a legendary sensor platform, which was originally developed by UC Berkeley and licensed to industry for commercialization.

Take TelosB developed from Crossbow as an example to illustrate its architecture and functionalities. The block diagram is shown in Figure 2.4.

Telos (ver. B) platform employs an 8 MHz TI MSP430 microcontroller with 10 kB RAM, a 1 MB external flash for data logging, an IEEE 802.15.4/ZigBee compliant RF transceiver with integrated on-board antenna, a USB interface for data collection and programming, and an optional on-board sensors of light, temperature, and humidity.

Comparing with Stargate and CerfCube, there are a few significant differences in processing unit between motes and embedded PC. The main differences are listed in Table 2.1.

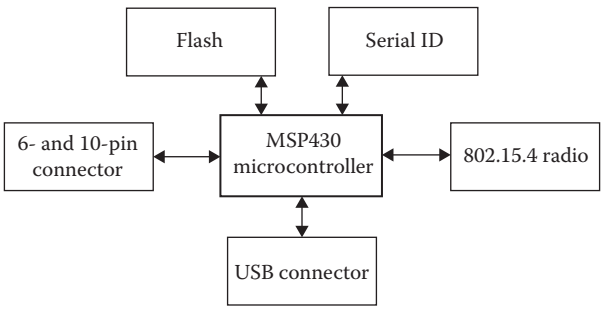


FIGURE 2.4 Block diagram of TelosB.

TABLE 2.1 Differences in Processing Unit between Motes and Embedded PC

		Motes	Embedded PC
Processor	Type	Microcontroller	Microprocessor
	CPU clock	Low, for example, 8–25 MHz for TI MSP430	High, for example, 100–400 MHz for PXA255
	Power consumption	Low, several milliwatts	High, several hundreds milliwatts
RAM		Integrated in microcontroller, a few kilobytes	External, dozens of megabytes
Flash		Integrated in microcontroller, size in kilobytes	External, size in megabytes

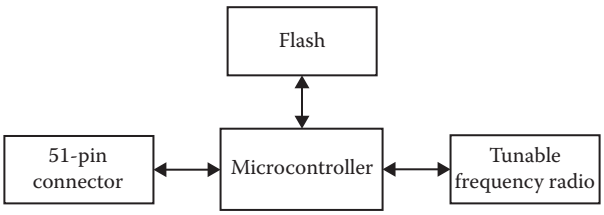


FIGURE 2.5 Block diagram of a MICA2 node.

The microcontroller is a small device on a single integrated circuit containing a processor core, memory, and programmable I/O peripherals. Flash memory and a small amount of RAM are also integrated on chip. Because it operates at low clock rate frequency and low bit words, it consumes far less power than microcontroller used in embedded PC, which enables it suitable for developing battery-powered sensor node.

The latest successor of TelosB is MICA2 mote. The block diagram of a MICA2 node is shown in Figure 2.5.

MICA2 features several improvements over its predecessors, including tunable multichannel transceiver, TinyOS distributed operating system, and support for wireless remote programming. MICA2 platform employs an Atmel ATmega128L microcontroller that runs TOS from its internal flash memory. Using TOS, the processor can be configured to run sensor application/processing, and the network/radio communications stack simultaneously. Its 51-pin expansion connector supports analog inputs, digital I/O, I2C, SPI, and UART interfaces. These interfaces make it easy to connect to a wide variety of external peripherals [3]. These features would enable it to be more suitable for commercial applications. However, it is still based on microcontroller architecture. Many varieties of this class of platforms are available on the market for research and commercial applications. Readers may find dozens of platforms easily on Internet or published papers. The most popular examples include BTnode [4], Nymph [5], iSense [6], and Smart Dust [7].

The latest development of WSN platform is the SoC platform. SoC is a further step from microcontroller-based platform to integrate all components into a single chip, including communication unit and sensing unit. A key advantage of SoC is that the size of device will be reduced, which is particularly attractive to design wearable sensor networks or body area networks. Here, an example of monitoring electrocardiography (ECG) by a SoC solution is illustrated in [8] to demonstrate its architecture design.

The platform architecture is shown in Figure 2.6. As shown in Figure 2.6, this SoC platform has five functional blocks, including (1) a system startup module (SSM) for remote battery control and initial frequency allocation of communication unit that a 5% duty-cycled body-channel communication

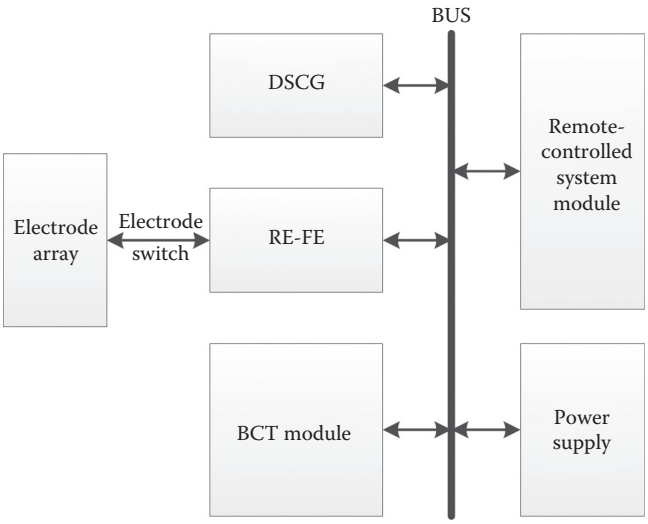


FIGURE 2.6 Platform architecture of a SoC design for monitoring ECG.

is used; (2) four reconfigurable electrode sensor front ends (RE-FE), with each connected to four voltage sensing electrodes to achieve reconfigurable sensing and digitization; (3) a differential sinusoidal current generator (DSCG) for a 5% duty-cycled high-quality balanced current injection; (4) a digital module containing a finite state machine controller with special-purpose registers, a 20 kB SRAM for data storage, a 10th-order FIR filter, an 8:1 compression block, and a packet encoder/decoder; and (5) a duty-cycled body-channel transceiver (BCT) for low-energy external data communication.

There are two significant differences between this SoC-based design and previously described embedded PC and microcontroller-based motes: one is its controller design, and the other is the communication design, which will be discussed later. The controller used in this is a state machine based one.

A state machine is generally defined as a mathematical model of computation used to design both computer programs and sequential logic circuits. It is conceived as an abstract machine that can be in one of a finite or infinite number of states. A finite state machine is a state machine with a finite number of states.

Generally, a state machine is not good for numerical calculation but good for any system that occupies various distinct states. In addition, a state machine is good for any applications with state changes, such as user interface design, pattern recognition tasks, and communication protocols. Also, state machines could be used to construct simple controllers for simple tasks, for example, the controller in ECG monitoring that we have just discussed.

A key advantage of a state machine is its ease to modify. This characteristic makes the state machines the backbone of field-programmable gate array (FPGA), which is a reconfigurable design. This forms another category of sensor platform design, which is a reconfigurable sensor platform.

SoC in designing sensor network platform is still in its infantry stage. A lot of efforts are required to evaluate the performance, flexibility, cost effectiveness, and also environmental resilience of this kind of design.

2.3 Reconfigurable WSN Platforms

Different from microprocessor-based WSN platform, reconfigurable, this is also referred to as reprogrammable, but reconfigurable is used throughout this chapter to reduce the confusion. WSN platform offers the flexibility in implementing application-driven functionalities and ability in reducing the energy consumption, which is critical to WSN applications.

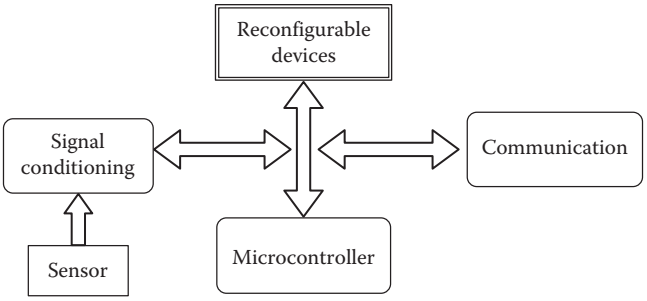


FIGURE 2.7 Block diagram with microcontroller and reconfigurable device.

Recent development has seen two types of hardware implementation in terms of reconfigurability: one is the combination of microcontroller with reconfigurable device, which I called half reconfigurability, and the other is pure reconfigurable device without microcontroller. Examples of the first type of implementation could be found in [9,10]. Its cooperation with other modules like microcontroller, sensing, and communication is illustrated in Figure 2.7.

As shown in Figure 2.7, a reconfigurable device is controlled by a microcontroller. Its function acts like a gate for microcontroller. It generates control signals for the signal conditioning unit. Whenever it detects data acquired from the sensor, it might trigger the microcontroller to accept it.

An example of a pure reconfigurable device–based WSN platform is demonstrated by Nokia wrist-attached wearable devices [11]. Its system architecture is shown in Figure 2.8. In this example, the reconfigurable device is FPGA, which has the full control of the platform.

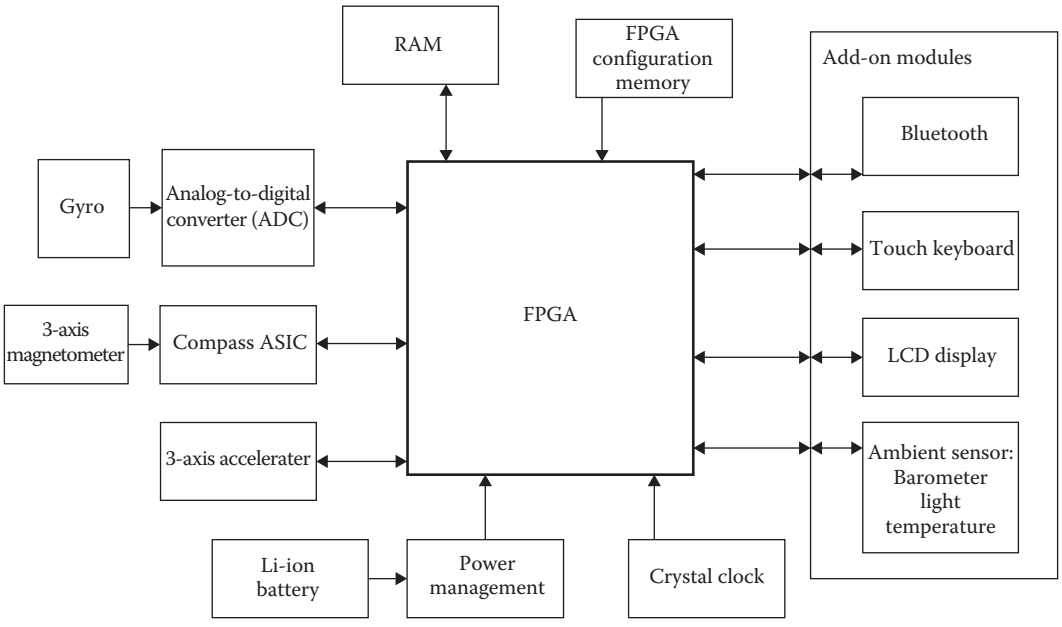


FIGURE 2.8 FPGA-based WSN platform.

2.4 Sensing

While processing unit is the core of a WSN platform that controls the data acquisition, transmission, and possible data processing, sensing unit is the one that interacts with the environment to convert the physical reading into a signal that the system can read and understand. With the development of electronic and manufacturing technologies, sensors are becoming smaller and more accurate. However, electronics and manufacturing are out of the scope of this book; further details about sensor technologies can be found from the handbook *Sensor Technology Handbook* [12].

2.5 Communications

The communication unit plays another important role in a WSN. It decides how the sensor nodes are working together and how the sensor nodes transmit the acquired physical data to a data receiving unit. This is the unit that forms wireless sensor nodes into a network.

The communication unit forms the network into different scales and formats, such as one-to-one simple network, one-to-many or many-to-one network, or many-to-many networks scaling from small to large with few to hundreds or thousands of nodes.

Strictly to WSNs, communication is done via wireless communication protocols over RF channel. A few popular wireless communication protocols have been developed and widely used in WSNs because of their characteristics in forming reliable, large-scale WSNs or commercial maturity in some applications. Some other protocols are also used in some critical environments.

In terms of WSN, ZigBee is possibly the number one protocol because of its low power consumption and capability in forming wireless mesh and ad hoc networks, which enrich the applications of WSNs. The WSN formed by ZigBee could reach a very large scale, theoretically supporting 65,535 network nodes [13]. However, in practice, it is impossible to reach such a high number because of various constraints. The largest network seen so far is more than 800 nodes, implemented in University of California, Berkeley, and the Intel Berkeley Research Lab [14]. Military networks may be larger, but no report is released.

The basis of ZigBee is the IEEE 802.15.4 standard, which specifies the physical layer and media access control for low-power wireless networks. Other wireless networking protocols, such as ISA100.11a, WirelessHART, and MiWi, are also based on this standard.

Bluetooth is possibly the second popular networking protocols in forming a WSN. The networking architecture of Bluetooth is quite different from ZigBee. Bluetooth uses master–slave model to form a network, where a master node can communicate with a maximum of seven slave nodes. This forms a *piconet*. However, the slave node can also act as a master node and communicate with another seven slave nodes, which forms a scatternet. However, because of the limitation of direct communication between slave nodes and formation of the network, the size of the network formed by Bluetooth normally is not big.

The popularity of applying Bluetooth in WSNs is because of commercial availability and support of many electronic devices. Bluetooth is widely used in health-care monitoring where a large sensor network is not necessary.

The advantage of Bluetooth is its higher transmission rate comparing with ZigBee, with the price of high power consumption. However, with the introduction of the latest version of Bluetooth low energy (BLE), Bluetooth still stands a strong candidate in WSNs in some applications like health-care monitoring.

Ultra wideband (UWB) is another technology quite often used in WSNs. One main characteristic of UWB is its high data transmission speed. This makes it a best choice in WSNs for video transmissions, that is, video surveillance [15]. Another characteristic of UWB is its ability in accurate localization, which makes it one of the best tools in location and tracking in WSNs, that is, outdoor sports [16]. The basis of UWB is IEEE 802.15.4a, which specifies the physical layer of UWB and also chirp spread spectrum (CSS).

Because of the advantages of WSNs and industrial needs, but limitations of technologies of ZigBee, Bluetooth, UWB, etc., in industrial environment, two specific industrial standards were developed, namely, WirelessHART [17] and ISA100.11a [18]. WirelessHART is the enabling wireless capability on existing HART protocol. The three key capabilities claimed by the developer are reliability, security and privacy, and effective power management, which make it a good option in process plant. At the same time, ISA100.11a aims to provide secure and reliable wireless communication for noncritical monitoring and control applications. They have used the similar physical layer technologies that are specified by the IEEE 802.15.4 standard but different architecture and technologies in data link layer and network layer, which enable them to support different topologies of the networking, however reliable and secure in process and manufacturing industries.

Both protocols compete in the same market to become the de facto standard. The competition is going on. The performance of these two protocols is being improved. However, who is going to win stands a mystery.

A detailed comparison between these two protocols is out of the scope of this chapter. However, readers can refer to the reference for further details [19].

IEEE 802.15.4 is the standard that specifies the physical layer and media access control layer for low-rate wireless personal area networks. It is the basis for the wireless protocols described earlier, such as ZigBee, Bluetooth/BLE, UWB, WirelessHART, and ISA100.11a. In order to enable these protocols to operate for IPv6 network addressing, a new protocol IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) was developed [20]. 6LoWPAN has defined the encapsulation and header compression mechanisms. Examples of applications of 6LoWPAN could be seen in hospital monitoring [21], home automation design [22], and agricultural environment monitoring [23].

The standard protocols discussed earlier are widely used in WSNs to form the single-hop or multihop wireless networks for monitoring and control. In order to enable them to work efficiently in WSNs, some other concerns such as mobility management, energy-efficient routing, and source scheduling should be considered as well. However, this is out of the scope of this chapter.

Standard protocols have obvious advantages in applications, that is, ease of implementation. However, the coexistence of some of the protocols is a big problem because of sharing of the same frequency band, that is, 2.4 GHz frequency band, which causes serious interference problems. This is more critical in some sensitive applications like health care. Therefore, nonstandard communication technologies are quite often used in this kind of applications. For example, in the aforementioned SoC implementation of monitoring cardiac disease, a 5% duty-cycled body-channel communication link is implemented for data transmission, which is more energy efficient.

2.6 Power Unit

Power unit is the simplest unit in a WSN platform but plays an important role in the determination of the lifetime of the WSNs. Because most of the sensor nodes are battery powered, how to make the battery work longer to extend the lifetime of the network node is becoming a core issue that the WSN designer, developer, and applicant have to consider. In order to reduce the power consumption in a WSN platform, quite a few measures have been taken, including scheduling wake-up and sleep pattern, adoption of low-power microcontroller and SoC design, application of low-power communication protocols, energy-efficient routing protocols, new design of cross-layer protocols, and task scheduling schemes. This involves the consideration of every aspect of a WSN platform.

Except those techniques mentioned earlier, one popular technique to make the battery/platform work longer is the application of energy harvesting technology by utilizing external energy sources. Energy harvesting, also known as power harvesting or energy scavenging, is a process by which energy is derived from external sources, for example, solar power, thermal energy, wind energy, salinity gradients, and kinetic energy, and stored for the wireless sensor node to work for longer time [24]. All these forms of energy harvesting have been successfully used in WSN platforms.

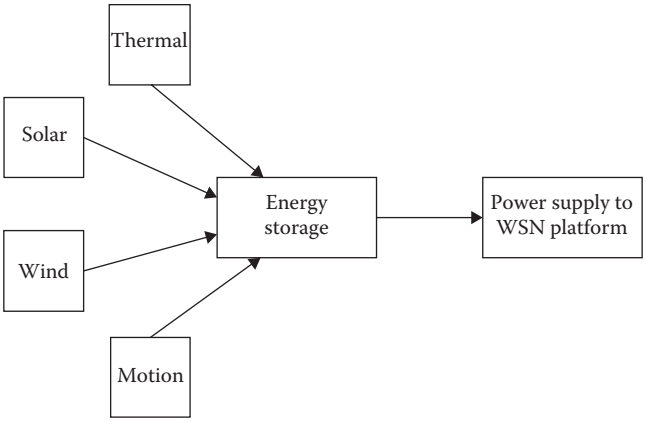


FIGURE 2.9 Schematic approach of energy harvesting.

The schematic approach of energy harvesting is simple, shown in Figure 2.9. Relevant devices convert solar, wind, thermal, or motion into energy. Then, the energy will be stored in a WSN platform by the form of recharging the battery or supercapacitor. Rechargeable battery or supercapacitor can work with or without battery on the WSN platform to supply power. Depending on the system requirement or environment, one or more forms of energy harvesting technologies could be used together such as solar and wind.

Solar power harvesting is possibly the most popular energy harvesting technology used for WSNs. Its solution is simple; normally, a solar panel is attached to the WSN platform. Examples of applying solar power harvesting include AmbiMax [25], Heliomote [26], Prometheus [27], and Everlast [28]. Although solar harvesting is a mature technology, one important issue still needs to be considered. This is efficiency. Efficiency is important in energy harvesting because it affects the cost, design, and lifetime of the system. In order to increase the efficiency, one key technique is the application of maximum power point tracking (MPPT) to obtain the maximum available power. MPPT is widely used in solar energy harvesting but also applicable in wind generation and all other energy harvesting.

While solar and wind are popular energy resources for WSNs in environment monitoring, they are hardly used in body area sensor networks for health-care monitoring because of the inability in deploying such kind of extra devices around the body. Energy harvesting for health-care monitoring depends on the energy generated from the human body itself, such as thermal energy and motion energy. For example, authors in [29] demonstrated a thermal energy harvesting technology to provide power for monitoring ECG, electromyography (EMG), or electroencephalography (EEG) signals, illustrated in Figure 2.10. They have used a commercial off-the-shelf (COTS) thermoelectric generator, which usually provides a very low output voltage (<100 mV) that is not high enough to power the sensor system to work. In order to solve this problem, the authors designed a boost converter of converting voltages as low as 30 mV. The energy harvested from the thermal is able to make the signal acquisition, signal processing, and transmission work without a battery. A batteryless sensor node is developed, which is ideal for health monitoring.

Another popular source of energy generated from the human body is the motion, for example, movement of limbs. The transformation from the motion to electrical power could be done by any kind of electromechanical transduction, such as electrostatic, electromagnetic, and piezoelectric. While design of electrostatic and electromagnetic transduction is more complex, piezoelectric becomes the main form for transferring body motion to electrical energy.

The phenomenon of piezoelectric is that the electric charge could be accumulated on some materials under mechanical stress, as illustrated in Figure 2.11. Therefore, to make piezoelectric happen,

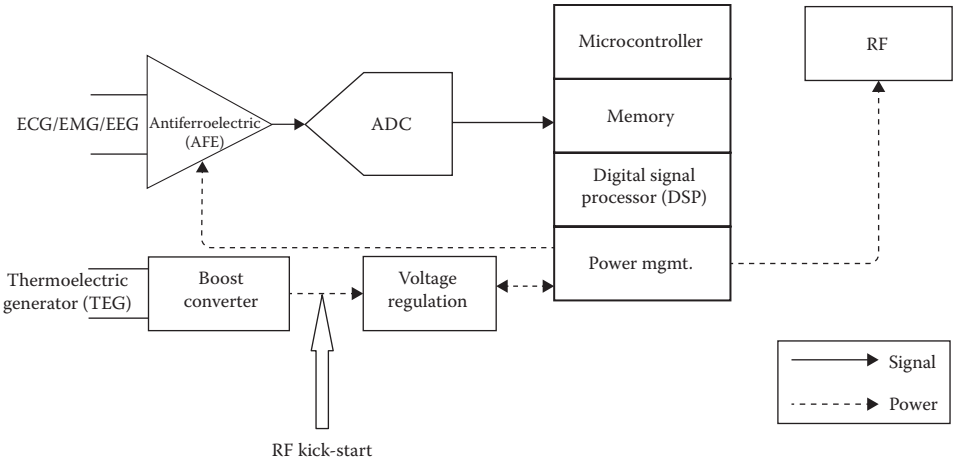


FIGURE 2.10 Thermal harvesting for body area sensor networks.

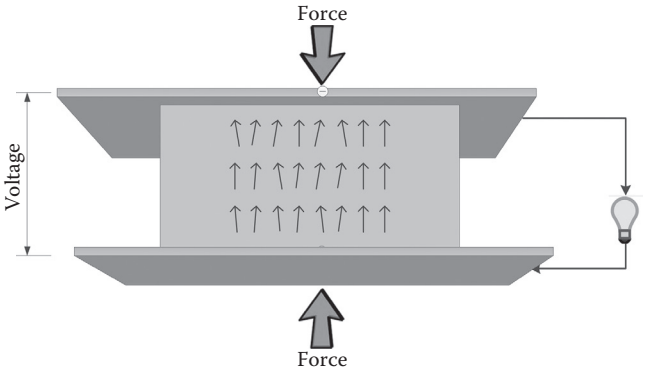


FIGURE 2.11 Illustration of piezoelectric.

the most important is to find the right material. One popular material used in piezoelectric harvesting is polyvinylidene fluoride (PVDF).

The history of piezoelectric energy harvesting is not long. Over the last decades, a few application technologies have been developed, by using the movement of the human body like walking and movement inside the human body like respiration and blood flowing. The first one to use it to harvest the energy from human walking to powering the electronic devices is researchers from the Media Lab at Massachusetts Institute of Technology (MIT) in the late 1990s [30]. At the same time, the pioneers of harvesting the energy from the motion of ribs when respiring [31] are researchers from Universität des Saarlandes and fluctuation of blood pressure [32] are from University of Pittsburgh, where these technologies are used to power implantable devices. Details of these technologies could be found from the references.

2.7 Operating Systems

Section 2.2 introduced the architecture and key components of a wireless sensor platform. These are the hardware part. How to make this hardware to work effectively is the job of operating system. This section will introduce the operating system for a wireless platform.

When talking about WSN operating systems, TinyOS is the one that cannot be missed. TinyOS, written in nesC language, is the legendary operating system, originally developed for Crossbow's mote sensor nodes.

Regarding TinyOS, there are numerous papers, books, and wikis online to introduce it. A research paper titled "TinyOS: an open operating system for wireless sensor networks" [33], written by researchers who developed this program, has rich sources about the techniques and features about TinyOS. Here, we briefly introduce the key features of TinyOS.

Because of the key characteristics of WSNs of low power, low cost, small size, distributed, concurrency of multiple operations on one board, and application driven, TinyOS was designed to meet the requirements of these characteristics. In order to do that, two approaches are adopted throughout the course of design and development, namely, event centric and platform for innovation, where flexibility is focused to enable the platform for various applications.

TinyOS is a component-based embedded system. A component is a computational entity that has one or more interfaces. Applications are developed by connecting the interfaces of the components.

Components abstract the hardware by a three-layer hierarchy, called hardware abstraction architecture (HAA), as shown in Figure 2.12.

HAA employs three sublayers, hardware presentation layer (HPL), hardware adaptation layer (HAL), and hardware interface layer (HIL). HPL is right on top of the hardware to represent the full functional capacity of the hardware. It is hardware dependent but hides the most hardware-dependent codes in order to develop a higher-level abstraction. The middle layer of HAA is HAL. HAL is built on top of HPL to build useful abstractions. In contrast to HPL, it has two key characteristics: one is that state is allowed to be maintained in this layer for performing arbitration and resource control, and the other is that hardware-specific applications can be built, as illustrated in Figure 2.12. On the other hand, the top layer HIL is doing the opposite job. It will convert the hardware-specific abstractions provided by HAL to hardware independent for cross-platform applications.

From the aforementioned description, it can be seen that two types of applications could be developed in TinyOS: one is hardware dependent, based on HAL layer, and the other is hardware independent, based on HIL. These two types of applications could be run in parallel. Now let us have a look at how TinyOS interacts with the four key modules in a WSN platform.

Currently, TinyOS supports three types of microcontrollers, namely, Atmel ATmega128L, Texas Instruments MSP430, and Intel XScale PXA271, and a few platforms developed by using one of the three types of microcontrollers, for example, the TelosB, MICA2, MICA2DOT, MICAz, TinyNode, and BTnode3.

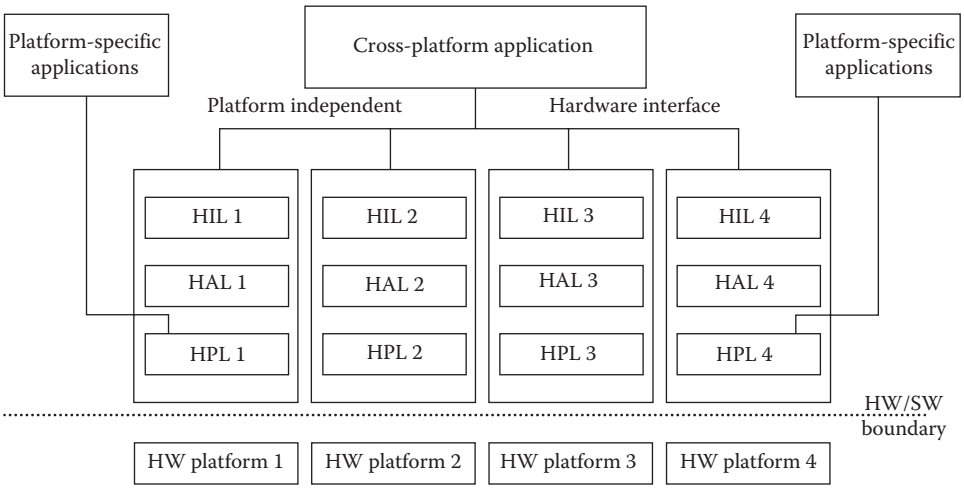


FIGURE 2.12 Hardware abstract architecture.

For sensors, data acquisition components are built on HIL interfaces. If a special component is required, for example, data acquisition for high frequency or accurate sampling, this component has to be built on HAL.

For communications, TinyOS adopts an unusual cross-layer optimization approach to develop a low-power stack. The current radio chips that it supports are CC1000, CC2420, CC2530, AT86RF212, and AT86RF230. In addition, various routing protocols have been developed for networking the sensor nodes into a WSN. In terms of power management, which is critical to the applications of WSNs, TinyOS divides the management into two types depending on the type of devices, namely, microcontroller power management and device power management.

Aside from TinyOS, some other tiny RTOSs have also gained the attention from a large community, such as UCOS [34], eCos [35], and μCLinux [36]. All these software packages are open source to support the research community. Some commercial ones are also available on the market, including MicroC/OS-II (Micrium), Nucleus (Mentor Graphics), and ThreadX (Express Logic). Different software packages have different features. Readers will find the details from the references.

2.8 Summary

This chapter describes the WSN platform and software. It started from describing the architecture of sensor nodes, scaling from embedded PC, microcontroller-based sensor node, to SoC. Then the details of each module are discussed, such as processing, communications, sensing, and power supply. Since the power consumption is a key element to the operation of WSNs, measures of energy harvesting are discussed to prolong the life of WSNs. Last, the operating system of WSN was discussed by taking TinyOS as an example.

References

1. Stargate platform. http://platformx.sourceforge.net/. Accessed on April 16, 2014.
2. CerfCube platform. http://ecos.sourceware.org/ecos/boards/cerfcube.html. Accessed on April 16, 2014.
3. MICA2 Wireless Measurement Systems. Document Part Number: 6020-0042-04. http://www.eol.ucar.edu/isf/facilities/isa/internal/CrossBow/DataSheets/mica2.pdf. Accessed on April 16, 2014.
4. Beutel, J., Kasten, O., and Ringwald, M., BTnodes—A distributed platform for sensor nodes, *First International Conference on Embedded Networked Sensor Systems, SenSys 2003*, Los Angeles, CA, November 5–7, 2003.
5. Chen, J., Fearing, R.S., Armstrong, B., and Burdick, J., NYMPH: A multiprocessor for manipulation applications, *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, CA, April 1986, Vol. 3, pp. 1731–1736.
6. Buschmann, C. and Pfisterer, D., iSense: A modular hardware and software platform for wireless sensor networks, 6. Fachgespräch Sensornetzwerke, 15, 2007, pp. 15–18.
7. Kahn, J.M., Katz, A.R.H., and Pister, A.K.S.J., Next century challenges: Mobile networking for "Smart Dust", *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Seattle, WA, 1999, pp. 271–278.
8. Yan, L., Bae, J., Lee, S., Roh, T., Song, K., and Yoo, H.-J., A 3.9 mW 25-electrode reconfigured sensor for wearable cardiac monitoring system, *IEEE Journal of Solid-State Circuits*, 46(1), 353–364, January 2011.
9. Portilla, J., Riesgo, T., and De Castro, A., A reconfigurable FPGA-based architecture for modular nodes in wireless sensor networks, *Third Southern Conference on Programmable Logic, 2007* (*SPL'07*), Mar del Plata, Argentina, February 28–26, 2007, pp. 203–206.
10. Jovanov, E., Milenkovic, A., Basham, S., Clark, D., and Kelley, D., Reconfigurable intelligent sensors for health monitoring: A case study of pulse oximeter sensor, *Proceedings of IEEE Engineering in Medicine & Biology Society*, 7, 4759–4762, 2004.

11. Ahola, T., Korpinen, P., Rakkola, J., Ramo, T., Salminen, J., and Savolainen, J., Wearable FPGA based wireless sensor platform, *Proceedings of IEEE Engineering in Medicine & Biology Society*, Lyon, France, pp. 2288–2291, August 22–26, 2007.
12. Wilson, J., *Sensor Technology Handbook*, Elsevier, Amsterdam, the Netherlands, 2004.
13. Pan, M.-S., Tsai, C.-H., and Tseng, Y.-C., The orphan problem in ZigBee wireless networks, *IEEE Transactions on Mobile Computing*, 8(11), 1573–1584, November 2009.
14. Largest Tiny Network Yet. http://webs.cs.berkeley.edu/800demo/, 2001. Accessed on April 16, 2014.
15. Huang, X., Dutkiewicz, E., Gandia, R., and Lowe, D., Ultra-wideband technology for video surveillance sensor networks, *2006 IEEE International Conference on Industrial Informatics*, Singapore, August 16–18, 2006, pp. 1012–1017.
16. Oppermann, I., Stoica, L., Rabbachin, A., Shelby, Z., and Haapola, J., UWB wireless sensor networks: UWEN—A practical example, *IEEE Communications Magazine*, 42(12), S27–S32, December 2004.
17. WirelessHART protocol. http://www.hartcomm.org. Accessed on April 16, 2014.
18. ISA100.11a protocol. http://www.isa100wci.org/. Accessed on April 16, 17, 2014.
19. Petersen, S. and Carlsen, S., WirelessHART versus ISA100.11a: The format war hits the factory floor, *IEEE Industrial Electronics Magazine*, 5(4), 23–34, December 2011.
20. http://6lowpan.net/. Accessed on April 16, 2014.
21. Jara, A.J., Zamora, M.A., and Skarmeta, A.F.G., HWSN6: Hospital wireless sensor networks based on 6LoWPAN technology: Mobility and fault tolerance management, *International Conference on Computational Science and Engineering, 2009* (*CSE'09*), Vancouver, British Columbia, Canada, August 29–31, 2009, Vol. 2, pp. 879–884.
22. Tudose, D.S., Voinescu, A., Petrareanu, M., Bucur, A., Loghin, D., Bostan, A., and Tapus, N., Home automation design using 6LoWPAN wireless sensor networks, *2011 International Conference on Distributed Computing in Sensor Systems and Workshops* (*DCOSS*), Barcelona, Spain, June 27–29, 2011, pp. 1–6.
23. Wang, X. and Yin, X., Agricultural environment real-time monitor and control system based on 6LoWPAN sensor networks, *Transactions of the Chinese Society of Agricultural Engineering*, 26(10), 224–228(5), October 2010.
24. Jain, A., and Bhullar, S., Emerging Dimensions in the Energy Harvesting, *IOSR Journal of Electrical and Electronics Engineering*, 3(1), 70–80, November and December, 2012.
25. Park, C. and Chou, P.H., AmbiMax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes, *Third Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2006* (*SECON'06*), Reston, VA, September 25–28, 2006, Vol. 1, pp. 168–177.
26. Raghunathan, V., Kansal, A., Hsu, J., Friedman, J., and Srivastava, M.B., Design considerations for solar energy harvesting wireless embedded systems, *Fourth International Symposium on Information Processing in Sensor Networks, 2005* (*IPSN'05*), Los Angeles, CA, April 15, 2005, pp. 457–462.
27. Jiang, X., Polastre, J., and Culler, D., Perpetual environmentally powered sensor networks, *Proceedings of the Fourth International Symposium on Information Processing in Sensor Networks*, Los Angeles, CA, 2005.
28. Simjee, F. and Chou, P.H., Everlast: Long-life, supercapacitor-operated wireless sensor node, low power electronics and design, *Proceedings of the 2006 International Symposium on Low Power Electronics and Design* (*ISLPED'06*), Tegernsee, Germany, October 4–6, 2006, pp. 197–202.
29. Zhang, F., Zhang, Y., Silver, J., Shakhsheer, Y., Nagaraju, M., Klinefelter, A., Pandey, J. et al., A batteryless 19 μW MICS/ISM-band energy harvesting body area sensor node SoC, *2012 IEEE International Solid-State Circuits Conference Digest of Technical Papers* (*ISSCC*), San Francisco, CA, February 19–23, 2012, pp. 298–300.
30. Kymissis, J., Kendall, C., Paradiso, J., and Gershenfeld, N., Parasitic power harvesting in shoes, *Proceedings of the Second IEEE International Conference on Wearable Computing*, Pittsburgh, PA, USA, 1998, pp. 132–139.
31. Hausler, E. and Stein, E., Implantable physiological power supply with PVDF film, *Ferroelectrics*, 60(1), 277–282, October 1984.

32. Ramsay, M.J. and Clark, W.W., Piezoelectric energy harvesting for bio-MEMS applications, *Proceedings of SPIE*, 4332, 429–438, 2001.
33. Levis, P., Madden, S., Polastre, J. Szewczyk, R., Whitehouse, K., Woo, A., Gay, D., Hill, J., Welsh, M., Brewer, E., and Culler, D., TinyOS: An Operating System for Sensor Networks, *Ambient Intelligence*, Springer Berlin Heidelberg, 2005, pp. 115–148.
34. UCOS operating system. http://micrium.com/rtos/ucosii/overview/. Accessed on April 16, 2014.
35. eCos operating system. http://ecos.sourceware.org/. Accessed on April 16, 2014.
36. μCLinux operating system. http://www.uclinux.org/. Accessed on April 16, 2014.

3

Mobile Crowd Sensing

3.1 Introduction **3**-1
3.2 Mobile Crowd Sensing Applications **3**-2
Smart Cities • Road Transportation • Health Care and Well-Being • Marketing/Advertising
3.3 Mobile Crowd Sensing Applications and Platforms **3**-3
3.4 Data Reliability Issues in Sensed Data **3**-4
Traffic Jam Alerts • Citizen Journalism • Environment
3.5 McSense: A Mobile Crowd Sensing Platform **3**-5
System Architecture and Processes Involved • Prototype Implementation • User Study and Tasks Developed for McSense
3.6 ILR in Crowd-Sensed Data **3**-8
Assumptions • Adversarial Model • ILR Design • Security Analysis • Related Work
3.7 Experimental Evaluation: Field Study **3**-12
Evaluating the ILR Scheme • Influence of the Task Price on Data Quality
3.8 Simulations **3**-14
Simulation Setup • Simulation Results
3.9 Field Study Insights and Improving the ILR Scheme **3**-16
Correlation of User Earnings and Fake Photos • Correlation of Location and Fake Photos • Malicious User: Menace or Nuisance? • Influence of Maintaining a Reputation Score • User's Survey Results and Observations • Battery Consumption
3.10 Summary **3**-20
Acknowledgment **3**-20
References **3**-20

Manoop Talasila
New Jersey Institute of Technology

Reza Curtmola
New Jersey Institute of Technology

Cristian Borcea
New Jersey Institute of Technology

3.1 Introduction

Mobile sensors such as smartphones and vehicular systems represent a new type of geographically distributed sensing infrastructure that enables mobile people-centric sensing (Riva and Borcea, 2007; Sensor Lab at Dartmouth, 2013; Urban Sensing at UCLA, 2013). According to a forecast for global smartphone shipments from 2010 to 2017, more than 1.5 billion phones are expected to be shipped worldwide (Statista, 2014). Smartphones already have several sensors: camera, microphone, GPS, accelerometer, digital compass, light sensor, and Bluetooth as proximity sensor (Mardenfeld et al., 2010; Reality Mining Project, 2014), and in the near future, they are envisioned to include health and pollution monitoring sensors (Garmin, 2014; *MIT News*, 2014). Vehicular systems have access to several hundred sensors embedded in cars, and recent vehicles come equipped with new types of sensors such as radar and camera. Compared to the tiny, energy-constrained sensors of static sensor networks,

smartphones and vehicular systems can support more complex computations, have significant memory and storage, and offer direct access to the Internet. Therefore, mobile people-centric sensing can be a scalable and cost-effective alternative to deploying static wireless sensor networks for dense sensing coverage across large areas.

Smartphones have already enabled a plethora of *mobile sensing* applications (Abdelzaher et al., 2007; Campbell et al., 2006; Honicky et al., 2008; Mohan et al., 2008) in gaming, smart environments, surveillance, emergency response, and social networks. Specially, activity recognition through mobile sensing and wearable sensors has led to many health-care applications, such as fitness monitoring, elder care support, and cognitive assistance (Choudhury et al., 2008). The expanding sensing capabilities of mobile phones have gone beyond the sensor networks' focus on environmental and infrastructure monitoring where people are now the carriers of sensing devices, the sources, and the consumers of sensed events (Azizyan et al., 2009; Kansal et al., 2007; Lu et al., 2009; Miluzzo et al., 2008; Siewiorek et al., 2003).

Despite its benefits, mobile people-centric sensing has two main issues: (1) incentivizing the participants and (2) reliability of the sensed data. *Mobile crowd sensing* has been proposed as a solution for the first issue. A mobile crowd sensing platform plays a similar role with the one played by Amazon's Mechanical Turk (MTurk) (Amazon Mechanical Turk, 2013) or ChaCha (2013) in *crowdsourcing* (Gupta et al., 2012; Narula et al., 2011): it allows individuals and organizations (clients) to access a sheer number of people (providers) willing to execute simple sensing tasks for which they are paid. Unlike the MTurk's tasks that are executed on personal computers and always require human work, mobile sensing tasks are executed on mobile devices that satisfy certain context/sensing requirements (e.g., location, time, and specific sensors) and many times do not require human work (i.e., automatic sensing tasks). Many organizations and individuals could act as crowd sensing clients. For example, local, state, and federal agencies could greatly benefit from this new sensing infrastructure as they will have access to valuable data from the physical world. Commercial organizations may be very interested in collecting mobile sensing data to learn more about customer behavior. Researchers in many fields of science and engineering could collect large amounts of sensed data for various experiments. Ultimately, all of us could act as clients through many mobile apps (e.g., find out the traffic conditions ahead on the highway).

Regarding the data reliability issue, the sensed data submitted by participants in crowd sensing are not always reliable as they can submit false data to earn money without executing the actual task. This problem will be illustrated in Section 3.4, and our solution will be discussed extensively in this chapter.

3.2 Mobile Crowd Sensing Applications

In the following, we present several application domains that can benefit from mobile crowd sensing as well as a number of applications (some of them already prototyped) for each domain.

3.2.1 Smart Cities

Worldwide, cities with high population density and a very large number of interconnected issues make effective city management a challenging task. Therefore, several significant government and industrial research efforts are currently underway to exploit the full potential of the sensing data by initiating smart city systems to improve city efficiency by deploying smarter grids, water management systems (London's Water Supply Monitoring, 2012), and ultimately the social progress (IBM Smarter Planet, 2014). Around the world, the government of South Korea is building the Songdo Business District, a green low-carbon area that aims at becoming the first full-scale realization of a smart city (Songdo Smart City, 2014). Despite their potential benefits, many of these efforts could be costly. Crowd sensing can reduce the costs associated with large-scale sensing and, at the same time, provide additional human-related data. For example, our recent work on ParticipACTION (Cardone et al., 2013) proposes to leverage crowd sensing to directly engage citizens in the management of smart cities; people can actively participate in sensing campaigns to make their cities safer and cleaner.

3.2.2 Road Transportation

Departments of transportation can collect fine-grain and large-scale data about traffic patterns in the country/state using location and speed data provided by GPS sensors embedded in cars. These data can then be used for traffic engineering, construction of new roads, etc. Drivers can receive real-time traffic information based on the same type of data collected from smartphones (Mobile Millennium Project, 2014). Drivers can also benefit from real-time parking data collected from cars equipped with ultrasonic sensors (Mathur et al., 2010). Transportation agencies or municipalities can efficiently collect pothole data using GPS and accelerometer sensors (Eriksson et al., 2008) in order to quickly repair the roads. Similarly, photos (i.e., camera sensor data) taken by people during/after snowstorms can be analyzed automatically to prioritize snow cleaning and removal.

3.2.3 Health Care and Well-Being

Wireless sensors worn by people for heart rate monitoring (Garmin, 2014) and blood pressure monitoring (MIT News, 2014) can communicate their information to the owners' smartphones. Typically, this is done for both real-time and long-term health monitoring of individuals. Mobile sensing can leverage these existing data into large-scale health-care studies that seamlessly collect data from various groups of people, which can be selected based on location, age, etc. A specific example involves collecting data from people who eat regularly fast food. Phones can perform activity recognition and determine the level of physical exercise done by people, which was proven to directly influence people's health. As a result of such study in a city, the municipality may decide to create more bike lanes to encourage people to do more physical activities. Similarly, phones can determine the level of social interaction of certain groups of people (e.g., using Bluetooth scanning, GPS, or audio sensor). For example, a university may discover that students (or students from certain departments) are not interacting with each other enough; consequently, it may decide to organize more social events on campus. The same mechanism coupled with information from *human sensors* can be used to monitor the spreading of epidemic diseases.

3.2.4 Marketing/Advertising

Real-time location or mobility traces/patterns can be used by vendors/advertisers to target certain categories of people (Google Mobile Ads, 2014; Mobads, 2014). Similarly, they can run context-aware surveys (function of location, time, etc.). For example, one question in such a survey could ask people attending a concert what artists they would like to see in the future.

3.3 Mobile Crowd Sensing Applications and Platforms

Recently, several mobile crowdsourcing projects tried to leverage traditional crowdsourcing platforms for mass adoption of people-centric sensing: Twitter (2014) has been used as a publish/subscribe medium to build a crowdsourced weather radar and a participatory noise-mapping application (Demirbas et al., 2010); mCrowd (Yan et al., 2009) is an iPhone-based platform that was used to build an image search system for mobile phones, which relies on Amazon's MTurk (Amazon Mechanical Turk, 2013) for real-time human validation (Yan, 2010). This has the advantage of leveraging the popularity of existing crowdsourcing platforms (tens of thousands of available workers) but does not allow for truly mobile sensing tasks to be performed by workers (i.e., tasks that can only be performed using sensors on mobile phones). PEIR is an application for participatory sensing that exploits mobile phones to evaluate if users have been exposed to airborne pollution, enables data sharing to encourage community participation, and estimates the impact of individual user/community behaviors on the surrounding environment (Mun et al., 2009). Medusa is a mobile crowd

sensing framework that uses a high-level domain-specific programming language to define sensing tasks and workflows that are promoted with monetary incentives to encourage user participation (Ra et al., 2012). So far, none of these existing applications and platforms have addressed the reliability of the sensed data.

3.4 Data Reliability Issues in Sensed Data

By leveraging smartphones, we can seamlessly collect sensing data from various groups of people at different locations using mobile crowd sensing. As the sensing tasks are associated with monetary incentives, participants may try to fool the mobile crowd sensing system to earn money. Therefore, there is a need for mechanisms to efficiently validate the collected data. In the following, we motivate the need for such a mechanism by presenting several scenarios involving malicious behavior.

3.4.1 Traffic Jam Alerts

Suppose that the Department of Transportation uses a mobile crowd sensing system to collect alerts from people driving on congested roads and then distributes the alerts to other drivers (Herrera et al., 2010; White et al., 2011). In this way, drivers on the other roads can benefit from real-time traffic information. However, the system has to ensure the alert validity because malicious users may try to proactively divert the traffic on roads ahead in order to empty these roads for themselves.

3.4.2 Citizen Journalism

Citizens can report real-time data in the form of photos, video, and text from public events or disaster areas (Photo Journalism, 2013; Toyama et al., 2003). In this way, real-time information from anywhere across the globe can be shared with the public as soon as the event happens. But malicious users may try to earn easy money by claiming that an event is happening at a certain location while being somewhere else.

3.4.3 Environment

Environment protection agencies can use pollution sensors installed in the phones to map with high accuracy the pollution zones around the country (Intel Labs, 2010; Sensordrone, 2013). The participants may claim *fake* pollution to hurt business competitors by submitting the sensed pollution data associated with false locations.

Ultimately, the validation of sensed data is important in a mobile crowd sensing system to provide confidence to its clients who use the sensed data. However, it is challenging to validate each and every sensed data point of each participant because sensing measurements are highly dependent on context. One approach to handle this issue is to validate the location associated with the sensed data point in order to achieve a certain degree of reliability on the sensed data. Still, we need to overcome a major challenge: how to validate the location of data points in a scalable and cost-effective way without help from the wireless carrier? Let us note that wireless carriers may not help with location validation for legal reasons related to user privacy or even commercial interests.

To achieve reliability on participants' location data, there are a few traditional solutions such as using Trusted Platform Modules (TPMs) (2013) on smartphones or duplicating the tasks among multiple participants. However, these solutions cannot be used directly for a variety of reasons. For example, it is not cost effective to have TPMs on every smartphone, while task replication may not be feasible at some locations due to a lack of additional users there. Another solution is to verify location through the use of secure location verification mechanisms (Capkun and Hubaux, 2005; Capkun et al., 2006; Sastry et al., 2003; Talasila et al., 2010) in real time when the participant is trying to submit the sensing data location.

Unfortunately, this solution requires infrastructure support or adds significant overhead on user phones if it is applied for each sensed data point.

The rest of the chapter is organized as follows: Section 3.5 presents the overview of McSense, our mobile crowd sensing platform, and its prototype implementation. Section 3.6 describes our improving location reliability (ILR) scheme to achieve data reliability in McSense and analyzes ILR's security. The experimental evaluation and simulation results for ILR are presented in Sections 3.7 and 3.8, respectively. In Section 3.9, we discuss a number of lessons learned from our McSense field study as well as potential improvements for ILR. Finally, Section 3.10 concludes the chapter.

3.5 McSense: A Mobile Crowd Sensing Platform

We have designed and implemented McSense (Talasila et al., 2013), a mobile crowd sensing platform that allows clients to collect many types of sensing data from smartphones carried by mobile users. The interacting entities in our mobile crowd sensing architecture are as follows:

- *McSense*: A centralized mobile crowd sensing system that receives sensing requests from clients and delivers them to providers; these entities are defined next.
- *Client*: The organization or group who is interested in collecting sensing data from smartphones using the mobile crowd sensing system.
- *Provider*: A mobile user who participates in mobile crowd sensing to provide the sensing data requested by the client.

3.5.1 System Architecture and Processes Involved

The architecture of McSense, illustrated in Figure 3.1, has two main components: (1) the server platform that accepts tasks from clients and schedules the individual tasks for execution at mobile providers and (2) the mobile platform (at the providers) that accepts individual tasks from the server, performs sensing, and submits the sensed data to the server. The communication among all these components takes place over the Internet. Next, we discuss the overall process in more detail.

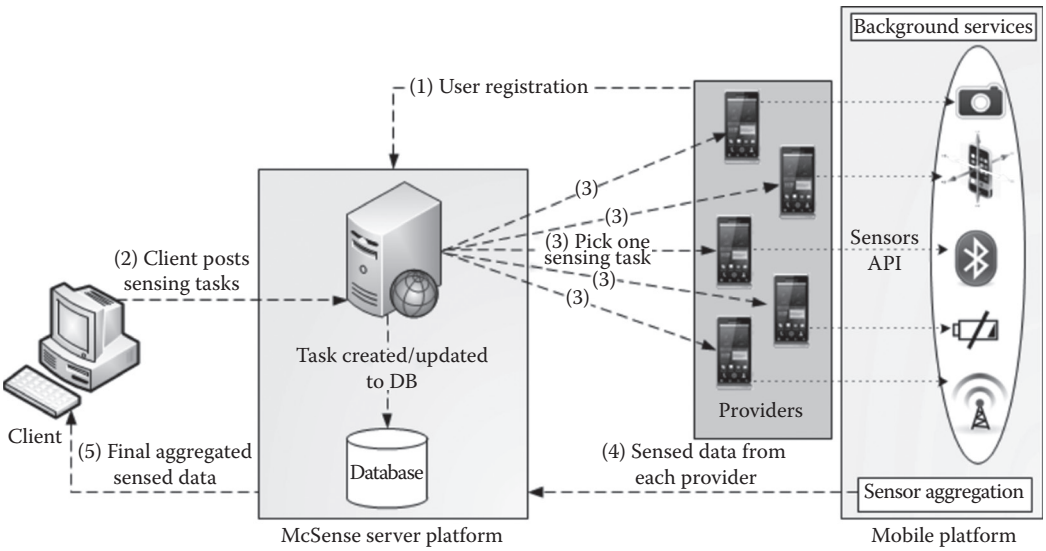


FIGURE 3.1 McSense architecture.

3.5.1.1 User Registration

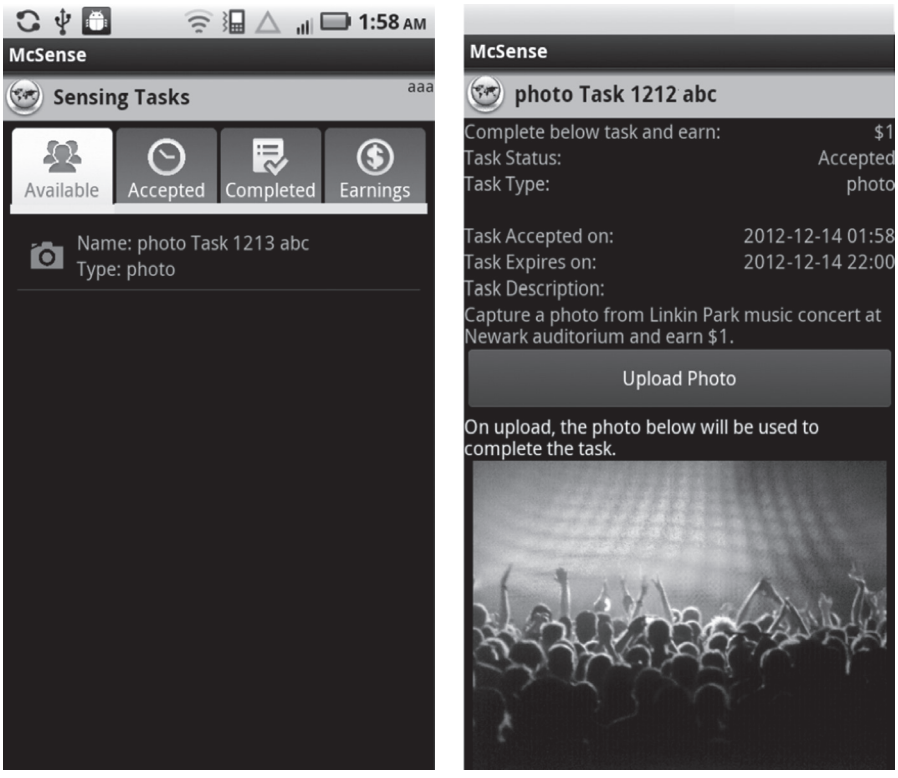
The McSense application on the smartphones shows a registration screen for first-time users, prompting them to enter an e-mail address and a password. During the registration process, the user phone's mobile equipment identifier (MEID) is captured and saved in the server's database along with the user's e-mail address and password. We chose to store the phone's MEID in order to restrict one user registration per device. In addition, the server also avoids duplicate registrations when users try registering with the same e-mail address again.

3.5.1.2 Posting New Sensing Tasks

New sensing tasks can be posted by clients using a web interface that's running on the McSense server. The sensing task details are entered on this web page by the client and submitted to the server's database. Once a new task is posted, the background notification service running on the provider's phone identifies the new available tasks and notifies the provider with a vibrate action on the phone. Providers can check the notification and can open the McSense application to view the new available tasks. When the application is loaded, the providers can see four tabs (Available, Accepted, Completed, and Earnings). The providers can view the list of tasks in the respective tabs (Figure 3.2a) and can click on each task from the list to view the entire task details (type, status, description, accepted time, elapsed time, completion time, expiration time, and payment amount).

3.5.1.3 Life Cycle of a Task

The life cycle starts from the Available tasks tab. When a provider selects an available task and clicks on the Accept button, the task is moved to the Accepted tab. Once a task is accepted, then that task is



(a) (b)

FIGURE 3.2 McSense android application showing tabs (a) and task screen for a photo task (b).

not available to others anymore (Figure 3.2b). When the accepted task is completed according to its requirements, the task is moved to the Completed tasks tab. Finally, the providers view their aggregated total dollars earned for successfully completed tasks under the Earnings tab. If the accepted task expires before completing successfully according to its requirements, it is moved to the Completed tasks tab and marked as unsuccessfully completed. The providers do not earn money for the tasks that are completed unsuccessfully.

3.5.1.4 Background Services on Phone

When the network is not available, a completed task is marked as pending upload. A background service on the phone periodically checks for the network connection. When the connection becomes available, the pending data are uploaded, and finally, these tasks are marked as successfully completed. If the provider phone is restarted manually or due to the mobile OS crash, then all the in-progress sensing tasks are automatically resumed by the Android's BroadcastReceiver service registered for the McSense application. Furthermore, the Accepted and the Completed tabs' task lists are cached locally and are synchronized with the server. If the server is not reachable, the users can still see the tasks that were last cached locally.

3.5.2 Prototype Implementation

The McSense application, shown in Figure 3.2, has been implemented in Android and is compatible with smartphones running Android OS 2.2 or higher. The application was tested successfully using Motorola Droid 2 phones, which have 512 MB RAM, 1 GHz processor, Bluetooth 2.1, Wi-Fi 802.11 b/g/n, 8 GB on-board storage, and 8 GB microSD storage. The McSense (2013) Android application was deployed to Google Play (2014) to make it available for campus students. The server side of McSense is implemented in Java/J2EE using the model–view–controller (MVC) framework. The Derby database is used to store the registered user accounts and assigned task details. The server side Java code is deployed on the GlassFish Application Server, which is an open-source application server.

3.5.3 User Study and Tasks Developed for McSense

To evaluate McSense and its data reliability protocol (see Section 3.7), we ran a user study at our campus for approximately 2 months. Over 50 students have participated in this study. Participants have been asked to download the McSense application from the Android market and install it on their phones. On the application server, we periodically posted various tasks. Some tasks have a monetary value associated with the task, which is paid on the task's successful completion; a few other tasks do not offer monetary incentives just to observe the participation of providers when collecting free sensing data. As tasks are submitted to the application server, they also appear on the phones where our application has been installed. Each task contains a task description, its duration, and a certain amount of money. The students use their phones to sign up to perform the task. Upon successful completion of the task, the students accumulate credits (payable in cash after the study terminated). The sensing tasks that we choose to use for this study fall into two categories:

1. Manual tasks, for example, photo tasks
2. Automated tasks, for example, sensing tasks using accelerometer and GPS sensors and sensing tasks using Bluetooth

3.5.3.1 Manual Photo Sensing Task

Registered users are asked to take photos from events on campus. Once the user captures a photo, she needs to click on the Complete Task button to upload the photo and to complete the task. When the photo is successfully uploaded to the server, the task is considered successfully completed. These uploaded photos can be used by the university news department for current news articles.

3.5.3.2 Automated Sensing Task Using Accelerometer and GPS Sensors

The accelerometer sensor readings and GPS location readings are collected at 1 min intervals. The sensed data are collected along with the user ID and timestamp, and they are stored into a file in the phone's internal storage, which can be accessed only by the McSense application. These data are then uploaded to the application server on task completion (which consists of many data points). Using the collected sensed data of accelerometer readings and GPS readings, we can identify user activities such as walking, running, and driving or locations that are important to the user. By observing the daily activities, we could find out how much exercise each student is getting daily and derive interesting statistics such as which departments have the most active and healthy students.

3.5.3.3 Automated Sensing Task Using Bluetooth Radio

In this automated sensing task, the user's Bluetooth radio is used to perform periodic Bluetooth scans (every 5 min) until the task expires; on its completion, the task reports the discovered Bluetooth devices with their location back to the McSense server. The sensed data from Bluetooth scans can provide interesting social information such as how often McSense users are near to each other. Also, it can identify groups who are frequently together to determine the level of social interaction of certain people (Mardenfeld et al., 2010).

3.5.3.4 Automated Resources Usage Sensing Task

In this automated sensing task, the usage of user's smartphone resources is sensed and reported back to the McSense server. Specifically, the report contains the mobile applications' usage, the network usage, the periodic Wi-Fi scans, and the battery level of the smartphone. While logging the network usage details, this automated task also logs the overall device network traffic and per-application network traffic.

3.6 ILR in Crowd-Sensed Data

This section presents ILR, a scheme which improves the location reliability of mobile crowd-sensed data with minimal human efforts. We also describe the validation process used by McSense to detect false location claims from malicious providers.

3.6.1 Assumptions

We assume that the sensed data are already collected by McSense from providers at different locations. However, these sensed data are awaiting validation before being sent to the clients who requested these data. We assume that every provider performs Bluetooth scans at each location where it is collecting sensing data. We also assume that the sensed data reported by providers for a given task always include location, time, and a Bluetooth scan. Note that Bluetooth scans can have a much lower frequency than the sensor sampling frequency.

3.6.2 Adversarial Model

We assume all the mobile devices are capable of determining their location using GPS. We also assume McSense is trusted and the communication between mobile users and McSense is secure. In our threat model, we consider that any provider may act maliciously and may lie about their location.

A malicious provider can program the device to spoof a GPS location (Humphreys et al., 2008) and start providing wrong location data for all the crowd sensing data requested by clients. Regarding this, we consider three threat scenarios, where (1) the provider does not submit the location and Bluetooth scan with a sensing data point, (2) the provider submits a Bluetooth scan associated with a sensing

task but claims a false location, and (3) the provider submits both a false location and a fake Bluetooth scan associated with a sensing data point. In Section 3.6.4, we will discuss how these scenarios are addressed by ILR.

We do not consider colluding attack scenarios, where a malicious provider colludes with other providers to show that she is present in the Bluetooth colocation data of others. In practice, it is not easy for a malicious provider to employ another colluding user at each sensing location. Additionally, these colluding attacks can be reduced by increasing the minimum node degree requirement, in the colocation data of each provider (i.e., a provider P must appear in the Bluetooth scans of at least a minimum number of other providers at her claimed location and time). Therefore, it becomes difficult for a malicious provider to create a false high node degree by colluding with real colocated people at a given location and time.

Finally, the other class of attacks that is out of scope for our current scheme is attacks in which a provider submits the right location and Bluetooth scan associated with this sensing task but is able to *fool* the sensors to create false readings (e.g., using the flame of a lighter to create the false impression of a high temperature).

3.6.3 ILR Design

The main idea of our scheme is to corroborate data collected from manual (photo) tasks with colocation data from Bluetooth scans. We describe next an example of how ILR uses the photos and colocation data.

3.6.3.1 Example of ILR in Action

Figure 3.3 maps the data collected by several different tasks in McSense. The figure shows 9 photo tasks (marked A–I) and 15 sensing tasks (marked 1–15) performed by different providers at different locations. For each of these tasks, providers also report neighbors discovered through Bluetooth scans. All these tasks are grouped into small circles using colocation data found in Bluetooth scans within a time interval t. For example, photo task A and sensing tasks 1, 2, and 3 are identified as colocated and grouped into one circle because they are discovered in each other's Bluetooth scans.

In this example, McSense does not need to validate all the photo tasks mapped in the figure. Instead, McSense first considers the photo tasks with the highest node degree (NodeDegree) by examining the colocated groups for photo task providers who have seen the highest number of other providers in Bluetooth scans around them. In this example, we consider NodeDegree ≥ 3. Hence, we see that photo tasks A, B, C, D, and G have discovered the highest number of providers around their location. Therefore, McSense chooses these five photo tasks for validation. These selected photo tasks are validated either manually or automatically (we discuss this in detail in Section 3.6.3.2). When validating these photo tasks, invalid photos are rejected, and McSense ignores the Bluetooth scans associated with them. If the photo is valid, then McSense considers the location of the validated photo as trusted because the validated photo is actually taken from the physical location requested in the task. However, it is not always possible to categorize every photo as a valid or a fake photo. Therefore, some photos are categorized as *unknown* when a decision cannot be made.

In this example, we assume that these five selected photos are successfully validated through manual verification. Next, using the transitive trust property, McSense extends the location trust of validated photos to other colocated providers' tasks, which are found in the Bluetooth scans of the A, B, C, D, and G photo tasks. For example, A extends the trust to the tasks 1, 2, and 3, while B extends the trust to tasks 4, 5, and 6. Then task 6 extends its trust to tasks 13 and 14. Finally, at the end of this process, McSense has 21 successfully validated tasks out of a total of 24 tasks. In this example, McSense required manual validation for just 5 photo tasks, but using the transitive trust property, it was able to extend the trust to 16 additional tasks automatically. Only 3 tasks (E, F, and 12) are not validated as they lack colocation data around them.

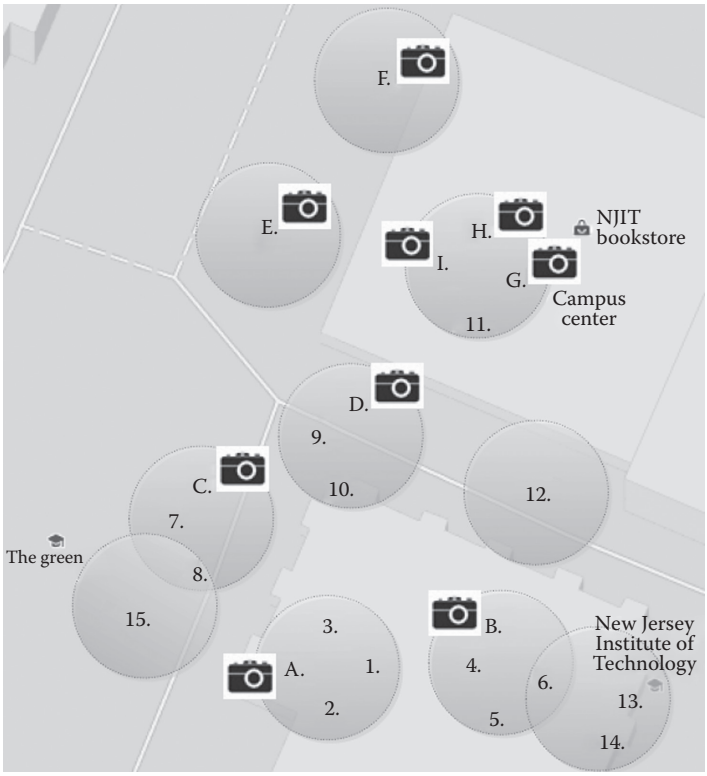


FIGURE 3.3 Example of McSense collected photo tasks [A–I] and sensing tasks [1–15] on the campus map, grouped using Bluetooth discovery colocation data.

3.6.3.2 ILR Phases

The ILR scheme has two phases as shown in Figure 3.4. Phase 1, photo selection, selects the photo tasks to be validated. And Phase 2, transitive trust, extends the trust to data points colocated with the tasks selected in Phase 1.

3.6.3.2.1 Phase 1—Photo Selection

Using collected data from the Bluetooth scans of providers, ILR constructs a connected graph of colocated data points for a given location and within a time interval t (these are the same groups represented in circles in Figure 3.3). From these graphs, we select the photo tasks that have node degrees greater than a threshold (NodeDegree).

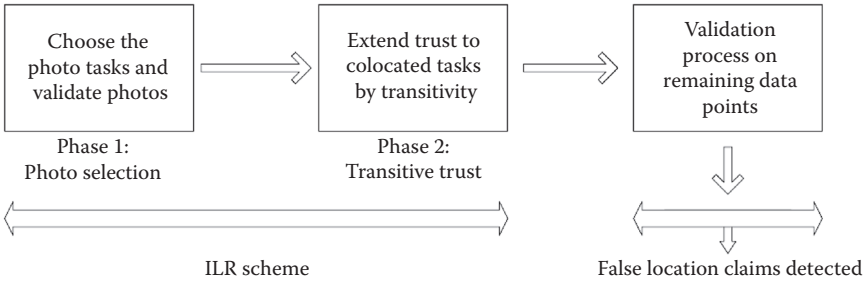


FIGURE 3.4 The phases of the ILR scheme.

These selected photo tasks are validated either by humans or by applying computer vision techniques. For manual validation, McSense could rely on other users recruited from Amazon MTurk (Amazon Mechanical Turk, 2013), for example. In order to apply computer vision techniques, first we need to collect ground truth photos to train image recognition algorithms. One alternative is to have trusted people collect the ground truth photos. However, if the ground truth photos are collected through crowd sensing, then they have to be manually validated as well. Thus, reducing the number of photos that require manual validation is an important goal for both manual and automatic photo recognition. Once the validation is performed, the location of the validated photo task is now considered to be reliable because the validated photos have been verified to be taken from the physical location requested in the task. For simplicity, we will refer to the participants who contributed valid photo tasks with reliable location and time as *validators*.

3.6.3.2.2 Phase 2—Transitive Trust

In this phase, we rely on the transitive trust property and extend the trust established in the validator's location to other colocated data points. In short, if the photo is valid, the trust is extended to colocated data points found in the Bluetooth scan of the validated photo task. In the current scheme, trust is extended until all colocated tasks are trusted or no other task is found; alternately, McSense can set a time to live (TTL) on extended trust. The following two steps are performed in this phase:

Step 1: Mark colocated data points as trusted: For each task colocated with a validated photo task, mark the task's location as trusted.
Step 2: Repeat Step 1 for each newly validated task until all colocated tasks are trusted or no other task is found.

3.6.3.3 Validation Process

After executing the two phases of ILR, all the colocated data points are validated successfully. If any malicious provider falsely claims one of the validated task's location at the same time, then the false claim will be detected in the validation step. Executing the validation process shown in Algorithm 3.1 will help us to detect the wrong location claims around the already validated location data points. For instance, if we consider task 12 from Figure 3.3 as a malicious provider claiming a false location exactly at photo task A's location and time, then task 12 will be detected in the validationProcess(), as it does not appear in the Bluetooth scans of photo task A. In addition to the validation process, McSense also performs a basic spatiotemporal correlation check to ensure that the provider is not claiming a location at different places at the same time.

Algorithm 3.1: ILR Validation Pseudo-Code

Notation:

TList: List of tasks that are not yet marked trusted after completing the first two phases of ILR.
T: Task submitted by a provider.
L: Location of the photo or sensing task (T).
t: Timestamp of the photo or sensing Task (T).
hasValidator(*L, t*): A function that checks if any validated data points already exist at task T's location and time.

validationProcess(): Run to validate the location of each task in *TList*

1. *for* each task T in TList *do*
2. *if* hasValidator(L, t) = = TRUE *then*
3. *Update* task T with false location claim at (L, t)

3.6.4 Security Analysis

The goal of the ILR scheme is to establish the reliability of the sensed data by validating the claimed location of the data points. In addition, ILR seeks to detect false claims made by malicious participants.

ILR is able to handle all the three threat scenarios presented in Section 3.6.2. In the first threat scenario, when there is no location and Bluetooth scan submitted along with the sensed data, the sensed data of that task are rejected, and the provider will not be paid by McSense.

In the second threat scenario, when a provider submits her Bluetooth discovery with a false location claim, ILR detects the provider in her neighbors' Bluetooth scans at a different location using the spatio-temporal correlation check and rejects the task's data.

Finally, when a provider submits a fake Bluetooth discovery with a false location claim, ILR looks for any validator around the claimed location, and if it finds anyone, then the sensed data associated with the false location claim are rejected. If there is no validator around the claimed location, then the data point is categorized as *unknown*.

As discussed in Section 3.6.2, sensed data submitted by malicious colluding attackers could be filtered to a certain extent in McSense by setting the node degree threshold (NodeDegree) to the minimum node degree requirement requested by the client.

3.6.5 Related Work

Trusted hardware represented by the TPM (Dua et al., 2009; Gilbert et al., 2011; Trusted Platform Module, 2013; Xu et al., 2011) has been leveraged to design new architectures for trustworthy software execution on mobile phones (McCune et al., 2008; Nauman et al., 2010; Schneider et al., 2011). Recent work has also proposed architectures to ensure that the data sensed on mobile phones are trustworthy (Gilbert et al., 2010; Saroiu and Wolman, 2010). When untrusted client applications perform transformations on the sensed data, YouProve (Gilbert et al., 2011) is a system that combines a mobile device's trusted hardware with software in order to ensure the trustworthiness of these transformations and that the meaning of the source data is preserved. YouProve describes three alternatives to combine the trusted hardware with software: The first two require to extend the trusted codebase to include either the code for the transformations or the entire application, whereas the third one requires building trust in the code that verifies that transformations preserve the meaning of the source data.

Relying completely on TPM is insufficient to deal with attacks in which a provider is able to *fool* the sensors (e.g., using the flame of a lighter to create the false impression of a high temperature). Recently, there have also been reports of successful spoofing of civilian GPS signals (Humphreys et al., 2008).

Orthogonal to the work in ILR, task pricing also helps in improving the data quality. A recent paper (Lee and Hoh, 2010) presents pricing incentive mechanisms to achieve quality data in participatory sensing application. In this work, the participants are encouraged to participate in the sensing system through a reverse auction based on a dynamic pricing incentive mechanism in which users can sell their sensing data with their claimed bid price.

The LINK protocol (Talasila et al., 2010, 2014) was recently proposed for secure location verification without relying on location infrastructure support. LINK can provide stronger guarantees than ILR but has a number of drawbacks if used for mobile sensing. LINK requires a provider to establish Bluetooth connections with her colocated users at each sensing location, which increases latency and consumes more phone battery. In addition, LINK is executed in real time to verify the users' locations, whereas ILR is executed on the collected data from mobile crowd sensing. Therefore, employing ILR helps providers in submitting sensed data quickly and also consumes less phone battery.

3.7 Experimental Evaluation: Field Study

The providers (students shown in Table 3.1) registered with McSense and submitted data together with their user ID. Both phases of ILR and the validation process are executed on data collected from the providers. In these experiments, we acted as the clients collecting the sensed data.

TABLE 3.1 Demographic Information of the Students

Total Participants	58
Males	90%
Females	10%
Ages 16–20	52%
Ages 21–25	41%
Ages 26–35	7%

3.7.1 Evaluating the ILR Scheme

The location data are mostly collected from the university campus (0.5 miles radius). The main goal of these experiments is to determine how efficiently can the ILR scheme help McSense validate the location data and detect false location claims. ILR considers the Bluetooth scans found within a 5 min interval of measuring the sensor readings for a sensing task.

Table 3.2 shows the total photo tasks that are submitted by students; only 204 photo tasks have Bluetooth scans associated with them. In this data set, we considered NodeDegree ≥ 1; therefore, we used all these 204 photo tasks with Bluetooth scans in Phase 1 to perform manual validation. In Phase 2, we are then able to automatically extend the trust to 148 new location data points through the transitive closure property of ILR.

To capture the ground truth, we manually validated all the photos collected by McSense in this study and identified that we have a total of 45 fake photos submitted to McSense from malicious providers, out of which only 16 fake photo tasks have Bluetooth scans with false location claims. We then applied ILR to verify how many of these 16 fake photos can be detected.

We were able to catch four users who claimed wrong locations to make money with fake photos, as shown in Table 3.3. Since the total number of malicious users involved in the 16 fake photo tasks is 10, ILR was able to detect 40% of them. Finally, ILR is able to achieve this result by validating only 11% of the photos (i.e., 204 out of 1784).

3.7.2 Influence of the Task Price on Data Quality

In the field study performed at New Jersey Institute of Technology (NJIT), a few tasks are posted with a high price (ranging between $2 and $10) in order to observe the impact on the sensing tasks. We have noticed a 15% increase in the task completion success rate for the high-priced sensing tasks compared

TABLE 3.2 Photo Task Reliability

	Number of Photo Tasks
Total photos	1784
Number of photos with Bluetooth scans (manually validated in ILR)	204
Trusted data points added by ILR	148

TABLE 3.3 Number of False Location Claims and Cheating Participants Detected by ILR

	Detected by ILR Scheme	Total	Percentage Detected
Tasks with false location claim	4	16	25
Cheating participants	4	10	40

with the low-priced sensing tasks. In addition, we have noticed an improvement in data quality for the high-priced photo tasks, with clear and focused photos compared with the low-priced photo tasks (the tasks priced with $1 or lower are considered low-priced tasks). Thus, our study confirms that task pricing influences the data quality. This result confirms that various task pricing strategies (Lee and Hoh, 2010) can be employed by McSense in parallel to the ILR scheme to ensure data quality for the sensing tasks.

3.8 Simulations

This section presents the evaluation of ILR using the ns-2 network simulator (ns-2 Simulator, 2014). The two main goals of the evaluation are (1) to estimate the right percentage of photo tasks needed in Phase 1 to bootstrap the ILR scheme and (2) to quantify the ability of ILR to detect false location claims at various node densities.

3.8.1 Simulation Setup

The simulation setup parameters are presented in Table 3.4. Given a simulation area of 100 m × 120 m, the node degree (i.e., average number of neighbors per user) is slightly higher than 5. We varied the simulation area to achieve node degrees of 2, 3, and 4. We consider low walking speeds (i.e., 1 m/s) for collecting photos. In these simulations, we considered all tasks as photo tasks. A photo task is executed every minute by each node. Photo tasks are distributed evenly across all nodes. Photo tasks with false location claims are also distributed evenly across several malicious nodes. We assume the photo tasks in ILR's Phase 1 are manually validated.

After executing the simulation scenarios described in the following, we collect each photo task's time, location, and Bluetooth scan. As per the simulation settings, we will have 120 completed photo tasks per node at the end of the simulation (i.e., 24,000 total photo tasks for 200 nodes). Over these collected data, we apply the ILR validation scheme to detect false location claims.

3.8.2 Simulation Results

In this set of experiments, we vary the percentage of photo tasks with false location claims. The resulting graph, plotted in Figure 3.5, has multiple curves as a function of the percentage of photo tasks submitting false location. This graph is plotted to gain insights on what will be the right percentage of photo tasks needed in Phase 1 to bootstrap the ILR scheme. Next, we analyze Figure 3.5.

3.8.2.1 Varying Percentage of False Location Claims

3.8.2.1.1 Low Count of Malicious Tasks Submitted

When 10% of total photo tasks are submitting false location, Figure 3.5 shows that the ILR scheme can detect 55% of the false location claims just by using 10% of the total photo tasks validated in Phase 1. This figure also shows that in order to detect more false claims, more photos need to be manually

TABLE 3.4 Simulation Setup for ILR

Parameter	Value
Number of nodes	200
% of tasks with false location claims	10, 15, 30, 45, 60
Bluetooth transmission range	10 m
Simulation time	2 h
User walking speed	1 m/s
Node density	2, 3, 4, 5
Bluetooth scan rate	1/min

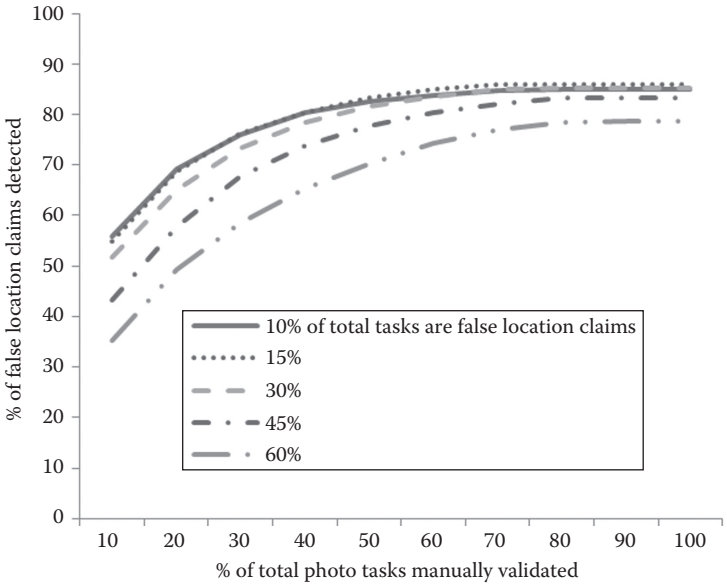


FIGURE 3.5 ILR performance as a function of the percentage of photos manually validated in Phase 1. Each curve represents a different percentage of photos with fake locations.

validated: for example, ILR uses up to 40% of the total photo tasks in Phase 1 to detect 80% of the false location tasks. Finally, Figure 3.5 shows that increasing the percentage of validated photo tasks above 40% does not help much as the percentage of detected false tasks remains the same.

3.8.2.1.2 High Count of Malicious Tasks Submitted

When 60% of the total photo tasks are submitting false locations, Figure 3.5 shows that ILR can still detect 35% of the false claims by using 10% of the total photo tasks in Phase 1. But in this case, ILR requires more validated photo tasks (70%) to catch 75% of the false claims. This is because by increasing the number of malicious tasks, the colocation data are reduced, and therefore, ILR cannot extend trust to more location claims in its Phase 2. Therefore, we conclude that the right percentage of photo tasks needed to bootstrap ILR is proportional to the expected false location claims (which can be predicted using the history of the users' participation).

3.8.2.2 Impact of Node Density on ILR

In this set of experiments, we assume that 10% of the total photo tasks are submitting false locations. In Figure 3.6, we analyze the impact of node density on the ILR scheme. We seek to estimate the minimum node density required to achieve highly connected graphs to extend the location trust transitively to more colocated nodes.

3.8.2.2.1 High Density

When simulations are run with node density of 5, Figure 3.6 shows that ILR can detect the highest percentage (85%) of the false location claims. The figure also shows similarly high results even for a node density of 4.

3.8.2.2.2 Low Density

When simulations are run with node density of 2, we can see that ILR can still detect 65% of the false location tasks using 50% of the total photo tasks in Phase 1. For this node density, even after increasing the number of validated photo tasks in Phase 1, the percentage of detected false claims

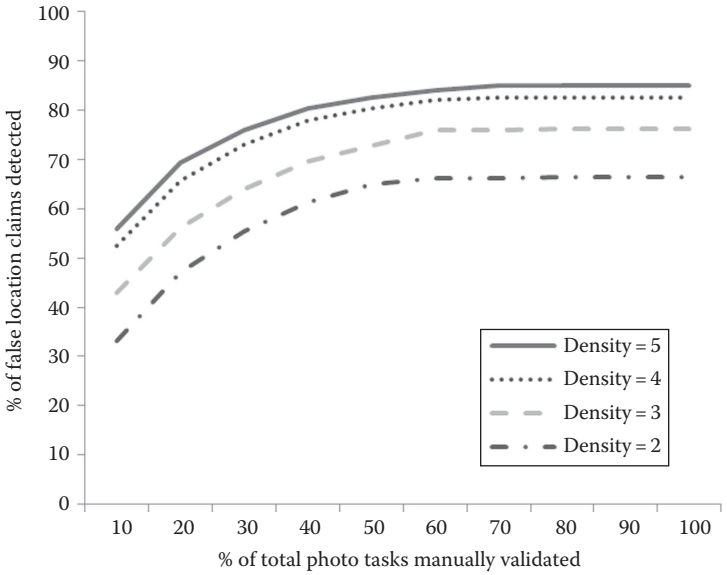


FIGURE 3.6 ILR performance as a function of the percentage of photos manually validated in Phase 1. Each curve represents a different network density represented as average number of neighbors per node.

does not increase. This is because there are fewer colocated users at low node densities. Therefore, we conclude that ILR can efficiently detect false claims with a low number of manual validations, even for low node densities.

3.9 Field Study Insights and Improving the ILR Scheme

In this section, we present our insights from the analysis of the data collected from the field study and discuss possible improvements of the ILR scheme based on these insights. In addition, we present observations from the survey that was administered to participants at the end of the field study to understand their opinion on location privacy and usage of phone resources.

3.9.1 Correlation of User Earnings and Fake Photos

To understand the correlation between the user earnings and the number of fake photos submitted, we plot the data collected from the McSense crowd sensing field study. The experimental results in Figure 3.7 show that the users who submitted most of the fake photos are among the top 20 high earners (with an exception of 4 low-earning users who submitted fake photos once or twice). This is an interesting observation that can be leveraged to improve the ILR scheme.

In the current ILR scheme, there are cases where the validation process cannot make a firm decision on some data points. Those data points fall under the *unknown* category as described in Section 3.6.3. If these *unknown* data points are too many (in millions), then it is challenging to validate all of them manually. Therefore, to improve the ILR scheme, we propose that these *unknown* cases of data points must go through an extra check to find whether the user is a high earner in the sensing system. If the user is a high earner, then there is a high probability that the user submitted data point is fake. Those photos should be manually validated. If the user is a mid or low-range earner, then there is a low probability of his or her data point being faked and the data point should be considered as valid. This method will help in reducing the number of photos that require manual validation.

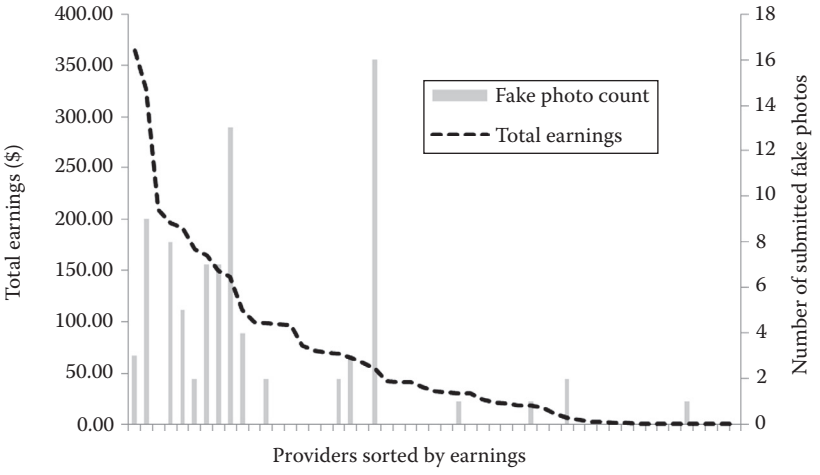


FIGURE 3.7 Correlation of earnings and fake photos.

3.9.2 Correlation of Location and Fake Photos

We ask the question "Is there any correlation between the amount of time spent by users on campus and the number of submitted fake photos?" As suspected, the users who spent less time on campus have submitted more fake photos. This behavior can be observed in Figure 3.8.

Figure 3.8 shows the number of fake photos submitted by each user, with the users sorted by the total hours spent on the NJIT campus. The participants' total hours recorded at the NJIT campus are the hours that are accumulated from the sensed data collected from *automated sensing task* described in Section 3.5.3. The NJIT location is considered to be a circle with a radius of 0.5 miles. If the user is in circle, then she is considered to be at NJIT. For most of the submitted fake photos with the false location claim, the users claimed that they are at a campus location where the photo task is requested, but they were in fact infrequent campus visitors.

This is an interesting observation, which can also be leveraged to improve the ILR scheme's validation process when there is a large number of data points classified as *unknown*. The intuition behind this argument is that users tend to fake the data mostly when they are not around the task's location. Therefore, to improve the ILR scheme, we propose to use a user's recorded location trail in the McSense

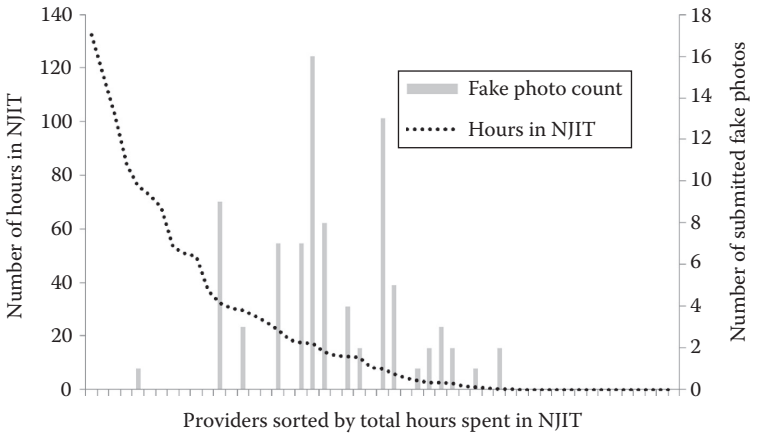


FIGURE 3.8 Correlation of user location and fake photos.

system in order to identify whether the user is or is not a frequent visitor of the task's location. If the user is not a frequent visitor of the claimed location, then there is a high probability that location claim is false, and the *unknown* data point should be manually checked. On the other hand, if the user is a frequent visitor of the claimed location, then her claim can be trusted. By reducing the number of photos that require manual validation, McSense can improve ILR's validation process for *unknown* data points.

3.9.3 Malicious User: Menace or Nuisance?

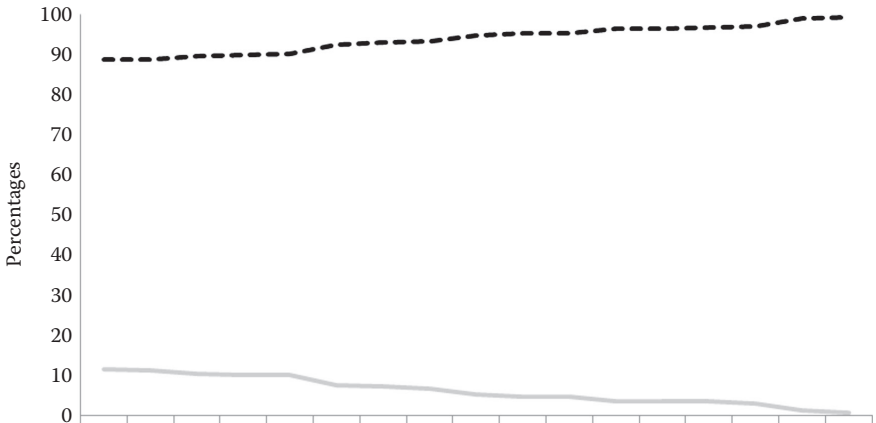
The photos submitted by malicious users are plotted in Figure 3.9. The data show that malicious users have submitted good photos at a very high rate compared to the fake photos. These malicious users are among the high earners, so they are submitting more data than the average user. Thus, it may not be a good idea to remove the malicious users from the system as soon as they are caught cheating.

Instead, it may be a better idea to identify the validity of the individual data points (which is exactly the same process done in the current ILR scheme discussed in Section 3.6.3). We conclude that malicious users are not a significant menace but may cause some confusion in the collected data. However, this can be filtered out by McSense through correlating the data with location and earnings as discussed earlier in the section.

3.9.4 Influence of Maintaining a Reputation Score

When a fake location claim is detected by ILR, McSense would benefit if the malicious user who submitted the fake claim receives a lower compensation upon completion of a task. In order to perform such a process, McSense should use a reputation module such as in Chu et al. (2010), which maintains a trust score for each user. This is similar with many other systems that rely on the participation of users. The trust score varies between 0 and 1. Initially, the trust score is given a default value for every user, and it evolves depending on the user participation. The trust score is reduced when the user is caught providing fake data and is increased when the user submits good data.

We propose that the McSense system maintains a trust score for every user, and then it uses this score for calculating the user payment upon task completion. For example, for a completed task that



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
% of good photos	88.5	88.6	89.5	89.7	90	92.5	92.9	93.2	94.7	95.2	95.3	96.4	96.5	96.6	97	98.8	99.2
% of fake photos	11.5	11.4	10.5	10.3	10	7.51	7.14	6.82	5.26	4.76	4.71	3.56	3.52	3.45	2.99	1.23	0.78

FIGURE 3.9 Photo counts of 17 cheating people.

is worth \$5, a user with a trust score of 0.9 will be paid only \$4.50. We envision that by maintaining a reputation score, the users providing fake data will eventually stop making false claims. We have seen earlier in the section that the malicious users also submit a significant amount of good data. But if their trust score drops to 0, then the malicious users will not participate anymore as they do not earn the task amount and will eventually leave the system. As we argued earlier in this section, it is not a good idea to entirely remove the malicious users from the system. Therefore, to avoid eliminating malicious users from the system, the trust score will not decrease anymore after reaching a minimum threshold (e.g., 0.2). Hence, the malicious user will only get 20% of the task dollars until she improves her trust score. Therefore, the McSense system does not need to worry about discarding good data that are submitted by malicious users.

3.9.5 User's Survey Results and Observations

At the end of the field study, we requested each user to fill a survey in order to understand the participants' opinion on location privacy and usage of phone resources. The survey contains 16 questions with answers on a five-point Likert scale (1 = *strongly disagree*, 2 = *disagree*, 3 = *neutral*, 4 = *agree,* 5 = *strongly agree)*. Out of 58 participants, 27 filled in the survey. Based on the survey answers, we provide next a few interesting observations that are directly or indirectly relevant in the context of data reliability:

1. One of the survey questions was "I tried to fool the system by providing photos from other locations than those specified in the tasks (the answer does not influence the payment)." By analyzing the responses for this specific question, we observe that only 23.5% of the malicious users admitted that they submitted the fake photos (4 admitted out of 17 malicious). This shows that the problem stated in this chapter on data reliability is real and it is important to validate the sensed data.
2. One survey question related to user privacy was "I was concerned about my privacy while participating in the user study." The survey results show that 78% of the users are not concerned about their privacy. This shows that many participants are willing to trade off their location privacy for paid tasks. The survey results are correlated with the collected McSense data points. We posted a few sensing tasks during weekends, which is considered to be private time for the participants who are mostly not in the campus at that time. We observe that 33% of the participants participated in the sensing and photo tasks, even when spending their personal time in the weekends. We conclude that the task price plays a crucial role (trading the user privacy) to collect quality sensing data from any location and time.
3. Another two survey questions are related to the usage of phone resources (e.g., battery) by sensing tasks: (1) "Executing these tasks did not consume too much battery power (I did not need to re-charge the phone more often than once a day)"; (2) "I stopped the automatic tasks (resulting in incomplete tasks) when my battery was low." The responses to these questions are interesting. Most of the participants reported that they were carrying chargers to charge their phone battery as required while running the sensing tasks and were keeping their phone always ready to accept more sensing tasks. This proves that phone resources, such as battery, are not a big concern for continuously collecting sensing data from different users and locations. We describe next the battery consumption measurements in detail.

3.9.6 Battery Consumption

We try to determine the amount of energy consumed by the user's phone battery for collecting sensing data that are required for ILR. Basically, ILR is executed on the server side over the collected data. But the collected data such as Bluetooth scans at each location are crucial for ILR. Next, we provide

measurements for the extra battery usage caused by keeping Bluetooth/Wi-Fi radios ON. We measured the readings using *Motorola Droid 2* smartphones running Android OS 2.2:

- With Bluetooth and Wi-Fi radios ON, the battery life of the *Droid 2* phone is longer than 2 days (2 days and 11 h).
- With Bluetooth OFF and Wi-Fi radio ON, the battery life of the *Droid 2* phone is longer than 3 days (3 days and 15 h).
- For every Bluetooth discovery, the energy consumed is 5.428 J. The total capacity of the *Droid 2* phone battery is 18.5 kJ. Hence, over 3000 Bluetooth discoveries can be collected from different locations using a fully charged phone.

3.10 Summary

This chapter presented the concept of mobile crowd sensing and its applications to everyday life. We described the design and implementation of McSense, our mobile crowd sensing platform, which was used to run a user study with over 50 users at the NJIT campus for a period of 2 months. We also discussed the data reliability issues in mobile crowd sensing by presenting several scenarios involving malicious behavior. We presented a protocol for location reliability as a step toward achieving data reliability in sensed data, namely, ILR. ILR also detects false location claims associated with the sensed data. Based on our security analysis and simulation results, we argue that ILR works well at various node densities. The analysis of the sensed data collected from the users in our field study demonstrates that ILR can efficiently achieve location data reliability and detect a significant percentage of false location claims. Therefore, we conclude this chapter with our belief that mobile crowd sensing will become a widespread method for collecting sensing data from the physical world once the data reliability issues are properly addressed.

Acknowledgment

Parts of this chapter were published in the *IJBDCN Journal* (Talasila et al., 2013. Reused with permission from the publisher).

References

Abdelzaher, T., Anokwa, Y., Boda, P., Burke, J.A., Estrin, D., Guibas, L., and Reich, J. (2007). Mobiscopes for human spaces. *Pervasive Computing, IEEE*, 6(2), 20–29.

Amazon Mechanical Turk. (2013). Retrieved from http://www.mturk.com. Accessed September 9, 2013.

Azizyan, M., Constandache, I., and Roy Choudhury, R. (2009). SurroundSense: Mobile phone localization via ambience fingerprinting. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking (MobiCom'09)*, Beijing, China, ACM, pp. 261–272.

Campbell, A.T., Eisenman, S.B., Lane, N.D., Miluzzo, E., and Peterson, R.A. (2006). People-centric urban sensing. In *Proceedings of the Second Annual International Workshop on Wireless Internet (WICON'06)*, Boston, USA, ACM, pp. 18.

Capkun, S., Cagalj, M., and Srivastava, M. (2006). Secure localization with hidden and mobile base stations. In *Proceedings of IEEE INFOCOM, (INFOCOM'06)*, Barcelona, Spain.

Capkun, S. and Hubaux, J. (2005). Secure positioning of wireless devices with application to sensor networks. In *24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'05)*, Miami, FL, pp. 1917–1928.

Cardone, G., Foschini, L., Bellavista, P., Corradi, A., Borcea, C., Talasila, M., and Curtmola, R. (2013). Fostering participaction in smart cities: A geo-social crowdsensing platform. *IEEE Communications Magazine*, 51(6), 112–119.

ChaCha. (2013). Your mobile bff. Retrieved from http://www.chacha.com. Accessed September 9, 2013.

Choudhury, T., Consolvo, S., Harrison, B., Hightower, J., LaMarca, A., LeGrand, L., and Haehnel, D. (2008). The mobile sensing platform: An embedded activity recognition system. *Pervasive Computing, IEEE*, 7(2), 32–41.

Chu, X., Chen, X., Zhao, K., and Liu, J. (2010). Reputation and trust management in heterogeneous peer-to-peer networks. *Springer Telecommunication Systems*, 44, 191–203.

Demirbas, M., Bayir, M.A., Akcora, C.G., Yilmaz, Y.S., and Ferhatosmanoglu, H. (2010). Crowd-sourced sensing and collaboration using twitter. In *2010 IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM)* Montreal, QC, Canada, IEEE, pp. 1–9.

Dua, A., Bulusu, N., Feng, W., and Hu, W. (2009). Towards trustworthy participatory sensing. *Proceedings of the Usenix Workshop on Hot Topics in Security (HotSec'09)*, Montreal, Canada.

Eriksson, J., Girod, L., Hull, B., Newton, R., Madden, S., and Balakrishnan, H. (2008). The pothole patrol: Using a mobile sensor network for road surface monitoring. In *Proceedings of the Sixth International Conference on Mobile Systems, Applications, and Services (MobiSys'08)*, Breckenridge, Colorado, USA, ACM, pp. 29–39.

Garmin. (2014). Edge 305. Retrieved from www.garmin.com/products/edge305/. Accessed May 28, 2014.

Gilbert, P., Cox, L., Jung, J., and Wetherall, D. (2010). Toward trustworthy mobile sensing. In *Proceedings of the 11th Workshop on Mobile Computing Systems & Applications (HotMobile'10)* Annapolis, Maryland, USA, ACM, pp. 31–36.

Gilbert, P., Jung, J., Lee, K., Qin, H., Sharkey, D., Sheth, A., and Cox, L. (2011). Youprove: Authenticity and fidelity in mobile sensing. In *Proceedings of the Ninth ACM Conference on Embedded Networked Sensor Systems (SenSys'11)*, Seattle, WA, USA, pp. 176–189.

Google Mobile Ads. (2014). Retrieved from http://www.google.com/ads/mobile/. Accessed May 28, 2014.

Google Play. (2014). Android app store. Retrieved from https://play.google.com/. Accessed May 28, 2014.

Gupta, A., Thies, W., Cutrell, E., and Balakrishnan, R. (2012). mClerk: Enabling mobile crowdsourcing in developing regions. In *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems (CHI'12)*, Austin, Texas, USA, ACM, pp. 1843–1852.

Herrera, J., Work, D., Herring, R., Ban, X., Jacobson, Q., and Bayen, A. (2010). Evaluation of traffic data obtained via GPS-enabled mobile phones: The mobile century field experiment. *Transportation Research Part C: Emerging Technologies*, 18(4), 568–583.

Honicky, R., Brewer, E.A., Paulos, E., and White, R. (2008). N-smarts: Networked suite of mobile atmospheric real-time sensors. In *Proceedings of the Second ACM SIGCOMM Workshop on Networked Systems for Developing Regions*, *(NSDR'08)*, SEATTLE, WA, USA, ACM, pp. 25–30.

Humphreys, T., Ledvina, B., Psiaki, M., O'Hanlon, B., and Kintner Jr., P. (2008). Assessing the spoofing threat: Development of a portable GPS civilian spoofer. In *Proceedings of the ION GNSS International Technical Meeting of the Satellite Division*, *(ION GNSS'08)*, Savannah, Georgia.

IBM Smarter Planet. (2014). Retrieved from http://www.ibm.com/smarterplanet/us/en/overview/ideas/. Accessed May 28, 2014.

Intel Labs. (2010). The mobile phone that breathes. Retrieved from http://scitech.blogs.cnn.com/2010/04/22/the-mobilephone-that-breathes/.

Kansal, A., Goraczko, M., and Zhao, F. (2007). Building a sensor network of mobile phones. In *Proceedings of the Sixth International Conference on Information Processing in Sensor Networks*, *(IPSN'07)*, Cambridge, MA, USA, ACM, pp. 547–548.

Lee, J. and Hoh, B. (2010). Dynamic pricing incentive for participatory sensing. *Pervasive and Mobile Computing, Elsevier*, 6(6), 693–708.

London's water supply monitoring. (2012). Retrieved from http://www.ucl.ac.uk/news/news-articles/May2012/240512-.

Lu, H., Pan, W., Lane, N.D., Choudhury, T., and Campbell, A.T. (2009). SoundSense: Scalable sound sensing for people-centric applications on mobile phones. In *Proceedings of the Seventh International Conference on Mobile Systems, Applications, and Services*, *(MobiSys'09)*, Kraków, Poland, ACM, pp. 165–178.

Mardenfeld, S., Boston, D., Pan, S.J., Jones, Q., Iamntichi, A., and Borcea, C. (2010). GDC: Group discovery using co-location traces. In *Proceedings of Second International Conference on Social Computing (SocialCom'10)*, Minneapolis, MN, USA, IEEE, pp. 641–648.

Mathur, S., Jin, T., Kasturirangan, N., Chandrasekaran, J., Xue, W., Gruteser, M., and Trappe, W. (2010). Parknet: Drive-by sensing of road-side parking statistics. In *Proceedings of the Eighth International Conference on Mobile Systems, Applications, and Services (MobiSys'10)*, San Francisco, CA, USA, ACM, pp. 123–136.

McCune, J., Parno, B., Perrig, A., Reiter, M., and Isozaki, H. (2008). Flicker: An execution infrastructure for TCB minimization. *SIGOPS Operating Systems Review*, 42(4), 315–328.

McSense. (2013). Android smart phone application. Retrieved from https://play.google.com/store/apps/details?id=com.mcsense.app. Accessed September 9, 2013.

McSense Project. (2013). Retrieved from http://web.njit.edu/~mt57/mcsense/.

Miluzzo, E., Lane, N.D., Fodor, K., Peterson, R., Lu, H., Musolesi, M., and Campbell, A.T. (2008). Sensing meets mobile social networks: The design, implementation and evaluation of the CenceMe application. In *Proceedings of the Sixth ACM Conference on Embedded Network Sensor Systems (SenSys'08)*, Raleigh, NC, USA, ACM, pp. 337–350.

MIT News. (2014). Retrieved from http://web.mit.edu/newsoffice/2009/blood-pressure-tt0408. html. Accessed May 28, 2014.

Mobads. (2014). Retrieved from http://www.mobads.com/. Accessed May 28, 2014.

Mobile Millennium Project. (2014). Retrieved from http://traffic.berkeley.edu/. Accessed May 28, 2014.

Mohan, P., Padmanabhan, V.N., and Ramjee, R. (2008). Nericell: Rich monitoring of road and traffic conditions using mobile smart phones. In *Proceedings of the Sixth ACM Conference on Embedded Network Sensor Systems (SenSys'08)*, Raleigh, NC, USA, ACM, pp. 323–336.

Mun, M., Reddy, S., Shilton, K., Yau, N., Burke, J., Estrin, D., Hansen, M., Howard, E., West, R., and Boda, P. (2009). PEIR, the Personal Environmental Impact Report, as a platform for participatory sensing systems research. In *Proceedings of the Seventh International Conference on Mobile Systems, Applications, and Services (MobiSys'09)*, Kraków, Poland, ACM, pp. 55–68.

Narula, P., Gutheim, P., Rolnitzky, D., Kulkarni, A., and Hartmann, B. (2011). MobileWorks: A mobile crowdsourcing platform for workers at the bottom of the pyramid. In *Human Computation* (*2011*) *AAAI Workshop*, *(HCOMP'11)*, San Francisco, CA, USA.

Nauman, M., Khan, S., Zhang, X., and Seifert, J. (2010). Beyond kernel-level integrity measurement: Enabling remote attestation for the android platform. *Trust and Trustworthy Computing*, 6101. 1–15.

ns-2 Simulator. (2014). Retrieved from http://nsnam.isi.edu/nsnam/index.php/Main_Page. Accessed May 28, 2014.

Photo Journalism. (2013). Retrieved from http://www.flickr.com/groups/photojournalism. Accessed September 9, 2013.

Ra, M.-R., Liu, B., La Porta, T., and Govindan, R. (2012). Medusa: A programming framework for crowd-sensing applications. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services* (*MobiSys'12*), Low Wood Bay, Lake District, United Kingdom ACM, pp. 337–350.

Reality Mining Project. (2014). Retrieved from http://reality.media.mit.edu/. Accessed May 28, 2014.

Riva, O. and Borcea, C. (2007). The Urbanet revolution: Sensor power to the people! *IEEE Pervasive Computing*, 6(2), 41–49.

Saroiu, S. and Wolman, A. (2010). I am a sensor, and I approve this message. In *Proceedings of the 11th Workshop on Mobile Computing Systems & Applications (HotMobile'10)*, Annapolis, Maryland, USA, ACM, pp. 37–42.

Sastry, N., Shankar, U., and Wagner, D. (2003). Secure verification of location claims. In *Proceedings of the Second ACM Workshop on Wireless Security (WiSe'03)*, San Diego, California, USA, ACM, pp. 1–10.

Schneider, F.B., Walsh, K., and Sirer, E.G. (2011). Nexus authorization logic (NAL): Design rationale and applications. *ACM Transactions on Information and System Security*, 14(1), 8:1–8:28.

Sensor Lab at Dartmouth. (2013). Smart phone sensing research. Retrieved from http://sensorlab.cs.dartmouth.edu/research.html. Accessed September 9, 2013.

Sensordrone. (2013). The 6th sense of your smart phone. Retrieved from http://www.sensorcon.com/sensordrone. Accessed September 9, 2013.

Siewiorek, D., Krause, A., Moraveji, N., Smailagic, A., Furukawa, J., Reiger, K., and Shaffer, J. (2003). Sensay: A context-aware mobile phone. In *2012 16th International Symposium on Wearable Computers (ISWC'03)*, Sanibel Island, Florida, USA, IEEE Computer Society, pp. 248–248.

Songdo Smart City. (2014). Retrieved from http://www.songdo.com. Accessed May 28, 2014.

Statista. (2014). Global smart phone shipments forecast 2010–2018. Retrieved from http://www.statista.com/statistics/263441/global-smart phone-shipments-forecast/. Accessed May 28, 2014.

Talasila, M., Curtmola, R., and Borcea, C. (2010). Link: Location verification through immediate neighbors knowledge. In *Proceedings of the Seventh International ICST Conference on Mobile and Ubiquitous Systems, (MobiQuitous'10)*, Sydeny, Australia, Springer, pp. 210–223.

Talasila, M., Curtmola, R., and Borcea, C. (2013). ILR: Improving location reliability in mobile crowd sensing. *International Journal of Business Data Communications and Networking (IJBDCN)*, 9(4), 65–85.

Talasila, M., Curtmola, R., and Borcea, C. (2014). Collaborative bluetooth-based location authentication on smart phones. *Pervasive and Mobile Computing*. doi: http://dx.doi.org/10.1016/j.pmcj.2014.02.004.

Toyama, K., Logan, R., and Roseway, A. (2003). Geographic location tags on digital images. In *Proceedings of the 11th ACM International Conference on Multimedia*, *(MULTIMEDIA'03)*, Berkeley, CA, USA, pp. 156–166.

Trusted Platform Module. (2013). Retrieved from http://www.trustedcomputinggroup.org/developers/trusted platform module. Accessed May 28, 2014.

Twitter. (2014). Retrieved from http://twitter.com/. Accessed May 28, 2014.

Urban Sensing at UCLA. (2013). University of California Los Angeles urban sensing research. Retrieved from http://research.cens.ucla.edu/urbansensing/. Accessed September 9, 2013.

White, J., Thompson, C., Turner, H., Dougherty, B., and Schmidt, D. (2011). Wreckwatch: automatic traffic accident detection and notification with smart phones. *Mobile Networks and Applications*, 16(3), 285–303.

Xu, G., Borcea, C., and Iftode, L. (2011). A policy enforcing mechanism for trusted ad hoc networks. *IEEE Transactions on Dependable and Secure Computing*, 8(3), 321–336.

Yan, T., Kumar, V., and Ganesan, D. (2010). Crowdsearch: Exploiting crowds for accurate real-time image search on mobile phones. In *Proceedings of the Eighth International Conference on Mobile Systems, Applications, and Services (SenSys'09)*, Berkeley, California, USA, ACM, pp. 77–90.

Yan, T., Marzilli, M., Holmes, R., Ganesan, D., and Corner, M. (2009). mCrowd: A platform for mobile crowdsourcing. In *Proceedings of the Seventh ACM Conference on Embedded Networked Sensor Systems (SenSys'09)*, Berkeley, California, USA, ACM, pp. 347–348.

II

Wireless Sensor Information Networks

4 In-Network Processing in Wireless Sensor Networks *Qiao Xiang and Hongwei Zhang*..... **4**-1
Introduction • Application-Dependent INP • Application-Independent INP • Research Challenges • Summary • References

5 Wireless Sensor Hierarchical Networks *Shancang Li and Xinheng (Henry) Wang*.....**5**-1
Introduction • Motivation of the Hierarchal Architecture • Summary • References

6 Efficient Distributed Bayesian Estimation in Wireless Sensor Networks *Andrew P. Brown, Ronald A. Iltis, and Hua Lee* **6**-1
Introduction • Local State Estimation • State Estimate Fusion • Optimal Distributed State Estimation • Approximate Distributed State Estimation • Nonlinear State Estimation • Distributed Position Estimation • Simulation Results and Discussion • Summary • References

4

In-Network Processing in Wireless Sensor Networks

4.1 Introduction **4**-1
4.2 Application-Dependent INP **4**-2
Data Aggregation • Data Fusion
4.3 Application-Independent INP **4**-8
Packet Packing • Network Coding
4.4 Research Challenges **4**-13
Systematic Modeling and Complexity Analysis • Joint Optimization of QoS and WSN-Specific INP • Cooperation of Different INP Methods in WSN • Theoretical Foundations of Algorithm Design
4.5 Summary **4**-15
References **4**-15

Qiao Xiang
McGill University

Hongwei Zhang
Wayne State University

4.1 Introduction

During the past decade, wireless sensor networks (WSNs) have been widely used in many applications, including environment monitoring, homeland security, industrial control, and disaster alarm. The community has been doing extensive research on different areas of this field, including embedded system design, communication protocol design, and in-network processing (INP) method. Different from traditional computer networks, WSNs are highly resource and environment constrained. Each sensor has a limited power capacity, a small memory, and a short transmission range. Due to all these constraints and other application-specific constraints, how to make sensors smartly use their resource, especially their energy, becomes a key challenge for research in WSN.

To solve this challenge, different INP methods have been proposed to improve energy efficiency and data delivery performance by reducing network traffic load and thus channel contention. Over the past years, many INP protocols have been proposed for query processing [8,44,45,63] and data collection [18,19,33,41,65].

In this chapter, we conduct a comprehensive survey on INP in WSN. We categorize INP protocols into two categories: application-dependent INP, that is, data aggregation and data fusion, and application-independent INP, that is, packet packing and network coding. We review representative protocols of each INP in WSN. Before we end this chapter, we also point out some open research challenges.

4.2 Application-Dependent INP

When designing INP protocols for WSN, researchers and engineers usually need to design customized INP functions and modules that fit requirements of targeting applications. These INPs are called *application-dependent INP*. In the following sections, we introduce the two most classic application-dependent INPs, data aggregation and data fusion.

4.2.1 Data Aggregation

Data aggregation is the most common INP method in multihop WSNs. Figure 4.1 shows an example of data aggregation. Instead of sending value v_1 through v_5 to the root node T, nodes F and G send the mean value they received from the last hop, respectively, to the sink. In this way, the data traffic in the network is reduced while the sink still receives useful data.

There are lots of literatures on data aggregation in WSN. Most of the early data aggregation research place energy efficiency as their main concern. Different protocols are proposed to increase the energy efficiency for different network architectures. Later on, quality of service (QoS) metrics in data aggregation have also drawn interests of the community, especially for end-to-end delivery latency. In the following, we discuss representative data aggregation protocols that focus on energy efficiency and latency, respectively.

4.2.1.1 Energy Efficiency

Chandrakasan et al. [6] first proposed a low-energy adaptive clustering hierarchy (LEACH) protocol for cluster-based WSN. In LEACH, the whole network is divided into several small clusters, each of which has a node functioning as a cluster head. Data within a cluster are collected by nodes in the cluster and transmitted to the head node. Nodes in the same cluster take turns to work as a cluster head. After the head node collects all the data, it aggregates all the data and sends the aggregated data to the sink.

Although LEACH improves the energy efficiency of the whole sensor network, it uses some strong assumptions that limit its development to real-world applications. LEACH assumes that each node has the same power level and collects data all at a fixed frequency, but in real-world applications, sensors' energy level may vary, and the data collection frequency is arbitrary. Some other cluster-based data aggregation protocols have also been proposed to overcome these drawbacks [2,24,66]. However, all these protocols are designed for the case when there is only one hop from the cluster head to the sink.

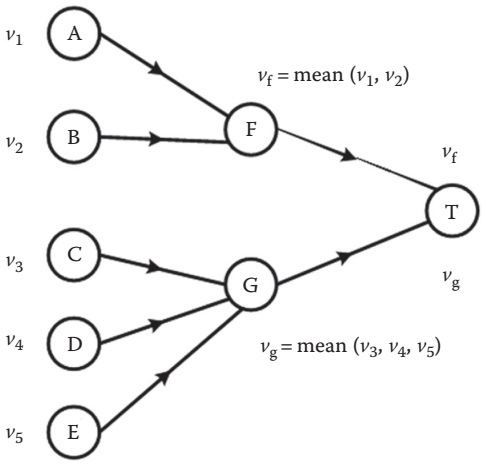


FIGURE 4.1 An example of data aggregations.

Besides cluster-based data aggregation, chain-based data aggregation protocols are also studied. A chain-based aggregation protocol called power-efficient gathering in sensor information systems (PEGASIS) is proposed in [40]. PEGASIS has each node only communicate with its closest neighbor during the transmission to the base station, which means the data are aggregated hop by hop. Two modified versions of PEGASIS are also proposed in [40], both of which aim to combine the idea of clustering with chain-based data aggregation. According to simulation results, PEGASIS can achieve 100%–200% better energy efficiency than LEACH. Nevertheless, both PEGASIS and its modified versions assume that each node has a global knowledge of all nodes in the network. And the delay to transmit data from the end of the chain to the sink may be very large, especially when the network size is large, even though the authors adopt CDMA technique to allow simultaneous transmissions.

Both cluster-based and chain-based data aggregation protocols organize the aggregation structure into a special tree, that is, a cluster tree or a chain. Therefore, designing data aggregation tree in WSNs becomes general focus in the data aggregation area. Ding et al. [9] designed an energy-aware distributed heuristic (EADAT) algorithm to construct a data aggregation tree in WSN. EADAT makes sensors choose their role in the whole tree based on their residual energy.

Tan and Körpeoğlu [49] propose another data aggregation tree construction algorithm called power-efficient data-gathering and aggregation protocol (PEDAP) based on minimum spanning tree. Although PEDAP achieves a better energy efficiency compared with EADAT, it assumes that each node has the global position knowledge of the whole network, which makes its communication overhead too high for general WSN.

There are also many other references talking about how to build data aggregation tree with different metrics [18,42]. Lu et al. [42] proposed a medium access control (MAC) protocol with data aggregation tree construction. Nodes in the tree take turns to be active and asleep. By doing this, not only can the energy efficiency be improved, but also the channel collision problem is alleviated.

Fan et al. [18] studied the scalability issue in data aggregation protocol design. The authors point out that maintaining the whole data aggregation tree structure in large-scale networks is resource and time consuming. To make data aggregation protocol scalable, the proposed protocol [18] first constructs several small shortest path trees as data aggregation trees. Then the root node of each small tree dynamically decides how to forward and further aggregate the aggregated data to the sink. In this way, the whole protocol becomes scalable, and the energy of nodes can be fully utilized since only several small tree structures need to be recorded and updated.

Another designing metric that is often used when designing data aggregation protocols is the network lifetime. In WSN, there is not a general definition for network lifetime. Currently, the most widely used definition is to define the network lifetime to be the time when the first node/the first fraction of nodes in the network run out of its/their energy. Some works have been done aiming to maximize the network lifetime of WSN with data aggregation [10,28,62].

Kalpakis et al. [28] designed an approximate algorithm to solve the maximum lifetime data gathering (MLDA) problem. The authors divide time into multiple time units and build a relaxed network flow model on this problem. Based on this model, data aggregation tree is constructed and updated. To make the whole algorithm scalable, a cluster-based MLDA problem is modeled and solved. The proposed model is compared with PEGASIS [40] using simulation. Results show that the network lifetime is significantly prolonged by the proposed MLDA-based protocol.

Xue et al. [62] modeled the problem of maximizing network lifetime in WSN as a multicommodity flow problem, in which a commodity represents the data generated from a sensor node and delivered to the sink. They designed an approximate algorithm with an approximation ratio of $1 - \varepsilon$ for any $\varepsilon > 0$. Based on this centralized algorithm, they further designed a distributed data aggregation algorithm. Simulation results show that the distributed algorithm has a much better performance on both energy efficiency and network lifetime than the minimum energy routing (MinEnergy) algorithm, whose goal is to minimize the energy consumption for each data unit routed through the network.

4.2.1.2 Latency

Yu et al. [68] studied the energy-latency trade-off for data gathering in WSN. Although this chapter still uses energy efficiency as the objective function, the authors put hard latency constraints on the problem definition. This research assumes that the data aggregation structure has already been built. In each data collection round, each nonroot node generates one piece of data, and every piece of data should be sent to the base station within its latency constraint. During the transmission, data from different sources can be aggregated such that only aggregated data need to be transmitted to the sink. The objective is to find a transmission and aggregation scheme for the whole data aggregation tree in each data collection round, such that the total energy consumption is minimized and every data is sent to the sink without violating the latency constraint. The authors build a nonlinear programming model for this problem and solve it using a numerical algorithm. Then a pseudopolynomial time approximation centralized algorithm based on dynamic programming is designed for this model. Furthermore, the authors implement an online distributed algorithm to adaptively control the transmission and aggregation policy of each node. It adopts a feedback control scheme to make nodes transmit faster if there are data violating the latency constraint. The proposed protocol is evaluated in simulation. Numerical results show that the distributed protocol can give a good approximated performance compared with the numerical algorithm and the dynamic programming centralized algorithm in terms of energy consumption. In the meantime, its adaptivity is also demonstrated through simulation.

As a starting chapter on the energy-latency trade-off in data aggregation, this chapter gives a good approximate algorithm to solve the problem modeled in this chapter. However, the problem definition is relatively simple since in each instance, only one data is generated at one source in each round. And the proposed distributed algorithm requires the cooperation from MAC layer protocols to minimize the interferences between nodes.

Becchetti et al. [3] systematically studied the complexity of latency-constrained data aggregation scheduling problem in WSN under different models. Different from [68], this chapter studies the complexity of latency-constrained data aggregation scheduling problem on different aggregation structures and different traffic patterns. Instead of minimizing the total energy consumption, the authors define two different objective functions. The first one is to minimize the total expected number of transmissions (ETX) given that each link has a constant ETX regardless of packet size, and the second one is to minimize the maximal total ETX in one node. This chapter proves that when the data aggregation structure is a tree, the whole problem is nondeterministic polynomial time (NP)-hard for both two objective functions with a reduction from the boolean satisfiability (SAT) problem. However, both problems are proved to be 2-approximative. The authors also give a polynomial dynamic programming algorithm to solve the problem with the first objective function in a chain data aggregation structure. Besides, this chapter also proposes a simple aggregation algorithm that evenly divides the spare waiting time for aggregation at different intermediate nodes along the transmission path. The authors analyze the competitive ratio of this algorithm and the upper bound for the competitive ratio of all possible algorithms for this problem on different aggregation structures.

The chapter mentioned earlier gives a complete theoretical analysis on the complexity of latency-constrained data aggregation in WSN, which builds a good theory foundation for the latency-guaranteed data aggregation research. The drawbacks of this chapter, however, are the following: (1) it does not evaluate the proposed simple packing scheme on either simulation or experiment and (2) the competitive ratio and the bounds have too many parameters, which make the ratio highly depend on specific data aggregation structures.

As a continuous work [3], Oswald et al. [47] proposed another approximate algorithm for the latency-constrained data aggregation problem. Instead of using energy efficiency as the objective function, this chapter defines the objective function to minimize the transmission cost. The authors define energy cost functions for energy consumption on transmissions and delay cost functions for nodes to hold data for further aggregation opportunity. The transmission cost is defined as the sum of energy cost and the delay cost. The chapter proposes an approximate algorithm to solve this problem. They derive a

competitive ratio ($h(c)$) of this algorithm for tree structure, where h is the tree's height and c is the transmission cost per edge, and a competitive ratio $\Theta(\min(\sqrt{h}, c))$ for chain structure. Both these two ratios are proved to be tight since the upper bound of the competitive ratio is proved to be at least $\Omega(\min(h, c))$. This chapter only focuses on theoretical analysis and does not give simulation or experiment evaluation for the proposed algorithm [3]. And the importance of this chapter is weakened because the objective function is not defined objectively. It would be more appropriate to define the objective function to be minimizing energy consumption.

Latency-constrained data aggregation is also studied in vehicular ad hoc networks (VANETs). Yu et al. [67] proposed a data aggregation protocol called CatchUp for data aggregation in VANETs. CatchUp dynamically controls the data forwarding delay in VANETs so that data can be fully aggregated during the transmission with an allowable delay. Different from data aggregation in WSN, where all data have the same base station as the destination, data aggregation model used in this chapter is described as follows. Each vehicle would broadcast its sensed data to every other vehicle in the network. CatchUp defines a rewards function for each node in the network to decide what action to take to have a maximal reward. The energy efficiency and latency constraint are not directly shown in the problem definition [47]. And CatchUp uses a local heuristic algorithm for each node to make decisions, which can only provide soft local latency guarantee.

Ye et al. [64] give a more systematical solution on local latency-constrained rewards maximizing algorithm [67]. This chapter models the problem on a single node in a WSN using data aggregation. In this chapter, the authors build a semi-Markov chain decision-making model for each node. The impact of latency constraint of data is defined as a negative-exponential rewards function. With the help of some important characteristics of semi-Markov chain, the chapter shows that once the statistics of the data arrival and the availability of the channel satisfy certain conditions, there exist optimal control-limit-type policies that are easy to implement in practice. In the case when the condition of the existence of optimal transmitting and waiting policy is not satisfied, the chapter provides two learning algorithms to solve a finite-state approximation model of the decision problem. Simulation results show that under two data aggregation schemes, the fixed aggregation scheme (FIX) scheme and on-demand aggregation scheme (OD) that are designed [23], both the optimal transmitting and waiting policy control algorithm and two approximate learning algorithms could effectively reduce the energy consumption while the data delay is guaranteed in a low value. Although it does not pose any hard latency constraint on the semi-Markov chain model, the fast decrease property of negative-exponential rewards function ensures that holding data for a long time for further data aggregation opportunities will not happen in the proposed algorithms.

4.2.2 Data Fusion

Data fusion is a collaborative signal processing technique that is widely used in distributed systems to enable the cooperation among multiple devices with limited sensing capability [56]. This technique has been widely studied for decades. Due to the limited sensing capability, the limited energy capacity, and the application background of WSN, data fusion has a wide application prospect in WSN applications.

Though it has a similar definition with data aggregation, data fusion is a more general technique that is more close to the application layer in WSN. In data aggregation, data from different sources are simply aggregated or compressed at some intermediate node so that the whole traffic in the network is reduced. On the contrary, in data fusion, not only are data aggregated or compressed, but also they are processed along the transmission to the sink to provide guarantee for data accuracy. Each individual sensor in the whole network can play the role as a decider. In other words, with data fusion technique, WSN can work as a distributed detection and decision-making system. Figure 4.2 gives a demonstration on how data fusion can be utilized in WSNs. In this network, nodes F and G each serves as a distributed fusion center and makes their own decision based on the information they received. And they only need to send their decisions d_F and d_G to the root node T.

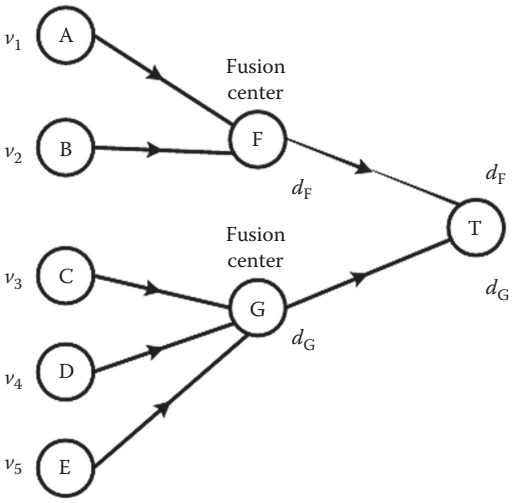


FIGURE 4.2 An example of data fusion.

Different data fusion architectures and systems are designed to fully use the limited resources in each sensor and in the meantime to guarantee the data accuracy. In the following, several representative works on data fusion in WSN are reviewed.

Thomopoulus et al. [55] studied the optimal data fusion in the sense of the Neyman–Pearson (N–P) test in a centralized fusion center. In the whole system, each sensor independently executes an N–P test and sends the decision result of the test back to the sink instead of sending the raw data. After receiving all decision results, the sink makes a final decision and adjusts the threshold of the whole test based on the final decision. This is an early work on data fusion in sensor networks. The whole system is built on sensor networks with powerful sensor, which does not take energy efficiency into account.

Niu et al. [46] proposed a distributed detection protocol in WSN [55]. In the proposed protocol, each sensor also individually and independently runs a hypothesis test and only sends the test result back to the sink. Consideration exists for the spatial correlation of data sensed by different sensors. The authors reach the conclusion that if the number of sensors is sufficiently large, the proposed fusion rule can provide a very good system level detection performance, in the absence of the knowledge of local sensors' performances and at low signal-to-noise ratio (SNR). Though the authors mention that sending only decisions from sensors to the sink could reduce the traffic in the networks, they do not formally take energy efficiency into account, either.

These two chapter are early work on applying data fusion into WSN, and they mainly focus on the data accuracy provided by fusing distributed decision value together. Energy efficiency is not the main design objective in these protocols, but only a by-product.

Clouqueur et al. [7] systematically compared the performance of distributed detection systems in WSN using value-based data fusion and decision-based data fusion. In value-based data fusion, raw data are directly sent back to the sink, and the sink fuses all raw data, abandons the outliers, and makes the final fusion. In decision-based data fusion, the chapter adopts a similar way [46,55], which sends back only decision calculated by each individual sensor to the sink. The authors conduct simulations to compare the performance of these two fusion schemes with robustness as the main metric. Results show that when the proportion of failed sensors in the whole network increases, decision-based fusion outperforms value-based fusion by providing a lower false decision probability, a lower power consumption, and a higher packet delivery probability.

Even though Clouqueur et al. [7] studied the energy efficiency of both value-based and decision-based data fusion in WSN, the authors do not discuss how to further reduce the traffic in WSN by

allowing data fusion in sensors. Kumar et al. [33] developed an architectural framework, DFuse, for distributed data fusion in WSN. There are two main components in the framework of DFuse. First, a fusion API is implemented so that the system can afford the development of complex sensor fusion applications. Second, the authors propose a heuristic algorithm to decide which set of sensors can play the role of fusion center. The idea of fusion center is similar as the cluster head in data aggregation. Not only does the fusion center is aware of the energy efficiency of the whole network, but also it helps distributed fusion operation in the network. The performance of DFuse is evaluated via simulation. Results show that DFuse can make sensor consume energy in an efficient way. The simulation also analyzes the latency caused by data fusion, but no bound of latency can be guaranteed in DFuse. Furthermore, although the evaluation studies the impact of different energy cost functions on DFuse, it does not talk about the impact of DFuse on different fusion applications.

Duarte and Hu [17] propose a distance-based decision fusion scheme for the collaborative target detection and classification of moving vehicles using acoustic spectral features. The authors design a new scheme to use the distance between the target and the sensor as a parameter to select sensors that can give a reliable detection result to participate decision fusion. This scheme makes use of an intuition that sensors far from the target will have a lower probability of making correct classification decisions. Therefore, only sensors close to the target can participate the target detection and classification. In this way, the communications within WSN is reduced so that energy efficiency is achieved. Simulation results show that the accuracy of target detection and classification is guaranteed and the energy efficiency is improved. Though data accuracy is guaranteed, the proposed scheme does not take other QoS requirements, for example, reliability and delay, into account.

Tan et al. [50] developed an analytical framework to study the real-time surveillance performance of large-scale WSN that is designed based on collaborative data fusion schemes. The authors define a delay metric called α-delay that is defined as the delay of detecting an intruder subject to the false alarm rate bound by α. The road map of this chapter is as follows: compared with intruder detection systems in WSN without data fusion, fusion-based systems require a smaller network density to achieve a false alarm rate α. Network density will further affect the end-to-end latency in WSN. Therefore, to achieve minimal α-delay, the ratio of network density of WSN with data fusion scheme and without data fusion scheme has an asymptotic tight bound of $\Theta(\text{SNR}/Q^{-1}(\alpha))$, where Q^{-1} is the inverse function of the complementary cumulative distribution function of the standard normal distribution. Simulations with realistic settings show that data fusion can reduce the network density by about 60% compared with the a general disc model without fusion while detecting any intruder within one detection period at a false alarm rate lower than 2% and guaranteeing that the detection delay is minimal.

Tan et al. [51,52] studied the calibration problem for fusion-based sensor networks. The authors propose an adaptive system-level calibration approach for sensor networks that employed collaborative data fusion for event/target detection. This calibration approach adopts a feedback control loop to adaptively mitigate the impact of physical uncertainties on the environment and the dynamics of the physical event/target of interest. The authors prove the stability and convergence of the proposed feedback control scheme. A routing algorithm for fusion-based sensor networks is also designed to minimize the impact of dynamics on fusion-based WSNs. Experiment and simulation results show that the proposed calibration system is able to maintain optimal detection performance under the presence of system and environmental dynamics.

Under the similar sensor measurement and data fusion model [51,52], Tan et al. [53,54] proposed a two-tier system level calibration approach for fusion-based sensor networks. The whole system is composed of two tiers. In the first tier, each sensor learns its local sensing model using in-place measurements and only transmits model parameters to the fusion cluster head. In the second tier, the fusion cluster head calibrates each sensor's model to a common sensing model. Using this two-tier approach, the communication overhead from sensors to fusion head is significantly decreased. A linear regression algorithm is proposed for first-tier local sensor model learning. And another algorithm is designed to calibrate biased local sensing models and maximize the system detection probability. The authors

evaluate the performance of this two-tier system using both experiment and simulation. Results show that the proposed approach can significantly improve the detection performance of sensor networks under different realistic settings.

4.3 Application-Independent INP

Besides data aggregation, there are also INP methods that do not require customized functions depending on specific applications in WSN. These methods are called *application-independent INP*. In this section, we introduce two most representative application-independent INPs in sensor networks, packet packing and network coding, and survey their recent research progress.

4.3.1 Packet Packing

Different from data aggregation, which aggregates spatial- or temporal-related packets into a packet while the size of the aggregated packet remains the same, packet packing technique simply puts information elements in packets together regardless of the correlation of packets. The length of the packed packet equals to the header plus the length of all information elements. Figure 4.3 gives an example on how packet packing works. Different from Figures 4.1 and 4.2, where data aggregation and data fusion sent aggregated/processed value to the sink, packet packing only assembles short packets into longer ones, for example, node F putting packets p_1 and p_2 together. In this way, nodes F and G can also send less number of packets to T such that the efficiency of the whole sensor network can be improved.

As a special INP method, packet packing has also been studied for WSN as well as general wireless and wired networks. In the following, we investigate the same representative ones. Jain et al. [27] studied the benefits of packet packing in ad hoc wireless networks under IEEE 802.11b standard. The authors point out that 802.11b networks have a high header overhead, which takes up a lot of bandwidth. They alleviate this high overhead by allowing a small delay on packets during the transmission so that intermediate nodes can pack different small packets into a larger packet before forwarding it to the next hop. In their protocol, they preconfigure a maximum aggregation delay to keep each packet wait at intermediate nodes for a while so that packets can be packed together. Based on both experiments on a wireless test bed and simulations on NS-2, their protocol can provide a significant improvement on network capacity compared with wireless networks without using packet packing. However, there are some drawbacks on this predefined waiting time. The end-to-end latency cannot be guaranteed. Meanwhile, by waiting at a fixed time at each intermediate node, a packet may lose the opportunity to pack more other packets at some certain nodes.

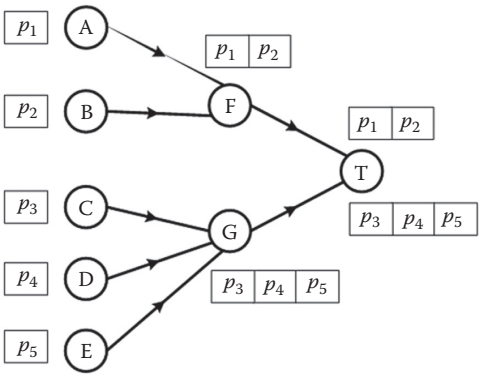


FIGURE 4.3 An example of packet packing.

Other work [27] also looks at the throughput gains by applying packet packing into MAC protocol design [36,38,43]. Li et al. [36] proposed an adaptive QoS-aware frame concatenation mechanism (AQCM) to control how long a packet should wait at an intermediate node. The AQCM is mainly designed for multimedia applications in multirate wireless ad hoc networks. AQCM controls the waiting time of every packet by detecting whether the required flow rate is satisfied and whether there is congestion in local traffic. The simulation results show that AQCM can achieve a desirable performance on multimedia multirate wireless ad hoc networks. However, AQCM only provides soft QoS guarantee.

Li et al. [38] and Lu et al. [43] use a packet packing to design a MAC protocol in ultra-wideband networks (UWBNs) and high-speed wireless local area networks (WLANs), respectively. Both of them adopt an opportunistic scheme to do packet packing. Using this scheme, the end-to-end delay of data flows is to some extent improved. But packets will still lose an opportunity to get further packed [27].

Kliazovich and Granelli [31] designed an internet protocol (IP) level packet packing scheme in WLANs. The authors categorize IP layer packets into two groups, with low priority and high priority, respectively. Packets in different groups cannot be packed. By this way, packets with high priority, that is, small delay constraints, will be packed and transmitted first. This scheme is easy to implement. Both experiment and simulation results show that the throughput can be improved using this grouping scheme compared with no packing scheme and the delay constraint can be satisfied. Though this packing strategy is easy and direct, the drawback of this scheme is that not only will packets lose packing opportunity but also the latency constraint can still be violated if packets in the same group have different latency constraints.

Then, Saket and Navet [48] studied packet packing in single-hop controller area networks (CANs) with a finite packet size. It gives a heuristic greedy algorithm that packs small packets into a single frame as many as possible. Different from the other authors [27,36,38,43], this chapter studies the impact of finite packet size on packet packing.

He et al. [23] developed a novel adaptive application-independent data aggregation (AIDA) protocol to provide soft latency guarantee for packet packing in WSN. AIDA is designed to be an independent layer between network layer and MAC layer. Packets can be packed in this layer under different packing schemes. The authors propose three different packing schemes. The first one is called fixed aggregation scheme (FIX), where AIDA packs a fixed number of network units into each AIDA payload. To ensure that network units do not wait an indefinite amount of time before being sent, a time-out threshold is predefined in the system. The second scheme is called on-demand (OD) scheme, which adopts an opportunistic packing policy. OD puts the real-time guarantee as the top concern. Packets at the same sensor can only be packed when the MAC layer is not available for transmission. In FIX scheme, the system can only provide a soft latency guarantee, and packets will lose opportunities to get further packed. In OD scheme, hard latency constraints are guaranteed, but packets have less opportunity to get packed than in FIX, which can increase energy consumption. To balance the energy efficiency and the latency requirement, the authors propose the third scheme called dynamic feedback (DYN) scheme. DYN implements a combination of OD scheme and FIX scheme where the number of packets packed in one sensor is adjusted dynamically via a feedback control from the output. In the case of low network traffic, DYN will default to the OD mechanism delivering packets to the MAC transmission queue as soon as they are ready. As network traffic builds up and the contention delays transmission, the feedback loop adjusts the threshold of number of packets that can be packed together to allow a greater degree of packing prior to sending. Simulation results indicate that DYN outperforms OD and FIX by providing a lower-average end-to-end delay, especially in heavy-load traffic. Nonetheless, the proposed DYN scheme has overreaction or underreaction on the change of MAC delay, which cannot provide hard latency guarantee for each single packet.

Dong et al. [13] studied the dynamic packet length control (DPLC) in sensor networks. The authors show that the length of packet can significantly affect the delivery performance in WSN. Therefore, this chapter proposed DPLC, a dynamic packet length adaption scheme with a lightweight and accurate

data-plane link estimation component. DPLC adaptively aggregated smaller packets into a larger one or fragments larger packets into smaller ones based on the physical channel conditions and interferences. Experiments on a 20-node test bed under a light traffic pattern showed that DPLC results in a 13% reduction in transmission overhead and a 41.8% reduction in energy consumption compared with collection tree protocol (CTP).

Xiang et al. [57,58] studied the joint optimization between packet packing and the latency of data delivery. They provide a comprehensive computational complexity analysis on this scheduling problem in sensor networks. The authors proved the strong NP-hardness of this problem via a reduction from SAT problem. They also show that certain special packing constraints make this problem polynomial solvable. Based on the complexity analysis, the authors designed a distributed, online protocol named tPack to make packing decisions to maximize the local utility of packet packing at each node. Experiment results on the NetEye test bed show that tPack is able to provide at least a reduction of 70% transmission cost in various heavy traffic patterns in a 120-node dense topology.

4.3.2 Network Coding

Network coding (NC) is first proposed for wired networks [1]. By mixing packets at intermediate nodes during the transmission, the bandwidth can be saved, and therefore, the throughput of the whole network can be significantly improved. During the past years, network coding has been one of the most popular research topics in computer networks. Different coding schemes are designed, categorized into linear network coding and nonlinear network coding. Compared with linear network coding, nonlinear network coding has been reported to outperform linear coding in several studies [15,16,34,35]. And there are multisource network coding problems for which nonlinear coding has a general better performance on throughput [16]. Nevertheless, according to the an analysis [37], linear network coding can provide a performance close to the best possible throughput, while only requiring a relatively low complexity, compared with the high complexity of nonlinear coding.

Due to the broadcast nature in wireless communication, each intermediate node can receive redundant packets during the transmission in wireless networks. Network coding is one of the best choices to make use of these redundancies. By mixing redundant packets together and forwarding the mixed packet, the throughput of wireless networks can be further improved. It is shown that linear coding functions can be designed randomly and independently at each node. Ho et al. [25,26] proposed a coding technique called random linear coding (RLC). Since RLC can be easily implemented in a distributed manner and it has a low complexity, it is widely used in wireless networks.

In sensor networks, network coding has been mainly applied in three scenarios, NC-based opportunistic routing, code dissemination, and NC-based network protection. In the following, we will introduce representative network coding protocols of these application scenarios.

4.3.2.1 NC-Based Opportunistic Routing

Opportunistic routing is proposed [4] with the protocol ExOR and has drawn the community's interests. Since opportunistic routing also makes use of the broadcast property in wireless communication, researchers have been working on the hybrid architecture of network coding and opportunistic routing in wireless networks.

Katti et al. [30] proposed COPE, a new architecture for wireless mesh networks. It is the first network coding that is implemented with the current network stack seamlessly. In the design of COPE, only interflow network coding is concerned. COPE adopts an opportunistic coding scheme, which does not delay packets' transmissions for further coding opportunity. According to the theoretical analysis, not only can network coding bring significant improvement on throughput, but also the MAC layer protocol can also improve the network throughput when it is combined with coding technique. COPE is implemented on a 20-node wireless network test bed. Experiment results show that COPE can increase the throughput of wireless mesh networks without modifying routing or higher layers.

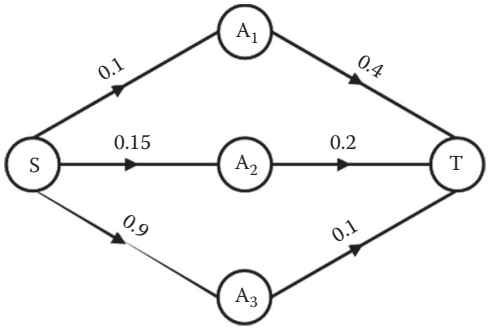


FIGURE 4.4 An example for NC-based opportunistic routing.

With continuous research [4,30], Chachulski et al. [5] combined intraflow RLC and the opportunistic routing protocol in [4] together to develop a new routing protocol called a MAC-independent opportunistic routing protocol (MORE) in wireless mesh networks. Figure 4.4 gives an example of intraflow NC-based opportunistic routing. Different from traditional shortest single path routing, where node S can only choose node A_3 as next hop, NC-based opportunistic routing makes use of all one-hop neighbor of node S, that is, nodes A_1–A_3. In this way, the routing diversity is fully utilized such that the network throughput can be improved. The contribution of MORE is multidimensional. First, it makes use of the broadcast property of wireless communication to improve the network throughput without modifying the existing MAC layer, that is, 802.11. Second, it adopts RLC for intraflow network coding. RLC has a low complexity and is easy to implement in a distributed system. Therefore, the network throughput is further improved. Third, both the memory overhead and the header overhead are bounded within a reasonable range. MORE is also evaluated in a 20-node test bed [4], and it outperforms ExOR in both unicast and multicast traffic flow with a higher throughput.

To further improve the throughput of wireless networks, Lin et al. [39] made use of hop-by-hop ACK and sliding window to allow different segments of packets to be transmitted in the network concurrently (CodeOR). However, it still adopts off-line ETX metric to decide how many coded packets to transmit to ensure the end-to-end decodability. To be adaptive to the dynamic of wireless links, Koutsonikolas et al. [32] designed a cumulative coded ACK (CCACK) scheme to allow nodes to notifying their upstream nodes that they have received enough coded packets in a simple and low overhead way. The throughput of CCACK is shown to be 45% better than that of MORE. The CCACK scheme gives a good solution to the problem *when should a sender stop broadcasting*. However, CCACK's major objective is to minimize the broadcast cost at each sender/forwarder. This approach cannot give a global minimization on transmission cost for NC-based opportunistic routing. Furthermore, CCACK requires a high memory space and a relatively complex computation process, which is not suitable for resource-constrained sensor networks.

Xiang et al. [59,60] studied the minimal cost NC-based routing problem in WSNs. The authors proposed the first mathematical framework on analytically measuring the cost of NC-based routing. They designed a greedy algorithm that can minimize the transmission cost of NC-based routing and prove its optimality. It is also shown in this work that the transmission cost of NC-based routing is upper bounded by the cost of shortest single path routing and that the shortest single path is not always chosen into the minimal cost forwarding set, as shown in Figure 4.5. Figure 4.5 has the same topology as Figure 4.4. In this example, the shortest path is $S \rightarrow A_3 \rightarrow T$, while the minimal cost NC-based routing braid is $S \rightarrow \{A_1, A_2\} \rightarrow T$. The authors proposed energy-efficient network-coding-based routing protocol (EENCR), an energy-efficient network coding-based routing protocol, which contains a distributed implementation of the greedy algorithm. Experiment results from the NetEye test bed demonstrated that EENCR significantly outperforms CTP, MORE, and CodeOR in terms of delivery reliability, delivery cost, and goodput. This work shows the routing diversity of wireless communication requires adaptively utilization in order to provide efficient and reliable service in WSNs.

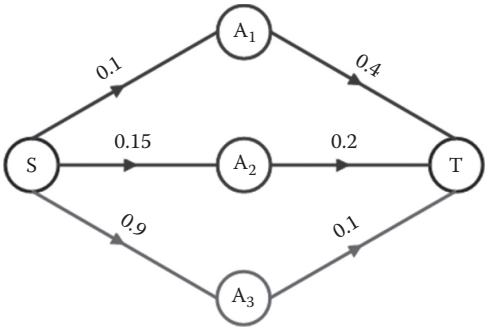


FIGURE 4.5 Minimal cost NC-based routing versus shortest path routing.

4.3.2.2 Code Dissemination and Reprogramming

Hagedorn et al. [22] proposed Rateless Deluge, the first WSN code reprogramming protocol that utilizes network coding. The use of network coding can effectively eliminate the need of feedback information about which packets need to be retransmitted and hence effectively increase the efficiency of code dissemination and reprogramming. Experiment results on Tmote Sky motes in both single-hop and multihop networks demonstrate that Rateless Deluge can significantly save the communication overhead in both data and control plan, that is, 15%–30% and 50%–80% than regular Deluge protocol, respectively.

Dong et al. [12] presented an analytical mathematical framework to analyze the performance of code/bulk data dissemination protocols in WSNs. This framework considers topology information, impact of contention, and pipelining. The authors compared the analytical results of this framework and simulation results in both square structures and linear structures. Results show that this mathematical framework fits accurately with the simulation results and much better than the analytical approach used in Deluge.

Dong et al. [11] studied the reprogramming problem in sensor networks and proposed ReXOR, a lightweight and density-aware reprogramming protocol for WSN. ReXOR uses XOR coding technique in the retransmission phase to reduce the transmission cost. The authors analyzed the advantage of ReXOR over Deluge in sparse and lossy network and its advantage over Rateless Deluge in dense network. ReXOR utilizes both the advantage of general network coding in reducing transmission cost and the near-zero decoding delay in XOR coding. In sparse and lossy network, ReXOR adaptively increases the interpage waiting time to improve the coding opportunity while in dense networks; the interpage time can be controlled to reduce transmission cost and propagation delay. Experiment results show that ReXOR has a much lower code dissemination completion time than both Deluge and Rateless Deluge in both sparse and dense grid topologies. In the meantime, it has a significant lower data traffic than both Deluge protocols, which is desirable in resource-constrained sensor networks.

Gao et al. [20,21] proposed a multithreaded design for network coding-based data dissemination. The protocol, MT-Deluge, separates the coding and radio operations into two threads. In the coding thread, an incremental decoding algorithm is proposed to shorten the waiting delay of radio thread. When the incremental decoding algorithm is executed, a packet-level thread synchronization mechanism is adopted to provide precise synchronization between multiple threads. When this algorithm is not being executed, a state-level thread synchronization mechanism would be enough. Experiment results in multihop line topology, multihop grid topology, and single-hop clique topology show that MT-Deluge can reduce the dissemination delay significantly in multihop topologies while single-thread dissemination topology such as Rateless Deluge is enough in single-hop topologies.

Dong et al. [14] proposed an efficient code dissemination protocol (ECD), an efficient code dissemination protocol in sensor networks. Leveraging the results [13], ECD supports configurable packet sizes, which improve the transmission efficiency in terms of transmission cost. ECD designs an accurate sender selection algorithm to mitigate the collision between transmissions. It also employs a simple

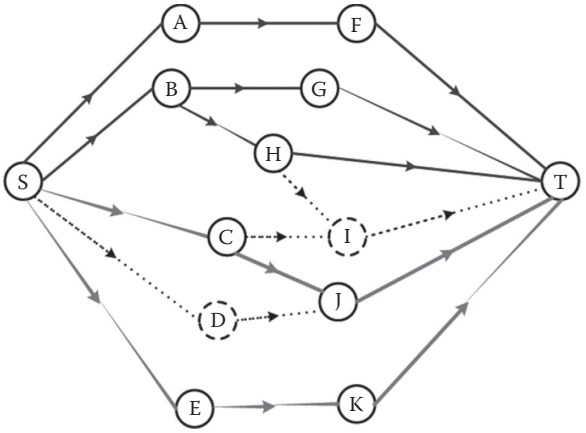


FIGURE 4.6 An example of node-disjoint NC-based braids.

impact-based backoff timer design to reduce the time spent in coordinating the transmissions of eligible senders. Experiment results show that ECD provides a better performance than Deluge and MNP, two classic code dissemination protocols, in terms of both data traffic and completion time. Results also demonstrate the impact of sender selection in code dissemination. Plus, network coding technique used in designing Rateless Deluge and ReXOR [11] can also be applied into ECD design and implementation.

4.3.2.3 NC-Based Network Protection

Kamra et al.'s [29] work is the first work to put resiliency as the major concern for network coding in WSN. The authors propose Growth Codes, a new class of network coding particularly suited to sensor networks where data collection is distributed. Unlike previous coding schemes, Growth Codes employs a dynamically changing codeword degree scheme that delivers data at a much faster rate to network data sinks. Furthermore, the coding algorithm is designed such that the sink is able to decode a substantial number of the received coded packets at any stage. Simulations in TOSSIM and experiments show that Growth Codes provide a high reliability in WSN where nodes are highly prone to failures.

Xiang et al. [61] studied the proactive NC-based protection problem to provide resiliency against transient network failures in sensor networks. The approach utilized in this work is to construct two node-disjoint braids to provide 1 + 1 protection. In Figure 4.6, for example, two node-disjoint braids marked in red and blue are constructed to deliver two copies of data from S to T. Different from constructing two node disjoints with minimal total cost, which is polynomial solvable, it is shown to be NP-hard to construct two node-disjoint network coding routing braids with minimal total cost in this work. The authors design a heuristic algorithm alternatively assigning nodes into two braids. Based on this algorithm, the authors designed ProNCP protocol and evaluated its performance on the NetEye test bed. Results show that ProNCP provides stable performance with close to 100% reliability under different transient failure models.

4.4 Research Challenges

From the previous discussion, we can find that, in resource-constrained wireless networks, INP effectively enhances messaging efficiency by reducing network traffic load. Nonetheless, most of the study of INP in wireless and sensor networks has ignored the issue of providing hard QoS such as the timeliness, reliability, and resiliency of data delivery when controlling the temporal and spatial data flow in networks. Therefore, how to provide QoS guarantee for INP in WSN is still a new area where only some preliminary works have been done. In this section, we point out some open issues and research challenges in this field.

4.4.1 Systematic Modeling and Complexity Analysis

Because QoS constraints are added to INP protocol design for WSN, the problem formulation will be different from that in existing research on energy-efficient INP design for traditional WSN and wireless networks. Some authors [64] have proposed some simple modeling frameworks for QoS-aware data aggregation protocol design based on semi–Markov chain. Nonetheless, frameworks like these are used for specific QoS constraints, such as reliability and latency, and only for data aggregation. And we still lack a general modeling framework across different QoS constraints and different INP methods to push the research on this area. Other authors [57,58] give an interval graph model for the latency-constrained packet packing problem, which may also apply to general QoS-aware INP scheduling problem in sensor networks. Still, other authors [60] propose the first mathematical framework on computing the transmission cost of NC-based routing and an optimal NC-based routing algorithm. But a more robust modeling framework for general INP methods is still needed.

Besides modeling issues, complexity analysis is also of great importance in this area. Complexity of problems may change due to the joining of new QoS constraints. Some QoS constraints may make the new problem easier, especially in a chain network [3,57,58], while some constraints may make the new problem even NP-hard to approximate [3,61]. A complete complexity analysis on QoS-aware INP problem will provide a guideline on how people can design and implement efficient approximate algorithm for sensor networks.

4.4.2 Joint Optimization of QoS and WSN-Specific INP

Existing work on QoS-aware INP design mainly consider how to provide service in WSN with guaranteed latency and reliability. Although in mission-critical real-time WSN these two metrics are the most important ones, there are other QoS metrics unexplored, for example, interactivity. Besides the aforementioned INP methods, there are also other WSN-specific INP methods including different degrees of data compression and local data filtering. Different INP methods in WSN will lead to different trade-offs among different QoS metrics. These trade-offs tend to be application specific. Studying the joint optimization on QoS and these new INP methods can provide support in the close-loop control in modern WSN.

4.4.3 Cooperation of Different INP Methods in WSN

The community has started to study the trade-off between QoS and single INP method in WSN. Some work [5,30,45] also proposes whole system architectures that cooperate INP with existing network protocol stack. However, how to apply different INP methods together in one system is still an open area. A simple example will show that this approach can further improve the system performance. Suppose intraflow coding is adopted in wireless networks. After a node did intraflow network coding for a few packets, it can further pack these coded packets together using packet packing method. In this way, the total ETX can be further reduced. Since INP methods all aim to reduce the traffic load in WSN, studying the cooperation between different INP methods is a promising direction to provide QoS-guaranteed performance for WSN. However, characteristics and major concerns of different INP methods can make the cooperation complex. For example, data fusion mainly aims to guarantee the data accuracy while data aggregation mainly considers how to minimize energy consumption. Thus, it is a challenging task to balance these two goals.

4.4.4 Theoretical Foundations of Algorithm Design

The research on QoS-aware INP design in WSN is still a developing area. Due to the different characteristic between WSN and other wireless networks, traditional network optimization theory is not enough to provide mathematical tools for this area. For example, traditional network flow perspective

mainly studied the static network flow model under which the conservation law always holds. Xiang et al. [60,61] studied the nonadditive network coding-based routing flow problem, showing that under NC-based routing pattern, the conservation law of network flow does not always hold. This shed lights for future research on QoS-assured INP data flow control in modern WSN.

4.5 Summary

After the past decade of active research and field trials, WSNs have started penetrating into many areas of science, engineering, and our daily life. They are also envisioned to be an integral part of cyber-physical systems such as those for alternative energy, transportation, and health care. However, most sensor nodes are highly resource constrained in terms of energy and computational capability. For resource-constrained WSN, INP improves energy efficiency and data delivery performance using light-weight and local computation to reduce network traffic load and thus channel contention. Over the past years, many INP methods have been proposed for query processing and general data collection. In this chapter, we introduce different INP methods in WSN, including data aggregation, packet packing, and network coding. By presenting the basic idea and discussing representative protocols of each category, we give a comprehensive tutorial on INP in WSNs.

In other words, in this chapter, we have presented a comprehensive survey of INP methods in WSN. INP techniques are adopted in WSN with the main goal of minimizing energy consumption. Different INP protocols are investigated in this chapter. We also point out some possible research directions in this area. With both the quick development of modern WSN and its wide use in cyber-physical systems, it is expected that INP continues to play an important role in modern WSN. Therefore, studying how to design QoS-aware INP protocols in WSN is a challenging and an important area for future research.

References

1. R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, July 2000.
2. A. Anandkumar, L. Tong, A. Swami, and A. Ephremides. Minimum cost data aggregation with localized processing for statistical inference. In *Proceedings of IEEE INFOCOM*, Phoenix, AZ, 2008, pp. 780–788.
3. L. Becchetti, P. Korteweg, A. Marchetti-Spaccamela, M. Skutella, L. Stougie, and A. Vitaletti. Latency constrained aggregation in sensor networks. In *ESA'06: Proceedings of the 14th Conference on Annual European Symposium*. Springer-Verlag, London, U.K., 2006, pp. 88–99.
4. S. Biswas and R. Morris. ExOR: Opportunistic multihop routing for wireless networks. *SIGCOMM Computer Communication Review*, 35(4):133–144, 2005.
5. S. Chachulski, M. Jennings, S. Katti, and D. Katabi. Trading structure for randomness in wireless opportunistic routing. In *SIGCOMM'07: Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. ACM, New York, 2007, pp. 169–180.
6. A.P. Chandrakasan, A.C. Smith, W.B. Heinzelman, and W.B. Heinzelman. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1:660–670, 2002.
7. T. Clouqueur, K. Saluja, and P. Ramanathan. Fault tolerance in collaborative sensor networks for target detection. *IEEE Transactions on Computers*, 53(3):320–333, March 2004.
8. A. Deshpande, C. Guestrin, S.R. Madden, J.M. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In *VLDB'04: Proceedings of the 30th International Conference on Very Large Data Bases*, Toronto, Ontario, Canada. VLDB Endowment, 2004, pp. 588–599.
9. M. Ding, X. Cheng, and G. Xue. Aggregation tree construction in sensor networks. In *2003 IEEE 58th Vehicular Technology Conference*, Orlando, Florida, US, October 2003, Vol. 4, pp. 2168–2172.

10. Q. Dong. Maximizing system lifetime in wireless sensor networks. In *IPSN'05: Proceedings of the Fourth International Symposium on Information Processing in Sensor Networks*. IEEE Press, Piscataway, NJ, 2005, p. 3.
11. W. Dong, C. Chen, X. Liu, J. Bu, and Y. Gao. A lightweight and density-aware reprogramming protocol for wireless sensor networks. *IEEE Transactions on Mobile Computing*, 10(10):1403–1415, 2011.
12. W. Dong, C. Chen, X. Liu, J. Bu, and Y. Liu. Performance of bulk data dissemination in wireless sensor networks. In *DCOSS'09: Proceedings of the Fifth IEEE International Conference on Distributed Computing in Sensor Systems*. Springer-Verlag, Berlin, Germany, 2009, pp. 356–369.
13. W. Dong, X. Liu, C. Chen, Y. He, G. Chen, Y. Liu, and J. Bu. DPLC: Dynamic packet length control in wireless sensor networks. In *2010 Proceedings of IEEE INFOCOM*, San Diego, California, US, 2010, pp. 1–9.
14. W. Dong, Y. Liu, C. Wang, X. Liu, C. Chen, and J. Bu. Link quality aware code dissemination in wireless sensor networks. In *ICNP: 2011 19th IEEE International Conference on Network Protocols*, Vancouver, BC Canada, 2011, pp. 89–98.
15. R. Dougherty, C. Freiling, and K. Zeger. Linearity and solvability in multicast networks. *IEEE Transactions on Information Theory*, 50(10):2243–2256, October 2004.
16. R. Dougherty, C. Freiling, and K. Zeger. Insufficiency of linear coding in network information flow. *IEEE Transactions on Information Theory*, 51(8):2745–2759, August 2005.
17. M. Duarte and Y.H. Hu. Distance based decision fusion in a distributed wireless sensor network. In *IPSN'03*: *Second International Workshop on Information Processing in Sensor Networks*, Palo Alto, CA, 2003, pp. 22–23.
18. K.-W. Fan, S. Liu, and P. Sinha. Scalable data aggregation for dynamic events in sensor networks. In *SenSys'06: Proceedings of the Fourth International Conference on Embedded Networked Sensor Systems*. ACM, New York, 2006, pp. 181–194.
19. Q. Fang, F. Zhao, and L. Guibas. Lightweight sensing and communication protocols for target enumeration and aggregation. In *MobiHoc'03: Proceedings of the Fourth ACM International Symposium on Mobile Ad Hoc Networking & Computing*. ACM, New York, 2003, pp. 165–176.
20. Y. Gao, J. Bu, W. Dong, C. Chen, L. Rao, and X. Liu. Exploiting concurrency for efficient dissemination in wireless sensor networks. In *DCOSS: 2011 International Conference on Distributed Computing in Sensor Systems and Workshops*, Barcelona, Spain, 2011, pp. 1–8.
21. Y. Gao, J. Bu, W. Dong, C. Chen, L. Rao, and X. Liu. Exploiting concurrency for efficient dissemination in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 24(4):691–700, 2013.
22. A. Hagedorn, D. Starobinski, and A. Trachtenberg. Rateless deluge: Over-the-air programming of wireless sensor networks using random linear codes. In *IPSN'08: Proceedings of the Seventh International Conference on Information Processing in Sensor Networks*. IEEE Computer Society, Washington, DC, 2008, pp. 457–466.
23. T. He, B.M. Blum, J.A. Stankovic, and T. Abdelzaher. AIDA: Adaptive application-independent data aggregation in wireless sensor networks. *ACM Transactions on Embedded Computing Systems*, 3(2):426–457, 2004.
24. W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1(4):660–670, October 2002.
25. T. Ho, R. Koetter, M. Medard, D. Karger, and M. Effros. The benefits of coding over routing in a randomized setting. In *Proceedings of the IEEE International Symposium on Information Theory*, Yokohama, Japan, July 2003, pp. 442.
26. S. Jaggi, P. Chou, and K. Jain. Low complexity algebraic multicast network codes. In *Proceedings of the IEEE International Symposium on Information Theory*, Yokohama, Japan, July 2003, pp. 368–368.
27. A. Jain, M. Gruteser, M. Neufeld, and D. Grunwald. Benefits of packet aggregation in ad-hoc wireless network. Technical Report CU-CS-960-03, Department of Computer Science, University of Colorado at Boulder, Boulder, CO, 2003.

28. K. Kalpakis, K. Dasgupta, and P. Namjoshi. Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks. *Computer Networks*, 42(6):697–716, 2003.
29. A. Kamra, V. Misra, J. Feldman, and D. Rubenstein. Growth codes: Maximizing sensor network data persistence. In *SIGCOMM'06: Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. ACM, New York, 2006, pp. 255–266.
30. S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft. XORs in the air: Practical wireless network coding. *IEEE/ACM Transactions on Networking*, 16(3):497–510, 2008.
31. D. Kliazovich and F. Granelli. Packet concatenation at the IP level for performance enhancement in wireless local area networks. *Wireless Networking*, 14(4):519–529, 2008.
32. D. Koutsonikolas, C.-C. Wang, and Y.C. Hu. CCACK: Efficient network coding based opportunistic routing through cumulative coded acknowledgments. In *2010 Proceedings of IEEE INFOCOM*, San Diego, California, US, 2010, pp. 1–9.
33. R. Kumar, M. Wolenetz, B. Agarwalla, J. Shin, P. Hutto, A. Paul, and U. Ramachandran. DFuse: A framework for distributed data fusion. In *SenSys'03: Proceedings of the First International Conference on Embedded Networked Sensor Systems*. ACM, New York, 2003, pp. 114–125.
34. A.R. Lehman. Network coding. PhD thesis, MIT Cambridge, MA. Supervisor-Sudan, Madhu, 2005.
35. A.R. Lehman and E. Lehman. Complexity classification of network information flow problems. In *SODA'04: Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2004, pp. 142–150.
36. M. Li, H. Zhu, Y. Xiao, I. Chlamtac, and B. Prabhakaran. Adaptive frame concatenation mechanisms for QoS in multi-rate wireless ad hoc networks. In *INFOCOM'08: The 27th Conference on Computer Communications*, Phoenix, Arizona, US, April 2008. IEEE, pp. 1112–1120.
37. S.-Y. Li, R. Yeung, and N. Cai. Linear network coding. *IEEE Transactions on Information Theory*, 49(2):371–381, February 2003.
38. T. Li, Q. Ni, D. Malone, D. Leith, Y. Xiao, and T. Turletti. Aggregation with fragment retransmission for very high-speed wlans. *IEEE/ACM Transactions on Networking*, 17(2):591–604, 2009.
39. Y. Lin, B. Li, and B. Liang. CodeOR: Opportunistic routing in wireless mesh networks with segmented network coding. In *ICNP'08: IEEE International Conference on Network Protocols*, Orlando, Florida, US, October 2008, pp. 13–22.
40. S. Lindsey, C. Raghavendra, and K.M. Sivalingam. Data gathering algorithms in sensor networks using energy metrics. *IEEE Transactions on Parallel and Distributed Systems*, 13(9):924–935, 2002.
41. J. Liu, M. Adler, D. Towsley, and C. Zhang. On optimal communication cost for gathering correlated data through wireless sensor networks. In *MobiCom'06: Proceedings of the 12th Annual International Conference on Mobile Computing and Networking*. ACM, New York, 2006, pp. 310–321.
42. G. Lu, B. Krishnamachari, and C.S. Raghavendra. An adaptive energy-efficient and low-latency MAC for tree-based data gathering in sensor networks: Research articles. *Wireless Communications and Mobile Computing*, 7(7):863–875, 2007.
43. K. Lu, D. Wu, Y. Qian, Y. Fang, and R.C. Qiu. Performance of an aggregation-based MAC protocol for high data-rate ultrawideband ad hoc networks. *IEEE Transactions on Vehicular Technology*, 56(1):312–321, January 2007.
44. S. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong. Tag: A tiny aggregation service for ad-hoc sensor networks. *SIGOPS Operating Systems Review*, 36(SI):131–146, 2002.
45. S.R. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong. TinyDB: An acquisitional query processing system for sensor networks. *ACM Transactions on Database Systems*, 30(1):122–173, 2005.
46. R. Niu, P.K. Varshney, and Q. Cheng. Distributed detection in a large wireless sensor network. *Information Fusion*, 7(4):380–394, 2006. (Special Issue on the *Seventh International Conference on Information Fusion—Part I*.)
47. Y.A. Oswald, S. Schmid, and R. Wattenhofer. Tight bounds for delay-sensitive aggregation. In *PODC'08: Proceedings of the 27th ACM Symposium on Principles of Distributed Computing*. ACM, New York, 2008, pp. 195–202.

48. R. Saket and N. Navet. Frame packing algorithms for automotive applications. *Journal of Embedded Computing*, 2(1):93–102, 2006.
49. H.O. Tan and I. Körpeoğlu. Power efficient data gathering and aggregation in wireless sensor networks. *SIGMOD Record*, 32(4):66–71, 2003.
50. R. Tan, G. Xing, B. Liu, and J. Wang. Impact of data fusion on real-time detection in sensor networks. In *RTSS*: *The 30th IEEE Real-Time Systems Symposium*, 2009.
51. R. Tan, G. Xing, X. Liu, J. Yao, and Z. Yuan. Adaptive calibration for fusion-based wireless sensor networks. In *2010 Proceedings of IEEE INFOCOM*, Washington DC, 2010, pp. 1–9.
52. R. Tan, G. Xing, X. Liu, J. Yao, and Z. Yuan. Adaptive calibration for fusion-based cyber-physical systems. *ACM Transactions on Embedded Computing Systems*, 11(4):80:1–80:25, January 2013.
53. R. Tan, G. Xing, Z. Yuan, X. Liu, and J. Yao. System-level calibration for fusion-based wireless sensor networks. In *RTSS*: *2010 IEEE 31st Real-Time Systems Symposium*, Washington DC, 2010, pp. 215–224.
54. R. Tan, G. Xing, Z. Yuan, X. Liu, and J. Yao. System-level calibration for data fusion in wireless sensor networks. *ACM Transactions on Sensor Networks*, 9(3):28:1–28:27, June 2013.
55. S. Thomopoulos, R. Viswanathan, and D. Bougoulias. Optimal decision fusion in multiple sensor systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23(5):644–653, September 1987.
56. P. Varshney. *Distributed Detection and Data Fusion*. Springer-Verlag, New York, Inc., 1996.
57. Q. Xiang, X. Liu, J. Xu, H. Zhang, and J.L. Rittle. When in-network processing meets time: Complexity and effects of joint optimization in wireless sensor networks. In *RTSS: The 30th IEEE Real-Time Systems Symposium*, Washington DC, 2009.
58. Q. Xiang, J. Xu, X. Liu, H. Zhang, and J.L. Rittle. When in-network processing meets time: Complexity and effects of joint optimization in wireless sensor networks. *IEEE Transaction of Mobile Computing* (*TMC*), 10(10):1488–1502, October 2011.
59. Q. Xiang and H. Zhang. QoS-aware in-network processing for mission-critical wireless cyber-physical systems. In *Doctoral Colloquium on 10th ACM Conference on Embedded Networked Sensor Systems* (*SenSys*), Toronto, Canada, 2012.
60. Q. Xiang, H. Zhang, J. Wang, and G. Xing. EENCR: An energy-efficient network coding based routing protocol, Technical Report, WSU-CS-DNC-TR-14-02, Wayne State University, Detroit, MI, 2014.
61. Q. Xiang, H. Zhang, J. Wang, and G. Xing. ProNCP: A proactive network coding based protection protocol, Technical Report, WSU-CS-DNC-TR-14-03, Wayne State University, Detroit, MI, 2014.
62. Y. Xue, Y. Cui, and K. Nahrstedt. Maximizing lifetime for data aggregation in wireless sensor networks. *Mobile Networks and Applications*, 10(6):853–864, 2005.
63. Y. Yao and J. Gehrke. The cougar approach to in-network query processing in sensor networks. *SIGMOD Record*, 31(3):9–18, 2002.
64. Z. Ye, A. Abouzeid, and J. Ai. Optimal policies for distributed data aggregation in wireless sensor networks. In *INFOCOM 2007: 26th IEEE International Conference on Computer Communications*, Alaska, US, May 2007. IEEE, pp. 1676–1684.
65. S. Yoon and C. Shahabi. The clustered aggregation (CAG) technique leveraging spatial and temporal correlations in wireless sensor networks. *ACM Transactions on Sensor Networks*, 3(1):3, 2007.
66. O. Younis and S. Fahmy. Heed: A hybrid, energy efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 3(4):366–379, October–December 2004.
67. B. Yu, J. Gong, and C.-Z. Xu. Catch-up: A data aggregation scheme for vanets. In *VANET'08: Proceedings of the Fifth ACM International Workshop on Vehicular Inter-NETworking*. ACM, New York, 2008, pp. 49–57.
68. Y. Yu, B. Krishnamachari, and V. Prasanna. Energy-latency tradeoffs for data gathering in wireless sensor networks. In *INFOCOM 2004: 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, Hong Kong, China, 1: 255, March 2004.

5

Wireless Sensor Hierarchical Networks

Shancang Li
University of the West of Scotland and University of Bristol

Xinheng (Henry) Wang
University of the West of Scotland

5.1 Introduction 5-1
Overview
5.2 Motivation of the Hierarchal Architecture 5-3
Issues in Designing a WSN Hierarchal Architecture • Principles for Designing Architecture for WSNs • Existing WSN Architectures • Architectures for Specific Areas: Multicore Device WSNs
5.3 Summary 5-15
References 5-16

5.1 Introduction

Wireless sensor networks (WSNs) have been believed to be the edge component of next-generation networks: Internet of Things (IoT) [1]. The hierarchical architecture is very popular in designing WSNs, which provides WSNs with scalability, ad hoc, and fault tolerance and enables easy integration of applications into WSNs. Figure 5.1 shows an example of basic hierarchical WSN, which roughly consists of three layers: sensing layer, networking layer, and application layer [1–3].

5.1.1 Overview

Like many advanced technologies, the WSNs originated in military applications. It was first developed in sound surveillance system (SOSUS) and air home warning and control system (AWACS) to detect and track submarines during the cold war [1,4]. Actually, the sensing technologies developed are still in use in many applications today such as undersea wildlife and volcanic activity monitoring and environment monitoring. In the 1960s and 1970s, the occurrence of networking technology motivated the research on distributed sensor networks (DSNs), and in the 1980s, the attempt of wirelessly connected sensor networks has been made [4,5]. In academia, many research interests have been attracted by DSNs, including CMU, MIT Lincoin Labs, and the WSNs technology soon attracted many attentions in academia and civilian scientific research [4]. In 1994, Professor Willam J. Kaiser submitted a research proposal, *low power wireless integrated microsensors*, to DARPA that promoted the attentions from military to industrial applications [5]. In 2001, the first European projects on WSNs were launched under the support of FP7 [6]. Since then, a number of projects were financed by EC aimed to develop key technologies for WSNs: communication protocols and architectural and technological solutions for WSNs, such as WISENTS [7], e-Sense [8], CRUSE [9], and CONET [10]. A lot of research efforts have

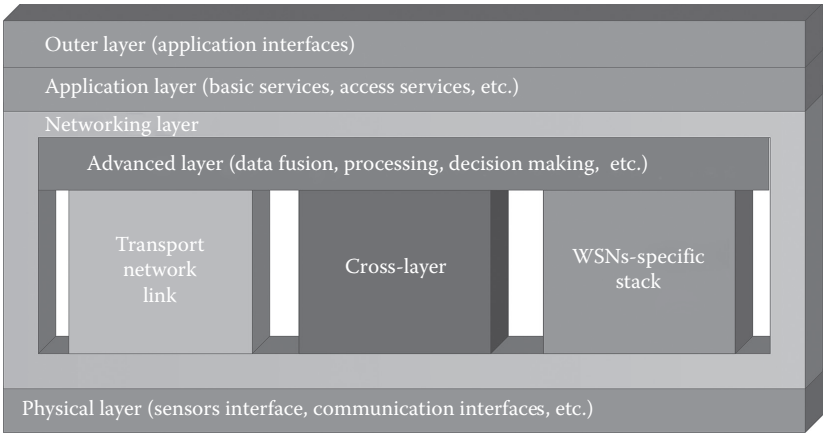


FIGURE 5.1 Hierarchical WSNs.

been done to develop the ad hoc networking technologies such as wireless low-energy communications, routing, media access control (MAC), size, cost, power consumption, and intelligent sensors. A number of WSN technologies have been developed which make it possible to build large-scale WSNs. Examples of these initiatives include [1–3,11,12]

- UCLA Wireless Integrated Network Sensors (1993)
- University of California at Berkeley PicoRadio program (1999)
- Micro Adaptive Multi-domain Power Aware Sensors program at MIT (2000)
- NSA Sensor Webs (2001)
- ZigBee Alliance (2002)
- Center for Embedded Network Sensing (2002)
- Radio-frequency identification (RFID) (2005)
- 6 LowPAN (2007)
- ANT short wireless communication (2012)
- Bluetooth lower energy (BLE) (2012)

Other technologies that are used in WSNs include ultra-wide band (UWB), Impulse Radio-UWB (IR-UWB) [13,14], and Z-Wave.

The goal of these research activities and standards is to enable high-volume deployment of low-cost and energy-efficient WSNs in industrial that can simplify development and maintenance tasks. Meanwhile, many research efforts have been focused on reducing the deployment costs on the following areas [2,15]:

- Development of intelligent sensors
- CMOS-based semiconductor devices
- Light networking protocols
- Energy-efficient technology

The culmination of these efforts is the development of intelligent WSNs, and a number of new concepts such as *smart city and smart planet* have been proposed based on these intelligent WSNs. Sensors are the heart of WSNs and much advancement has been made in the past decade. A number of low-cost and power-efficient hardware device required by WSN applications have been developed, such as system on chip (SoC) and BLE, which provide the application processing, networking, and wireless communication capabilities simultaneously [3,6,15].

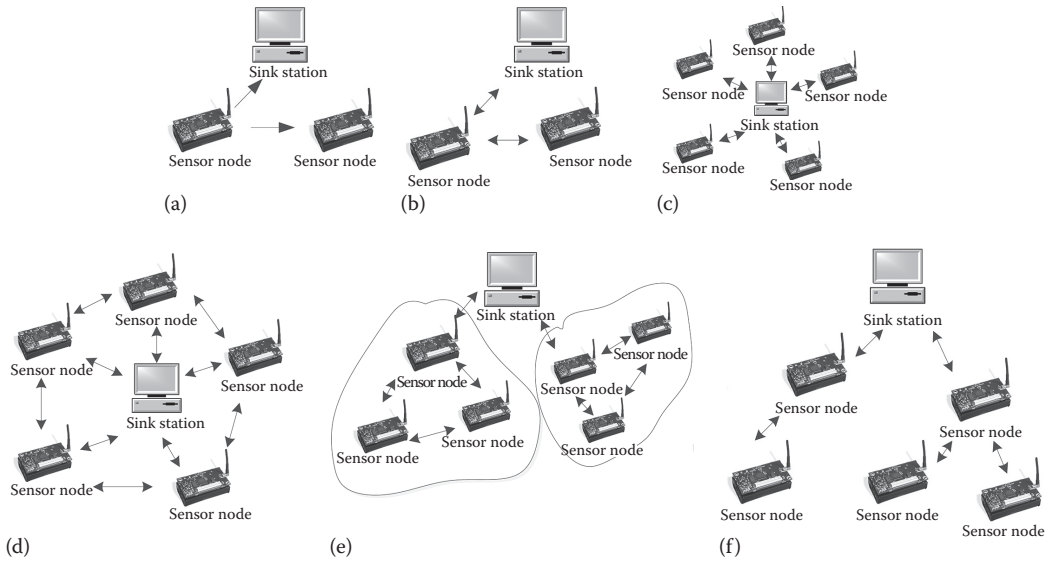


FIGURE 5.2 Topologies of WSNs. (a) One way, (b) bidirectional, (c) start, (d) mesh, (e) cluster, and (f) tree.

The limitation of transmission range of nodes requires WSNs be an ad hoc network (multihop routing, self-organizing, and self-management) [1–3,16–18]. In general, the WSN topologies roughly fall into six categories, as shown in Figure 5.2:

- One way
- Bidirectional
- Start
- Mesh networks
- Cluster
- Tree

The one-way mode only provides one-way communication links. It is commonly used in applications such as tire pressure monitoring system, garage door operator, TV remote control, Bluetooth mouse, and wireless keyboard. The widely used IEEE 802.15.4 in WSNs also supports topologies, such as *bidirectional, mesh, cluster, and tree topologies*, which are described in the ZigBee Alliance specification [6].

5.2 Motivation of the Hierarchal Architecture

To support a wide range of applications of WSNs, it is necessary to well organize individual nodes to form a meaningful network. The conventional ad hoc networks and the nonstandard WSNs are widely used as a basic architecture of a WSN.

A standard architecture should be able to powerfully enlarge the range of applications of WSNs. Unfortunately, it is still an open issue: a number of WSNs have been developed, and there are still no standard architecture and protocol stacks that can be used for different application scenarios. An ideal architecture is necessary that can be rapidly deployed in different application scenarios. Although many commonly used sensor nodes might be involved, the architectures used in different scenarios might be diverse. The main reasons are multicaused [19]:

1. A WSN may be designed for specific applications or scenarios where the performances are optimized. This optimization might cross the layers of architecture of the WSN.

2. A WSN is normally designed based on components that are suitable for specific application scenarios; when the application scenarios changed, the incompatibility might affect the performance of the system.
3. Separately developing may cause multiple of WSNs systems outputs or information processing, and the interaction with existing networks such as the Internet might be different, meanwhile decrease the reusability of WSNs.

The traditional hierarchy network architectures, that is, open systems interconnection model (OSI) or transmission control protocol (TCP)/internet protocol (IP), are based on its modularization, which enables the network to well integrate the protocols on different layers and hence extend the functionalities on one layer without affecting other layers [19]. In existing hierarchy networks, that is, ad hoc network, the upper layer can call services lie at lower layers. By doing this, the network can build more powerful services than at its own layer. In this architecture, the intermediate nodes are unable to separately process packages at different layers and only can simply store/forward packets. In this case, WSNs cannot directly exploit the architecture of the Internet, and a new hierarchal architecture becomes necessary [2,3,6,7]:

- WSNs might be deployed in heavy industrial application scenarios, which make it possible to get power supply or recharge. The limitation on power supply requires that all layers in a WSN must take into account the power-efficient issue, and an optimization scheme must be applied for different layers to further optimize energy consumption in a whole system review.
- The communication and computation ability limit sensor node from heavy communication load and computation load.
- For some applications, the large-scale multihop WSNs, an optimized architecture can reduce the probability of occurrence of *dead node*, where it might be impossible to replace or repair a node.
- In general, the WSNs are applied as an edge network of IoT, which focuses on sensing or collecting specific types of information. The commonly used store/forward mechanism is incapable to process multiple sources information in WSNs, where the intermediate nodes are required to be able to perform data fusing to further reduce the communication load and energy consumption. The data fusion mechanism should be able to process information depending on the content of information, which is different with traditional network hierarchy architecture. This makes it impossible to directly apply IP routing protocols in WSNs.
- WSNs are designed to intelligently process data and self-organize the sensor nodes, which require the network architecture to self-maintain the network based on adaptive cycle-loop style. This requires the borders between layers to be destroyed.

In a word, the traditional network architectures and protocols are no longer suitable for WSNs, which might cause difficulties in coordinating operations from different layers. In order to overcome the resource limitations or to further widen the range of application scenarios, designing new hierarchal architecture for WSNs is a challenge and an urgent task.

5.2.1 Issues in Designing a WSN Hierarchal Architecture

As mentioned earlier, the new features of WSNs require specific mechanisms to support the functionalities; most of these mechanisms are difficult to implement based on traditional networking architectures [20,21]. Even so, the performance might be decreased due to the incompatibility between new features and the old network architectures. The new features and services of WSNs bring us new problems in designing network architecture for WSNs such as the following: Shall we take the hierarchy structure? How to define the functionalities of each layer; how to deal with the specific needs of WSNs, such as packets aggregations, data inner processing, data query, distributed data storage; or how to define the

abstracted definition in functionality layers? Can the IP-based addressing and routing protocols be used in WSNs? To answer these questions, we summarized the following issues to illustrate the problems faced by WSNs in architecture design.

5.2.1.1 Resources Allocation and Management

In most monitoring applications using WSNs, the power supply, network bandwidth, computation capability, and storage are constantly scarce resources. An effective method to allocate and use these resources can be very helpful to prolong the lifetime of WSNs. Meanwhile, improving the network transmission, computation capabilities, and storage becomes a big concern in designing a WSN.

An effective power management scheme is necessary in architecture design for WSNs, which is involved in all layers of WSNs. Recently, a lot of research efforts have moved to the low-energy hardware (such as low-energy SoC), effective MAC protocols, and routing protocols [22–25]. Since the idle states of each layer are not well synchronous, it is required to manage the power distribution between layers. For example, a management module in MAC should be provided to swap the states of RF module, and in routing layer, the energy scheme should be able to effectively schedule the network flows to avoid congestion.

To prolong the time of use, many research efforts have focused on the energy-efficient issue. Meanwhile, the limitation of bandwidth has caused difficulties in data-streaming-based application of WSNs. Fortunately, the new low-energy communication technologies, such as BLE and ANT, have significantly widened the range of applications for WSNs [24]. However, the implementation of large-scale WSNs is still difficult due to the bandwidth and management. The incensement of information dries up the bandwidth resources. As a result, using data fusion technology to reduce the communication load and enhancing the usage of bandwidth are two important topics in designing the architecture of WSNs.

5.2.1.2 Protocols

In developing a WSN, the developers designing for different layers collaboratively work for a general optimization goal (such as reduce the energy consumption, improve the transmission rate, and decrease the BER), which coupling between layers: the upper layers have to know the QoS of lower layers [26,27]. Meanwhile, the performance of protocols at lower layers must be guided by instructions from upper layers. This totally departs the structure of traditional networks (such as ISO and TCP/IP) and will probably cause extra complexity and make the system modular with a low exchangeability. For example, the data link layer might need the capacity, power consumption, signal channel, and transmission directions of physical layer; however, the network might instruct the lower layers work by matching the standard transmission.

Although this cross-layer hierarchal architecture design can cause system complexity, it is the most effective method to improve the whole system performance so far. To implement the cross-layer control, comprehensive interfaces must be designed to provide well understanding between layers.

5.2.1.3 In-Network Data Processing

In the traditional network architectures (such as TCP/IP and OSI), the data are processed at the edge of the networks [23], which significantly simplified the architecture of network and gained the scalability of network. However, in WSN, the in-network data processing technology is able to house data transformation function to consolidate the data streaming from the data sources before forwarding the processed stream to the sink nodes (receivers). It can reduce the communication load and hence make less resource consumption. On the other hand, repackaging data at intermediate nodes can reduce the network congestion. For example, when congestion is found, the high rate of in-network data fusion can reduce the congestion by finding another way or scheduling the data fusion mechanisms.

In designing the hierarchal architecture for WSNs, the intermediate nodes must be designed to be able to effectively filter and fuse data, which is particularly useful in dynamic monitoring environment. The in-network data processing mechanism might increase the complexity of architecture design, but it is still worth to implement the new features of WSNs.

5.2.1.4 Data Integrity

It has been widely used to acquire information in the emerging IoT in the coming era of *big data*. Recently, a number of *smart* stuff have been proposed based on WSNs, RFID, NFC, and other new network technologies. In WSNs, a wide range of sensors can be used, from temperature sensor, color sensors, RFID, to intelligent biosensors; most of them work in limited conditions [26]. Therefore, it is difficult to real-time calibrate. In designing the hierarchal architecture, one must take into account the integrity of information. The data integrity in WSNs has been researched as a security topic.

In designing a WSN, the architecture should be able to provide the so-called *data purification* mechanism to prevent the transmission from integrity. Particularly, data purification requires the WSN to aggregate and regulate the integrity of data. When invalid data are found at some sensor nodes, the WSN should be able to automatically correct the operations of nodes according to the context. In addition, the WSN must perform the information security check to ensure that all instructions are correctly dispatched.

5.2.1.5 Security and Privacy

The traditional network architectures are very widely used today, which was developed carefully, but there are still a number of serious security flaws inherent in the protocols because hosts rely on IP source address for authentication [22,28]. Thus, security has become one of the most intractable problems facing the modern Internet. Due to the lack of physical layer protection, the resource constraints in the sensor nodes and multihop communications in open wireless channel make the security of WSNs an even greater challenge. In the design of new WSNs architectures, security protection mechanisms should be used to provide protection against possible attacks (such as wireless communication channel attacks and information theft easier transmission) from various layers [29].

These mechanisms must be designed from the bottom up through the various layers of the architecture of WSNs, and the following protections should be provided [30]: (1) data confidentiality (a sensor node should not reveal its data to the neighboring nodes and should be able to establish and maintain confidentiality); (2) data integrity (it ensures that no message can be altered by intermediate nodes); (3) data availability (it ensures that the services provided by a WSN is always available even in the presence of attacks); (4) authentication (it ensures that the information sources are legal); and (5) data freshness (it means that the architecture should be able to provide recent data and old information should not be replayed by attackers).

5.2.1.6 Network-Layer Diversification

The widely used networking-layer protocols are IP based, where each terminal node can achieve end-to-end communication via IP layers. However, in WSNs, many new communication interfaces and intelligent sensors will be required to enable efficient information exchange across diverse network layers and nodes [31]. It will inevitably lead to the diversification of its network protocols, and the sample IP-based end-to-end communication is no longer applicable. For example, in a subnet of WSN, the collaborative actions at nodes and the data broadcasts for data aggregations/distributions should be performed with similar style that is defined by the gateway [32]. The hierarchal architecture of WSNs should be able to deal with the diverse network protocols and shield the internal to the external network, while providing the means to achieve seamless information exchange with external networks.

5.2.1.7 Standardization of WSNs

The standardization is a key issue in designing architecture for WSNs, which can reduce the development and maintenance costs, increase interoperability, and speed up their utilization in markets.

The standardization of WSNs involves hardware design, communication protocols, and system integration. In fact, there will be no *single* standard protocol for WSNs, but good compatibilities can be helpful to well integrate WSNs with other systems. In certain aspects of communication, the IEEE 802.15.4 is proposed for short-range communications in WSNs to provide applications with flexible throughput and QoS requirements. Most of the recently developed WSNs platforms comply with the IEEE 802.15.4 standard. Meanwhile, several standardization attempts such as ZigBee, WINA, SP100.11a, and 6LoWPAN are expected to improve the compatibilities between WSNs and existing networks. On the other hand, the standardization attempts, such as software architecture and data aggregations, are also forming.

5.2.2 Principles for Designing Architecture for WSNs

Based on the previous analysis, it should be noticed that it is critical to design the hierarchal architecture of WSNs to optimize the resources and simplify the architecture to further improve the performance of WSNs. A challenging task is how to build the complicate hierarchal architecture for WSNs with simple existing protocol stacks. In addition, hierarchal architecture designing for large-scale WSNs, quasistatic topologies, cross-layer design, inner-data processing, address/names schemes, and routing protocols are also important issues. In the following subsections, we will discuss the details of these issues and give an overview of the basic principles in hierarchal architecture design of WSNs.

5.2.2.1 Layered-Architecture and Modularity

Modular design is the basic principle to build a complex system, but each module can be independently created and then used in different architectures [33]. To meet the requirements of diverse applications, the WSN protocol family could be quite complex, in which the modular design can be used to well implement the following issues that must be solved: (1) standardized interfaces between the same type of protocol packages, (2) module reuse in different kinds of protocols, and (3) application optimization configuration for different protocols.

5.2.2.2 Cross-Layer Optimization

In traditional OSI architecture, each layer is responsible for specific functionalities of a task, which makes it easier to design efficient protocol due to the strict, clear boundaries between the layers [27,33,34]. However, in resource-limited WSNs, a task could be implemented through cross-layer design in which specific optimization goals (such as efficient energy consumption) for the whole network could be achieved. This could be implemented by modifying the protocols at each layer, which violates the OSI model [34]. The main idea of cross-layer design is to exploit information from multiple layers to jointly optimize performance of each layer to achieve specific optimal goals. Figure 5.3 shows an example of a cross-layer design in WSNs.

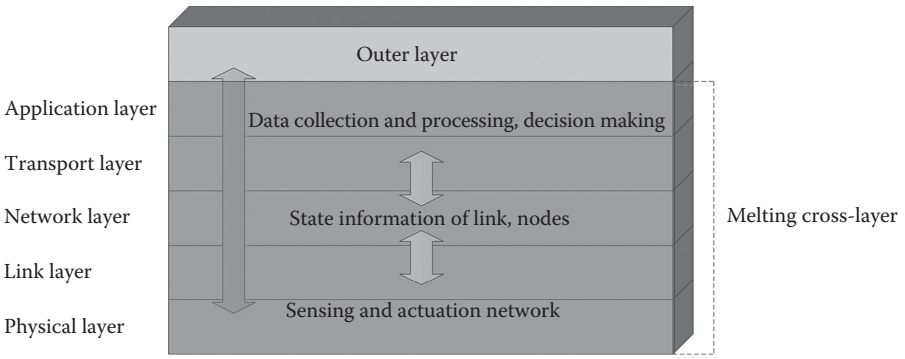


FIGURE 5.3 Cross-layer design.

In cross-layer design, the new, nonstandard interfaces between layers merge functionalities to jointly calibrate optimization goals, in which the following issues should be considered:

- Unintended cross-layer interaction (the new created interactions between layers could lead to unforeseen decencies)
- Stability, for a given system
- Long-term sustainability
- Position and time synchronization (a well-designed cross-layer architecture should be able to provide the position and time synchronization at different layers)

In addition, switching between different layers becomes very necessary in cross-layer hierarchal architecture. For example, to achieve reliable paths at the network layer, the routing protocols may need to know the link quality of physical layers.

5.2.2.3 Configurable Applications

WSNs are invented for specific applications and it can be said to be application-oriented networks, in which the requirements of applications might be very different: some applications may require the reliability of data transmission, and some can tolerate a small amount of packets loss, while reliable transmission requires more controllable structures and data transmission (such as data redundancy, the command response [ACK]) and hence causes more energy consumption. Therefore, the applications should be able to independently configure the characteristics of the underlying protocols according to their needs, and trade-offs between various goals could be achieved [35].

5.2.2.4 In-Network Data Processing

To achieve optimal system performance, a WSN requires each node to process collected data. For example, data integration data flows need to be compared and integrated to reduce network traffic and save energy consumption. In some cases, it is difficult to determine whether to send the entire raw data to a base station. Furthermore, in view of monitoring, the data changes in adjacent space and time may be very small. In this case, an in-network data processing scheme could be very useful to aggregation data to improve the data transmission efficiency and reduce the communication load. In general, a WSN has relatively sloppy time traffic in long-term applications (most of the time, the node is in the sleep state). Therefore, the nodes have to forward the collected data immediately, but can process for reducing transmission redundancy [28].

5.2.2.5 Adaptability

In WSNs, the robustness design of hierarchal architecture can guarantee the ability of WSNs to resist change without adapting its initial stable configuration. A WSN must be robust and resilient to failures such as *hardware failure, links failure, network failure*, and temporal variation of the wireless channels and interfaces [33]. An adaptive hierarchal architecture can automatically adapt to changes in the external environment, network topology, and quality of the radio channel. In addition, the changes must immediately be made for application or management system, especially for multitasking peer network and converged network operations. As a result, in architecture design for WSNs to optimize the limited resources, a lot of adaptive scheme has been proposed, such as bandwidth-adaptive routing scheme, energy-adaptive MAC protocol, adaptive communication scheme, and data-driven WSNs.

5.2.2.6 Distributed Architectures

Compared with tradition networks, collaboration within the WSNs is very important, which is needed by both infrastructures and applications. In monitoring applications, if adjacent nodes detected an event, a collaboration scheme is very helpful to make a decision before sending the results to base station or run some data fusion tasks.

5.2.2.7 Data-Centric Architecture

Most traditional networks are address centric or nodes centric, in which the data sources and the data receiver identifier (IP address) are very important. However, in WSNs, the content of data is more important. The sources of data in WSNs may not be as important as the content of the data. The key to data-centric architecture design is to separate data from behaviors. With data-centric approach, a WSN can be built from independent components with loosely coupled behaviors. The data changes in WSNs drive the integrations between components.

5.2.3 Existing WSN Architectures

In 2003, the IEEE standard for low-rate wireless personal area networks (LR-WPANs) was first published, which has been widely accepted as the standard for WSNs. Since then, IEEE has created a number of substandards. The ZigBee Group set up IEEE 802.15.4b for WSNs communication. However, the architecture of WSNs is still affected with a lack of a uniform standard due to most WSNs architectures are from its prototype systems.

Most early WSNs architectures are designed by modifications of the architecture of the Internet, which use a five-layer protocol stack [20,21]: *application layer, transmission layer, network layer, data link layer, and physical layer*. With the advances of WSNs, some basic elements are constantly being added to the stack, which forms the current complex and diverse architectures.

The existing WSN architectures can be roughly categorized into two classes depending on the application areas: (1) the general-purpose architecture and (2) design for particular field applications (such as underwater WSNs) The general-purpose architecture considers all aspects of network functions and is normally designed in modular where the components could be added or removed depending on the requirements of application. The latter architecture is designed for specific areas, which has a limited flexibility.

Currently, a number of typical WSN prototypes are available that contains proven architectures, which are very helpful to standardize architecture for research. However, these systems tend to be practical for a special task, and major modifications are needed when the task changes. In the following subsections, we will introduce the typical WSN architectures that have been successfully used in specific applications.

5.2.3.1 VigilNet

VigilNet WSN system [36] has been successfully designed, demonstrated, and delivered in battlefield monitoring applications, which applies multiple existing WSN nodes for different tasks, including MICA2, MICA2DOT, XMS2, and XSM. VigilNet supports large-scale WSN and can be used in long-term military applications and proved to be able to provide good surveillance quality parameters and network lifetime [37]. Figure 5.4 shows the details of VigilNet architecture.

VigilNet was designed in a systematic approach and includes a number of subsystems: *sensing subsystem, communication subsystem, energy management subsystem, reconfiguration subsystem, content-aware subsystem (synchronization, positioning, and packing), application subsystem, and debug subsystem*. In VigilNet, the energy efficiency is considered by all protocols, which makes it possible to build a large-scale WSN with longer lifetime enough for long-term monitoring applications. All these subsystems are designed in modular and without obvious dependencies.

From the point of view of architecture design, VigilNet is a relatively simple system with the following flaws: (1) it does not contain a cross-level intermodule design, in which the energy management subsystem independently applies a multidimensional subregional scheme, which limits the energy management lies at node level and makes it difficult to further optimize the energy consumption between subsystems; (2) it lacks a security protection scheme; and (3) it does not consider the data flow optimization within the network, such as data fusion or things like this.

Essentially, VigilNet is still built up based on the idea of traditional network architectures. It is actually implemented that the WSN functions as applications upon the traditional network hierarchy.

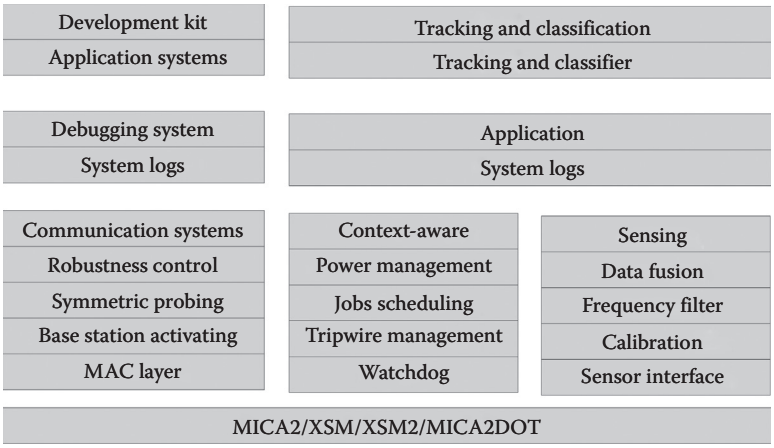


FIGURE 5.4 Hierarchy of VigilNet.

5.2.3.2 INSIGHT

INSIGHT [38] is developed to monitor the living conditions such as humidity, temperature, light intensity, and other parameters. As designed for applications within a relatively small range, the architecture of INSIGHT is very simple. The INSIGHT has the following features: long life cycle, remote query and configure, easy to deploy, and high reliability. Figure 5.5 shows the basic structure of INSIGHT.

To further reduce the energy consumption and make it easier to implement, INSIGHT is designed to keep a simple architecture and only support a single-hop star network structure. Meanwhile, in order to make INSIGHT accessible online, a web server and SQL server are configured at base station to facilitate remote query. For the reliability of INSIGHT, a watchdog timer to restart the failed node is designed at the node as well as at the base station, but the latter is designed to restart web server and database.

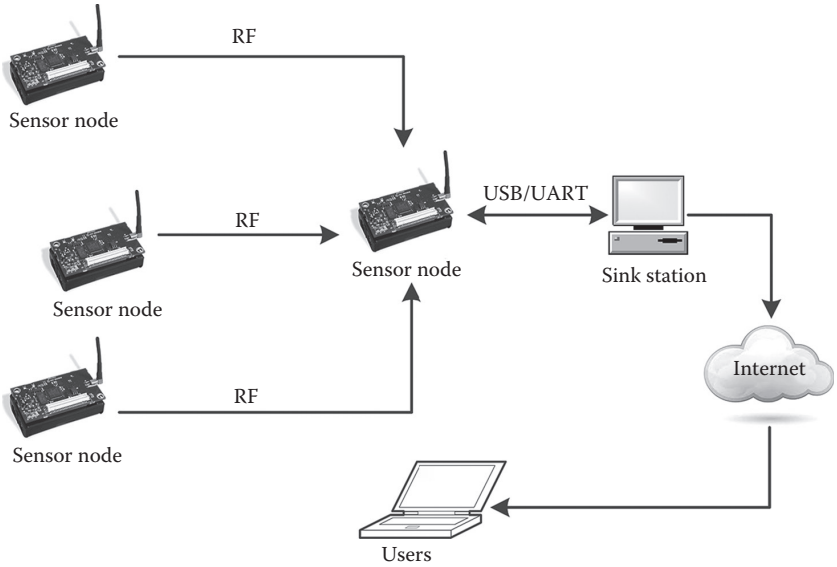


FIGURE 5.5 The architecture of INSIGHT.

5.2.3.3 MANNA Management System

MANNA [39,40] proposed a set of network management for WSN architectures. In traditional networks, the network management and service management are being run separately. However, in WSNs, all characteristics of the network elements, services, configurations, sensing, processing, and communication are dependent on the configuration of the networks [41]. For example, the failure of a node in traditional networks could be fixed manually, but for WSNs, it is impossible. A WSN is a self-organizing, self-healing, and self-configuring wireless network whose topology might be changed in real time. It requires failures in the networks be automatically repaired when it happens. From the viewpoint of management, this self-healing characteristic needs to be included in the architecture design of WSNs.

Functionally, the MANNA architecture includes three management features [40]: functional areas, management levels, and WSN functionalities. As a result, MANNA is designed based on the three-level functional architecture.

In MANNA, three functional sets are defined from top to bottom: *services set, functions set, and WSN models set*. At the top layer, a service consists of multiple functionalities set. In general, a service is called by WSN model(s) according to the configuration of the application. In [42], an example, namely, *design and evaluation of fire detection systems*, is proposed that demonstrates the application of MANNA. The system is able to provide *self*-management services: the self-organization, self-configuration, self-service, and self-maintenance. It illustrated a layered heterogeneous WSN design and analysis of distributed self-management programs.

5.2.3.4 Narrow-Waist Sensor Net Protocol

It is obvious that the rapid development of the Internet is largely dependent on the compatibility of IP protocol [32]. The IP protocol supports a variety of upper-layer protocols, while also running on a variety of underlying protocols. Similarly, a *narrow-waist* structure could be very helpful to widen the range of applications of WSNs, which can well facilitate cross-platform protocols or component transplant in WSNs. Therefore, the key issue is to fully consider the characteristics of various needs in WSNs and design the *waist protocol* to better bridge the upper and bottom protocols.

In [32], David reported that a *waist* protocol could be built above the data link layer of WSNs (Figure 5.6). It could be a best-effort (BE) single-hop broadcast protocol with a set of interface support protocols at upper layer and compatible with possible bottom-layer protocols. The reason why it locates at data link layer rather than at network layer like the Internet is mainly because different applications require different embodiments of combinations of protocols; meanwhile, the addressing and routing schemes are very different.

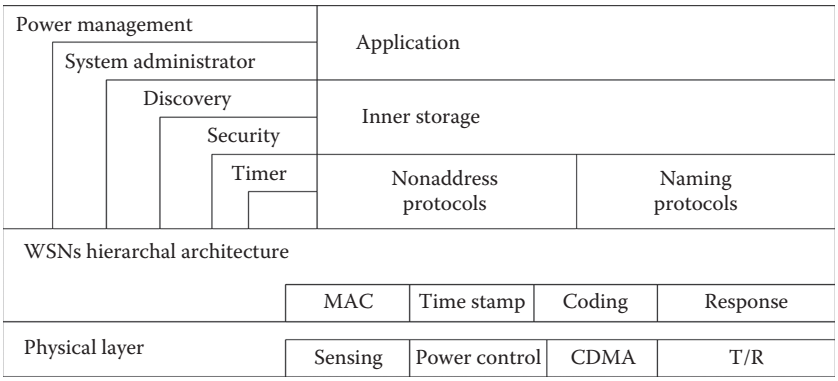


FIGURE 5.6 Narrow-waist architecture.

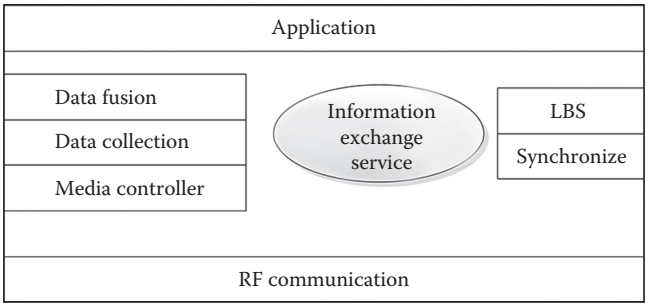


FIGURE 5.7 Communication protocols with IES structure.

5.2.3.5 Information Exchange Service

Unlike traditional networks, the characteristics of WSNs require different communication protocol stacks [43]. Therefore, in designing the communication protocol stacks for WSNs, one must consider the cross-level functional optimization, adaptability, addressing, and data integration. Figure 5.7 shows an example of WSN communication protocols. It is a simple stack but provides information exchange service (IES) dependent from the communication stack. In addition, it also enables node self-configuration.

5.2.3.6 Agent-Based Architecture

To simplify the implementation and maintenance of applications, agent-based architectures are proposed [44]. Functionally, the agent-based architecture could be classified into four classes: (1) Interface agent is responsible for receiving the query messages from the users and returning the structure of message to the users; (2) regional agent is responsible for receiving regional query message and forwarding it to cluster agents; (3) cluster agent is responsible for dispatching the query message to cluster members and processing data fusion; (4) query agent is responsible for performing query request at nodes. Figure 5.8 shows an example of agent-based WSN architecture.

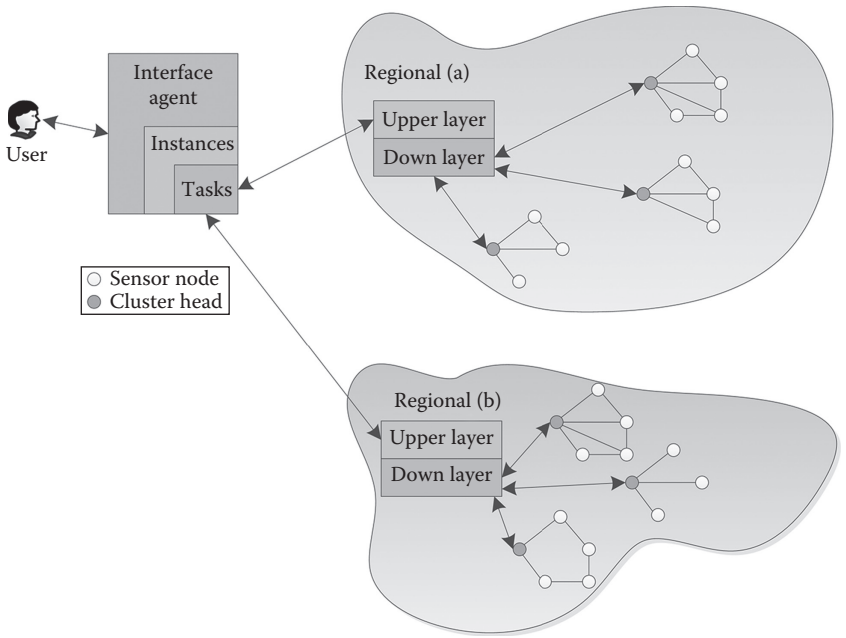


FIGURE 5.8 Agent-based hierarchal architecture.

The four-agent architecture can significantly reduce the complexity of application design and maintenance of WSNs and hence brings a lot of convenience for the implementation of WSNs. However, it also brings some additional technical challenges: *cluster algorithms, data fusion algorithms, query processing, etc.*

The agent-based architecture could improve the flexibility of a WSN to accommodate variability (such as the application changes, failures of nodes, and environmental impact) to provide network scalability. The agent-based architecture is able to adaptively add or delete agents according to the requirements of application. Meanwhile, it can automatically expand and contract according to the range of cover.

5.2.4 Architectures for Specific Areas: Multicore Device WSNs

Recently, the advances of multicore technology have motivated the development of embedded WSNs, which enable the WSNs to cope with heavy information processing (e.g., audio and video) but have low energy consumption. The multicore technology can significantly improve the capability of intelligent sensor node (computation, communication, storage, low energy consumption) and can simplify the structure of WSNs. The symmetric multiprocessors (SMPs) and tiled many-core architectures (TMAs) could also be used in designing WSN nodes for some specific applications [44–46].

The multicore devices could be interconnected as well as single-core sensors, with which the application range can be significantly widened: (1) Real-time WSNs, such as WSNs with smart wireless cameras, video-/image-centric WSNs, and scalable streaming WSNs. (2) Satellite-based WSNs—by providing connections with satellites, the multicore devices can dramatically reduce the costs of traditional missions. The GPS-related sensors can enable WSNs provide location-based services. (3) Reliable industrial WSNs—the multicore sensor nodes could provide WSNs with fault tolerance.

5.2.4.1 Multitask WSNs

In most existing WSN architectures, nodes are designed to perform the same task [47,48]. But in many cases, WSNs may need to perform multiple tasks simultaneously. This requires the WSN architecture to be capable of supporting multiple tasks, and nodes could work on different tasks. Meanwhile, each node can easily switch from one task to another when needed. Figure 5.9 shows an example of multiple-task architecture for WSNs.

In multitask WSN architectures, the *scope state* layers store the *scope state* of nodes, and the routing layer is designed to charge control message (such as create and delete) [48].

As an edge network of the next-generation network, WSNs support multitasks, and it could be reinforced from the following two aspects: (1) for multiple requests, it could be further optimized with refine grain; (2) inner-network division, fusion, and collaborations.

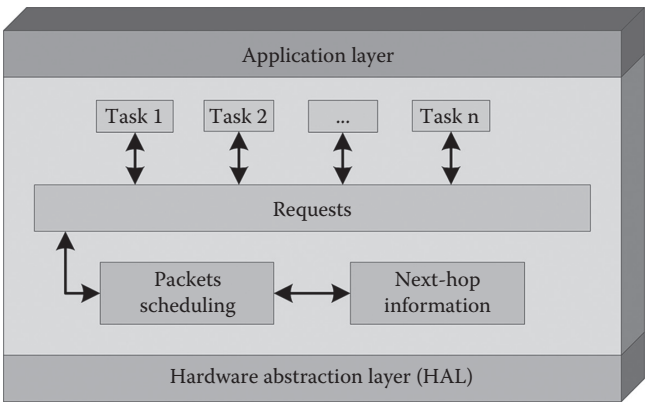


FIGURE 5.9 Multitask architecture.

5.2.4.2 Self-Securing and Self-Healing Architecture

The TV show ConWe believes that for most of the WSNs architecture, the robust guarantee and self-healing schemes (security, fault tolerant, etc.) are not enough for practical applications and lack of error models for global robustness and self-recovery capability [49,50]. Integrating self-securing and self-healing into WSNs architectures can improve not only the robustness of WSNs but also the self-learning capability to cope with unknown errors. This kind of architectures has the following characteristics [49,50]:

1. *Antivulnerability*: It means that the failure of component, nodes, or links cannot cause the breakdown of the whole network and the effects should be reduced as much as possible.
2. *Autonomy*: Without the drive from system, the architecture should be able to dependently cope with errors.
3. *Multilayer protection*: Protections should be provided to protect operations at different layers.
4. *Compactness*: The system must have a resource consumption limitation to guarantee the resource allocations.
5. *Adaptability*: Similar to the immune system, the architecture should be able to analysis and store unknown system error and hence to learn more about it.

5.2.4.3 Abstract Region

Energy consumption and storage capability of sensor nodes are two key resources of WSNs [18,36]. Recent advances in WSN architecture design often require these limitations to extend the lifetime of WSNs be jointly considered. The data fusion is able to extend the lifetime and improve the coverage of WSNs by reducing the traffic load and conserving energy of the sensors [51]. The data fusion involves three aspects: (1) the sensing range of a node, (2) how a node fuses multiple reports into a single one, and (3) the data fusion architecture. In general, a WSN for data processing is often able to provide some local resource primitives that can greatly simplify the design of mechanisms and applications. The *abstract region* proposed a set of primitive operations for the local area, which involve neighbor discovery, neighbor queries, data sharing, and data integration. Abstract region is designed based on a variety of different shaped WSN build numbers, which is similar to clustering but more flexible and efficient. Meanwhile, the abstract model allows the applications to be able to configure the bottom parameters according to the requirements to obtain a global optimization.

5.2.4.4 Heterogeneous Architectures

The intelligent devices interconnected with WSNs would increase the complexity of WSNs. A heterogeneous architecture (see Figure 5.10) could leverage capability (computing, processing, storage, etc.) to meet performance and energy requirements of applications. It is able to integrate powerful sensor nodes (computers) as well as single MCU sensors, with which applications arranged could be widened.

5.2.4.5 Integrated Architecture

Integrating with emerging technologies is a trend of WSNs. Recently, the smart wearable systems (SWS), RFID devices, and smartphones are reported to be integrated with WSNs for intelligent services, such as intelligent health-care systems [52]. This integrated architecture (see Figure 5.11) is multiparametric and involves multiple wireless sensing platforms based on the uPnP framework, which is able to easily interconnect third part devices (sensor, RFID Tags, etc.). The wireless body sensor networks (WBSNs), body area sensor network (WBAN), intelligent home network, and near-me area networks (NANs) are the successful examples of this architecture. In this architecture, the data processing on application layer is challenging that focused on real-time data representation, data source control, analyses, storage, etc.

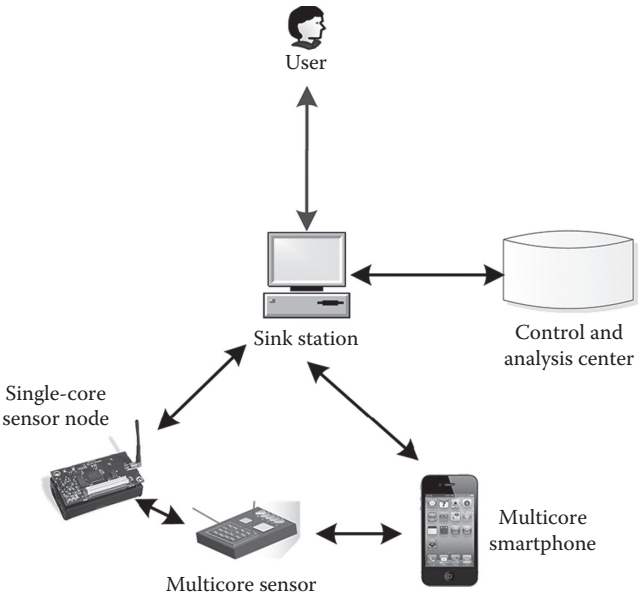


FIGURE 5.10 Multicore architecture.

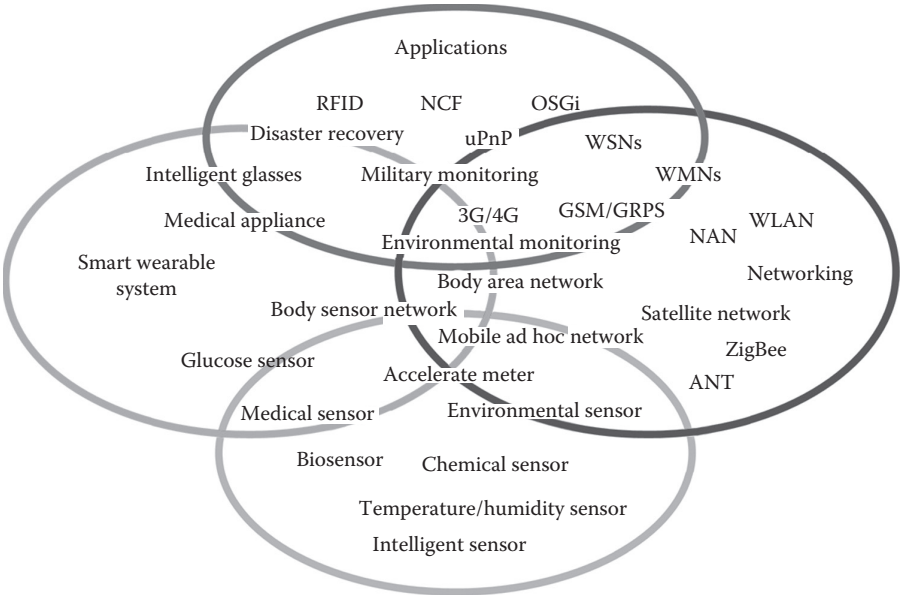


FIGURE 5.11 Integrated architecture.

5.3 Summary

In summary, the diversity of applications causes the design and implementation of WSN architectures that is a complex task. As an edge network of next-generation network, the WSN is bound to be integrated in the Internet. Based on the experience of previous studies, we propose the following suggestions for the future design of WSN architectures.

First, the communication stack is the key in designing of WSN architecture. The nature of WSN communication architecture determines it is bound to be different with existing network architecture, and the new features of WSNs must be considered throughout the entire hierarchy structure (cross-layer). Therefore, we believe that its structure is a multidimensional structure, rather than a linear structure. For example, the energy management must have corresponding implementation at all layers. Similarly, data security at all layers must need appropriate treatment. At a dimension, the communication between layers is implemented by interfaces and events. Meanwhile, it should also provide messaging and multidimensional function calls inside and provide as much available simple interfaces to the application.

Second, the WSN architectures must be designed with high configurability, which could significantly improve the universality for all kinds of applications. For example, the WSN architecture should be designed to be able to be reprogrammed via a wireless channel. The applications may configure the services and resources as well as perform executable tasks.

Third, in order to further improve the application range of WSNs, it must be designed to provide flexible debugging capabilities and management systems. Debug interfaces enable the user to well monitor the implementation and deployment of the WSNs as well as handle network errors and some maintenance functions. To improve the security protection, the network management and application services should be separately designed. The network management module provides network automatic maintenances, such as self-organization, error handlings, and information updating; meanwhile, the applicant services are designed to provide user application interfaces.

References

1. Li S. et al. 2013. Compressed sensing signal and data acquisition in wireless sensor networks and internet of things. *IEEE Transactions on Industrial Informatics*, 9(4), 2177–2186.
2. Matin M.A. *Wireless Sensor Networks: Technology and Protocols*. InTech, September 6, 2012.
3. Silicon Labs. 2013. The evolution of wireless sensor networks. Available online: http://www.silabs.com/Support%20Documents/TechnicalDocs/evolution-of-wireless-sensor-networks.pdf. Accessed on April 15, 2014.
4. Chong C.-Y. 2003. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91, 1247–1256.
5. Kaiser W.J. Low power wireless integrate microsensors. Available online: http://www.janet. ucla.edu/WINS/wiminnovative.htm. Accessed on September 15, 2014.
6. Buratti C., Conti A., Dardari D., Verdone R. 2009. An overview on wireless sensor networks technology and evolution. *Sensor*, 9, 6869–6896.
7. Marron P.J., Minder D., Consortium E.W. 2006. *Embedded WiseNts Research Roadmap*. Information Society Technologies, Berlin, Germany.
8. EC Project e-SENSE, FP6. Capturing ambient intelligence for mobile communications through wireless sensor networks.2006. Available online: http://www.ist-esense.org. Accessed on April 14, 2014.
9. EC Project CRUISE, FP6. Creating Ubiquitous Intelligent Sensing Environments (CRUISE).2006. Available online: http://www.ist-cruise.eu. Accessed on April 14, 2014.
10. Marron P.J. 2008. *Cooperating Objects Network of Excellence*. University of Bonn, Zentrum, Germany.
11. Pottie G.J. 1998. Wireless sensor networks. In *IEEE Information Theory Workshop*, Killarney, Ireland, pp. 1239–1240.
12. *Business Week Online*. Tech wave 2: The sensor revolution. Available online: http://www.businessweek.com/magazine/content/03_34/b3846622.htm. Accessed on April 15, 2014.
13. Zhang J. et al. 2009. UWB systems for wireless sensor networks. *Proceedings of the IEEE*, 97(2), 313–331.

14. Handziski V., Polastre J., Hauer J.H., Sharp C. 2004. Flexible hardware abstraction of the TI Msp430 MCU in Tinyos. In *Proceedings of Sensys'04: Proceedings of the Second International Conference on Embedded Networked Sensor Systems*, Baltimore, MD, November 2004. ACM Press, New York, NY, USA, pp. 277–278.
15. Van Der Werff T. 10 Emerging technologies that will change the world. Available online: http://www.globalfuture.com/mittreands2001.htm. Accessed on September 15, 2014.
16. Wetherall D. 1999. Active network vision and reality: Lessons form a capsule-based system. In *Symposium on Operating Systems Principles*, Kiawah Island, SC, pp. 64–79.
17. Hill J., Szewczyk R., Woo A., Hollar S., Culler D.E., Pister K.S.J. 2000. System architecture directions for networked sensors. In *Architecture Support for Programming Languages and Operating Systems* (*ASPLOS'00*), Cambridge, MA, pp. 12–24.
18. Estrin D., Govindan R., Heidemann J., Kumar S. 1999. Next century changes: Scalable coordinate in sensor network. In *Proceedings of the Fifth ACM/IEEE International Conference on Mobile Computing and Networking*, Seattle, WA, USA, pp. 263–270.
19. Xu L., He W., Li S. Internet of things in industries: A survey. 2014. *IEEE Transactions on Industrial Informatics*, published online, 10.1109/TII.2014.2300753.
20. Kim J., Hee K., Lee J., Kang H., Kim D. 2014. An ID/Locator separation-based mobility management architecture for WSNs. *IEEE Transactions on Mobile Computing*, published online, 10.1109/TMC.2013.142.
21. Alagheband M.R., Aref M.R. 2012. Dynamic and secure key management model for hierarchical heterogeneous sensor networks. *IET Information Security*, 6(4), 271–280.
22. Li Y., Jia Z., Liu F., Xie S. 2012. Hardware reconfigurable wireless sensor network node with power and area efficiency. *IET Wireless Sensor Systems*, 2(3), 247–252.
23. Yu S., Ren K., Lou W. 2011. FDAC: Toward fine-grained distributed data access control in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 22(4), 673–686.
24. Filho D.F.L., Amazonas J.R. 2013. TCNet: Trellis coded network—Implementation of QoS-aware routing protocols in WSNs. *IEEE Latin America Transactions*, 11(3), 969–974.
25. Abdulla A.E.A.A., Nishiyama H., Yang J., Ansari N., Kato N. 2012. HYMN: A novel hybrid multi-hop routing algorithm to improve the longevity of WSNs. *IEEE Transactions on Wireless Communications*, 11(7), 2531–2541.
26. Barachi M., Kadiwal A., Glitho R., Khendek F., Dssouli R. 2010. The design and implementation of architectural components for the integration of the IP multimedia subsystem and wireless sensor networks. *IEEE Communications Magazine*, 48(4), 42–50.
27. Vuran M.C., Akyildiz I.F. 2010. XLP: A cross-layer protocol for efficient communication in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 9(11), 1578–1591.
28. Han K., Kim K., Park J., Shon T. 2011. Efficient sensor node authentication in third generation-wireless sensor networks integrated networks. *IET Communications*, 5(12), 1744–1754.
29. Raychaudhuri D., Gerial M. (eds.). 2005. New architectures and disruptive technologies for the future Internet: The wireless, mobile and sensor network perspective. *Report of NSF Wireless Mobile Planning Group (WMPG) Workshop*, New Brunswick, NJ, USA, August 2005.
30. Isha A.M., Raj G. 2013. DOS attacks on TCP/IP layers in WSN. *International Journal of Computer Networks and Communications Security*, 1(2), 40–50.
31. Stankovic J.A., Wood A.D., He T. 2011, Realistic Applications for Wireless Sensor Networks, Theoretical Aspects of Distributed Computing in Sensor Networks, Springer, pp. 835–863.
32. Culler D. et al. 2005. Towards a sensor networks architecture: Lowering the waist ling. In *Proceedings of the 10th Workshop on Hot Topics in Operating System* (*HotOS X*). Santa Fe, NM, USA.
33. Erol-Kantarci M., Mouftah H.T., Oktug S. 2011. A survey of architectures and localization techniques for underwater acoustic sensor networks. *IEEE Communications Surveys & Tutorials*, 13(3), 487–502.

34. Mehlman J. 2013. Cross-layer design: A case for standardization. Research Report Tech- report. Accessed on September 15, 2014.
35. Li Y., Jia Z., Liu F., Xie S. 2012. Hardware reconfigurable wireless sensor network node with power and area efficiency. *IET Wireless Sensor Systems*, 2(3), 247–252.
36. Demirbas M., Chow K.Y., Wan C.S. 2014. INSIGHT: Internet-sensor integration for habitat monitoring. Tech-report. Available online: http://www.cse.buffalo.edu/tech-reports/2006-02.pdf.
37. Ruiz L.B., Nogueira J.M., Loureiro A.A.F. 2003. MANNA: A management architecture for wireless sensor networks. *IEEE Communications Magazine*, 41(2), 116–125.
38. He T. et al. 2006. VigilNet: An integrated sensor networks system for energy efficient surveillance. *ACM Transactions on Sensor Networks*, 2(1), 1–38.
39. He T. et al. 2005. An overview of the VigilNet Architecture. *11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*. Hong Kong, China. http://www.tc.ait.ac.th/faculty/teerapat/AT77.9019_Cross-Layer_Design_for_Wireless_Networks/Reading%20Assignments/Cross-layer%20Design_A_case_standardization.pdf. Accessed on September 15, 2014.
40. Ruiz L.B., Nogueira J.M., Loureiro A.A.F. 2003. MANNA: A management architecture for wireless sensor networks. *IEEE Communication Magazine*, 41, 116–125.
41. Ruiz L.B., Nogueira J.M., Loureiro A.A.F. 2003. MANNA: A management architecture for wireless sensor networks. *IEEE Communications Magazine*, 41(2), 116–125.
42. Nogueira J.M., Loureio A.A.F. 2005. On the design of a self-managed wireless sensor networks. *IEEE Communications Magazine*, 43(8), 95–102.
43. Erol-Kantarci M., Mouftah H.T. 2012. Suresense: Sustainable wireless rechargeable sensor networks for the smart grid. *IEEE Wireless Communications*, 19(3), 30–36.
44. Brannstrom M., Jungert E. 2003. A scalable agent architecture for a dynamic sensor networks. *Electronics Notes in Theoretical Computer Science* 68(3), 268–283.
45. Munir A., Gordon-Ross A., Ranka S. (May 2014). Multi-core embedded wireless sensor networks: Architecture and applications. 2014. *IEEE Transactions on Parallel and Distributed Systems*, published online, 10.1109/TPDS.2013.219.
46. Watfa M.K. 2010. Operating system design in future wireless sensor networks. *Journal of Networks*, 5(10), 1201–1214.
47. Handziski V. et al. 2003. A common wireless sensor network architecture. In H. Karl (ed.), *Proceedings of 1. GI/ITG Fachgespr*, Berlin, Germany, pp. 10–17.
48. Schaust S. 2005. Mobile sinks in ADHOC Sensor networks. Diploma thesis, Department of Mathematics and Computer Science, Freie University, Berlin, Germany.
49. Blumenthal J., Handy M., Golatowski F., Haase M., Timmermann D. 2003. Wireless sensor networks—New challenges in software engineering. *Proceedings of the Ninth IEEE International Conference on Emerging Technologies and Factory Automation*, Lisbon, Portugal, September 2003.
50. Kottapalli V.A. et al. 2003. Two-tiered wireless sensor network architecture for structural health monitoring. *SPIE, 10th International Symposium on Smart Structures and Materials*, San Diego, CA, March 2–6, 2003.
51. Xing G., Tan R., Liu B., Wang J., Jia X., Yi C.-W. 2009. Data fusion improves the coverage of wireless sensor networks. *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*, Beijing, China, pp. 157–168.
52. Lewandowski J., Arochena H., Naguib R., Chao K., Garcia-Perez A. 2014. Logic-centred architecture for ubiquitous health monitoring. *IEEE Journal of Biomedical and Health Informatics*, 18, 1525–1532.

6

Efficient Distributed Bayesian Estimation in Wireless Sensor Networks

Andrew P. Brown
Toyon Research Corporation

Ronald A. Iltis
University of California

Hua Lee
University of California

6.1 Introduction **6**-1
6.2 Local State Estimation **6**-3
6.3 State Estimate Fusion **6**-5
6.4 Optimal Distributed State Estimation **6**-6
6.5 Approximate Distributed State Estimation **6**-10
Computation and Memory Resource Efficiency • Communications Resource Efficiency
6.6 Nonlinear State Estimation **6**-14
6.7 Distributed Position Estimation **6**-18
6.8 Simulation Results and Discussion **6**-19
6.9 Summary **6**-22
References **6**-23

6.1 Introduction

In most current wireless sensor network (WSN) implementations, information is communicated (via multiple hops if necessary) to a sink or central processor, which then computes the central (global) estimate of the phenomena of interest. However, in large networks, nodes that are near the sink are heavily burdened by relaying packets from more distant nodes. Such inequitable load sharing can quickly lead to traffic congestion and node energy depletion. Instead, it is beneficial to distribute computation throughout the network, with each node performing data fusion and compression. In the distributed estimation approach presented here, raw measurements collected at each sensor are processed locally to generate local estimates of the states of interest. The sufficient statistics of these estimates provide a lossless compression of all measurement data used to generate the local estimates. Information packets based on these sufficient statistics are then transmitted to other nearby nodes and fused. The communication and computation burden is shared equally across nodes, leading to improved network longevity and scalability. Additional benefits of this distributed architecture include real-time situational awareness within the network, robustness to node and sink failures, and the capability of ad hoc operation in the absence of any infrastructure.

Work on distributed estimation dates back at least to the 1970s [6] and has received renewed interest [35,44,45,54,55]. The development of practical algorithms for distributed estimation is a challenging problem, since typically the errors in quantities (e.g., local estimates) to be fused are correlated. If not properly accounted for, these correlations lead to reuse of common information, resulting in inaccurate estimation and instability [57]. The use of information graphs to identify common measurements

in local estimates was introduced in [13], and this framework has been used to develop a variety of distributed/decentralized estimation architectures and algorithms [12,17,30]. Most of the existing practical algorithms are approximate when applied to the estimation of time-varying states (i.e., for state-space models with nonzero process noise) [36]. The decentralized information filter [17], which provides the minimum-mean-square Kalman filter (KF) estimate in fully connected networks, is a notable exception. The primary difficulty in the decentralized information filter framework arises when the internode communication delays are significant [46]. This case appears very often in real-world applications, for example, in large ad hoc networks (due to medium access contention or multihop routing).

In this chapter, we present a unified Bayesian framework for distributed estimation of time-varying states under the assumption of linear Gaussian state and measurement models. The resulting algorithms are resource efficient and are valid for arbitrary, time-varying network connectivity. Significant attention is devoted to the problem of delayed information fusion, for which optimal and approximate algorithms are introduced and compared. Furthermore, the problem of aggregating information for time-varying states is addressed with the introduction of an approximate algorithm that trades off improved communications efficiency for increased latency in the rate at which information propagates through the network.

To demonstrate the distributed estimation techniques presented in this chapter, we apply the algorithms to the problem of distributed position estimation of mobile nodes in a wireless ad hoc network. A critical challenge often encountered in the development of wireless communications and sensor networks is the requirement for reliable indoor/outdoor localization (terminal or sensor node position estimation). This technology enables location-dependent services [42,53] and facilitates efficient multihop routing in ad hoc networks [33,52]. Node localization is also required for providing context to collected data in sensor networks [14,23,43,47,50]. In military applications, geolocation is required for tracking military personnel and resources to enable effective command, control, and coordination [1,16,39,43,56,57]. Localization devices may be embedded in handheld or wearable radios carried by warfighters, mounted on vehicles/aircraft, or contained in unattended sensors distributed on the battlefield. Military wireless networks are ad hoc in nature; the ability to form a robust, dynamic network of warfighters, vehicles/aircraft, and sensors is highly desirable.

GPS cannot always be relied upon due to node design constraints including cost, power consumption, and/or form factor, possibly hostile or unintentional jamming and reliability (e.g., position estimate unavailability indoors or in urban canyons due to shadowing and multipath). Thus, GPS-less distributed localization is an active research area in WSNs for both civilian [14,43,47,50] and military [16,23,30,39,57] applications. Low-cost GPS-less *radiolocation* can be implemented via various metrics such as signal strength [49], connectivity [8,14,22], angle of arrival (AOA), time of arrival (TOA), time difference of arrival (TDOA) [9], and round-trip time (RTT) [24,25,31,32,39]. Received signal strength indicator (RSSI) measurements are available from existing radios, but this positioning method [19–21,49] is vulnerable to multipath fading [9] and requires channel-specific calibration to map RSSI to distance. Connectivity measurements are also readily available but provide only coarse position information. When AOA measurements are used, a position can be estimated with only two reference nodes (or with only one reference node if range measurements are available). However, performance is severely degraded by scattering near antennas [9]. The use of antenna arrays also corresponds to increased sensor node cost, power consumption, size, and complexity. The use of TOA and RTT measurements for ranging and radiolocation dates back at least to the 1970s Department of Defense (DOD) joint tactical information distribution system (JTIDS) program. Distributed EKF-based positioning algorithms for JTIDS based on TOA are described in [57]. A disadvantage of pure TOA-based range estimation is that each sensor node must have a common network clock. Using round-trip travel time for ranging eliminates the need for such network synchronization. Communication handsets implementing RTT-based ranging using direct-sequence waveforms are described in [39]. The use of RTT range measurements for sensor positioning applications has also been explored in [25,50,51]. In this chapter, we present node

position estimation using extended KF (EKF)- and unscented KF (UKF)-based algorithms for local measurement filtering, with collaboration between nodes to perform distributed nonlinear estimation.

This chapter is organized as follows: The Bayesian framework for local state estimation and fusion is presented in Sections 6.2 and 6.3, respectively. Optimal distributed estimation in this framework is presented along with the problem of delayed information fusion in Section 6.4. Resource-efficient approximate algorithms for information aggregation and delayed information fusion are then given in Section 6.5. Extensions to nonlinear estimation are presented in Section 6.6. The application of mobile node position estimation in ad hoc networks is presented in Section 6.7. Finally, simulation results and discussion are provided in Section 6.8.

6.2 Local State Estimation

Consider a network of sensor nodes that obtain state measurements and communicate sufficient statistics for these measurements to a subset of neighboring nodes with which they are *connected* (e.g., via a shared wireless channel). Information packets may also be communicated multihop to other more distant nodes. The linear state dynamics model for $\mathbf{x}(n) \in \mathbb{R}^{N_x}$ assumed in the sequel is

$$\mathbf{x}(n) = \mathbf{F}(n, n-1)\mathbf{x}(n-1) + \mathbf{v}(n-1), \tag{6.1}$$

(see, e.g., [5]), where the initial state $\mathbf{x}(0)$ is assumed Gaussian distributed, $\mathbf{F}(n, n-1)$ is the state transition matrix, and $\mathbf{v}(n-1) \in \Re^{N_x}$ is white Gaussian process noise with covariance $\mathbf{Q}(n-1)$. In the following, it will be convenient to write the state transition from discrete time index m to n (for $m \leq n$) as

$$\mathbf{x}(n) = \mathbf{F}(n, m)\mathbf{x}(m) + \mathbf{v}(n, m), \tag{6.2}$$

where the white Gaussian process noise $\mathbf{v}(n, m)$ has covariance $\mathbf{Q}(n, m)$. The matrices $\mathbf{F}(n, m)$ and $\mathbf{Q}(n, m)$ can be obtained recursively using

$$\mathbf{F}(n, m) = \mathbf{F}(n, n-1)\mathbf{F}(n-1, m), \tag{6.3}$$

$$\mathbf{Q}(n, m) = \mathbf{F}(n, n-1)\mathbf{Q}(n-1, m)\mathbf{F}^T(n, n-1) + \mathbf{Q}(n-1), \tag{6.4}$$

with $\mathbf{F}(n, n) = \mathbf{I}$, $\mathbf{Q}(n, n) = 0$, and $\mathbf{Q}(n+1, n) = \mathbf{Q}(n)$. However, for computational efficiency, $\mathbf{F}(n, m)$ and $\mathbf{Q}(n, m)$ should be computed directly, for example, as demonstrated for a particular choice of linear state dynamics model in Section 6.7. The linear measurement model is

$$\mathbf{z}(n) = \mathbf{H}(n)\mathbf{x}(n) + \mathbf{w}(n) \in \mathbb{R}^{N_z}, \tag{6.5}$$

where

$\mathbf{H}(n)$ is the observation matrix

$\mathbf{w}(n)$ is white Gaussian measurement noise with covariance $\mathbf{R}(n)$

The index n references the time t_n when some event (relevant to the distributed estimation) occurs asynchronously in the network; the delay $t_{n+1} - t_n$ between consecutive events is arbitrary. Note that for estimation of time-varying states, network-wide clock synchronization is required, but if unavailable, the clock biases can be estimated jointly with $\mathbf{x}(n)$ [57].

The cumulative measurement set $\mathbf{Z}_i^m$ is defined as the set of all measurements recorded at node i, along with the set of all measurements for which sufficient statistics are received via communication with other nodes, up to and including time (index) m. That is, $p(\mathbf{x}(n) | \mathbf{Z}_i^m)$ denotes the *a posteriori* probability

distribution on $\mathbf{x}(n)$, given the cumulative information available at node i at time m. Then the state estimate, the state estimate error, and the covariance of this error are defined by

$$\hat{\mathbf{x}}\left(n \mid Z_i^m\right) = E\left\{\mathbf{x}(n) \mid Z_i^m\right\}, \tag{6.6}$$

$$\tilde{\mathbf{x}}\left(n \mid Z_i^m\right) = \mathbf{x}(n) - \hat{\mathbf{x}}\left(n \mid Z_i^m\right), \tag{6.7}$$

$$P\left(n \mid Z_i^m\right) = Cov\left\{\tilde{\mathbf{x}}(n \mid Z_i^m)\right\}. \tag{6.8}$$

For $n \geq m$, $\hat{\mathbf{x}}(n \mid Z_i^m)$ and $P(n \mid Z_i^m)$ can be efficiently calculated using the KF algorithm [2,4,29] running locally at node i. Together, this conditional mean and covariance constitute sufficient statistics [40] for the *a posteriori* density $p(\mathbf{x}(n) \mid Z_i^m)$, under the assumption of linear Gaussian state and measurement models. Equivalent sufficient statistics are given by the information state vector and information matrix [38]:

$$\hat{\mathbf{y}}\left(n \mid Z_i^m\right) = P^{-1}\left(n \mid Z_i^m\right)\hat{\mathbf{x}}\left(n \mid Z_i^m\right) \tag{6.9}$$

and

$$Y\left(n \mid Z_i^m\right) = P^{-1}\left(n \mid Z_i^m\right), \tag{6.10}$$

respectively. These quantities are especially useful in distributed data fusion and can be efficiently calculated using the information form [17] of the KF. The prediction from time m to time n is given by

$$Y\left(n \mid Z_i^m\right) = \left[F(n,m)Y^{-1}\left(m \mid Z_i^m\right)F^T(n,m) + Q(n,m)\right]^{-1}, \tag{6.11}$$

$$\hat{\mathbf{y}}\left(n \mid Z_i^m\right) = Y\left(n \mid Z_i^m\right)F(n,m)Y^{-1}\left(m \mid Z_i^m\right)\hat{\mathbf{y}}\left(m \mid Z_i^m\right), \tag{6.12}$$

which in the following is denoted by

$$p\left(\mathbf{x}(n) \mid Z_i^m\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(m) \mid Z_i^m\right). \tag{6.13}$$

Given new information contained in $\mathbf{z}(n)$, the update is

$$\hat{\mathbf{y}}\left(n \mid Z_i^n\right) = \hat{\mathbf{y}}\left(n \mid Z_i^m\right) + \mathbf{i}(n), \tag{6.14}$$

$$Y\left(n \mid Z_i^n\right) = Y\left(n \mid Z_i^m\right) + \mathbf{I}(n), \tag{6.15}$$

where the new information is

$$\mathbf{i}(n) = \mathbf{H}^T(n)\mathbf{R}^{-1}(n)\mathbf{z}(n), \tag{6.16}$$

with

$$\mathbf{I}(n) = \mathbf{H}^T(n)\mathbf{R}^{-1}(n)\mathbf{H}(n). \tag{6.17}$$

6.3 State Estimate Fusion

Consider a data fusion event at node i and time $n \geq m$: node i receives a transmission from node j containing sufficient statistics for $p(\mathbf{x}(m)|\mathcal{Z}_j^m)$, with $\mathcal{Z}_j^m$ denoting some set of measurements (though not necessarily a *cumulative* measurement set*) up to time m. Let $\hat{\mathbf{x}}(n|\mathcal{Z}_i^{n-}), P(n|\mathcal{Z}_i^{n-})$ denote sufficient statistics for node i's current estimate at time n, where $\mathcal{Z}_i^{n-}$ does not include the new information $\mathcal{Z}_j^m$ to be fused. Using the KF prediction equations, node i can propagate $\hat{\mathbf{x}}(m|\mathcal{Z}_j^m), P(m|\mathcal{Z}_j^m)$ to the current time n: $p(\mathbf{x}(m)|\mathcal{Z}_j^m) \overset{\text{Pred}}{\square} p(\mathbf{x}(n)|\mathcal{Z}_j^m)$. Node i's task is then to fuse the resulting sufficient statistics $\hat{\mathbf{x}}(n|\mathcal{Z}_j^m), P(n|\mathcal{Z}_j^m)$ from node j with its previous local information $\hat{\mathbf{x}}(n|\mathcal{Z}_i^{n-}), P(n|\mathcal{Z}_i^{n-})$ to form $\hat{\mathbf{x}}(n|\mathcal{Z}_i^n), P(n|\mathcal{Z}_i^n)$ (sufficient statistics for $p(\mathbf{x}(n)|\mathcal{Z}_i^n)$), where $\mathcal{Z}_i^n = \mathcal{Z}_i^{n-} \cup \mathcal{Z}_j^m$.

To fuse two Gaussian posterior densities $p(\mathbf{x}(n)|\mathcal{Z}_1)$ and $p(\mathbf{x}(n)|\mathcal{Z}_2)$ conditioned on measurement sets $\mathcal{Z}_1$ and $\mathcal{Z}_2$, the fusion formula

$$p(\mathbf{x}(n)|\mathcal{Z}_1 \cup \mathcal{Z}_2) \propto \frac{p(\mathbf{x}(n)|\mathcal{Z}_1)p(\mathbf{x}(n)|\mathcal{Z}_2)}{p(\mathbf{x}(n)|\mathcal{Z}_1 \cap \mathcal{Z}_2)} \tag{6.18}$$

from [13] can be used if the measurements are conditionally independent given the state, that is,

$$\begin{aligned} p(\mathcal{Z}_1 \cup \mathcal{Z}_2|\mathbf{x}(n)) = p(\mathcal{Z}_1 - \mathcal{Z}_2|\mathbf{x}(n))p(\mathcal{Z}_2 - \mathcal{Z}_1|\mathbf{x}(n)) \\ p(\mathcal{Z}_1 \cup \mathcal{Z}_2|\mathbf{x}(n)), \end{aligned} \tag{6.19}$$

where " ," " ," and "–" denote the set union, intersection, and reduction operators. The fusion (6.18) can be computed efficiently using a lemma from [11], which states that sufficient statistics for the product of Gaussian densities

$$p(\mathbf{x}(n)|\mathcal{Z}_4) \sim \frac{p(\mathbf{x}(n)|\mathcal{Z}_1)p(\mathbf{x}(n)|\mathcal{Z}_2)}{p(\mathbf{x}(n)|\mathcal{Z}_3)} \tag{6.20}$$

are given by

$$P(n|\mathcal{Z}_4)^{-1} = P(n|\mathcal{Z}_1)^{-1} + P(n|\mathcal{Z}_2)^{-1} - P(n|\mathcal{Z}_3)^{-1}, \tag{6.21}$$

$$\hat{\mathbf{x}}(n|\mathcal{Z}_4) = P(n|\mathcal{Z}_4)\left[P(n|\mathcal{Z}_1)^{-1}\hat{\mathbf{x}}(n|\mathcal{Z}_1) + P(n|\mathcal{Z}_2)^{-1}\hat{\mathbf{x}}(n|\mathcal{Z}_2) - P(n|\mathcal{Z}_3)^{-1}\hat{\mathbf{x}}(n|\mathcal{Z}_3)\right]. \tag{6.22}$$

Using (6.9) and (6.10) in (6.21) and (6.22), the equivalent sufficient statistics $\hat{\mathbf{y}}(n|\mathcal{Z}_4), \mathbf{Y}(n|\mathcal{Z}_4)$ for $p(\mathbf{x}(n)|\mathcal{Z}_4)$ are obtained as

$$\mathbf{Y}(n|\mathcal{Z}_4) = \mathbf{Y}(n|\mathcal{Z}_1) + \mathbf{Y}(n|\mathcal{Z}_2) - \mathbf{Y}(n|\mathcal{Z}_3), \tag{6.23}$$

$$\hat{\mathbf{y}}(n|\mathcal{Z}_4) = \hat{\mathbf{y}}(n|\mathcal{Z}_1) + \hat{\mathbf{y}}(n|\mathcal{Z}_2) - \hat{\mathbf{y}}(n|\mathcal{Z}_3). \tag{6.24}$$

Again, the fusion formula (6.18) holds exactly (i.e., condition (6.19) is satisfied) if the measurement sets $\mathcal{Z}_1 - \mathcal{Z}_2$, $\mathcal{Z}_2 - \mathcal{Z}_1$, and $\mathcal{Z}_1 \cap \mathcal{Z}_2$ are independent, given $\mathbf{x}(n)$. This is not generally the case for time-varying $\mathbf{x}(n)$. For example, if $\mathcal{Z}_1 - \mathcal{Z}_2$ contains the measurement $\mathbf{z}^{k_1}(m_{k_1})$ obtained at node k_1 at time $t_{m_{k_1}}$, and

* For example, in Section 6.4, $\mathcal{Z}_j^m = \mathbf{z}_j(m)$, and in Section 6.5.2, $\mathcal{Z}_j^m$ is a collection of recent measurements at node j.

$\mathcal{Z}_2 - \mathcal{Z}_1$ contains the measurement $\mathbf{z}^{k_2}(m_{k_2})$ obtained at node k_2 at time $t_{m_{k_2}}$, where $t_n > t_{m_{k_1}} > t_{m_{k_2}}$, then $\mathbf{z}^{k_1}(m_{k_1})$ and $\mathbf{z}^{k_2}(m_{k_2})$ are dependent when conditioned on $\mathbf{x}(n)$, since $\mathbf{x}(m_{k_2})$ and $\mathbf{x}(m_{k_1})$ are dependent. Condition (6.19) *does* hold for estimation of the joint posterior density of the cumulative trajectory $\mathbf{x}^n = \{\mathbf{x}(n), \mathbf{x}(n-1), \ldots, \mathbf{x}(0)\}$ at all measurement times:

$$p(\mathcal{Z}_1 \cup \mathcal{Z}_2 \,|\, \mathbf{x}^n) = p(\mathcal{Z}_1 - \mathcal{Z}_2 \,|\, \mathbf{x}^n) \cdot p(\mathcal{Z}_2 - \mathcal{Z}_1 \,|\, \mathbf{x}^n) \cdot p(\mathcal{Z}_1 \cap \mathcal{Z}_2 \,|\, \mathbf{x}^n). \tag{6.25}$$

However, fusion of densities defined over $\mathbf{x}^n$ has prohibitive complexity, growing as n^3.

The following analysis gives insight into how optimal fusion can be achieved with reduced computation, communication, and storage requirements. Using

$$\begin{aligned} p\left(Z_1^n \,|\, \mathbf{x}^n\right) &= p\left(\bigcup_{\{k,m_k\}:\mathbf{z}^k(m_k)\in Z_1^n} \mathbf{z}^k(m_k) \,|\, \mathbf{x}^n\right) \\ &= \prod_{\{k,m_k\}:\mathbf{z}^k(m_k)\in Z_1^n} p(\mathbf{z}^k(m_k) \,|\, \mathbf{x}^n) \\ &= \prod_{\{k,m_k\}:\mathbf{z}^k(m_k)\in Z_1^n} p(\mathbf{z}^k(m_k) \,|\, \mathbf{x}(m_k)) \end{aligned} \tag{6.26}$$

and Bayes' rule,

$$p\left(\mathbf{x}^n \,|\, Z_1^n\right) \sim p\left(\mathbf{x}^n\right) \prod_{\{k,m_k\}:\mathbf{z}^k(m_k)\in Z_1^n} \frac{p\left(\mathbf{x}(m_k) \,|\, \mathbf{z}^k(m_k)\right)}{p\left(\mathbf{x}(m_k)\right)}. \tag{6.27}$$

Thus, sufficient statistics for $p(\mathbf{x}^n \,|\, Z_1^n)$ can be obtained by computing, storing/transmitting, and fusing sufficient statistics for densities $p(\mathbf{x}(m_k) \,|\, \mathbf{z}^k(m_k))$ defined over N_x-dimensional state spaces (the *a priori* densities $p(\mathbf{x}(m_k))$ and $p(\mathbf{x}^n)$ are assumed known.* Though it appears that fusion with $p(\mathbf{x}^n)$, defined over an $(n+1)$ N_x-dimensional state space, is required, the next section shows how to avoid this difficulty.

6.4 Optimal Distributed State Estimation

Consider the data processing required of node j after obtaining measurements $\mathbf{z}^j(m)$ at time m. From (1.27), it is clear that any node i can compute sufficient statistics for $p(\mathbf{x}^n \,|\, Z_i^n) = p(\mathbf{x}^n \,|\, Z_i^{n-} \cup \mathbf{z}^j(m))$ and thus for the desired density $p(\mathbf{x}(n) \,|\, Z_i^n)$, given sufficient statistics for $p(\mathbf{x}(m) \,|\, \mathbf{z}^j(m))$ and for $\{p(\mathbf{x}(m_k) \,|\, \mathbf{z}^k(m_k))\}\forall\, \mathbf{z}^k(m_k) \in Z_i^{n-}$. For optimal Bayesian data fusion at node i, it is therefore sufficient for node j to compute and transmit sufficient statistics for $p(\mathbf{x}(m) \,|\, \mathbf{z}^j(m))$.

When node j obtains measurements $\mathbf{z}^j(m)$ at time m, it processes them in its local KF (or nonlinear filter such as the EKF or UKF) to compute sufficient statistics for $p(\mathbf{x}(m) \,|\, Z_j^m)$, given sufficient statistics for $p(\mathbf{x}(m) \,|\, Z_j^{m-1})$. Node j can then use (6.18) to calculate sufficient statistics for the posterior density based only on the new measurements $\mathbf{z}^j(m)$:

$$p\left(\mathbf{x}(m) \,|\, \mathbf{z}^j(m)\right) \sim \frac{p(\mathbf{x}(m))\, p\left(\mathbf{x}(m) \,|\, Z_j^m\right)}{p\left(\mathbf{x}(m) \,|\, Z_j^{m-1}\right)}, \tag{6.28}$$

* In practice, the *a priori* densities may be estimated from an initial set of measurements.

where condition (6.19) holds, since $\mathbf{z}^j(m)$ is independent of $\mathbf{Z}_j^{m-1}$, given $\mathbf{x}(m)$, under the assumption of independent measurement noise in (6.5). As discussed in Section 6.5, (6.28) yields better performance in the nonlinear case, compared with direct estimation of $p(\mathbf{x}(m)\,|\,\mathbf{z}^j(m))$ in the EKF or UKF/SUKF, since $p(\mathbf{x}(m)\,|\,\mathbf{z}^j(m))$ is subject to divergence in these filters, while $p(\mathbf{x}(m)\,|\,\mathbf{Z}_j^{m-1})$ and $p(\mathbf{x}(m)\,|\,\mathbf{Z}_j^{m})$ are well approximated if the EKF or UKF/SUKF is tracking the states. From (6.20), (6.23) and (6.24), and (6.9) and (6.10), sufficient statistics for $p(\mathbf{x}(m)\,|\,\mathbf{z}^j(m))$ are obtained as

$$\mathbf{Y}\left(m\,|\,\mathbf{z}^j(m)\right)=\mathbf{Y}(m)+\mathbf{Y}\left(m\,|\,\mathbf{Z}_j^{m}\right)-\mathbf{Y}\left(m\,|\,\mathbf{Z}_j^{m-1}\right), \tag{6.29}$$

$$\hat{\mathbf{y}}\left(m\,|\,\mathbf{z}^j(m)\right)=\bar{\mathbf{y}}(m)+\hat{\mathbf{y}}\left(m\,|\,\mathbf{Z}_j^{m}\right)-\hat{\mathbf{y}}\left(m\,|\,\mathbf{Z}_j^{m-1}\right), \tag{6.30}$$

where $\bar{\mathbf{y}}(m)$ and $\mathbf{Y}(m)$ are the mean and covariance of the *a priori* distribution. Due to the simplicity of (6.29) and (6.30), it is more efficient to employ the information form of the KF. After computing $\hat{\mathbf{y}}(m\,|\,\mathbf{z}^j(m))$, $\mathbf{Y}(m\,|\,\mathbf{z}^j(m))$, node j transmits these quantities in a sufficient statistics packet (SSP) including the origin node index, j, and asynchronous measurement time, t_m. The algorithm for computing $\hat{\mathbf{y}}(m\,|\,\mathbf{z}^j(m))$, $\mathbf{Y}(m\,|\,\mathbf{z}^j(m))$ is specified in Figure 6.1. In some cases, improved communications efficiency may be achieved by forming the SSP with $\mathbf{z}^j(m)$ and the information required to specify $\mathbf{H}(m)$ and $\mathbf{R}^{-1}(m)$,

Measure:

$$\mathbf{z}^j(m).$$

Given:

$$\mathbf{F}(m,m-1),\mathbf{Q}(m,m-1),\mathbf{H}(m),\mathbf{R}(m),\bar{\mathbf{y}}(m-1),\mathbf{Y}(m-1),$$
$$\hat{\mathbf{y}}\left(m-1\,|\,\mathbf{Z}_j^{m-1}\right),\mathbf{Y}\left(m-1\,|\,\mathbf{Z}_j^{m-1}\right).$$

Compute:

$$\mathbf{Y}(m)=\left[\mathbf{F}(m,m-1)\mathbf{Y}^{-1}(m-1)\mathbf{F}^T(m,m-1)+\mathbf{Q}(m,m-1)\right]^{-1}$$
$$\bar{\mathbf{y}}(m)=\mathbf{Y}(m)\mathbf{F}(m,m-1)\mathbf{Y}^{-1}(m-1)\bar{\mathbf{y}}(m-1)$$
$$\mathbf{Y}\left(m\,|\,\mathbf{Z}_j^{m-1}\right)=\left[\mathbf{F}(m,m-1)\mathbf{Y}^{-1}\left(m-1\,|\,\mathbf{Z}_j^{m-1}\right)\mathbf{F}^T(m,m-1)+\mathbf{Q}(m,m-1)\right]^{-1}$$
$$\hat{\mathbf{y}}\left(m\,|\,\mathbf{Z}_j^{m-1}\right)=\mathbf{Y}\left(m\,|\,\mathbf{Z}_j^{m-1}\right)\mathbf{F}(m,m-1)\mathbf{Y}^{-1}\left(m-1\,|\,\mathbf{Z}_j^{m-1}\right)\hat{\mathbf{y}}\left(m-1\,|\,\mathbf{Z}_j^{m-1}\right)$$
$$\mathbf{Y}\left(m\,|\,\mathbf{z}^j(m)\right)=\mathbf{Y}(m)+\mathbf{H}^T(m)\mathbf{R}^{-1}(m)\mathbf{H}(m)$$
$$\hat{\mathbf{y}}\left(m\,|\,\mathbf{z}^j(m)\right)=\bar{\mathbf{y}}(m)+\mathbf{H}^T(m)\mathbf{R}^{-1}(m)\mathbf{z}^j(m)$$
$$\mathbf{Y}\left(m\,|\,\mathbf{Z}_j^{m}\right)=\mathbf{Y}\left(m\,|\,\mathbf{Z}_j^{m-1}\right)+\mathbf{Y}\left(m\,|\,\mathbf{z}^j(m)\right)-\mathbf{Y}(m)$$
$$\hat{\mathbf{y}}\left(m\,|\,\mathbf{Z}_j^{m}\right)=\hat{\mathbf{y}}\left(m\,|\,\mathbf{Z}_j^{m-1}\right)+\hat{\mathbf{y}}\left(m\,|\,\mathbf{z}^j(m)\right)-\bar{\mathbf{y}}(m)$$

Store:

$$\bar{\mathbf{y}}(m),\mathbf{Y}(m),\hat{\mathbf{y}}\left(m\,|\,\mathbf{Z}_j^{m}\right),\mathbf{Y}\left(m\,|\,\mathbf{Z}_j^{m}\right).$$

Transmit:

$$\text{SSP containing }\hat{\mathbf{y}}\left(m\,|\,\mathbf{z}^j(m)\right),\mathbf{Y}\left(m\,|\,\mathbf{z}^j(m)\right),j,t_m.$$

FIGURE 6.1 Measurement node processing/SSP generation.

rather than with $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid \mathbf{z}^j(m))$. The sufficient statistics are then reconstructed at the destination nodes using, from (6.29) to (6.30) and (6.14) to (6.17),

$$\hat{\mathbf{y}}\left(m \mid \mathbf{z}^j(m)\right) = \bar{\mathbf{y}}(m) + \mathbf{H}^T(m)\mathbf{R}^{-1}(m)\mathbf{z}^j(m), \tag{6.31}$$

and

$$\mathbf{Y}(m \mid \mathbf{z}^j(m)) = \mathbf{Y}(m) + \mathbf{H}^T(m)\mathbf{R}^{-1}(m)\mathbf{H}(m). \tag{6.32}$$

Note that in the nonlinear case, it would be necessary to approximate $\mathbf{H}(m)$, for example, using the Jacobian.

Suppose node $i \neq j$ receives an SSP containing new information $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid \mathbf{z}^j(m))$. If node i's most recent state estimate is stored as $\hat{\mathbf{y}}(n \mid Z_i^n)$, $\mathbf{Y}(n \mid Z_i^n)$, with $n \leq m$, then fusing the new information to form sufficient statistics $\hat{\mathbf{y}}(m \mid Z_i^m)$, $\mathbf{Y}(m \mid Z_i^m)$ for $p(x(m) \mid Z_i^m) = p(x(m) \mid Z_i^n \cup z_j^m)$ is straightforward:

$$p\left(x(m) \mid Z_i^n\right) \overset{\text{Pred}}{\Leftarrow} p\left(x(n) \mid Z_i^n\right), \tag{6.33}$$

$$p\left(x(m) \mid Z_i^m\right) \sim \frac{p\left(x(m) \mid Z_i^n\right) p\left(x(m) \mid z^j(m)\right)}{p(x(m))}. \tag{6.34}$$

The algorithm for computing $\hat{\mathbf{y}}(m \mid Z_i^m)$, $\mathbf{Y}(m \mid Z_i^m)$ is specified in Figure 6.2. If node i receives redundant copies of $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid \mathbf{z}^j(m))$, it can identify (based on the index pair (j, m)) and discard them.

Receive:

SSP containing $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid z^j(m))$, j, t_m.

If Not Delayed ($n \leq m$):

Given:

$\mathbf{F}(m,n)$, $\mathbf{Q}(m,n)$, $\bar{\mathbf{y}}(n)$, $\mathbf{Y}(n)$, $\hat{\mathbf{y}}(n \mid \mathbf{Z}_i^n)$, $\mathbf{Y}(n \mid \mathbf{Z}_i^n)$.

Compute:

$$\mathbf{Y}(m) = [\mathbf{F}(m,n)\mathbf{Y}^{-1}(n)\mathbf{F}^T(m,n) + \mathbf{Q}(m,n)]^{-1}$$

$$\bar{\mathbf{y}}(m) = \mathbf{Y}(m)\mathbf{F}(m,n)\mathbf{Y}^{-1}(n)\bar{\mathbf{y}}(n)$$

$$\mathbf{Y}(m \mid \mathbf{Z}_i^n) = [\mathbf{F}(m,n)\mathbf{Y}^{-1}(n \mid \mathbf{Z}_i^n)\mathbf{F}^T(m,n) + \mathbf{Q}(m,n)]^{-1}$$

$$\hat{\mathbf{y}}(m \mid \mathbf{Z}_i^n) = \mathbf{Y}(m \mid \mathbf{Z}_i^n)\mathbf{F}(m,n)\mathbf{Y}^{-1}(n \mid \mathbf{Z}_i^n)\hat{\mathbf{y}}(n \mid \mathbf{Z}_i^n)$$

$$\mathbf{Y}(m \mid \mathbf{Z}_i^m) = \mathbf{Y}(m \mid \mathbf{Z}_i^n) + \mathbf{Y}(m \mid \mathbf{z}^j(m)) - \mathbf{Y}(m)$$

$$\hat{\mathbf{y}}(m \mid \mathbf{Z}_i^m) = \hat{\mathbf{y}}(m \mid \mathbf{Z}_i^n) + \hat{\mathbf{y}}(m \mid \mathbf{z}^j(m)) - \bar{\mathbf{y}}(m)$$

Store:

$\bar{\mathbf{y}}(m)$, $\mathbf{Y}(m)$, $\hat{\mathbf{y}}(m \mid \mathbf{Z}_i^m)$, $\mathbf{Y}(m \mid \mathbf{Z}_i^m)$.

FIGURE 6.2 Fusion node processing/SSP fusion: EIF algorithm, undelayed case.

Note that the distributed estimation algorithm (6.29) and (6.30) and (6.33) and (6.34) is a generalization of the decentralized information filter [17].

In-network processing and communication delays (e.g., in local state estimation, medium access, or routing) are often nonnegligible, especially in multihop ad hoc networks. Thus, the information to be fused may be delayed with respect to the most recent state estimate at the fusion node, that is, $m < n$. For optimal information fusion, node i must fuse the *delayed* information $\hat{\mathbf{y}}(m|\mathbf{z}^j(m))$, $\mathbf{Y}(m|\mathbf{z}^j(m))$ with $\hat{\mathbf{y}}(n|\mathcal{Z}_i^{n-})$, $\mathbf{Y}(n|\mathcal{Z}_i^{n-})$ which denote sufficient statistics for node i's current estimate at time n. $\mathcal{Z}_i^{n-}$ is not necessarily the same as $\mathcal{Z}_i^{n-1}$, since $\mathcal{Z}_i^{n-}$ is the information at node i at time n prior to any communication from other nodes; hence, it may include its own measurement $\mathbf{z}^i(n)$. $\mathcal{Z}_i^n$ is the information at node i at time n after all communication from other nodes has taken place. The fusion task is to compute the updated information $\hat{\mathbf{y}}(n|\mathcal{Z}_i^n)$, $\mathbf{Y}(n|\mathcal{Z}_i^n)$, corresponding to sufficient statistics for $p(\mathbf{x}(n)|\mathcal{Z}_i^n) = p(\mathbf{x}(n)|\mathcal{Z}_i^{n-} \cup \mathcal{Z}_j^m)$. Optimal algorithms have been obtained for the closely related problem of out-of-sequence measurement (OOSM) fusion [3] in best linear unbiased estimation [58] and generalized smoothing [37] frameworks. However, compared with SSP reprocessing, the existing optimal fusion algorithms are more complicated to implement and do not provide improvements in computation and memory efficiency. Thus, an algorithm based on SSP reprocessing will be presented using the following notation:

$$\mathcal{Z}_i^{n-} = \mathcal{Z}_i^{m-1} \cup \left\{ \mathbf{z}^{k_1}(m_{k_1}), \mathbf{z}^{k_2}(m_{k_2}), \ldots, \mathbf{z}^{k_K}(m_{k_K}) \right\}, \tag{6.35}$$

where $t_m < t_{m_{k_1}} \le t_{m_{k_2}} \le \cdots \le t_{m_{k_K}}$ and $\mathcal{Z}_i^{m-1}$ is the set of measurements up through time

$$t_{m-1} = \max \left\{ t_{m_{k'}} : \mathbf{z}^{k'}(m_{k'}) \in \mathcal{Z}_i^n, t_{m_{k'}} \le t_m \right\}. \tag{6.36}$$

and $\{\mathbf{z}^{k_1}(m_{k_1}), \mathbf{z}^{k_2}(m_{k_2}), \ldots, \mathbf{z}^{k_K}(m_{k_K})\}$ is the set of measurements at later times.*,† Given $\hat{\mathbf{y}}(m-1|\mathcal{Z}_i^{m-1})$, $\mathbf{Y}(m-1|\mathcal{Z}_i^{m-1})$ and $\{\hat{\mathbf{y}}(m_k|\mathbf{z}^k(m_k))\}$, $\{\mathbf{Y}(m_k|\mathbf{z}^k(m_k))\}$, for $k = k_1, k_2, \ldots, k_K$, optimal fusion of $\hat{\mathbf{y}}(m|\mathbf{z}^j(m))$, $\mathbf{Y}(m|\mathbf{z}^j(m))$ is performed as follows:

$$p\left(\mathbf{x}(m)|\mathcal{Z}_i^{m-1}\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(m-1)|\mathcal{Z}_i^{m-1}\right), \tag{6.37}$$

$$p\left(\mathbf{x}(m)|\mathcal{Z}_i^{m}\right) \sim \frac{p\left(\mathbf{x}(m)|\mathcal{Z}_i^{m-1}\right) p\left(\mathbf{x}(m)|\mathbf{z}^j(m)\right)}{p\left(\mathbf{x}(m)\right)}, \tag{6.38}$$

and for $k = k_1, k_2, \ldots, k_K$, with $m_0 = m$,

$$p\left(\mathbf{x}(m_k)|\mathcal{Z}_i^{m_k-1}\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(m_{k-1})|\mathcal{Z}_i^{m_{k-1}}\right), \tag{6.39}$$

$$p\left(\mathbf{x}(m_k)|\mathcal{Z}_i^{m_k}\right) \sim \frac{p\left(\mathbf{x}(m_k)|\mathcal{Z}_i^{m_k-1}\right) p\left(\mathbf{x}(m_k)|\mathbf{z}^k(m_k)\right)}{p\left(\mathbf{x}(m_k)\right)}, \tag{6.40}$$

---

* $\mathcal{Z}_i^{m-1}$ is not necessarily a *cumulative* measurement set, since it may contain measurements (obtained at other nodes at times $\le t_{m-1}$) for which information communication was delayed such that the information was received at node j at times $> t_{m-1}$.

† Recall that $t_{m-1} \le t_m$, since measurement is performed asynchronously throughout the network.

finally obtaining $\hat{y}(n|Z_i^n)$, $Y(n|Z_i^n)$. In cases where $t_{m_k} = t_{m_{k-1}}$ due to simultaneous measurements, the prediction step (6.39) may be omitted. Due to SSP reprocessing (6.40), the computation and memory requirement of the optimal algorithm is $\mathcal{O}(K+1)$ times greater in the case $m < n$, compared with the case $m \geq n$.

6.5 Approximate Distributed State Estimation

The optimal distributed estimation algorithms of Section 6.4 are sufficiently resource efficient for many practical applications, but there will be other applications for which greater efficiency is required. In Section 6.5.1, two approximate delayed information fusion algorithms for improved computation and memory resource efficiency are presented, and in Section 6.5.2, information aggregation is introduced for improved communications efficiency.

6.5.1 Computation and Memory Resource Efficiency

For optimal fusion of delayed SSPs ($m < n$) using (6.37) through (6.40), the computational complexity is $\mathcal{O}((K+1)N_x^3)$, which can be reduced to $\mathcal{O}(N_x^3)$, with a similar reduction in memory requirements, using the following *predicted information fusion* (PIF) algorithm:

$$p\left(x(n)|z^j(m)\right) \overset{\text{Pred}}{\Leftarrow} p\left(x(m)|z^j(m)\right), \tag{6.41}$$

$$p\left(x(n)|Z_i^n\right) \sim \frac{p\left(x(n)|Z_i^{n-}\right)p\left(x(n)|z^j(m)\right)}{p\left(x(n)\right)}. \tag{6.42}$$

From Section 6.3, (6.42) results from the conditional independence approximation

$$p\left(Z_i^{n-} \cup z^j(m)|x(n)\right) \approx p\left(Z_i^{n-}|x(n)\right)p\left(z^j(m)|x(n)\right), \tag{6.43}$$

which holds exactly if the states are time invariant. Notice that for $m = n$, (6.42) is the same as (6.34); that is, the fusion is optimal if the SSP communication delay is 0.

As an alternative to the PIF algorithm, consider the following, which is based on the approximate OOSM fusion algorithm A*l*1 of [3]. For $m < n$, compute $\hat{y}(m|\mathcal{Z}_i^{m-1})$, $Y(m|\mathcal{Z}_i^{m-1})$ and $\hat{y}(n|\mathcal{Z}_i^{m-1})$, $Y(n|\mathcal{Z}_i^{m-1})$ via prediction:

$$\begin{aligned} p\left(x(m)|\mathcal{Z}_i^{m-1}\right) &\overset{\text{Pred}}{\Leftarrow} p\left(x(m-1)|\mathcal{Z}_i^{m-1}\right), \\ p\left(x(n)|\mathcal{Z}_i^{m-1}\right) &\overset{\text{Pred}}{\Leftarrow} p\left(x(m)|\mathcal{Z}_i^{m-1}\right). \end{aligned} \tag{6.44}$$

Then form an equivalent measurement [3] $\mathbf{z}^e(n)$ to represent the information in $\{\mathbf{z}^{k_1}(m+1), \mathbf{z}^{k_2}(m+2), \ldots, \mathbf{z}^{k_K}(n)\}$:

$$p\left(x(n)|z^e(n)\right) \approx \frac{1}{c}\frac{p\left(x(n)\right)p\left(x(n)|Z_i^{n-}\right)}{p\left(x(n)|\mathcal{Z}_i^{m-1}\right)}, \tag{6.45}$$

where t_{m-1} is defined such that $t_{m-1} \le t_m$, for asynchronous measurement times. The assumed equivalent measurement model is

$$\mathbf{z}^e(n) \approx \mathbf{H}^e(n)\mathbf{x}(n) + \mathbf{w}^e(n), \tag{6.46}$$

where

$\mathbf{H}^e(n) = \mathbf{I}$ (i.e., $\mathbf{z}^e(n)$ is a direct observation of the state)
$\mathbf{w}^e(n)$ is white Gaussian noise

However, while $\mathbf{w}^e(n)$ is uncorrelated with $\mathbf{v}(m')$, for $m' \in \{0, \ldots, m-1\}$, $\mathbf{w}^e(n)$ is only approximately uncorrelated with the process noise $\mathbf{v}(n, m)$. The primary benefit of forming the equivalent measurement is that the delayed SSP $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid \mathbf{z}^j(m))$ can be fused (optimally) at time m, the time at which it provides the most information on the state:

$$p\left(\mathbf{x}(m) \mid \mathcal{Z}_i^m\right) = \frac{p\left(\mathbf{x}(m) \mid \mathcal{Z}_i^{m-1}\right) p\left(\mathbf{x}(m) \mid \mathbf{z}^j(m)\right)}{p\left(\mathbf{x}(m)\right)}, \tag{6.47}$$

since $\mathbf{z}^j(m)$ conditioned on $\mathbf{x}(m)$ is independent of $\mathcal{Z}_i^{m-1}$. Hence, the prediction, including information from the delayed SSP, can be formed at the current time n:

$$p\left(\mathbf{x}(n) \mid \mathcal{Z}_i^m\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(m) \mid \mathcal{Z}_i^m\right). \tag{6.48}$$

Finally, given the assumed equivalent measurement model (6.46), in which $\mathbf{z}^e(n)$ is independent of $\mathcal{Z}_i^m$ if $\mathbf{x}(n)$ is given, the fusion formula (6.18) can be employed:

$$p\left(\mathbf{x}(n) \mid \mathbf{Z}_i^n\right) \sim \frac{p\left(\mathbf{x}(n) \mid \mathcal{Z}_i^m\right) p\left(\mathbf{x}(n) \mid \mathbf{z}^e(n)\right)}{p\left(\mathbf{x}(n)\right)}. \tag{6.49}$$

In this algorithm, SSP reprocessing is avoided. However, subsequent to delayed SSP fusion, the stored sufficient statistics for $p(\mathbf{x}(m'') \mid \mathcal{Z}_i^{m''})$, $m < m'' < n$, are no longer valid, unless they are updated to include the information in $\mathbf{z}^j(m)$. If the updates are performed, the computational complexity of this approximate fusion algorithm is greater than for optimal SSP reprocessing. Thus, we introduce the following equivalent information fusion (EIF) algorithm, which overcomes this difficulty to provide accurate, efficient delayed information fusion.

In large networks, it may be frequently necessary to fuse delayed SSPs. With n denoting the current time, let $n - L$ denote the earliest time of SSP origin for which delayed SSPs will be fused. The selection of $n - L$ is based on a trade-off between estimation accuracy and computation and storage efficiency. For times of SSP origin earlier than $n - L$, delayed SSPs received at a fusion node may be discarded (if the information is nearly irrelevant to the current state) or fused with the PIF algorithm, which does not require storing past state estimates or SSPs. The EIF algorithm provides efficient fusion of $\mathbf{z}_j^m$ into $\mathbf{Z}_i^{n-}$ to form $\mathbf{Z}_i^n = \mathbf{Z}_i^{n-} \bigcup \mathbf{z}_j^m$, for $n - L \le m \le n$. It is assumed that sufficient statistics for $p(\mathbf{x}(n-L) \mid \mathcal{Z}_i^{n-L})$, $p(\mathbf{x}(n) \mid \mathbf{Z}_i^{n-})$, and $p(\mathbf{x}(n-L+l) \mid \mathbf{z}^{kj}(n-L+l))$, for $l = 1, 2, \ldots, L$, are stored. Sufficient statistics for the equivalent measurement $\mathbf{z}^{e'}(n)$ representing the information contained in $\{\mathbf{z}^{kj}(n-L+l)\} \forall l$ are computed, following prediction of the stored state estimate:

$$p\left(\mathbf{x}(n) \mid \mathcal{Z}_i^{n-L}\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(n-L) \mid \mathcal{Z}_i^{n-L}\right), \tag{6.50}$$

$$p\left(\mathbf{x}(n)\,|\,\mathcal{Z}_i^{n-L}\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(m)\,|\,\mathcal{Z}_i^{n-L}\right), \tag{6.51}$$

$$p\left(\mathbf{x}(n)\,|\,\mathbf{z}^{e'}(n)\right) \sim \frac{p\left(\mathbf{x}(n)\right)p\left(\mathbf{x}(n)\,|\,Z_i^{n-}\right)}{p\left(\mathbf{x}(n)\,|\,\mathcal{Z}_i^{n-L}\right)}. \tag{6.52}$$

Fusion of the delayed SSP at time m is given by

$$p\left(\mathbf{x}(m)\,|\,\mathcal{Z}_i^{n-L} \cup \mathbf{z}^j(m)\right) \propto \frac{p\left(\mathbf{x}(m)\,|\,\mathcal{Z}_i^{n-L}\right)p\left(\mathbf{x}(m)\,|\,\mathbf{z}^j(m)\right)}{p\left(\mathbf{x}(m)\right)}. \tag{6.53}$$

Then the updated current state estimate is computed as

$$p\left(\mathbf{x}(n)\,|\,\mathcal{Z}_i^{n-L} \cup \mathbf{z}^j(m)\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(m)\,|\,\mathcal{Z}_i^{n-L} \cup \mathbf{z}^j(m)\right), \tag{6.54}$$

$$p\left(\mathbf{x}(n)\,|\,Z_i^{n}\right) \propto \frac{p\left(\mathbf{x}(n)\,|\,\mathcal{Z}_i^{n-L} \cup \mathbf{z}^j(m)\right)p\left(\mathbf{x}(n)\,|\,\mathbf{z}^{e'}(n)\right)}{p\left(\mathbf{x}(n)\right)}. \tag{6.55}$$

Finally, the stored state estimate is updated as

$$p\left(\mathbf{x}(n-L+1)\,|\,\mathcal{Z}_i^{n-L}\right) \overset{\text{Pred}}{\Leftarrow} p\left(\mathbf{x}(n-L)\,|\,\mathcal{Z}_i^{n-L}\right), \tag{6.56}$$

$$p\left(\mathbf{x}(n-L+1)\,|\,\mathcal{Z}_i^{n-L+1}\right) \propto \frac{p\left(\mathbf{x}(n-L+1)\,|\,\mathcal{Z}_i^{n-L}\right)p\left(\mathbf{x}(n-L+1)\,|\,\mathbf{z}^{k_1}(n-L+1)\right)}{p\left(\mathbf{x}(n-L+1)\right)}. \tag{6.57}$$

To maintain the constant storage requirement, the last entry in $(p\mathbf{x}(n-L+l)\,|\,\mathbf{z}^{k_l}(n-L+l))$, $l = 1, 2, \ldots, L$, is then deleted, and the entries are shifted, as needed, to accommodate storage of (sufficient statistics for) $p(\mathbf{x}(m)\,|\,\mathbf{z}^j(m))$. Note that the stored state estimate is updated optimally (by reprocessing one SSP during each delayed SSP fusion cycle), but the current state estimate is updated approximately, due to the independence assumptions on the equivalent measurement noise. The EIF algorithm is specified in detail in Figures 6.3 and 6.4.

The EIF algorithm can be easily extended such that after a specified large number of fusion cycles, the current state estimate is updated optimally by reprocessing the stored $p(\mathbf{x}(n-L+l)\,|\,\mathbf{z}^{k_l}(n-L+l))$, $l = 1, 2, \ldots, L$. For the experiments described in Section 6.8, the performance of the EIF algorithm was evaluated with and without infrequent reprocessing of stored SSPs, and in the time frame examined, the difference in performance was insignificant. The benefits of the EIF algorithm, compared with the optimal algorithm, are a nearly twofold reduction in memory requirements (for large L) and $\mathcal{O}(N_x^3)$ computational complexity. Furthermore, for the experiments described in Section 6.8, the difference in estimation performance for the EIF and optimal algorithms was insignificantly small.

6.5.2 Communications Resource Efficiency

The algorithms of Sections 6.4 and 6.5.1 can be modified to obtain an approximate distributed state estimation algorithm, which is parameterizable in the trade-off of improved communications efficiency

Receive:

$$\text{SSP containing } \hat{\mathbf{y}}\left(m \mid \mathbf{z}^j(m)\right), \mathbf{Y}\left(m \mid \mathbf{z}^j(m)\right), j, t_m.$$

If Delayed ($n > m$):

Given:

$$\mathbf{F}(n-L+1, n-L), \mathbf{Q}(n-L+1, n-L), \mathbf{F}(m, n-L), \mathbf{Q}(m, n-L),$$
$$\mathbf{F}(n,m), \mathbf{Q}(n,m), \bar{\mathbf{y}}(n-L+1), \mathbf{Y}(n-L+1), \bar{\mathbf{y}}(n), \mathbf{Y}(n),$$
$$\hat{\mathbf{y}}\left(n \mid \mathbf{Z}_i^{n-}\right), \mathbf{Y}\left(n \mid \mathbf{Z}_i^{n-}\right), \hat{\mathbf{y}}\left(n-L \mid \mathbf{Z}_i^{n-L}\right), \mathbf{Y}\left(n-L \mid \mathbf{Z}_i^{n-L}\right),$$
$$\hat{\mathbf{y}}\left(n-L+l \mid \mathbf{z}^{k_l}(n-L+l)\right), \mathbf{Y}\left(n-L+l \mid \mathbf{z}^{k_l}(n-L+l)\right), \quad l = 1, 2, \ldots, L.$$

Compute:

$$\mathbf{Y}\left(m \mid \mathbf{Z}_i^{n-L}\right) = \left[\mathbf{F}(m, n-L)\mathbf{Y}^{-1}\left(n-L \mid \mathbf{Z}_i^{n-L}\right)\mathbf{F}^T(m, n-L) + \mathbf{Q}(m, n-L)\right]^{-1}$$
$$\hat{\mathbf{y}}\left(m \mid \mathbf{Z}_i^{n-L}\right) = \mathbf{Y}\left(m \mid \mathbf{Z}_i^{n-L}\right)\mathbf{F}(m, n-L)\mathbf{Y}^{-1}\left(n-L \mid \mathbf{Z}_i^{n-L}\right)\hat{\mathbf{y}}\left(n-L \mid \mathbf{Z}_i^{n-L}\right)$$
$$\mathbf{Y}\left(n \mid \mathbf{Z}_i^{n-L}\right) = \left[\mathbf{F}(n,m)\mathbf{Y}^{-1}\left(m \mid \mathbf{Z}_i^{n-L}\right)\mathbf{F}^T(n,m) + \mathbf{Q}(n,m)\right]^{-1}$$
$$\hat{\mathbf{y}}\left(n \mid \mathbf{Z}_i^{n-L}\right) = \mathbf{Y}\left(n \mid \mathbf{Z}_i^{n-L}\right)\mathbf{F}(n,m)\mathbf{Y}^{-1}\left(m \mid \mathbf{Z}_i^{n-L}\right)\hat{\mathbf{y}}\left(m \mid \mathbf{Z}_i^{n-L}\right)$$
$$\mathbf{Y}\left(n \mid \mathbf{z}^{e'}(n)\right) = \mathbf{Y}(n) + \mathbf{Y}\left(n \mid \mathbf{Z}_i^{n-}\right) - \mathbf{Y}\left(n \mid \mathbf{Z}_i^{n-L}\right)$$
$$\hat{\mathbf{y}}\left(n \mid \mathbf{z}^{e'}(n)\right) = \bar{\mathbf{y}}(n) + \hat{\mathbf{y}}\left(n \mid \mathbf{Z}_i^{n-}\right) - \hat{\mathbf{y}}\left(n \mid \mathbf{Z}_i^{n-L}\right)$$
$$\mathbf{Y}\left(m \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m)\right) = \mathbf{Y}\left(m \mid \mathbf{Z}_i^{n-L}\right) + \mathbf{Y}\left(m \mid \mathbf{z}^j(m)\right) - \mathbf{Y}(m)$$
$$\hat{\mathbf{y}}\left(m \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m)\right) = \hat{\mathbf{y}}\left(m \mid \mathbf{Z}_i^{n-L}\right) + \hat{\mathbf{y}}\left(m \mid \mathbf{z}^j(m)\right) - \bar{\mathbf{y}}(m)$$

FIGURE 6.3 Fusion node processing/SSP fusion: EIF algorithm, delayed case (Part A).

for increased latency in the rate at which information propagates through the network. The approach is essentially the same, except that transmitted SSPs contain information for *blocks* of N_b consecutive measurements at a single node, rather than for a single measurement vector obtained at one node and one time.

SSP formation for improved communications efficiency is accomplished as follows. As each new measurement vector $\mathbf{z}^j(m)$ is obtained, the new information is extracted via (6.28) using the predicted and updated KF estimates. The extracted information $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid \mathbf{z}^j(m))$ is then fused into the SSP sufficient statistics $\hat{\mathbf{x}}(m \mid \mathcal{Z}_{\text{SSP}}^{m-1})$, $\mathbf{P}(m \mid \mathcal{Z}_{\text{SSP}}^{m-1})$ (predicted to the current time via the KF prediction equations), where $\mathcal{Z}_{\text{SSP}}^{m-1}$ is the set of measurements whose information has been previously aggregated into the SSP. The fusion of new information into the SSP is accomplished via another application of (6.18):

$$p\left(\mathbf{x}(m) \mid \mathcal{Z}_{\text{SSP}}^{m}\right) \propto \frac{p\left(\mathbf{x}(m) \mid \mathbf{z}^j(m)\right) p\left(\mathbf{x}(m) \mid \mathcal{Z}_{\text{SSP}}^{m-1}\right)}{p(\mathbf{x}(m))}. \tag{6.58}$$

When the full block of N_b measurements has been processed, formation of the SSP block is complete, and it is transmitted to nearby nodes. When node $i \neq j$ receives the SSP, it uses one of the approximate fusion algorithms of Section 6.5.1 to update its local estimate $\hat{\mathbf{y}}(m \mid \mathbf{Z}_i^n)$, $\mathbf{Y}(m \mid \mathbf{Z}_i^n)$ using $\hat{\mathbf{y}}(m \mid \mathcal{Z}_{\text{SSP}}^{m})$, $\mathbf{Y}(m \mid \mathcal{Z}_{\text{SSP}}^{m})$. The fusion algorithm assumes that $\hat{\mathbf{y}}(m \mid \mathcal{Z}_{\text{SSP}}^{m})$, $\mathbf{Y}(m \mid \mathcal{Z}_{\text{SSP}}^{m})$ can be fused in the same way as

$$\mathbf{Y}(n \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m))$$
$$= [\mathbf{F}(n,m)\mathbf{Y}^{-1}(m \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m))\mathbf{F}^T(n,m) + \mathbf{Q}(n,m)]^{-1}$$
$$\hat{\mathbf{y}}(n \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m))$$
$$= \mathbf{Y}(n \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m))\mathbf{F}(n,m)\mathbf{Y}^{-1}(m \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m))\hat{\mathbf{y}}(m \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m))$$
$$\mathbf{Y}(n \mid \mathbf{Z}_i^{n}) = \mathbf{Y}(n \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m)) + \mathbf{Y}(n \mid \mathbf{z}^{e^j}(n)) - \mathbf{Y}(n)$$
$$\hat{\mathbf{y}}(n \mid \mathbf{Z}_i^{n}) = \hat{\mathbf{y}}(n \mid \mathbf{Z}_i^{n-L} \cup \mathbf{z}^j(m)) + \hat{\mathbf{y}}(n \mid \mathbf{z}^{e^j}(n)) - \bar{\mathbf{y}}(n)$$
$$\mathbf{Y}(n-L+1 \mid \mathbf{Z}_i^{n-L})$$
$$= [\mathbf{F}(n-L+1,n-L)\mathbf{Y}^{-1}\left(n-L \mid \mathbf{Z}_i^{n-L}\right)\mathbf{F}^T(n-L+1,n-L)$$
$$+ \mathbf{Q}(n-L+1,n-L)]^{-1}$$
$$\hat{\mathbf{y}}(n-L+1 \mid \mathbf{Z}_i^{n-L})$$
$$= \mathbf{Y}\left(n-L+1 \mid \mathbf{Z}_i^{n-L}\right)\mathbf{F}(n-L+1,n-L)\mathbf{Y}^{-1}\left(n-L \mid \mathbf{Z}_i^{n-L}\right)$$
$$\times \hat{\mathbf{y}}\left(n-L \mid \mathbf{Z}_i^{n-L}\right)$$
$$\mathbf{Y}\left(n-L+1 \mid \mathbf{Z}_i^{n-L+1}\right)$$
$$= \mathbf{Y}\left(n-L+1 \mid \mathbf{Z}_i^{n-L}\right) + \mathbf{Y}\left(n-L+1 \mid \mathbf{z}^{k_1}(n-L+1)\right) - \mathbf{Y}(n-L+1)$$
$$\hat{\mathbf{y}}(n-L+1 \mid \mathbf{Z}_i^{n-L+1})$$
$$= \hat{\mathbf{y}}(n-L+1 \mid \mathbf{Z}_i^{n-L}) + \hat{\mathbf{y}}\,(n-L+1 \mid \mathbf{z}^{k_1}(n-L+1)) - \bar{\mathbf{y}}(n-L+1)$$

Store:

$$\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m)), \mathbf{Y}(m \mid \mathbf{z}^j(m))$$
$$\text{in } \hat{\mathbf{y}}(\mathrm{n}-L+l \mid \mathbf{z}^{k_l}(n-L+l)), \mathbf{Y}(n-L+l \mid \mathbf{z}^{k_l}(n-L+l)), \quad l = 1, 2, ..., L.$$

FIGURE 6.4 Fusion node processing/SSP fusion: EIF algorithm, delayed case (Part B).

sufficient statistics $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid \mathbf{z}^j(m))$ from a single measurement, and hence, the fusion is suboptimal (the error in $\hat{\mathbf{y}}(\mathrm{n} \mid \mathcal{Z}_{\mathrm{SSP}}^{\mathrm{m}})$ is correlated with the process noise). However, the algorithm provides a trade-off that can be desirable in some cases for achieving improvements in resource efficiency. In Section 6.8, simulation results are presented for various choices of N_b, and it is shown that significant improvement in communications efficiency may be possible with minimal degradation of estimation accuracy.

6.6 Nonlinear State Estimation

The extension to nonlinear state estimation is now discussed, and it is shown in this and subsequent sections that the distributed algorithms of Sections 6.4 and 6.5 are well suited for the nonlinear case. Practical nonlinear state estimation algorithms are by necessity approximate, with the most popular algorithms being the EKF [2], UKF [26,28], Gaussian sum filters [2], and particle/sequential Monte Carlo filters [15,23,48]. The latter two require significantly more computation and cannot be employed in the efficient distributed estimation framework of Sections 6.4 and 6.5 without further approximations. In contrast, the EKF and UKF are computationally efficient and can be conveniently employed in distributed estimation, as shown in the following.

We consider the state dynamics and measurement models*

$$\mathbf{x}(n) = \mathbf{f}(\mathbf{x}(n-1)) + \mathbf{v}(n-1) \tag{6.59}$$

and

$$\mathbf{z}(n) = \mathbf{h}(\mathbf{x}(n)) + \mathbf{w}(n), \tag{6.60}$$

where **f** and **h** are linear or nonlinear functions. More generally, we denote the nonlinear component (**f** or **h**) of the state or measurement equation by

$$\mathbf{x}_g(n) = \mathbf{g}(\mathbf{x}(m)). \tag{6.61}$$

The EKF and UKF rely on linearization of $g(\cdot)$:

$$\mathbf{x}_g(n) \cong \mathbf{A}(n|m)\mathbf{x}(m) + \mathbf{b}(n|m). \tag{6.62}$$

In the EKF, the linearization is analytical and is based on a truncated (usually first-order) Taylor series expansion about the most recent local state estimates, $\hat{\mathbf{x}}(n-1|Z_1^{n-1})$ and $\hat{\mathbf{x}}(n|Z_1^{n-1})$. The EKF equations are obtained by applying the standard KF for the linearized model, and the effects of linearization errors are ignored. In contrast, it has been shown [34] that the UKF performs a *statistical linearization* (via weighted linear regression [18]) of the nonlinear function. The UKF was designed [27,28] to approximate the mean and covariance of the *a posteriori* distribution $p(\mathbf{x}_g(n)|Z_1^m)$ transformed through the true nonlinear function $g(\cdot)$. The approximation is based on selecting a set of $R \geq N_x + 1$ *sigma points* $\mathcal{X}_r(n-1|n-1)$ with sample mean $\hat{\mathbf{x}}(n-1|Z_1^{n-1})$ and covariance $\mathbf{P}_x(n-1|Z_1^{n-1})$ prior to propagation through the dynamics equation or $\mathcal{X}_r(n|n-1)$ with sample mean $\hat{\mathbf{x}}(n|Z_1^{n-1})$ and covariance $\mathbf{P}_x(n|Z_1^{n-1})$ prior to propagation through the measurement equation. In the following, $\mathcal{X}_r(m|l)$ is used to generically represent both cases, with appropriate substitutions in the prediction and update cases.

For the UKF implementation for radiolocation (Sections 6.6 and 6.7), we use the SUKF [26,41], in which the distances of the sigma points from the current state estimate are scaled by a parameter $0 \leq \alpha \leq 1$ and also by parameters $\kappa = 0$ (usually), $\beta = 2$ (for Gaussian distributions). The inclusion of α is necessary to ensure that the estimated covariance matrices are positive definite and because in the standard UKF the distances to the sigma points increase with increasing state dimension (resulting in nonlocal sampling of the nonlinear function). With the parameter choices $\kappa = 0$ and $\alpha = 1/\sqrt{N_x}$, the distances to the sigma points are made invariant to the state dimension N_x.

To compute the sigma points and weights, first the Cholesky decomposition is employed to calculate the square root **A** of $\mathbf{P}_x(m|Z_1^l)$:

$$\mathbf{A}^T\mathbf{A} = \mathbf{P}_x(m|Z_1^l), \tag{6.63}$$

where below we denote the rth row of **A** by $\mathbf{A}_r$. Then, following [41], Section 3.2.2,

$$l = \alpha^2(N_x + \kappa) - N_x, \tag{6.64}$$

$$\mathcal{X}_0(m|l) = \hat{\mathbf{x}}(m|Z_1^l), \tag{6.65}$$

* If the process or measurement noise is an argument of the nonlinear function, then Equations 6.59 and 6.60, and following, may be extended by augmenting the state with the noise terms.

$$\omega_0^m = \lambda/(N_x + \lambda), \tag{6.66}$$

$$\omega_0^c = \lambda/(N_x + \lambda) + 1 - \alpha^2 + \beta, \tag{6.67}$$

and for $r = 1, \ldots, N_x$:

$$\mathcal{X}_r(m \mid l) = \hat{x}(m \mid Z_i^l) + (\sqrt{N_x + \lambda})A_r^T, \tag{6.68}$$

$$\mathcal{X}_{r+N_x}(m \mid l) = \hat{x}(m \mid Z_i^l) - (\sqrt{N_x + \lambda})A_r^T, \tag{6.69}$$

$$\omega_r^m = 1/(2(N_x + \lambda)), \tag{6.70}$$

$$\omega_r^c = \omega_r^m, \omega_{r+N_x}^m = \omega_r^m, \omega_{r+N_x}^c = \omega_r^c. \tag{6.71}$$

Each sigma point is transformed according to

$$\mathcal{Y}_r(n \mid l) = g(\mathcal{X}_r(m \mid l)); \tag{6.72}$$

the resulting sample mean, covariance, and cross-covariance are given by

$$\hat{x}_g\left(n \mid Z_i^l\right) = \sum_{r=1}^{R} \omega_r \mathcal{Y}_r(n \mid l), \tag{6.73}$$

$$P_{x_g}\left(n \mid Z_i^l\right) = \sum_{r=1}^{R} \omega_r \left\{\mathcal{Y}_r(n \mid l) - \hat{x}_g\left(n \mid Z_i^l\right)\right\}\left\{\mathcal{Y}_r(n \mid l) - \hat{x}_g\left(n \mid Z_i^l\right)\right\}^T, \tag{6.74}$$

$$P_{xx_g}\left(n \mid Z_i^l\right) = \sum_{r=1}^{R} \omega_r \left\{\mathcal{X}_r(m \mid l) - \hat{x}\left(m \mid Z_i^l\right)\right\}\left\{\mathcal{Y}_r(n \mid l) - \hat{x}_g\left(n \mid Z_i^l\right)\right\}^T. \tag{6.75}$$

The statistical linearization can then be computed as [34]

$$A(n \mid m) = P_{xx_g}^T\left(n \mid Z_i^l\right) P_x^{-1}\left(m \mid Z_i^l\right), \tag{6.76}$$

$$b(n \mid m) = \hat{x}_g\left(n \mid Z_i^l\right) - A(n \mid m)\hat{x}\left(m \mid Z_i^l\right). \tag{6.77}$$

It can be shown [34] that this solution for $\mathbf{A}(n \mid m)$ and $\mathbf{b}(n \mid m)$ minimizes the weighted sum of squared errors:

$$J(n \mid m) = \sum_{r=1}^{R} \omega_r \left\{e_r^T(n \mid m) e_r(n \mid m)\right\}, \tag{6.78}$$

where

$$e_r(n\,|\,m) = \mathcal{Y}_r(n\,|\,1) - A(n\,|\,m)\mathcal{X}_r(m\,|\,1) - b(n\,|\,m). \tag{6.79}$$

Also,

$$P_{x_g}(n\,|\,1) = A(n\,|\,m)P_x^{-1}\left(m\,|\,Z_i^1\right)A(n\,|\,m)^T + P_{ee}(n\,|\,m), \tag{6.80}$$

with the first term quantifying the spread in uncertainty that is explained by the linearized model and the second term,

$$P_{ee}(n\,|\,m) = \sum_{r=1}^{R} \omega_r e_r(n\,|\,m)e_r^T(n\,|\,m), \tag{6.81}$$

quantifying the estimated increase in uncertainty due to the approximation employed. In the EKF, $\mathbf{P}_{ee}(n\,|\,m) = 0$ is assumed, which can quickly result in large errors in the state covariance estimate. This is one reason why the UKF/SUKF is expected to outperform the EKF. Another reason is that for general probability distributions, the linearization in the UKF/SUKF is accurate to the second order of the Taylor series expansion [28], whereas the EKF is only accurate to the first order. In Section 6.8, the superiority of the SUKF is clearly demonstrated for distributed radiolocation.

Whether the EKF or SUKF is employed, the information form of the KF prediction and update Equations 6.11 and 6.12 and 6.14 and 6.15 is then given by

$$Y\left(n\,|\,Z_i^{n-1}\right) = \left[A(n\,|\,m)Y^{-1}\left(n-1\,|\,Z_i^{n-1}\right)A^T(n\,|\,m) + Q(n,n-1)\right]^{-1}, \tag{6.82}$$

$$\hat{y}\left(n\,|\,Z_i^{n-1}\right) = Y\left(n\,|\,Z_i^{n-1}\right)\left[A(n\,|\,m)Y^{-1}\left(n-1\,|\,Z_i^{n-1}\right)\hat{y}\left(n-1\,|\,Z_i^{n-1}\right) + b(n\,|\,m)\right], \tag{6.83}$$

and

$$\hat{y}\left(n\,|\,Z_i^{n}\right) = \hat{y}\left(n\,|\,Z_i^{n-1}\right) + A^T(n\,|\,m)R^{-1}(n)\left[z(n) - b(n\,|\,m)\right], \tag{6.84}$$

$$Y\left(n\,|\,Z_i^{n}\right) = Y\left(n\,|\,Z_i^{n-1}\right) + A^T(n\,|\,m)R^{-1}(n)A(n\,|\,m). \tag{6.85}$$

Note that $\mathbf{A}(n\,|\,m)$ and $\mathbf{b}(n\,|\,m)$ are computed separately for the prediction and update equations, based on the nonlinear functions, $g = f$ and $g = h$, in the state and measurement equations, respectively. Furthermore, the distributed estimation algorithms are extended to the linearized case via the following substitutions:

Prediction equations

$$\mathbf{F}(n, m) \rightarrow \mathbf{A}(n\,|\,m), \tag{6.86}$$

$$F(n,m)Y^{-1}\left(m\,|\,Z_i^m\right)\hat{y}\left(m\,|\,Z_i^m\right) \rightarrow A(n\,|\,m)Y^{-1}\left(m\,|\,Z_i^m\right)\hat{y}\left(m\,|\,Z_i^m\right) + b(n\,|\,m), \tag{6.87}$$

Update equations

$$\mathbf{H}(n) \rightarrow \mathbf{A}(n \mid m), \tag{6.88}$$

$$\mathbf{z}(n) \rightarrow \mathbf{z}(n) - \mathbf{b}(n \mid m), \tag{6.89}$$

and proper substitutions of i, m, and n.

Due to linearization, the estimates obtained using the distributed algorithms of Sections 6.4 and 6.5 are approximations to the optimal minimum variance estimates, and the estimation errors are only approximately Gaussian. However, the effect of linearization is small if the local EKF or SUKF is closely tracking the states (which is aided by fusion of information communicated from other nodes). In fact, the utility of (6.28) for calculating sufficient statistics $\hat{\mathbf{y}}(m \mid \mathbf{z}^j(m))$, $\mathbf{Y}(m \mid \mathbf{z}^j(m))$ based on the current measurement vector $\mathbf{z}^j(m)$ is most evident in the context of nonlinear estimation. The reason for this can be seen as follows. The calculated density $p(\mathbf{x}(m) \mid \mathbf{z}^j(m))$ is the *a posteriori* distribution based on a single measurement vector $\mathbf{z}^j(m)$ and the *a priori* distribution $p(\mathbf{x}(m))$. For the sake of argument, ignore linearization errors. Then $p(\mathbf{x}(m) \mid \mathbf{z}^j(m))$ could be calculated by predicting the *a priori* density (which in practice may be based on some initial measurements shared among all nodes) to the current time and then calculating the updated density using $\mathbf{z}^j(m)$ in the local EKF or SUKF update equations. However, bringing linearization error back into the picture, in the nonlinear case, the estimation of $p(\mathbf{x}(m) \mid \mathbf{z}^j(m))$ would be severely inaccurate because the *a priori* mean predicted to the current time would likely be very far from the true state, and the predicted *a priori* covariance would be very large [2].* In contrast, (6.28) can provide an accurate method for estimating sufficient statistics for $p(\mathbf{x}(m) \mid \mathbf{z}^j(m))$. The utility of (6.28) is demonstrated in Section 6.8 for distributed EKF- and SUKF-based node localization using internode range measurements (nonlinear transformations of the node positions).

The method of Section 6.5.2 for improving communications efficiency at the expense of increased delays in information propagation should also be evaluated in the context of nonlinear estimation. Increasing the number N_b of measurements represented by one SSP could cause the local estimation errors at each node to increase, since the most recent measurements obtained at other nodes are unavailable locally. This increase in instantaneous estimation errors could make the linearization less accurate, leading to further increases in estimation errors. However, this effect may be negligible as long as the local estimation errors are kept small (e.g., by ensuring a sufficient local measurement rate). For the simulations described in Section 6.8, N_b was varied between 1 and 20.

6.7 Distributed Position Estimation

In wireless ad hoc networks in which GPS is unavailable to all or some fraction of the network nodes, localization can be based on a variety of measurements, as discussed at the beginning of this chapter. In the simulations described in Section 6.8, node positions are estimated using range measurements between pairs of nodes. Such range measurements can be obtained, for example, from RTT measurements. For a master node j located at $\mathbf{x}_j(m)$ obtaining range measurements to N_z ($N_z = 1$ for the simulations discussed) reference nodes $\mathbf{i} = i_1, i_2, \ldots, i_{N_z}$ located at $\mathbf{x}_i(m)$, the nonlinear measurement equation is given by

$$\mathbf{z}^j(m) = \mathbf{h}^m(\mathbf{x}(m)) + \mathbf{w}(m) \in \mathbb{R}^{N_z \times 1}, \tag{6.90}$$

where

$$\mathbf{h}^m(\mathbf{x}(m)) = \left[\rho_{j,i_1}(m)\ \rho_{j,i_2}(m) \ldots \rho_{i,i_{N_z}}(m)\right]^T \tag{6.91}$$

* In [2], it is shown that the linearization is highly accurate if the state estimate covariance is very small.

and $\rho_{j,i}(m) = ||\mathbf{x}_i(m) - \mathbf{x}_j(m)||$. Node position estimation is thus complicated by the nonlinearity and state coupling inherent in the measurement equation. In this chapter, time-varying node positions are estimated and tracked using the EKF or SUKF. Other approaches to position estimation based on range measurements have included methods based on semidefinite programming [14] and local coordinate system construction based on geometry [10]. Combinations of the local coordinate and an EKF approach were developed in [50,51]. Indeed, for time-varying states, a KF-based algorithm is best able to provide the required position tracking capabilities.

For the state dynamics, a common linear model, the discretized *continuous white noise acceleration model* [5], is used for the simulations presented in Section 6.8. In this model, position $y(n)$ and velocity $\upsilon_y(n)$ in one coordinate are described by

$$\mathbf{y}_{1D}(n) = \mathbf{F}(n, m)\mathbf{y}_{1D}(m) + \mathbf{v}(n, m), \tag{6.92}$$

where

$$\mathbf{y}_{1D}(n) = \left[y(n), \upsilon_y(n)\right]^T, \tag{6.93}$$

$$\mathbf{F}(n,m) = \begin{bmatrix} 1 & t_n - t_m \\ 0 & 1 \end{bmatrix}, \tag{6.94}$$

and $\mathbf{v}(n, m)$ has covariance

$$\mathbf{Q}(n,m) = \begin{bmatrix} \frac{1}{3}(t_n - t_m)^3 & \frac{1}{2}(t_n - t_m)^2 \\ \frac{1}{2}(t_n - t_m)^2 & (t_n - t_m) \end{bmatrix} \sigma^2_{\Delta\upsilon}. \tag{6.95}$$

The overall state vector $\mathbf{x}(n)$ is of dimension $N_x = 4N_n$, for a network of N_n nodes distributed in two dimensions. Each node estimates the entire vector $\mathbf{x}(n)$ locally. The preceding is necessary due to the coupled nature of the node pair positions in range measurements, if there is uncertainty in both node locations. The resulting cross-correlations between position and velocity coordinate estimates for all nodes must be accounted for in order to avoid information loops leading to instability [57]. Since the complexity of the overall distributed estimation is $\mathcal{O}(N_x^3)$, in large networks, the formation of smaller subnetworks, though suboptimal, is a practical necessity. The linearization of (6.90), as required for the EKF algorithm, is specified in [7,30] and is specified for the SUKF in the previous section.

6.8 Simulation Results and Discussion

The distributed estimation algorithms of Sections 6.4 and 6.5 and the EKF- and SUKF-based position estimation and tracking algorithms of Sections 6.6 and 6.7 were simulated for a network of $N_n = 20$ nodes. All nodes were randomly deployed in a network area of 667 by 667 m and moved according to the white noise acceleration model of Section 6.7. The node initial velocity standard deviation was 10 m/s, and the acceleration standard deviation was $\sigma_{\Delta\upsilon} = 1$ m/s^2. The measurement and one-hop communication range of each node was limited to 275 m (a shorter range could be used in denser networks), and each node ranged (with measurement errors of 5 m standard deviation) to its nearest five neighbors (*if within range*) twice per second, on average (intermeasurement times were drawn from a Rayleigh distribution). Nodes communicated SSPs to neighbors located a maximum of $N_h = 1$, 2, 3, or 4 hops away, where the number of hops was modeled according to the distance $d(m)$ between source and destination

nodes: $0 \le d \le 275 \rightarrow 1$ hop, $275 < d \le 478 \rightarrow 2$ hops, $478 < d \le 662 \rightarrow 3$ hops, and $662 < d \le 834 \rightarrow 4$ hops. Source nodes communicated SSPs to one-hop neighbors with a delay (including processing delay) of approximately 0.15 s, and the communications delay for any additional hops was approximately 0.1 s, per hop. The standard deviation for initial node position and velocity estimates was 150 m and 5 m/s, respectively, for 70% of the nodes. The other 30% of nodes were *assisted* (e.g., via GPS) and obtained independent estimates of their own position and velocity once per second with error standard deviations of 10 m and 0.333 m/s, respectively. The existence of *assisted* nodes was assumed in order to enable geographic, as opposed to relative, localization (the assisted nodes serve a role similar to that of *anchor* or *beacon* nodes in the localization literature). Because estimation errors are coupled across nodes (as discussed in Section 6.7), assisted nodes communicated measurements of their own positions to other nodes up to N_h hops away, with the same communication delays as for SSPs.

The performance of the EKF- and SUKF-based node localization algorithms was first compared for centralized estimation in which all measurements were available without delays. The resulting mean absolute position estimation error traces, averaged over 500 ensemble runs, are shown versus measurement time index in Figure 6.5. Clearly, the SUKF-based algorithm outperformed the EKF-based localization algorithm, with the best performance obtained for smaller values (0.01–0.5) of the SUKF scaling parameter α. In the best case, 6.9 m mean absolute position estimation error was obtained in steady state for the SUKF-based algorithm, compared with 38.3 m for the EKF-based algorithm. In the remaining simulations, $\alpha = 1/\sqrt{N_x} \approx 0.12$ was used. To demonstrate the robustness of the SUKF algorithm, the range measurement standard deviation was varied from 1 to 50 m, and the resulting mean absolute position estimation error traces, averaged over 100 ensemble runs, are shown versus measurement time index in Figure 6.6.

The performance of the approximate delayed PIF and EIF distributed algorithms was then compared with that of the centralized SUKF. In Figure 6.7, mean absolute position estimation error traces, averaged over 100 ensemble runs, are shown. Notably, nearly identical performance was obtained by the distributed EIF and centralized SUKF algorithms for an SSP block size of N_b =1 measurement, even though communication delays were included for the distributed algorithm, while the centralized algorithm processed all measurements with no delays. Furthermore, the estimation performance of the EIF algorithm is shown to degrade gracefully with increasing SSP block size, and the EIF outperforms the PIF algorithm by a large margin (Figure 6.7).

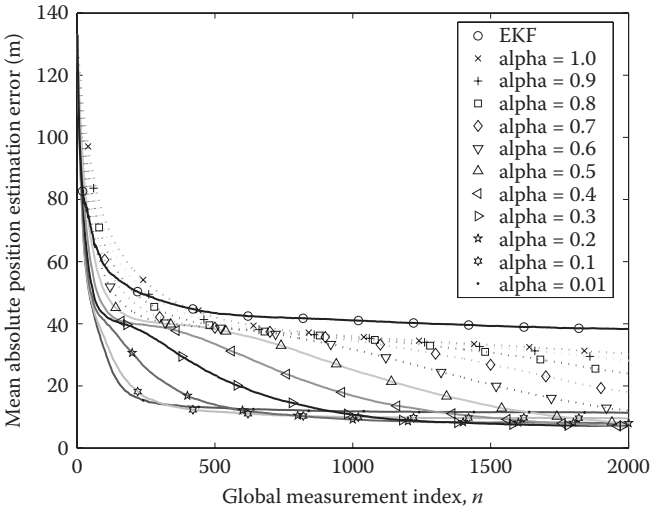


FIGURE 6.5 Comparison of the EKF- and SUKF-based node localization algorithms, for centralized estimation in which all measurements were available without delays (α is the SUKF scaling parameter).

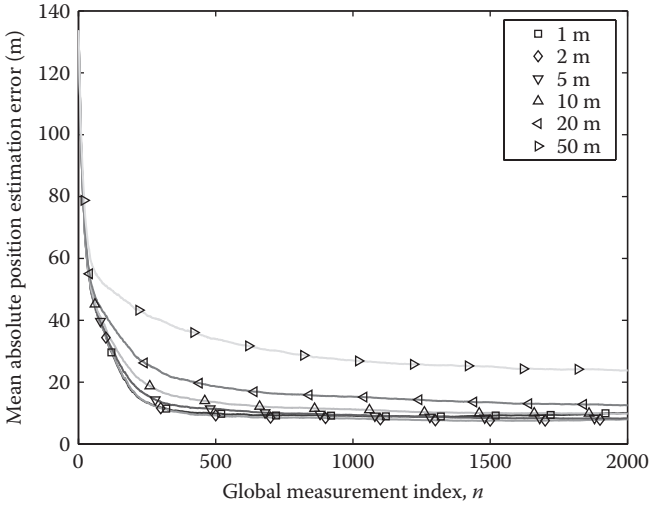


FIGURE 6.6 Comparison of SUKF-based centralized estimation performance for different values of range measurement standard deviation (values shown in the legend).

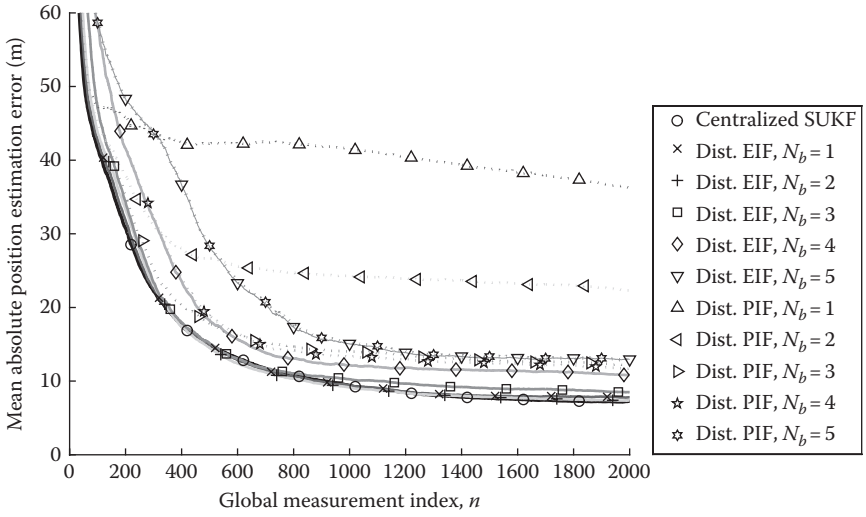


FIGURE 6.7 Comparison of the centralized SUKF and approximate delayed PIF and EIF distributed algorithms.

Finally, the performance of the distributed EIF and PIF algorithms was compared versus SSP block size and the maximum number of hops, N_h, over which information was communicated. The final/steady-state mean absolute position estimation errors are plotted in Figures 6.8 and 6.9 for $N_h = 1$ (Figure 6.8) and $N_h = 2, 3, 4$ (Figure 6.9). In these experiments, significant savings in communication resources are achieved for smaller values of N_h and/or larger values of N_b, and little degradation in estimation accuracy is incurred for SSP block sizes of up to $N_b = 5$, using the EIF algorithm. Furthermore, the performance of the EIF algorithm degrades gracefully with increasing N_b. Interestingly, these results suggest that at larger values of N_b ($\geq$20), the PIF algorithm may achieve nearly the same estimation accuracy as the EIF algorithm. Thus, at larger values of N_b, the PIF algorithm may be preferred due to computational and memory resource efficiency, while at smaller values of N_b, the EIF algorithm is preferred when accurate estimation is required. Together, these results demonstrate the utility of the

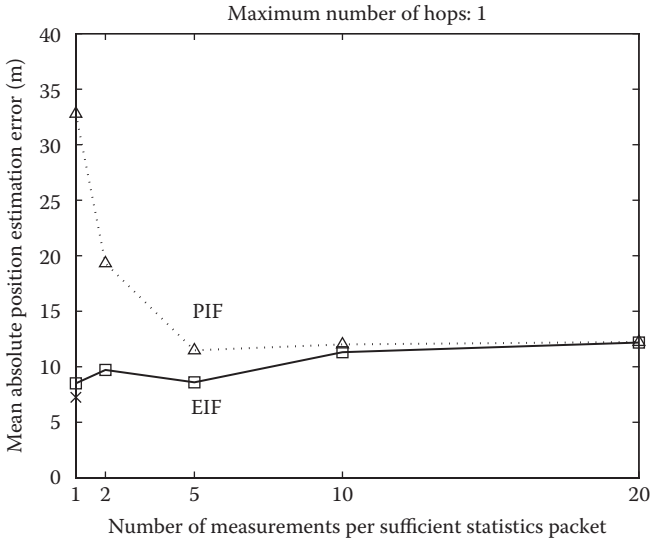


FIGURE 6.8 Comparison of the distributed EIF and PIF algorithms versus SSP block size, for information communication over a single hop.

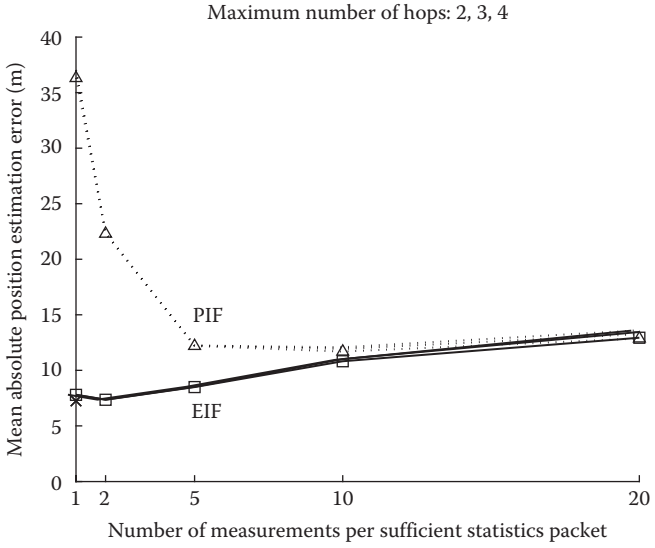


FIGURE 6.9 Comparison of the distributed EIF and PIF algorithms versus SSP block size, for information communication over a maximum of 2, 3, or 4 hops.

Bayesian-fusion-based distributed estimation framework and verify the performance of the approximate distributed algorithms for fusing delayed information packets in ad hoc networks of mobile nodes.

6.9 Summary

This chapter provides a detailed discussion of distributed estimation algorithms, with an example application of sensor node localization in mobile ad hoc networks. The new contributions of this chapter include (1) generalized and simplified formulations of optimal and approximate information aggregation

and fusion algorithms well suited to nonlinear estimation, (2) a new approximate EIF algorithm for fusion of delayed information, with accuracy approaching that of the optimal fusion algorithm, and (3) demonstration of resource-efficient, accurate distributed estimation of mobile node positions using simulated range measurements in an ad hoc network with significant communication delays. Metrics considered include measures of estimation accuracy, as well as resource efficiency, in terms of computation, storage, and communication requirements.

References

1. Akyildiz, I.F., W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communication Magazine*, 40(8):102–114, August 2002.
2. Anderson, B. and J. Moore. *Optimal Filtering.* Prentice-Hall, Englewood Cliffs, NJ, 1979.
3. Bar-Shalom, Y., H. Chen, and M. Mallick. One-step solution for the multistep out-of-sequence-measurement problem in tracking. *IEEE Transactions on Aerospace and Electrical System*, 40(1):27–37, January 2004.
4. Bar-Shalom, Y. and T.E. Fortmann. *Tracking and Data Association.* Academic Press, Orlando, FL, 1988.
5. Bar-Shalom, Y., X.R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. Wiley-Interscience, New York, 2001.
6. Bar-Shalom, Y. and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11(5):451–460, September 1975.
7. Brown, A.P. and R.A. Iltis. Distributed terrestrial radiolocation using the RLS algorithm. In *Proceedings of the International Telemetering Conference, ITC 2002,* San Diego, CA, October 2002.
8. Bulusu, N., J. Heidemann, and D. Estrin. GPS-less low-cost outdoor localization for very small devices. *IEEE Personal Communication*, 7(5):28–34, October 2000.
9. Caffery, J.J. Jr. *Wireless Location in CDMA Cellular Radio Systems.* Kluwer Academic, Boston, MA, 2000.
10. Capkun, S., M. Hamdi, and J. Hubaux. GPS-free positioning in mobile ad-hoc networks. *Cluster Computing*, 5:157–167, 2002.
11. Chong, C.Y. Hierarchical estimation. In *Proceedings of the MIT/ONR Workshop on C3*, Monterey, CA, 1979.
12. Chong, C.Y., S. Mori, W.H. Barker, and K.C. Chang. Architectures and algorithms for track association and fusion. *IEEE AES Systems Magazine*, 15(1):5–13, January 2000.
13. Chong, C.Y., E. Tse, and S. Mori. Distributed estimation in networks. In *Proceedings of the 1983 American Control Conference*, San Francisco, CA, September 1983, vol. 1, pp. 294–300.
14. Doherty, L., K.S.J. Pister, and L.E. Ghaoui. Convex position estimation in wireless sensor networks. In *Proceedings IEEE INFOCOM 2001*, Anchorage, AK, 2001, vol. 3, pp. 1655–1663.
15. Doucet, A., N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice.* Springer, New York, 2001.
16. Farley, M. and S. Carlson. A new pseudolite battlefield navigation system. In *IEEE 1998 Position Location and Navigation Symposium*, New York, April 1998, pp. 208–217.
17. Grime, S. and H.F. Durrant-Whyte. Data fusion in decentralized sensor networks. *Control Engineering Practice*, 2(5):849–863, 1994.
18. Grob, J. *Linear Regression.* Springer, New York, 2003.
19. Haeberlen, A., E. Flannery, A. Ladd, A. Rudys, D. Wallach, and L. Kavraki. Practical robust localization over large-scale 802.11 wireless networks. In *MobiCom 2004*, Philadelphia, PA, September 2004.
20. Hata, M. and T. Nagatsu. Mobile location using signal strength measurements in a cellular system. *IEEE Transactions on Vehicular Technology*, vt-29:245–252, May 1980.
21. Hellebrandt, M. and R. Mathar. Location tracking of mobiles in cellular radio networks. *IEEE Transactions on Vehicular Technology*, 48(5):1558–1562, September 1999.

22. Hu, L. and D. Evans. Localization for mobile sensor networks. In *MobiCom 2004*, Philadelphia, PA, September 2004.
23. Ihler, A.T., J.W. Fisher, R.L. Moses, and A.S. Willsky. Nonparametric belief propagation for self-calibration in sensor networks. In *Proceedings of the IEEE ICASSP*, Montreal, Quebec, Canada, May 2004.
24. Iltis, R.A. and S. Kim. Geometric derivation of expectation-maximization and generalized successive interference cancellation algorithms with CDMA channel estimation. *IEEE Transactions on Signal Processing*, 51(5):1367–1377, May 2003.
25. Joa-Ng, M. and I.-T. Lu. A novel spread spectrum-based synchronization and location determination method for wireless system. *IEEE Communications Letters*, 3(6):177–179, June 1999.
26. Julier, S.J. The scaled unscented transformation. In *Proceedings of American Control Conference*, Anchorage, AK, 2002, pp. 4555–4559.
27. Julier, S.J. and J.K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *Proceedings of AeroSense: 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, Orlando, FL, 1997, pp. 182–193.
28. Julier, S.J. and J.K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, March 2004.
29. Kalman, R.E. A new approach to linear filtering and prediction problems. *Transactions ASMEJ Basic Engineering*, 82:32–45, March 1960.
30. Kim, S., A.P. Brown, T. Pals, R.A. Iltis, and H. Lee. Geolocation in ad hoc networks using DS-CDMA and generalized successive interference cancellation. *IEEE Journal on Selected Areas in Communications*, 23(5):984–998, May 2005.
31. Kim, S. and R.A. Iltis. A matching pursuit/GSIC-based algorithm for DS-CDMA sparse channel estimation. *IEEE Signal Processing Letters*, 11(1):12–15, January 2004.
32. Kim, S., T. Pals, R.A. Iltis, and H. Lee. CDMA multipath channel estimation using generalized successive interference cancellation algorithm for radiolocation. *Proceedings of 37th Annual Conference on Information Sciences and Systems*, Princeton, NJ, March 2002.
33. Lee, S.J., J. Hsu, R. Hayashida, M. Gerla, and R. Bagrodia. Selecting a routing strategy for your ad hoc network. *Computer Communications*, 26(7):723–733, May 2003.
34. Lefebvre, T., H. Bruyninckx, and J.D. Schutter. Comment on 'a new method for the nonlinear transformation of means and covariances in filters and estimators'. *IEEE Transactions on Automatic Control*, 47(8):1406–1408, August 2002.
35. Li, X.R., Y. Zhu, J. Wang, and C. Han. Optimal linear estimation fusion–I: Unified fusion rules. *IEEE Transactions on Information Theory*, 49(9):2192–2208, September 2003.
36. Liggins, M.E., C.Y. Chong, I. Kadar, M.G. Alford, V. Vannicola, and S. Thomopoulos. Distributed fusion architectures and algorithms for target tracking. *Proceedings of the IEEE*, 85(1):95–107, January 1997.
37. Mallick, M. and K. Zhang. Optimal multiple-lag out-of-sequence measurement algorithm based on generalized smoothing framework. In *Proceedings of SPIE*, San Diego, CA, August 2–4, 2005, vol. 5913.
38. Maybeck, P.S. *Stochastic Models, Estimation, and Control*, vol. 1. Academic Press, New York, 1979.
39. McCrady, D.D., L. Doyle, H. Forstrom, T. Dempsey, and M. Martorana. Mobile ranging using low-accuracy clocks. *IEEE Transactions on Microwave Theory and Techniques*, 48(6):951–958, June 2000.
40. Mendel, J.M. *Lessons in Estimation Theory for Signal Processing, Communications, and Control.* Prentice Hall, Upper Saddle River, NJ, 1995.
41. Merwe, R.V.D. *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models.* PhD thesis, Oregon Health and Science University, Portland, OR, 2004. Supervisor: Eric A. Wan.
42. Mohapatra, D. and S.B. Suma. Survey of location based wireless services. In *2005 IEEE International Conference on Personal Wireless Communications*, Piscataway, NJ, 2005, pp. 358–362.

43. Moses, R.L., D. Krishnamurthy, and R.M. Patterson. A self-localization method for wireless sensor networks. *EURASIP Journal on Applied Signal Processing*, 2003(4):348–358, March 2003.
44. Mutambara, A.G.O. and H.F. Durrant-Whyte. Estimation and control for a modular wheeled mobile robot. *IEEE Transactions on Control Systems Technology*, 8(1):35–46, January 2000.
45. Nebot, E.M., M. Bozorg, and H.F. Durrant-Whyte. Decentralized architecture for asynchronous sensors. *Autonomous Robots*, 6(2):147–164, April 1999.
46. Nettleton, E.W. and H.F. Durrant-Whyte. Delayed and asequent data in decentralised sensing networks. *Proceedings of SPIE*, 4571:1–9, 2001.
47. Patwari, N., A.O. Hero, M. Perkins, N.S. Correal, and R.J. O'Dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on Signal Processing*, 51(8):2137–2003, August 2003.
48. Ristic, B., S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications.* Artech House, Boston, MA, 2004.
49. Savarese C., J.M. Rabaey, and J. Beutel. Locationing in distributed ad-hoc wireless sensor networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, Salt Lake City, UT, 2001, vol. 4, pp. 2037–2040.
50. Savvides, A., C.C. Han, and M.B. Srivastava. Dynamic fine-grained localization in ad-hoc wireless sensor networks. In *Proceedings of the Fifth International Conference on Mobile Computing and Networking (MobiCom)*, Rome, Italy, July 2001, pp. 166–179.
51. Savvides, A., H. Park, and M.B. Srivastava. The bits and flops of the N-hop multilateration primitive for node localization problems. In *First ACM International Workshop on Sensor Networks and Applications*, Atlanta, GA, 2002.
52. Stojmenovic, I. Position-based routing in ad hoc networks. *IEEE Communications Magazine*, 40:128–134, July 2002.
53. Tayal, M. Location services in the GSM and UMTS networks. In *2005 IEEE International Conference on Personal Wireless Communications*, Piscataway, NJ, pp. 373–378, 2005.
54. Vadigepalli, R. and F.J. Doyle. III. Structural analysis of large-scale systems for distributed state estimation and control applications. *Control Engineering Practice*, 11(8):895–905, August 2003.
55. Vercauteren, T. and X. Wang. Decentralized sigma-point information filters for target tracking in collaborative sensor networks. *IEEE Transactions on Signal Processing*, 53(8):2997–3009, August 2005.
56. Wang, A. and A. Chandrakasan. Energy-efficient DSPs for wireless sensor networks. *IEEE Signal Processing Magazine*, 19(4):68–78, July 2002.
57. Widnall, W.S. and G.F. Gobbini. Stability of the decentralized estimation in the JTIDS relative navigation. *IEEE Transactions of Aerospace and Electronic Systems*, AES-19(2):240–249, March 1983.
58. Zhang K., X.R. Li, and Y. Zhu. Optimal update with out-of-sequence measurements for distributed filtering. In *Proceedings of Fifth International Conference Information Fusion*, Annapolis, MD, July 2002, pp. 1519–1526.

III

Sensor Networking Data Manipulation and Advanced Hardware

7 **Constructing Load-Balanced Data Aggregation Trees in Probabilistic Wireless Sensor Networks** *Jing (Selena) He* **7**-1
Introduction • Related Work • Network Model and Problem Definition • Connected Maximal Independent Set • Load-Balanced Data Aggregation Tree • Performance Evaluation • Summary • References

8 **Biometric Sensors and How They Work** *Sinjini Mitra* **8**-1
Introduction • Biometric Sensors and Authentication • Different Types of Biometric Sensors • Biometric Sensors in Consumer Electronics and Mobile Devices • Challenges and Issues with Biometric Sensors • Future of Biometric Sensor Technology • Summary • References

9 **Radio Frequency Identification Device Readers and Tags** *C.J. Wiemer* **9**-1
Introduction • RFID Tags • RFID Readers • RFID Standards • RFID Implementations • RFID Concerns • Summary • References

7

Constructing Load-Balanced Data Aggregation Trees in Probabilistic Wireless Sensor Networks

7.1 Introduction 7-1
7.2 Related Work 7-4
Energy-Efficient Aggregation Scheduling • Minimum-Latency Aggregation Scheduling • Maximum-Lifetime Aggregation Scheduling • Remarks
7.3 Network Model and Problem Definition 7-6
Assumptions • Network Model • Problem Definition • Remarks
7.4 Connected Maximal Independent Set 7-10
Formulation of LBMIS • Approximation Algorithm • Connecting LBMIS • LBPNA for Non-Leaf Nodes
7.5 Load-Balanced Data Aggregation Tree 7-16
ILP Formulation of LBPNA for Leaf Nodes • Randomized Approximation Algorithm
7.6 Performance Evaluation 7-20
Simulation Environment • Scenario 1: Change Side Length of Square Area • Scenario 2: Change Node Transmission Range • Scenario 3: Change Total Number of Nodes
7.7 Summary 7-23
References 7-24

Jing (Selena) He
Kennesaw State University

7.1 Introduction

In wireless sensor networks (WSNs), sensor nodes periodically sense the monitored environment and send the information to the sink (or base station), at which the gathered/collected information can be further processed for end-user queries. In this *data gathering* process, *data aggregation* can be used to fuse data from different sensors to eliminate redundant transmissions, since the data sensed by different sensors have spatial and temporal correlations [1]. Hence, through this in-network data aggregation technique, the amount of data that needs to be transmitted by a sensor is reduced, which in turn decreases each sensor's energy consumption so that the whole network lifetime is extended. For continuous monitoring applications with a periodical traffic pattern, a tree-based topology is often adopted to gather and aggregate sensing data because of its simplicity. Compared with an arbitrary network topology, a tree-based topology conserves the cost of maintaining a routing table at each node,

which is computationally expensive for the sensor nodes with limited resources. For clarification, data gathering trees capable of performing aggregation operations are also referred to as *data aggregation trees* (DATs), which are directed trees rooted at the sink and have a unique directed path from each node to the sink. Additionally, in a DAT, sensing data from different sensors are combined at intermediate sensors according to certain aggregation functions including COUNT, MIN, MAX, SUM, and AVERAGE [2].

Due to the dense sensor deployment, many different DATs can be constructed to relay data from the monitored area to the sink. According to the diverse requirements of different applications, the DAT-related works can be roughly classified into three categories: energy-efficient aggregation scheduling [3–5], minimum-latency aggregation scheduling [6–8], and maximum-lifetime aggregation scheduling [9–11]. It is worth mentioning that aggregation scheduling attracts a lot of interests in the current literatures. However, unlike most of the existing works that spend lots of efforts on aggregation scheduling, we mainly focus on the DAT construction problem.

Furthermore, most of the existing DAT construction works are based on the ideal *deterministic network model* (DNM), where any pair of nodes in a WSN is either connected or disconnected. Under this model, any specific pair of nodes are neighbors if their physical distance is less than the transmission range, while the rest of the pairs are always disconnected. However, in most real applications, the DNM cannot fully characterize the behaviors of wireless links due to the existence of the *transitional region phenomenon* [12]. It is revealed by many empirical studies [12,13] that, beyond the *always connected* region, there is a *transitional region* where a pair of nodes is probabilistically connected via the so-called lossy links [12]. Even without collisions, data transmissions over lossy links cannot be guaranteed. Moreover, as reported in [12], usually there are many more lossy links (sometimes [14] 90% more) than fully connected links in a WSN. Therefore, in order to well characterize WSNs with lossy links, a more practical network model is the *probabilistic network model* (PNM). Under this model, there is a *transmission success ratio* (ι_{ij}) associated with each link connecting a pair of nodes v_i and v_j, which is used to indicate the probability that a node can successfully deliver a package to another. An example is shown in Figure 7.1a, in which the number over each link represents its corresponding transmission success ratio, and v_0 is the sink. For convenience, the WSNs considered under the DNM are called *deterministic WSNs*, whereas the WSNs considered under the PNM are called *probabilistic WSNs*. When $\iota_{ij} = 1$, DNM can be viewed as a special case of PNM.

On the other hand, all the aforementioned works did not consider the *load-balance* factor when they construct a DAT. Without considering balancing the traffic load among the nodes in a DAT, some heavy-loaded nodes may quickly exhaust their energy, which might cause network partitions or malfunctions. For instance, for aggregating the sensing data from eight different nodes to the sink node v_0, a shortest-path-based DAT for the probabilistic WSN (Figure 7.1a) is shown in Figure 7.1b. The intermediate node v_4 aggregates the sensing data from four different nodes, whereas v_7 only aggregates one sensing data from v_8. For simplicity, if every link shown in Figure 7.1 is always there and every node has the same amount of data to be transferred through the intermediate nodes with a fixed data rate, heavy-loaded v_4 must deplete its energy much faster than v_7. From Figure 7.1b, we know that the intermediate nodes usually aggregate the sensing data from neighboring nodes in a shortest-path-based DAT. Actually, the number of neighboring nodes of an intermediate node is a potential indicator of the traffic load on each intermediate node. However, it is not the only factor to impact the traffic load on each intermediate node. The criterion to assign a parent node, to which data are aggregated for each node on a DAT, is also critical to balance traffic load on each intermediate node. We refer to the procedure that assigns a unique parent node for each node in the network as the parent node assignment (PNA) in this chapter. Two PNAs different from Figure 7.1b are depicted in Figure 7.1c and d. Evidently, with respect to load balance, the PNA shown in Figure 7.1d is the best (although it may induce high aggregation delay), which also implies the load-balanced DAT (LBDAT) shown in Figure 7.1d can extend network lifetime notably compared with the DATs shown in Figure 7.1b and c, since the traffic load is evenly distributed over all the intermediate nodes.

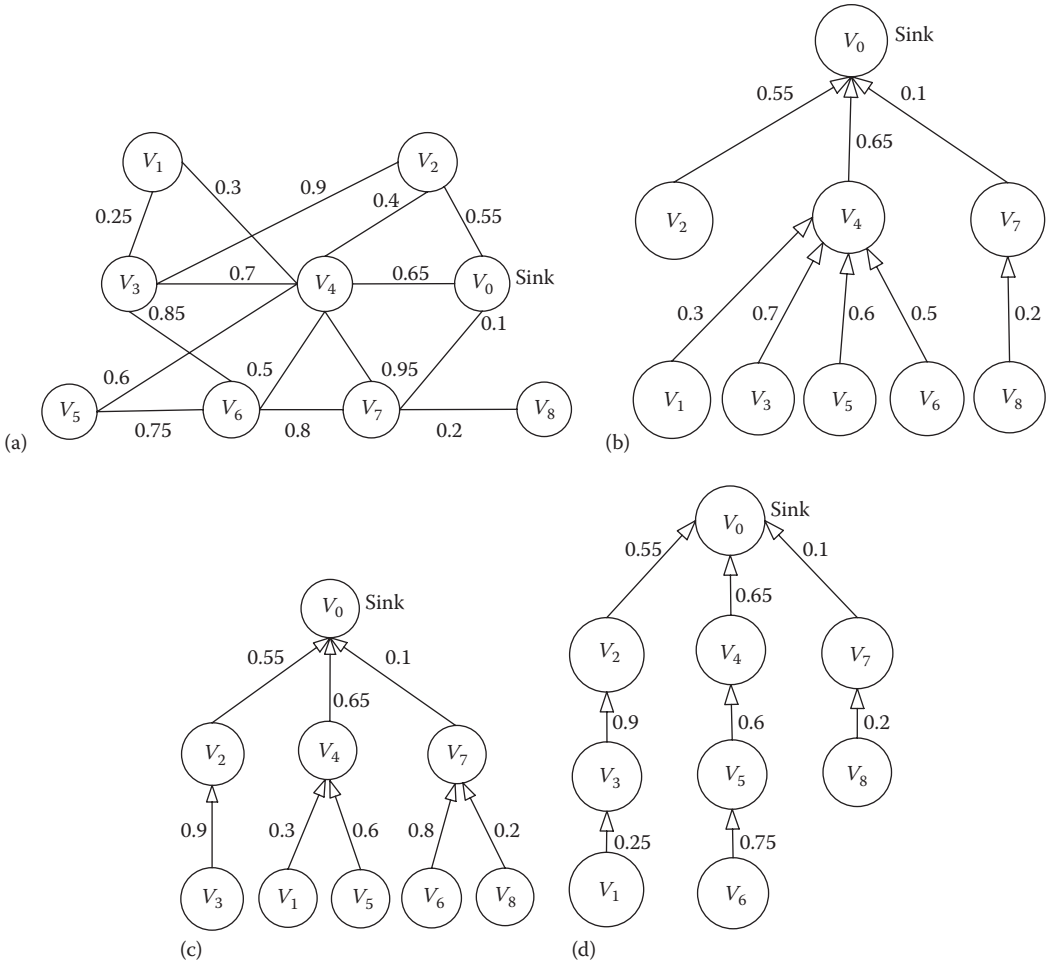


FIGURE 7.1 (a) A probabilistic wireless sensor network (WSN), (b) A imbalanced data aggregation tree (DAT), (c) another imbalanced DAT, (d) A load-balanced DAT (LBDAT).

In summary, the investigated problem in this chapter is distinguished from all the prior works in three aspects. First, most of the current literature investigates the DAT construction problem under the DNM, whereas our work is suitable for both DNM and PNM. Second, the *load-balance* factor is not considered when constructing a DAT in most of the aforementioned works. Third, the DAT construction problem is our major concern, whereas the prior works focus on the aggregation scheduling problem. Therefore, in this chapter, we explore the DAT construction problem under the PNM considering balancing the traffic load among all the nodes in a DAT. To be specific, in this chapter, we construct a LBDAT under the PNM in three phases. We first investigate how to construct a load-balanced maximal independent set (LBMIS). A maximal independent set (MIS) can be defined formally as follows: given a graph = (,), an independent set (IS) is a subset ⊆ such that for any two vertex v_1, $v_2 \in$, they are not adjacent, that is, $(v_1, v_2) \notin$. An IS is called an MIS if we add one more arbitrary node to this subset; the new subset will not be an IS anymore. After obtaining an LBMIS, we attempt to find a minimum-sized set of nodes called LBMIS connector set to make this LBMIS connected, which is called the connected MIS (CMIS) problem. Finally, we seek a load-balanced parent node assignment.

The main contributions of this chapter are summarized as follows:

1. We identify and highlight the use of lossy links when constructing a DAT. Moreover, in order to measure the load balance of the nodes in a DAT under the PNM, we define two new metrics: *potential load* and *actual load*.
2. The LBDAT construction problem is an NP-complete problem and we solve it in three phases. First, we propose an approximation algorithm by using the linear relaxation and random rounding techniques to solve the LBMIS problem, which is an NP-hard problem. Theoretical analysis shows that this algorithm yields a solution upper bounded by $O(\ln(n))opt_{\text{LBMIS}}$, where opt_{LBMIS} is the optimal result of LBMIS and n is the number of sensors in a WSN. Subsequently, a minimum-sized set of nodes are identified to make the LBMIS connected. Finally, to solve LBDAT, we present a randomized approximation algorithm to find an LBPNA. The approximation algorithm produces a solution in which the actual traffic load on each intermediate node is upper bounded by $O(\log(n))opt_{\text{LBPNA}}$, where opt_{LBPNA} is the optimal result.
3. We also conduct simulations to validate our proposed algorithms.

The rest of this chapter is organized as follows: in Section 7.3, we introduce the network model and formally define the LBDAT construction problem. The design of algorithms, theoretical analysis of the CMIS problem, and the LBDAT construction problem are presented in Sections 7.4 and 7.5, respectively. The simulation results are presented in Section 7.6 to validate our proposed algorithms. Finally, the chapter is concluded in Section 7.7.

7.2 Related Work

The problem of data gathering and aggregation in WSNs has been extensively investigated in the literature. Moreover, a tree-based topology to periodically aggregate collected data in WSNs is widely adopted because of its simplicity. However, most of existing works concentrated on the aggregation scheduling problem in deterministic WSNs, which is very different from our research problem. To be specific, we focus on constructing an LBDAT to perform data aggregation in probabilistic WSNs in this chapter. Therefore, in this section, we review the most related works. Based on the different user requirements, the existing DAT-related works can be roughly divided into three categories: energy-efficient aggregation scheduling [3,4,15], minimum-latency aggregation scheduling [6,7,16–18], and maximum-lifetime aggregation scheduling [9,10,19–21].

7.2.1 Energy-Efficient Aggregation Scheduling

As to battery-powered sensors in WSNs, energy efficiency is always the major concern. Hence, it is important to minimize the total energy consumed by a WSN when designing a DAT. The authors proposed a power-efficient data gathering and aggregation protocol (PEDAP) in [3], in which a near optimal minimum energy cost spanning tree is constructed for data aggregation. At first, only the sink node is included in the tree. Then, it keeps selecting nodes not in the tree one by one to join the current tree iteratively. The selected node is the one that can transmit packets to one of the nodes in the current tree with the minimum energy cost. However, PEDAP does not consider each node's energy and thus cannot achieve energy awareness. Therefore, PEDAP and power-aware (PEDAP–PA) are proposed in [3] to improve PEDAP by considering the remaining energy of the sender. Later, the authors tried to construct an energy-balanced minimum degree spanning tree (MDST) in [15]. It starts from an arbitrary tree and tries to balance degrees of nodes in the tree according to their energy. However, a node with fewer children in a DAT does not mean it relays fewer data. Differed from the previous centralized algorithms, in [4], the authors proposed a localized, self-organizing, robust, and energy-efficient DAT for a WSN called localized power-efficient data aggregation protocol (L-PEDAP). The proposed approach consists of two phases. In the first phase, it computes a sparse topology over the original graph of the WSN using the 1-hop neighborhood information.

In the second phase, it constructs a DAT over the edges of the computed sparse topology. Moreover, L-PEDAP is adaptive since it considers the dynamic changes when constructing a routing tree.

7.2.2 Minimum-Latency Aggregation Scheduling

The minimum-latency data aggregation problem was proven to be an NP-hard problem in [6]. Moreover, Chen et al. [6] designed a $(\Delta - 1)R$ approximation algorithm based on a shortest path tree for data aggregation, where Δ is the maximum degree of the network graph and R is the network radius. Subsequently, the first-fit algorithm is proposed by Huang et al. [16], in which a connected dominating set (CDS)-based tree is first constructed, and then a maximal interference-free set of links is scheduled in each time slot. The latency of Huang's approach is bounded by $23R + \Delta - 18$. However, the already scheduled transmissions could also interfere with the candidate links neglected in [16]. Hence, as a successor, Wan et al. [7] developed a $15R + \Delta - 4$ approximation algorithm called sequential aggregation scheduling (SAS) to solve the minimum-latency aggregation schedule (MLAS) problem. Similar to Huang's work, Wan et al. [7] also divided the aggregation process into the tree construction phase and the scheduling phase. The main difference is that the parents of leaf nodes are dynamically determined during the scheduling process. Subsequently, Xu et al. [17] developed an approximation algorithm with bound $16R' + \Delta - 14$, where R' is the inferior network radius that is smaller than R. Recently, Li et al. proposed a distributed scheduling algorithm named Clu-DDAS based on a novel cluster-based aggregation tree in [18] whose latency bound is $4R' + 2\Delta - 2$.

All these works devoted efforts to find a *data aggregation schedule* for each link on the constructed DAT that leads to the minimum *data aggregation latency*. Hence, all these studies differed from our work. We mainly focus on the load-balanced tree construction in probabilistic WSNs.

7.2.3 Maximum-Lifetime Aggregation Scheduling

Wu et al. [15] proved that constructing an arbitrary aggregation tree with the maximum lifetime is NP-complete. Hence, a huge number of approximation algorithms are proposed to construct a DAT with maximum lifetime. Xue et al. [9], using linear programming (LP) formulation, modeled this problem as a multicommodity flow problem, where a commodity represents the data generated from a sensor node and delivered to a base station. A fast approximate algorithm is presented, which is able to compute $(1 - \varepsilon)$ approximation to the optimal lifetime for any $\varepsilon > 0$. Lin et al. considered a more general network model in which the transmission power levels of sensors are heterogeneous and adjustable in [10]. The proposed algorithm starts from an arbitrary spanning tree rooted at the base station. Subsequently, one of the heavily loaded nodes is reduced to normalized load by partially rearranging the current tree to create a new tree. The upper bound on the lifetime of the constructed DAT is also presented in [10]. The authors in [19] proposed a combinatorial iterative algorithm for finding an optimal continuous solution to the maximum-lifetime data gathering with aggregation (MLDA) problem that consists of up to $n - 1$ aggregation trees and achieves lifetime T_0. They obtained an α-approximate optimal integral solution, where $\alpha = T_{0-n+1}/T_0$, and n is the number of sensors in a WSN. The decentralized lifetime maximizing tree (DLMT) with energy consideration is proposed in [20]. Recently, Luo et al. proposed a distributed shortest-path-based DAT in [21]. The authors transformed the problem of maximizing the lifetime of DATs into a general version of semimatching problem and showed that the problem can be solved by a min-cost max-flow approach in polynomial time.

7.2.4 Remarks

All the existing works considered designing a data aggregation schedule or constructing a DAT under the DNM. To the best knowledge of us, however, none of them attempt to construct an LBDAT under the PNM, which is more realistic. This is the major motivation of this research work. Moreover, all the

aforementioned works were focused on constructing energy-efficient aggregation scheduling, minimum-latency aggregation scheduling, or maximum-lifetime aggregation scheduling. Unfortunately, they do not consider the load-balance factor when constructing a DAT. In contrast, in this chapter, we first show an example to illustrate that an imbalanced DAT cannot prolong the network lifetime by reducing the communication cost. Instead, it actually leads to the reduction of network lifetime. Based on this observation, we then build an LBDAT for more practical probabilistic WSNs. Approximation algorithms are proposed in the chapter followed by comprehensive theoretical analysis.

7.3 Network Model and Problem Definition

In this section, we give an overview of the LBDAT construction problem under the PNM. We first present the assumptions and then introduce the PNM. Finally, we give the problem definitions and make some remarks for the proposed problems.

7.3.1 Assumptions

We assume a static connected WSN with the set of n nodes $_s = \{v_1, v_2, \ldots, v_n\}$ and one sink node v_0. All the nodes have the same transmission range. The *transmission success ratio* ι_{ij} associated with each link connecting a pair of nodes v_i, v_j is available, which can be obtained by periodic hello messages or be predicted using link quality index (LQI) [22]. We also assume that the ι_{ij} values are fixed. This assumption is reasonable as many empirical studies have shown that LQI is pretty stable in a static environment [23]. Furthermore, no node failure is considered since it is equivalent to a link failure case. No duty cycle is considered either. We do not consider packet collisions or transmission congestion, which are left to the MAC layer.

We further assume that the n nodes monitor the environment in the deployed area and periodically report the collected data to the sink node v_0 along the LBDAT (the formal definition of LBDAT will be given later). Every node produces a data package of B bits during each report interval. Moreover, an intermediate node can aggregate multiple incoming 5-bit packets, together with its own package into a single outgoing 5-bit package. Furthermore, we assume the data receiving rate of each node v_i is γ_i, and R denotes the maximum data receiving rate of all the nodes. Finally, the degree of a node v is denoted by d_i, whereas δ/Δ denotes the minimum/maximum node degree in the network.

7.3.2 Network Model

Under the *probabilistic network model* (PNM), we model a WSN as an undirected graph , (, , ( )), where = $_s$ $\{v_0\}$ is the set of $n + 1$ nodes, denoted by v_i, where $0 \le i \le n$. i is called the node ID of v_i in the chapter. is the set of lossy links. $\forall v_i, v_j \in$, there exists a line (v_i, v_j) in if and only if (1) v_i and v_j are in each other's transmission range and (2) $l_{ij} > 0$. For each link, $(v_i, v_j) \in \mathbb{E}$, ι_{ij} indicates the probability that node v_i can successfully directly deliver a packet to node v_j, and ( ) = $\{l_{ij} | (v_i, v_j) \in , 0 \le \iota_{ij} \le 1\}$. We assume the links are undirected (bidirectional), which means two linked nodes are able to transmit and receive information from each other with the same ι_{ij} value. Because of the introduction of ι_{ij}, we define the one-hop neighborhood and the h-hop neighborhood as follows.

Definition 7.1: 1-Hop Neighborhood ($_1(v_i)$)

1-hop neighborhood ($_1(v_i)$). $\forall v_i \in$; the one-hop neighborhood of node v_i is defined as $N_1(v_i) = \{v_j | v_j \in , \iota_{ij} > 0\}$.

The physical meaning of 1-hop neighborhood is the set of the nodes that can be directly reached from node v_i.

Definition 7.2: *h*-Hop Neighborhood ($_1(v_i)$)

h-hop neighborhood ($_h(v_i)$). $\forall v_i \in$; the *h*-hop neighborhood of node v_i is defined as

$$N_h(v_i) = N_{h-1}(v_i) \cup \left\{ v_k \mid \exists v_j \in N_{h-1}(v_i), v_k \in N_1(v_j), v_k \notin \bigcup_{i=1}^{h-1} N_i(v_i) \right\}.$$

The physical meaning of the *h*-hop neighborhood is the set of the nodes that can be reached from node v_i with some probability by passing at most *h* links.

7.3.3 Problem Definition

Since load balance is the major concern of this work, the measurement of the traffic load balance under the PNM is critical to solve the LBDAT construction problem. Hence, in this subsection, we first define a novel metric called *potential load* to measure the potential traffic load on each node.

As we mentioned in Section 7.1, the number of neighboring nodes of a node (i.e., $|_1(v_i)|$) is a potential indicator of the traffic load on each node. However, it is not the only factor to indicate the potential traffic load on each node in probabilistic WSNs. For example, if $\iota_{ij} = 0.5$, then the expected number of transmissions to guarantee v_i to deliver one packet to v_j is $1/0.5 = 2$. The less the i_{ij} value, the more potential traffic load on v_j from v_i. Therefore, a more reasonable and formal definition of the potential load is given as follows:

Definition 7.3

Potential load (ρ_i). $\forall v_i \in$ $_s$; the potential load of v_i is defined as

$$\rho_i = \sum_{v_j \in N_1(v_i)} \left\lceil \frac{B}{\gamma_i} \right\rceil \frac{1}{\iota_{ij}}.$$

We solve the LBDAT construction problem in three phases in this chapter. First, we construct an LBMIS, and then we select additional nodes to connect the nodes in LBMIS, denoted by the CMIS problem. Finally, we acquire an LBPNA. After LBPNA is determined, by assigning a direction of each link in the constructed tree structure, we obtain an LBDAT. In this subsection, we formally define the LBMIS, CMIS, LBPNA, and LBDAT construction problems sequentially.

Definition 7.4: LBMIS Problem

For a probabilistic WSN represented by graph (, , ( )), the *LBMIS* problem is to find a node set $\subseteq$ such that the following conditions apply:

1. $v_0 \in M$.
2. $\forall u \in$ and $u \notin$, $\exists v \in$, such that $(u, v) \in$.
3. $\forall u \in$, $\forall v \in$, and $u \neq v$, such that $(u, v) \in$.
4. There exists no proper subset or superset of *M* satisfying the conditions 1, 2, and 3.
5. Maximize $\min\{\rho_i | \forall v_i \in \}$.*

* Maxmin and minmax can achieve the load-balance objective similarly according to (Boyd and Vandenberghe, [24]) for the specific communication model data aggregation. In this chapter, minmax is also applicable.

Taking the load-balance factor into consideration, we are seeking an MIS in which the minimum potential load of the nodes in the constructed LBMIS is maximized. In other words, the potential traffic load on each node in the LBMIS is as balanced as possible. Now, we are ready to define the CMIS problem.

Definition 7.5: CMIS Problem

For a probabilistic WSN represented by graph (, , ( )) and an *LBMIS* , the *CMIS* problem is to find a node set ⊆ \ such that the following conditions apply:

1. The induced graph [∪] on *G* is connected.
2. Minimize | |, where | | is the cardinality of set .

For convenience, the nodes in set are called *independent nodes*, whereas the nodes in set are called *LBMIS connectors*. Moreover, the nodes in the set \(∪) are called *leaf nodes*. Furthermore, $\forall v_i \in$ ∪ , v_i is also called a *non-leaf node*. Hence, the set of non-leaf nodes is denoted by = ∪ .

Constructing a load-balanced connected topology is just one part of the work to build an LBDAT. In order to measure the actual traffic load, one more important task needed to be resolved is how to do PNA for leaf nodes in the network. Since the actual traffic load of each node in a DAT is dependent on the number of its children, which are composed of leaf nodes and non-leaf nodes, we give the formal definition of the PNA for leaf nodes to non-leaf nodes as follows:

Definition 7.6: PNA for Leaf Nodes (A_L)

For a probabilistic WSN represented by graph (, , ( )) and a CMIS = $\{v_1, v_2, \ldots, v_m\}$, we need to find *m* disjoint sets on , denoted by (v_1), (v_2), …, (v_m), such that the following conditions apply:

1. Each set (v_i) ($1 \le i \le m$) contains exactly one non-leaf node v_i.
2. $\bigcup_{i=1}^{m}$ (v_i) = , and (v_i) ∩ (v_j) = ∅ ($1 \le i \ne j \le m$).
3. $\forall v_u \in$ (v_i) ($1 \le i \le m$) and $vu \ne v_i$, such that $(v_u v_i) \in E$.
4. Assign $v_i (1 \le i \le m)$ as the parent node of the nodes in (v_i)\{v_i}. A PNA *for leaf nodes* is $A_L = \{$ (v_i)$|\forall v_i \in$; $1 \le i \le m\}$.

Definition 7.7: PNA for non-leaf nodes (A_I)

For a probabilistic WSN represented by graph (, , ( )) and a CMIS = $\{v_1, v_2, \ldots, v_m\}$, we need to find *m* sets on , denoted by (v_1), (v_2), …, (v_m), such that the following conditions apply:

1. $\forall v_i \in$; the set (v_i) contains exactly one independent node v_i.
2. $\forall v_i \in$; the set (v_j) contains exactly one LBMIS connector v_j.
3. $\forall v_i \in$; $1 \le |\{$ (v_j)$|v_i \in I(v_j); j = 1, 2, \ldots, m\}| \le 2$.
4. $\bigcup_{i=1}^{m}$ (v_i) = .

5. $\forall v_i \in$ (v_i) $(1 \le i \le m)$ and $vu \ne v_i$, such that $(v_u, v_i) \in$.
6. Assign v_i $(1 \le i \le m)$ as the parent node of the nodes in $(v_i)\backslash\{v_i\}$.

A *PNA for non-leaf nodes* is

$$\mathbf{A_I} = \left\{ \mathbf{I}(\mathbf{v_i}) \,|\forall \mathbf{v_i} \in \mathbf{D}\,;\ \mathbf{1} \le \mathbf{i} \le \mathbf{m} \right\}.$$

A_L and A_I together are called a PNA A. According to the preceding definitions, as to each set (v_i) in A_L, v_i is the *parent node* of the nodes in set $(v_i)\backslash\{v_i\}$, whereas the nodes in set $(v_i)\backslash\{v_i\}$ are called the *leaf child nodes* of v_i. Similarly, as to each set (v_i) in A_I, v_i is the *parent node* of the nodes in set $(v_i)\backslash\{v_i\}$, whereas the nodes in set $(v_i)\backslash\{v_i\}$ are called the *non-leaf child nodes* of v_i. As we have already known, ρ_i is only the indicator of the potential traffic load on each non-leaf node. The actual traffic load only can be determined when a PNA, that is, $A = \{A_L, A_I\}$, is decided. In other words, the number of leaf children and non-leaf child nodes (i.e., $|\ (v_i)| - 1$ and $|\ (v_i)| - 1$) along with the corresponding ι_{ij} is the indicators of the actual traffic load on each non-leaf node v_i. According to this observation, we give the following definition:

Definition 7.8: Actual Load (α_i)

The actual load of a non-leaf node v_i is

$$\forall \mathbf{v_i} \in \mathbf{D}\ ,\ \alpha_i = \sum_{v_j \in \{L(v_i) \cup I(v_i) | i \ne j\}} \left\lceil \frac{B}{\gamma_i} \right\rceil \frac{1}{\iota_{ij}}.$$

Load balance is our major concern; hence, when doing PNA, we must take it into consideration. The formal definition of load-balanced PNA is as follows:

Definition 7.9: LBPNA ($\mathcal{A}^*$)

For a probabilistic WSN represented by graph (, , ( )) and a *CMIS* $= \{v_1, v_2, \ldots, v_m\}$, the *LBPNA* problem is to find a PNA $\mathcal{A}^*$ for , such that $\min\{\alpha_i | \forall v_i \in \}$ is maximized under $\mathcal{A}^*$.

After a $\mathcal{A}^*$ is decided, every node in the network has a unique parent node. Hence, a tree structure (LBDAT) is established.

7.3.4 Remarks

Since finding an MIS is a well-known NP-hard problem [25] in graph theory, CMIS is NP-hard as well. Therefore, CMIS cannot be solved in polynomial time unless P = NP. Consequently, we propose an approximation algorithm by using linear relaxation and random rounding technique to obtain an approximate solution. Additionally, the key aspect to solve the LBDAT construction problem is to find an LBPNA $\mathcal{A}^*$, and obtaining an LBPNA is NP-complete. Then we formulate it as an equivalent binary programming. We also present a randomized approximation algorithm to find the approximate solution to $\mathcal{A}^*$. After specifying the direction of each link in $\mathcal{A}^*$, we obtain an LBDAT .

7.4 Connected Maximal Independent Set

In this section, we first introduce how to solve the LBMIS problem. We formulate the LBMIS problem as an integer nonlinear programming (INP). Subsequently, we show how to obtain an $O(\ln(n))$ approximation solution by using LP relaxation techniques. Finally, we present how to find a minimum-sized set of LBMIS connectors to form a CMIS .

7.4.1 Formulation of LBMIS

For convenience, we assign a decision variable for each sensor $v_i \in$, which is allowed to be 0/1 value. This variable sets to 1 if and only if the node is an independent node, that is, $\forall v_i \in$, $\omega_i = 1$. Otherwise, it sets to 0.

It is well known that in graph theory, an MIS is also a dominating set (DS). A DS is defined as a subset of nodes in a WSN such that each node in the network is either in the set or adjacent to some node in the set. Hence, we formally model LBMIS as an INP as follows:

Sink node constraint: All aggregated data are reported to the sink node; hence, the sink node is deliberately set to be an independent node, that is, $\omega_0 = 1$.

DS property constraint: Since an MIS is also a DS, we should formulate the DS constraint for the LBMIS problem first. The DS property states that each nonindependent node must reside within the 1-hop neighborhood of at least one independent node. We therefore have $\omega_i + \sum_{v_j \in N_1(v_i)} \omega_j \geq 1, \forall v_i \in V$.

IS property constraint: Since the solution of the LBMIS problem is at least an IS, the IS property is also a constraint of LBMIS. The IS property indicates that no two independent nodes are adjacent, that is, $\forall v_i, v_j \in$, $(v_i, v_j) \notin$. In other words, we have $\sum_{v_j \in N_1(v_i)} \omega_i \cdot \omega_j = 0, \forall v_i \in V$.

Consequently, the objective of the LBMIS problem is to maximize the minimum potential load (ρ_i) of all the independent nodes ($\forall v_i \in$). We denote v as the objective of the LBMIS problem, that is, $v = \min_{v_i \in M} (\rho_i)$. Mathematically, the LBMIS problem can be formulated as an INP ($\text{INP}_{\text{LBMIS}}$) as follows:

$$
\begin{aligned}
&\max \quad v = \min\{\rho_i \mid \forall v_i \in M\},\\
&\text{s.t.} \quad \omega_0 = 1;\\
&\qquad \omega_0 + \sum_{v_j \in N_1(v_i)} \omega_j \geq 1;\\
&\qquad \sum_{v_j \in N_1(v_i)} \omega_i \cdot \omega_j = 0; \quad (\text{INP}_{\text{LBMIS}})\\
&\qquad \omega_i, \omega_i \in \{0,1\}, \quad \forall v_i, v_i \in V.
\end{aligned}
$$

Since the IS property constraint is quadratic, the formulated integer programming $\text{INP}_{\text{LBMIS}}$ is not linear. To linearize $\text{INP}_{\text{LBMIS}}$, the quadratic constraint is eliminated by applying the techniques proposed in [26]. More specifically, the product $\omega_i \cdot \omega_j$ is replaced by a new binary variable $\bar{\omega}_{ij}$, on which several additional constraints are imposed. As a consequence, we can reformulate $\text{INP}_{\text{LBMIS}}$ exactly to an integer linear programming ($\text{ILP}_{\text{LBMIS}}$) by introducing the following linear constraints:

1. $\sum_{v_j \in N_1(v_i)} \bar{\omega}_{ij} = 0$
2. $\omega_i \geq \bar{\omega}_{ij}, \omega_j \geq \bar{\omega}_{ij}$.
3. $\omega_i + \omega_j - 1 \leq \bar{\omega}_{ij}, \bar{\omega}_{ij} \in \{0,1\}, \forall v_i, v_j \in V$.

According to Definition 7.3, the potential load of an independent node v_i is $\rho_i = \sum_{j:\omega_j t_{ij}>0} B/\gamma_i(1/t_{ij})$ Moreover, by relaxing the conditions $\omega_j \in \{0, 1\}$ and $\varpi_{ij} \in \{0,1\}$ to $\omega_j \in [0, 1]$ and $\varpi_{ij} \in [0,1]$, correspondingly, we obtain the following relaxed LP (LP^*_{LBMIS}):

$$
\begin{aligned}
&\max v = \min\left\{\rho_i = \sum_{j:\omega_j t_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\frac{1}{t_{ij}}\middle|\forall v_i \in V_s\right\},\\
&\text{s.t.}\quad \omega_0 = 1;\\
&\qquad \omega_i + \sum_{v_i \in N_1(v_i)} \omega_j \geq 1;\\
&\qquad \sum_{v_j \in N_1(v_i)} \bar{\omega}_{ij} = 0; \quad \left(LP^*_{LBMIS}\right)\\
&\qquad \omega_i \geq \bar{\omega}_{ij};\quad \omega_j \geq \bar{\omega}_{ij};\\
&\qquad \omega_i + \omega_j - 1 \leq \bar{\omega}_{ij};\\
&\qquad \omega_i, \omega_j, \bar{\omega}_{ij} \in [0,1],\quad \forall v_i, v_j \in V_s.
\end{aligned}
$$

7.4.2 Approximation Algorithm

Due to the relaxation enlarging the optimization space, the solution of LP^*_{LBMIS} corresponds to an upper bound to the objective of INP_{LBMIS}. Given an instance of LBMIS modeled by the INP (INP_{LBMIS}), we propose an approximation algorithm as shown in Algorithm 7.1 to search for an LBMIS.

Algorithm 7.1: Approximation Algorithm for LBMIS

1 Solve LP^*_{LBMIS}. Let (ω^*, v^*) be the optimum solution, where

$$\omega^* = \langle \omega_1^*, \omega_2^*, \ldots, \omega_n^* \rangle, v^* = \min\left\{\sum_{j:\omega_j^* t_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\frac{1}{t_{ij}}\middle|\forall v_i \in V\right\};$$

2 Sort all the sensor nodes by the ω_i^* value in the decreasing order. The sorted node ID i is stored in the array by $A[n]$

3 $\hat{\omega}_0 = 1$;

4 **for** $i = 1$ to **do**

5 $\hat{\omega}_i = 0$;

6 *counter* = 0;

7 **while** $counter \leq \tau, where \tau = \dfrac{7\ln(n)}{\min\left\{\omega_i^* \middle| v_i \in V, \omega_i^* > 0\right\}}$ **do**

8 $k = 0$;

9 **while** $k < n$ **do**

10 $i = A[k]$;

11 **If** $\forall v_j \in N_1(v_i), \omega_j = 0$ **then**

12 $\hat{\omega}_i = 1$ with probability $p_i = \omega_i^*$;

13 $k = k + 1$;

14 *counter* = *counter* + 1;

15 **return** $\left(\hat{\omega}, \hat{v} = \min\left\{\sum_{j:\hat{\omega}_j t_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\frac{1}{t_{ij}}\middle|\forall v_i \in V\right\}\right)$

The basic idea of Algorithm 7.1 is as follows: first, solve the relaxed LP (LP^{*}_{LBMIS}) to get an optimal fractional solution, denoted by (ω^*, v^*), where $\omega^* = \langle \omega_1^*, \omega_2^*, \ldots, \omega_n^* \rangle$, and then round ω_i^* to integer $\hat{\omega}_i$ according to the six steps shown in lines 2–14 of Algorithm 7.1:

1. Sort sensor nodes by the ω_i^* value (where $1 \le i \le n$) in the decreasing order.
2. Set the sink node to be the independent node, that is, $\hat{\omega}_0 = 1$.
3. Set all $\hat{\omega}_i$ to be 0.
4. Start from the first node in the sorted node array A. If there is no node that has been selected as an independent node in v_i's 1-hop neighborhood, then let $\hat{\omega}_i = 1$ with probability $p_i = \omega_i^*$.
5. Repeat step 4 till you reach the end of array A.
6. Repeat steps 4 and 5 for $(7\ln(n))/(\min\{\omega_i^* \mid v_i \in V, \omega_i^* > 0\})$ times.

Next, the correctness of our proposed approximation algorithm (Algorithm 7.1) is proven, followed by the performance ratio analysis. Before showing the correctness of Algorithm 7.1, two important lemmas are given as follows:

Lemma 7.1

For a probabilistic WSN represented by (, , ( )), if a subset ⊆ is a DS and meanwhile is also an IS, then this subset is an MIS of .

Proof: If ⊆ is a DS of , it implies that $\forall v_i \in$ \ ; there exists at least one node $v_j \in$ in v_i's 1-hop neighborhood. Moreover, if is also an IS, it implies that no two nodes in are adjacent, that is, $\forall v_s, v_t \in$; $(v_s, v_t) \in$ E.

Suppose is *not* an MIS. In other words, we can find at least one more node, which does not violate the DS property and the IS property of , to be added into . Suppose v_i is such a node. Based on the DS property, we know that $\exists v_j \in$ and $v_j \in$ $_1(v_i)$. According to the hypothesis, $v_i \in$ and considering the fact that $v_j \in$ $_1(v_i)$, we conclude that there are two nodes (v_i and v_j) adjacent in (i.e., $(v_i, v_j) \in$), which are contradicted to the IS property. Hence, the hypothesis is false and Lemma 7.1 is true. ■

Lemma 7.2

The set $M = \{v_i \mid \hat{\omega}_i = 1, 0 \le i \le n\}$, where $b\omega_i$ is derived from Algorithm 7.1, is a DS almost surely.

Proof: Suppose $\forall v_i \in$, $|$ $_1(v_i)| = k_i$, where $|$ $_1(v_i)|$ is the cardinality of the set $_1(v_i)$. Let the random variable W_i denote the event that no node in the set $_1(v_i) \cup \{v_i\}$ is selected as an independent node. Additionally, we denote $W = \max\{\omega_j^* \mid v_j \in N_1(v_i) \cup \{v_i\}\}$. For the probability of W_i happening, we have

$$
\begin{aligned}
P(W_i) &= \left[(1-\omega_1)(1-\omega_2)\cdots(1-\omega_{k_i})(1-\omega_i)\right]^{\tau} \le (1-W)^{\tau} \\
&\le \left(1-\min\left\{\omega_i^* \mid v_i \in V, \omega_i^* > 0\right\}\right)^{\tau} \le \left(e^{-\min\{\omega_i^* \mid v_i \in V, \omega_i^* > 0\}}\right)^{\tau} \\
&= e^{-\frac{7\min\{\omega_i^* \mid v_i \in V, \omega_i^* > 0\}\ln(n)}{\min\{\omega_i^* \mid v_i \in V, \omega_i^* > 0\}}} \le e^{-7\ln(n)} = 1/n^7.
\end{aligned}
$$

■

Thus, according to the Borel–Cantelli lemma, $P(W_i) \sim 0$, which implies that there exists one independent node in the set $_1(v_i) \cup \{v_i\}$ almost surely, that is, it is almost sure that the set

$M = \{v_i \mid \hat{\omega}_i = 1, 0 \le i \le n\}$ derived from Algorithm 7.1 is a DS. Then, it is reasonable that we consider is a DS of in the following.*

Based on Lemmas 7.1 and 7.2, the following theorem can be obtained. ■

Theorem 7.1

The set $M = \{v_i \mid \hat{\omega}_i = 1, 0 \le i \le n\}$, where $\hat{\omega}_i$ is derived from Algorithm 7.1, is an MIS.

Proof: According to Algorithm 7.1, no two nodes can both be set as independent nodes in the 1-hop neighborhood. This guarantees the IS property of , that is, $\forall v_i, v_j \in$, $(v_i, v_j) \in$. Moreover, is a DS as proven in Lemma 7.2. Hence, based on Lemma 7.1, we conclude that M is an MIS.

From Theorem 7.1, we know that the solution of Algorithm 7.1 is an MIS. Subsequently, we analyze the approximation factor of Algorithm 7.1 in Theorem 7.2. ■

Theorem 7.2

Let opt_{LBMIS} denote the optimal solution of the LBMIS problem. The proposed algorithm yields a solution of $O(\ln(n))\ opt_{\text{LBMIS}}$.

Proof: The expected *pi* of the independent node v_i found by Algorithm 7.1 is

$$E\left[\sum_{j:\omega_j l_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\frac{1}{l_{ij}}\right] = \left\lceil\frac{B}{\gamma_i}\right\rceil\sum_{v_j\in N_1(v_i)}\frac{E[\hat{\omega}_i]}{l_{ij}} \ge \left\lceil\frac{B}{\gamma_i}\right\rceil\sum_{v_j\in N_1(v_i)}\frac{\omega_i^*\times 1+(1-\omega_i^*)\times 0}{l_{ij}}$$

$$= \left\lceil\frac{B}{\gamma_i}\right\rceil\sum_{v_j\in N_1(v_i)}\frac{\omega_i^*}{l_{ij}} \ge v^*.$$ ■

Applying the Chernoff bound, we obtain the following bound:

$$\Pr\left[\sum_{j:\omega_j l_{ij}>0}\frac{1}{l_{ij}} \le (1-\sigma)\tau v^*\right] \le e^{-(\sigma^2/2)\tau v^*},$$

for arbitrary $0 \sigma 1$. To simplify this bound, let $\sigma = \sqrt{6/7}$; we get

$$\Pr\left[\sum_{j:\omega_j l_{ij}>0}\frac{1}{l_{ij}} \le (1-\sigma)\tau v^*\right] \le e^{-(3/7)\tau v^*} \le e^{-(3/7)\tau}$$

$$\le e^{-\frac{3\ln(n)}{\min\{\omega_i^* \mid v_i\in V,\ \omega_i^*>0\}}}$$

$$\le e^{-3\ln(n)} = \frac{1}{n^3}.$$

* It is almost impossible that is not a DS of . If not, we repeat the entire rounding process.

Applying the union bound, we get the probability that some independent node has the potential load $\hat{v}$ less than $(1-\sigma)\tau v^*$:

$$\Pr[\hat{v} \le (1-\sigma)\tau v^*] \le n\frac{1}{n^3} = \frac{1}{n^2}.$$

Again, since $\sum_{n>0}(1/n^2)$ is a particular case of the Riemann zeta function, then $\sum_{n>0}(1/n^2)$ is bounded, that is, $\sum_{n>0}(1/n^2)$ is the Borel–Cantelli lemma, $P[\hat{v} \le (1-\sigma)\tau v^*] \sim 0$.

In summary, we get

$$\Pr[\text{a node is selected to be an independent node in 1-hop neighborhood} \cap \hat{v} \ge (1-\sigma)\tau v^*]$$

$$= 1 \cdot \left(1 - \frac{1}{n^2}\right) \sim 1, \quad \text{when } n \sim \infty.$$

Furthermore, the minimum potential load on all the independent nodes produced by Algorithm 7.1 is upper bounded by $(1-\sigma)\tau v^*$ with probability 1, where v^* is the optimum solution of LP^*_{LBMIS} and $\tau = 7\ln(n)/\min\{\omega_i^* \mid v_i \in V, \omega_i^* > 0\}$. Hence, Theorem 7.2 is proven.

7.4.3 Connecting LBMIS

To solve the CMIS problem, one more step is needed after constructing an LBMIS, which is making the LBMIS connected. Next, we introduce how to find a minimum-sized set of LBMIS connectors to connect the constructed LBMIS by a similar procedure as in [7].

We first divide the LBMIS into disjoint node sets according to the following criterion:

$$M_0 = \{v_0\} \quad \text{and} \quad M_1 = \left\{ v_i \mid v_i \in M, \exists v_j \in M_{l-1}, v_i \in N_2(v_j), v_i \notin \bigcup_{k=0}^{l-1} M_k \right\}.$$

The sink node is put into $_0$. Clearly, $|_0| = 1$. All the independent nodes in the 2-hop neighborhood of the nodes in $_{l-1}$ are put into $_l$. Hence, l is called the *level* of an independent node. $_l$ represents the set of independent nodes of level l in with respect to the node in $_0$. Additionally, suppose the maximum level of an independent node is L. For each $0 \le i \le 1$, let $_i$ be the set of the nodes adjacent to at least one node in $_i$ and at least one node in $_{i+1}$. Subsequently, compute a minimum-sized set of nodes $_I \subseteq _i$ to cover the nodes in set $_{i+1}$. Let $C = \bigcup_{i=0}^{L-1} C_i$ and therefore, $D = M \cup C$ is a CMIS of the original graph G. We use the WSN shown in Figure 7.2a as an example to explain the construction process of a CMIS. In Figure 7.2a, each circle represents a sensor node. As we mentioned earlier, the construction process consists of two phases. In the first phase, it solves the LBMIS problem by Algorithm 7.1 to obtain , which is shown in Figure 7.2b by black circles. The number beside each independent node is the level of that node with respect to the sink node v_0. In the second phase, we choose the appropriate LBMIS connectors ( ), shown by gray nodes in Figure 7.2c, to connect all the nodes in M to form a CMIS ( ).

Next, we analyze the number of non-leaf nodes $|D|$ produced by our proposed algorithms. The following lemma presents some additional properties of the constructed CMIS [7]:

Lemma 7.3

The following statements are true:

1. For each $0 \le i \le L$, each LBMIS connector in is adjacent to at most four independent nodes in $_{i+1}$.
2. For each $1 \le i \le 1$, each independent node in $_i$ is adjacent to at most 11 LBMIS connectors in $_i$.
3. $|_0| \le 12$. ■

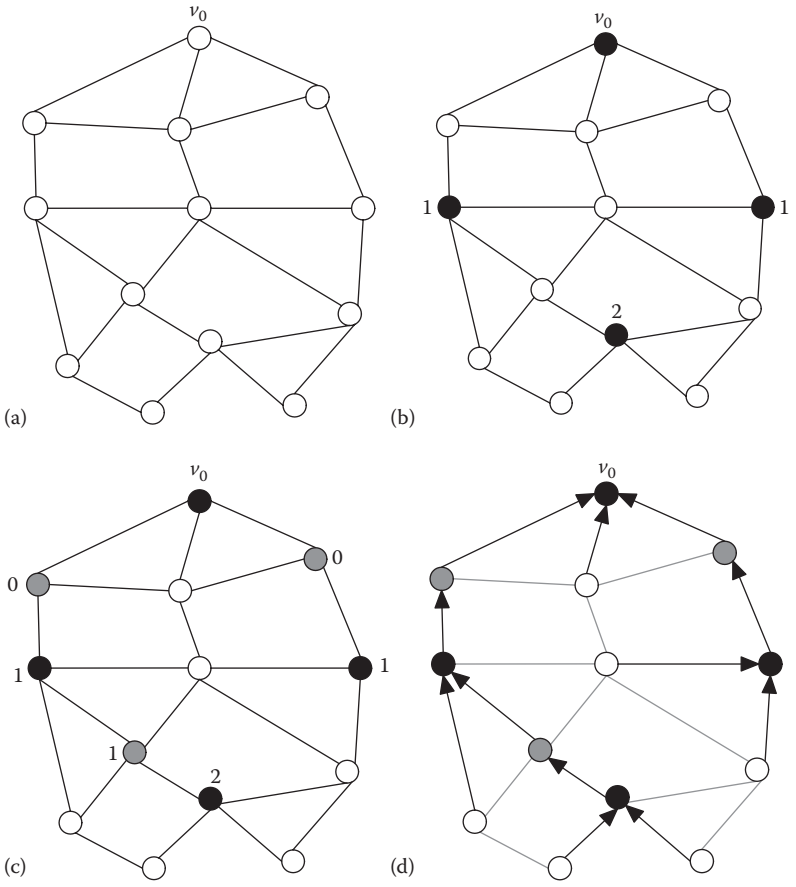


FIGURE 7.2 Illustration of (a) the topology of a WSN, (b) a LBMIS (marked by black nodes), (c) a CMIS (marked by gray nodes), and (d) a LBDAT construction process.

Theorem 7.3

The number of non-leaf nodes satisfies $|M| + (M-1)/4 \le |D| \le 2|M|$.

Proof: According to **Algorithm 7.1**, each LBMIS connector connects the independent nodes in M_i and M_{i+1}. Hence, $|C| = |\bigcap_{i=0}^{L-1} C_i| \le \sum_{i=0}^{L-1} \max\{|M_i|, |M_{i+1}|\} \le |M|$. Moreover, based on Lemma 7.3, $|C| \ge (M-1)/4$. Finally, we get $|M| + (M-1)/4 \le |M \cup C| = |D| \le |M| + |C| \le 2|M|$. ■

7.4.4 LBPNA for Non-Leaf Nodes

After CMIS is constructed, we find an LBPNA for non-leaf nodes. The procedure is as follows:

1. $\forall v_i \in$ $_0$; their parent is the sink node v_0.
2. According to the ID increasing order, for every $v_i \in$ $_l$, and $l > 0$, its parent is assigned to be the adjacent node $v_j \in$ $_{l-1}$ with the minimum traffic load.
3. According to the ID increasing order, for every $v_i \in$ $_l$, and $l > 0$, its parent is assigned to be the adjacent node $v_j \in$ $_{l-1}$ with the minimum traffic load.

We use the node ID to break the tie (small ID with higher priority). After applying this PNA scheme to all the non-leaf nodes, $\forall v_i \in$, its parent node is decided. Furthermore, for each $v_i \in$, the traffic load of v_i introduced by its non-leaf child nodes is denoted by φ_i. Based on Lemma 7.3, $\forall v_i \in$, it has at most 12 non-leaf children, and $\forall v_i \in$, it has at most 4 non-leaf children. Considering that for $v_i \in$, it has as many as $O(n)$ leaf children. We focus on studying the PNA scheme for leaf nodes in the next section since any PNA method for non-leaf nodes yields a constant approximation of at most 12.

7.5 Load-Balanced Data Aggregation Tree

A tree structure is decided after the LBPNA $\mathcal{A}^*$ is produced, which includes LBPNA for non-leaf nodes and leaf nodes. By assigning a direction of each link in the constructed tree from the children node to the parent node, we obtain an LBDAT. We already illustrate how to find a PNA for non-leaf nodes. Hence, in this section, we first formulate the LBPNA for leaf nodes as an ILP. Then, we present an approximation algorithm by applying the linear relaxation and random rounding technique. Finally, we exploit an example to illustrate how to build an LBDAT.

We already illustrate how to find a PNA for non-leaf nodes. In this section, we study the LBPNA for leaf nodes.

7.5.1 ILP Formulation of LBPNA for Leaf Nodes

As we have already known, constructing an arbitrary aggregation tree with the maximum lifetime is NP-complete [15]. Through a similar proving procedure, it can be shown that LBPNA is also an NP-complete problem. In this subsection, we first model LBPNA as an ILP.

We define a binary variable β_i to indicate whether the sensor v_i is a non-leaf node or not. β_i sets to be 1 *iff* the sensor v_i is a non-leaf node. Otherwise, β_i sets to be 0. Additionally, we assign a random variable ξ_{ij} for each link connecting a non-leaf node v_i and a leaf node v_j on the graph modeled from a probabilistic WSN, that is,

$$\xi_{ij} = \begin{cases} 1, & \text{if non-leaf node } v_i \text{ is assigned to be} \\ & \text{the parent of leaf node } v_j, \\ 0, & \text{otherwise.} \end{cases}$$

Consequently, LBPNA can be formulated as an ILP ($\text{ILP}_{\text{LBPNA}}$) as follows:

$$\max \vartheta = \min \left\{ \alpha_i = \sum_{v_j \in L(v_i) \setminus \{v_i\}} \left\lceil \frac{B}{\gamma_i} \right\rceil \frac{1}{l_{ij}} + \varphi_i \middle| \forall v_i \in D \right\},$$

$$\text{s.t.} \sum_{\substack{v_j \in N_1(v_j) \\ \xi_{ij} \in \{0,1\}}} \beta_i \xi_{ij} = 1, \quad \forall v_j \notin D \quad (\text{ILP}_{\text{LBPNA}}).$$

The objective function ϑ is the minimum actual load (α_i) among all the non-leaf nodes. The first constraint states that each leaf node can be allocated to only one non-leaf node, whereas the second constraint indicates that ξ_{ij} is a binary variable. According to Definition 7.8, the number of *leaf child nodes* and the number of *non-leaf child nodes* both contribute to the actual load of a non-leaf node. The leaf child nodes of parent node v_i can be represented by v_j: $\beta_i \xi_{ij} > 0$. The traffic load introduced by non-leaf children to v_i is denoted by φ_i. Moreover, as stated in Lemma 7.3, the number of non-leaf children nodes of an independent parent node $v_i \in$ is no more than 12, whereas the number

of non-leaf children nodes of an LBMIS connector parent node $v_i \in$ is no more than 4. Therefore, for simplicity, we assume that the total actual load of leaf children nodes is approximated to $12\lceil B/R \rceil$ (i.e., $\sum_{v_j \in \{(v_i)i \neq j\}} \lceil B/l_{ij} \rceil (l_{ij}) \simeq 12\lceil B/R \rceil$).* By relaxing variable $\xi_{ij} \in \{0,1\}$ to $\xi_{ij} \in [0, 1]$, we get the relaxed formulation that falls into a standard LP problem, denoted by $\mathrm{LP}^*_{\mathrm{LBPNA}}$ as follows:

$$\max \vartheta = \min\{\alpha_i \mid \forall v_i \in D\},$$

$$\text{s.t.} \sum_{v_j \in N_1(v_i)} \beta_i \xi_{ij} = 1, \quad \forall v_j \notin D,$$

$$0 \le \varphi_i \le 12\left\lceil \frac{B}{R} \right\rceil, \quad \left(\mathrm{LP}^*_{\mathrm{LBPNA}}\right)$$

$$\xi_{ij} \in [0,1].$$

In $\left(\mathrm{LP}^*_{\mathrm{LBPNA}}\right)$, $\alpha_i = \sum_{v_j \in L_1(v_i) \setminus \{v_i\}, j \neq i, \xi_{ij} > 0} \lceil B/l_{ij} \rceil \max\{1, (\beta_i \xi_{ij}/l_{ij})\} + \varphi_i$. Using $\max\{1, (\beta_i \xi_{ij}/l_{ij})\}$ is mainly because if v_j has some data ($\xi_{ij} > 0$ percent of v_j's data) that have been forwarded by v_i (v_i is the parent node of v_j), v_j must transmit at least one data packet to v_i since data packets are the basic communication units in a WSN. Due to the relaxation enlarging the optimization space, the solution of $\mathrm{LP}^*_{\mathrm{LBPNA}}$ corresponds to an upper bound of the objective of $\mathrm{ILP}_{\mathrm{LBPNA}}$.

7.5.2 Randomized Approximation Algorithm

Given an instance of LBPNA modeled by the ILP ($\mathrm{ILP}_{\mathrm{LBPNA}}$), the sketch of the randomized approximation algorithm is shown in Algorithm 7.2. We summarize Algorithm 7.2 as follows: first, solve the relaxed LP ($\mathrm{LP}^*_{\mathrm{LBPNA}}$) to get an optimal fractional solution, denoted by ($\xi^* v^*$), where

Algorithm 7.2: Approximation Algorithm for LBPNA

1 Solve $\mathrm{LP}^*_{\mathrm{LBPNA}}$. Let (ξ^*, ϑ^*) be the optimum solution;
2 Sort the ξ^*_{ij} values in each row (for each i) of ξ^* in the decreasing order and then store the corresponding j(vj's ID) in a 2D array denoted by $A[n][m]$;
3 $\hat{\xi}_{ij} = 0$;
4 **while** $k \le \kappa = \dfrac{6\log(n)}{\delta^2 \min\{\xi^*_{ij} \mid 1 \le i \le n, 1 \le j \le m, \xi^*_{ij} > 0\}}$ **do**
5 $k = 0, l = 0$;
6 **while** $l < m$ **do**
7 $i = k$;
8 **while** $l < m$ **do**
9 $j = A[k][l]$;
10 **if** $v_j \in N_1(v_i)$ and $\hat{\xi} = 0$ **then**
11 $\hat{\xi}_{ij} = 1$ with probability ξ^*_{ij};
12 break;
13 $k = k + 1$;
14 **return** $(\hat{\xi}, \alpha = \min\{\alpha_i \mid \forall v_i \in D\})$

* It loses only a constant factor.

$\mathcal{X}^* = \langle\langle \xi^*_{11}, \ldots, \xi^*_{1m}, \xi^*_{21}, \ldots, \xi^*_{2m}, \ldots, \xi^*_{n1}, \ldots, \xi^*_{nm} \rangle\rangle$, and then round ξ^*_{ij} to integers $\hat{\xi}_{ij}$ by a random rounding procedure, which consists of five steps as shown in lines 2–13 of Algorithm 7.2:

1. Sort the ξ^*_{ij} values in each row of $\mathcal{X}^*$ (for every $1 \le i \le n$) in the decreasing order and store the corresponding j (v_j's ID) in a 2D array $A[n][m]$.
2. Set all $\hat{\xi}_{ij}$ to be 0.
3. Start from the first row in the sorted array A. If there is no parent node assigned to v_i in its 1-hop neighborhood, then let $\hat{\xi}_{ij} = 1$ with probability ξ^*_{ij}. Then, go to the next row in A.
4. Repeat step 3 till you reach the end of array A.
5. Repeat steps 3 and 4 for κ times, where

$$\kappa = \kappa = \frac{6\log(n)}{\delta_2 \min\{\xi^*_{ij} \mid 1 \le i \le n \cdot 1 \le j \le m, \xi^*_{ij} > 0\}},$$

and δ is any constant satisfying $0 < \delta > 1$.

Next, the correctness of the proposed approximation algorithm (Algorithm 7.2) is proven in the following lemma. Due to space limitation, the proof of Lemma 7.4 is omitted. ■

Lemma 7.4

$\forall v_i \in \ \setminus$ is assigned a parent non-leaf node in its 1-hop neighborhood almost surely after executing Algorithm 7.2.

Proof: We first denote $\chi = \max\{\xi^*_{ij} \mid v_i \in N_1(v_j), v_i \in D, \xi^*_{ij} > 0\}$. Additionally, denote the probability that a leaf node $v_j \in \ \setminus$ is not assigned a parent non-leaf node in its 1-hop neighborhood after executing κ times as . Then,

$$P = \prod_{v_j \in N_1(v_j), v_i \in D} \left(1 - \xi^*_{ij}\right)^{\kappa} \le e^{-\chi\kappa} = e^{-\chi \cdot \frac{6\log(n)}{\delta_2 \min\{\xi^*_{ij} \mid 1 \le i \le n, 1 \le j \le m, \xi^*_{ij} > 0\}}} \le e^{-6\log(n)} \le \frac{1}{n^6}.$$

Now, the probability that a leaf node is not assigned a parent non-leaf node in its 1-hop neighborhood after the random rounding is $1/n^6$, which implies $Pr[\text{a leaf node has no neighboring non-leaf node}] \le n(1/n^6) = (1/n^5)$.

Similarly, according to the Borel–Cantelli lemma, this probability is 0 almost surely, which implies that it is almost sure that every leaf node is assigned a parent non-leaf node in its 1-hop neighborhood after executing Algorithm 7.2. ■

Subsequently, we analyze the approximation factor of Algorithm 7.2 in Theorem 7.4.

Theorem 7.4

Let $\text{opt}_{\text{LBPNA}}$ denote the optimal solution of LBPNA. Algorithm 7.2 yields an optimal fractional solution of $O(\log(n))opt_{\text{LBPNA}}$ with probability 1.

Proof: Considering any non-leaf node v_i and leaf node v_j, the expected actual load of v_i is as follows:

$$E\left[\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\hat{\xi}_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\max\left\{1,\frac{\beta_i\xi_{ij}}{\iota_{ij}}\right\}+\varphi_i\right] \tag{7.1}$$

$$=E\left[\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\hat{\xi}_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\max\left\{1,\frac{\beta_i\xi_{ij}}{\iota_{ij}}\right\}+\varphi_i\right] \tag{7.2}$$

$$=\left\lceil\frac{B}{\gamma_i}\right\rceil E\left[\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\hat{\xi}_{ij}>0}\max\left\{1,\frac{\beta_i\xi_{ij}}{\iota_{ij}}\right\}\right]+\varphi_i \tag{7.3}$$

$$=\left\lceil\frac{B}{\gamma_i}\right\rceil\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\hat{\xi}_{ij}>0}\max\left\{1,\frac{\beta_i}{\iota_{ij}}E[\hat{\xi}_{ij}]\right\}+\varphi_i \tag{7.4}$$

$$\geq\left\lceil\frac{B}{\gamma_i}\right\rceil\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\hat{\xi}_{ij}>0}\max\left\{1,\frac{\beta_i}{\iota_{ij}}\xi^*_{ij}\right\}+\varphi_i \tag{7.5}$$

$$=\vartheta^*. \tag{7.6}$$

In the derivation, we exploit the facts that β_i and $\hat{\xi}_{ij}$ are independent, and the procedure setting $\hat{\xi}_{ij}=1$ with probability is repeated k times. Hence, $E[\hat{\xi}_{ij}]\geq 1\times\xi^*_{ij}+0\times(1-\xi^*_{ij})=\xi^*_{ij}$.

Applying the Chernoff bound, we obtain the following bound:

$$Pr\left[\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\hat{\xi}_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\max\left\{1,\frac{\beta_i\hat{\xi}_{ij}}{\iota_{ij}}\right\}+\varphi_i\leq(1-\delta)\kappa\vartheta^*\right]\leq e^{-\frac{\delta^2}{2}\kappa\vartheta^*}, \tag{7.7}$$

for arbitrary $0<\delta<1$. Then, we have

$$Pr\left[\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\hat{\xi}_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\max\left\{1,\frac{b\beta_i\hat{\xi}_{ij}}{\iota_{ij}}\right\}+\varphi_i\leq(1-\delta)\kappa\vartheta^*\right]\leq e^{-(\delta^2/2)\kappa} \tag{7.8}$$

$$=e^{-\chi\frac{6\log(n)}{\delta^2\min\{\xi^*_{ij}\mid 1\leq i\leq n\cdot 1\leq j\leq m\ \xi^*_{ij}>0\}}\frac{\delta^2}{2}}\leq e^{-3\log(n)}\leq\frac{1}{n^3}. \tag{7.9}$$

Inequality (7.8) holds, since

$$\vartheta^*=\min\left\{\sum_{v_j\in L(v_i)\setminus\{v_i\},j\beta_i\xi^*_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\max\left\{1,\frac{\beta_i\xi^*_{ij}}{\iota_{ij}}\right\}+\varphi_i\,\middle|\,\forall v_i\in D\right\}\geq 1;$$

we have

$$Pr\left[\sum_{j\beta_i\hat{\xi}_{ij}>0}\left\lceil\frac{B}{\gamma_i}\right\rceil\frac{1}{\iota_{ij}}+\varphi_i+1\leq(1-\lambda)\kappa\vartheta^*\right]\leq e^{-\lambda\kappa\vartheta^*}\leq e^{-\lambda\kappa}=e^{-(3\Delta\log(n)/\Delta)}\leq\frac{1}{e^{3\ln(n)}}\leq\frac{1}{n^3}.$$

Summing over all non-leaf nodes $v_i \in$, we obtain the probability that some non-leaf node has the actual load less than $(1 - \delta)\kappa\vartheta^*$ as follows: $\Pr[\hat{\vartheta} \le (1-\delta)\kappa\vartheta^*] \le n(1/n^3) = (1/n^2)$. Again, according to Borel–Cantelli lemma, $\Pr[\hat{\vartheta} \le (1-\delta)\kappa\vartheta^*] \sim 0$.

Then, considering Lemma 7.4, we have *Pr*[each leaf node is assigned to a parent non-leaf node $\check{}$ $\hat{y} \ge (1-\delta)\kappa(\vartheta^*)] \ge (1-(1/n^5))((1-1/n^2))\sim$, when $n \sim \infty$ for $0 < \delta < 1$, and

$$\kappa = \frac{6\log(n)}{\delta^2 \min\{\xi_{ij}^* \mid 1 \le i \le n \cdot 1 \le j \le m, \xi_{ij}^* > 0\}}.$$

Hence, Algorithm 7.2 yields a solution upper bounded by $O(\log(n))opt_{\mathrm{LBPNA}}$. Moreover, this bound can be verified in polynomial time. After A^* is decided, a tree can be obtained by assigning each link a direction from the children to the parent.

7.6 Performance Evaluation

Since there are no existing works studying the LBDAT construction problem for probabilistic WSNs currently, in the simulations, LBDAT is compared with the recently published DS-based data aggregation algorithm [7] denoted by DAT. We compare both algorithms in terms of the number of non-leaf nodes: network lifetime, which is defined as the time duration until the first non-leaf node runs out of energy, and the network residual energy.

7.6.1 Simulation Environment

We build our own simulator where all the nodes have the same transmission range of 50 m, and all the nodes are deployed uniformly and randomly in a square area of size 300 m × 300 m. For each specific setting, 100 instances are generated. The results are averaged over these 100 instances (all results are rounded to integers). Moreover, a random value between 0.5 and 0.98 is assigned to the transmission success ratio (ι_{ij}) value associated with a pair of nodes (v_i and v_j) inside the transmission range. Otherwise, a random value between 0 and 0.5 is assigned to ι_{ij} associated with a pair of nodes beyond the transmission range. Every sensor node produces a packet with size 1 during each report time interval. The data receiving rate γ_i of each node v_i is randomly generated from the value between 0 and 10. The energy consumption model is that every node has the same initial 1000 units of energy. Receiving a packet consumes one unit of energy, while transmitting a packet consumes two units of energy. In the simulation, we consider the following tunable parameters: the node transmission range and the total number of nodes deployed in the square area.

7.6.2 Scenario 1: Change Side Length of Square Area

In this scenario, all nodes have the same transmission range of 20 m, and 100 nodes are deployed uniformly and randomly in a square area. The side length of the square area is incremented from 100 m to 150 m by 10 m. The impact of the area side length on the number of non-leaf nodes, the network lifetime, and the network residual energy of both algorithms is presented in Figure 7.3.

From Figure 7.3a, we can see that with the increase in the area of the network deployed region, the number of non-leaf nodes increases for both algorithms (DAT and LBDAT). This is because the probabilistic WSN becomes thinner and more non-leaf nodes are needed to maintain the connectivity of the constructed CMIS. There is no obvious trend showing which algorithm might produce more non-leaf nodes when constructing a DAT.

From Figure 7.3b, we know that the network lifetime increases for both algorithms with the side length of the deployed area increasing. It is obvious that the density of the network becomes much

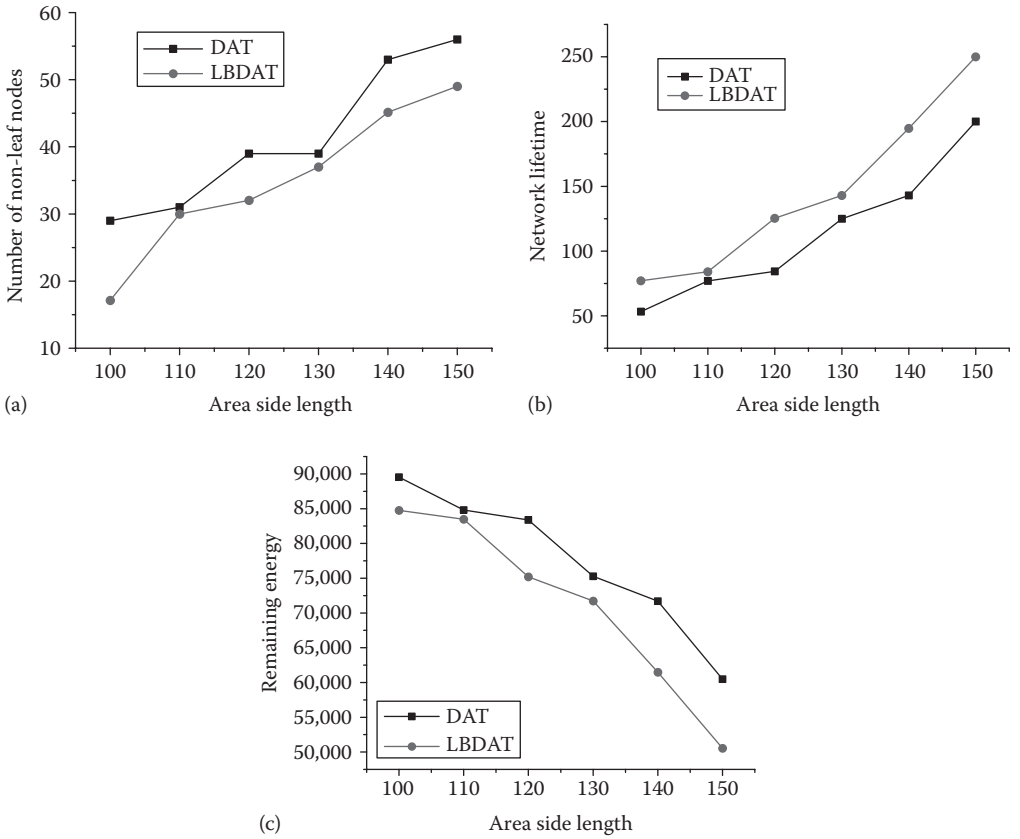


FIGURE 7.3 Simulation results for Scenario 1: (a) number of non-leaf nodes, (b) network lifetime, and (c) remaining energy.

thinner with the side length of the deployed area increasing. As to data aggregation, the thinner the network, the fewer neighbors of each non-leaf node. In other words, the aggregated data are less on each non-leaf node when the network becomes thinner. Hence, network lifetime is increasing for both algorithms. Additionally, LBDAT prolongs the network lifetime by 32% on average compared with DAT. The results demonstrate that load-balanced PNA can improve the network lifetime.

Figure 7.3c shows the remaining energy over the whole network of both algorithms. With the increase in the side length of the deployed area, the remaining energy decreases for both algorithms. As the WSN becomes thinner, more nodes are selected as non-leaf nodes to maintain the connectivity of the constructed CMIS. Additionally, the traffic load on a non-leaf node is less as mentioned earlier; hence, the remaining energy decreases with the deployed area increasing. From Figure 7.3c, LBDAT has less remaining energy than DAT. This is because LBDAT considers the load-balance factor when building a CMIS and doing PNA. Thus, the lifetime of the whole network is extended, which means the remaining energy of the network is less than DAT. In summary, Figure 7.3 indicates that constructing an LBDAT can balance the energy consumption on each non-leaf node and make the lifetime of the whole network prolonged considerably.

7.6.3 Scenario 2: Change Node Transmission Range

When 100 nodes are deployed uniformly and randomly in a square area of 300 m × 300 m, the impacts of the node transmission range on the number of non-leaf nodes, the network lifetime, and the network residual energy of LBDAT and DAT are shown in Figure 7.4.

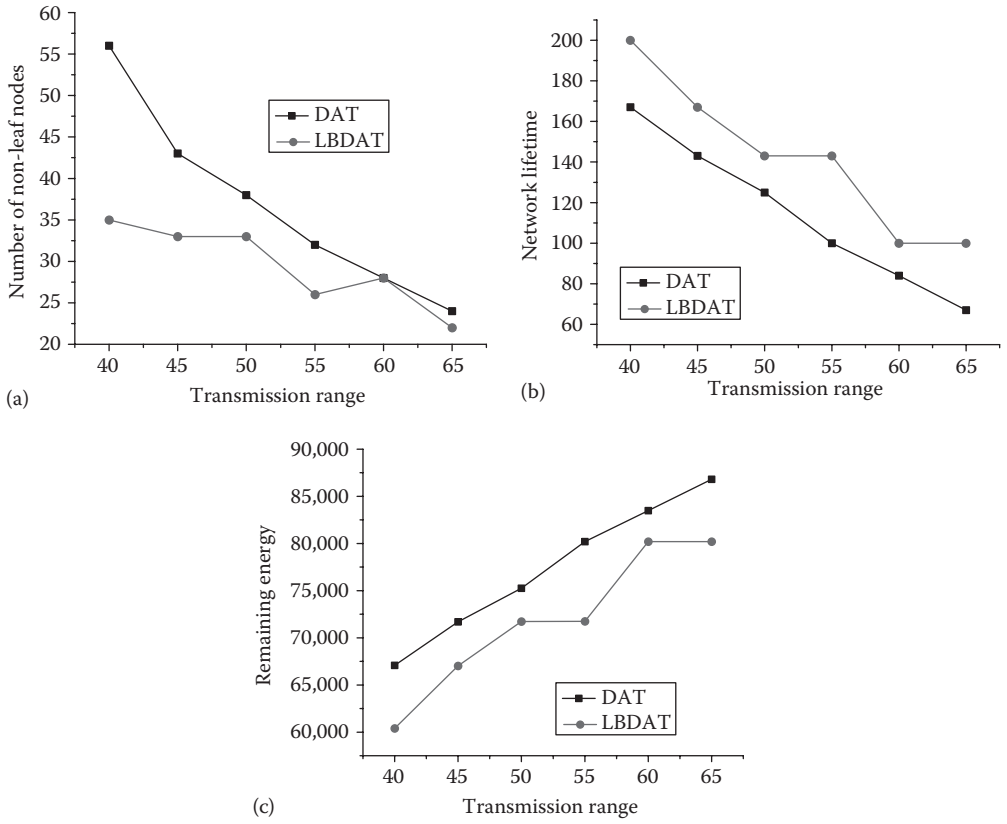


FIGURE 7.4 Simulation results for Scenario 2: (a) number of non-leaf nodes, (b) network lifetime, and (c) remaining energy.

From Figure 7.4a, we can see that with the increase in the node transmission range, the number of non-leaf nodes decreases for both algorithms (DAT and LBDAT). This is because there are more nodes in the circle with the node transmission range increasing and for the network to become denser. Hence, the connectivity of the constructed CMIS can still be maintained even using fewer non-leaf nodes.

From Figure 7.4b, we know that the network lifetime decreases for both algorithms with the node transmission range increasing. The fact is that the network becomes denser with the node transmission range increasing. The denser the network is, the more neighbors of each non-leaf node. Since we perform data aggregation, the aggregated data are increasing on each non-leaf node when the network becomes denser. Hence, network lifetime is decreasing for both algorithms. Moreover, LBDAT prolongs the network lifetime by 28% on average compared to DAT.

Figure 7.4c shows the remaining energy over the whole network of both algorithms. With the increase in the node transmission range, the remaining energy increases for both algorithms. This is because a bunch of redundant sensors exist in the more and more crowded network. Thus, the remaining energy increases for both algorithms with the network lifetime decreasing.

7.6.4 Scenario 3: Change Total Number of Nodes

Suppose that all the nodes have the same transmission range of 50 m and all the nodes are deployed uniformly and randomly in a square area of 300 m × 300 m. The impacts of the number of nodes

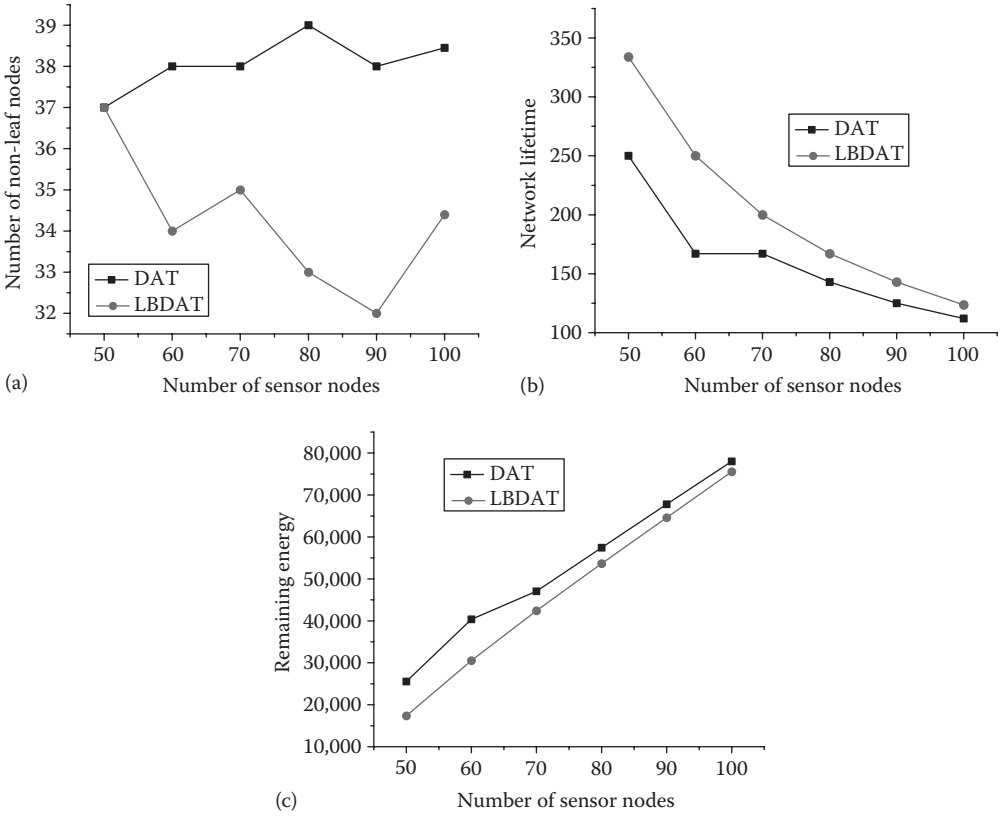


FIGURE 7.5 Simulation results for Scenario 3: (a) number of non-leaf nodes, (b) network lifetime, and (c) remaining energy.

on the number of non-leaf nodes, the network lifetime, and the network residual energy of both algorithms are shown in Figure 7.5.

From Figure 7.5a, we can see that with the increase in the number of the sensor nodes, the number of non-leaf nodes almost remains stable (from 32 to 39) for both algorithms (DAT and LBDAT). This is because the area of the network deployed region and the transmission range of n nodes are kept fixed.

From Figure 7.5b, we can see that the network lifetime decreases for both algorithms with the number of nodes increasing. This is because we perform data aggregation in a more and more crowded network. Additionally, we can see that LBDAT prolongs the network lifetime by 42% on average compared to DAT. The results demonstrate that the LBPNA can improve the network lifetime significantly.

Figure 7.5c shows the remaining energy over the whole network of both algorithms. With the increase in the number of nodes, the remaining energy increases for both algorithms. This is because the WSN becomes denser, and a lot of redundant sensor nodes exist in the WSN.

7.7 Summary

In this chapter, we address the fundamental problems of constructing an LBDAT in probabilistic WSNs. We first solve the CMIS problem, which is NP-hard, in two phases. In the first phase, we aim to find the optimal MIS such that the minimum *potential load* of all the independent nodes is maximized. To this end, a near optimal approximation algorithm is proposed. In the second phase, the minimum-sized set of LBMIS connectors are found to make the LBMIS connected. The theoretical lower and upper bounds of the number of non-leaf nodes are analyzed as well. Subsequently, we study the LBDAT construction

problem and propose an approximation algorithm by using the linear relaxing and random rounding techniques. After an LBPNA is decided by assigning a direction to each link, we obtain an LBDAT. The simulation results show that the proposed algorithms can extend the network lifetime significantly.

Our next step is to come up with a sophisticated model to integrate the aforementioned three phases together and analyze the overall performance of the LBDAT construction problem. This is because the three phases of the algorithm might lead to performance loss/improvement since we did not investigate the correlations among them. Another direction is to design distributed algorithms for the LBDAT construction problem under both DNM and PNM.

References

1. R. Cristescu, B. Beferull-Lozano, and M. Vetterli. On network correlated data gathering. In *23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, Hong Kong, China, 2004, Vol. 4, pp. 2571–2582.
2. S. Madden, R. Szewczyk, M.J. Franklin, and D. Culler. Supporting aggregate queries over ad-hoc wireless sensor networks. In *Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications*. Callicoon, New York, IEEE, 2002, pp. 49–58.
3. H.Ö. Tan and I. Körpeoglu. Power efficient data gathering and aggregation in wireless sensor networks. *ACM SIGMOD Record*, 32(4):66–71, 2003.
4. H.O. Tan, I. Korpeoglu, and I. Stojmenovic. Computing localized power-efficient data aggregation trees for sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 22(3):489–500, 2011.
5. S. Ji and Z. Cai. Distributed data collection and its capacity in asynchronous wireless sensor networks. In *2012 Proceedings of IEEE INFOCOM*, Orlando, FL, 2012, pp. 2113–2121.
6. X. Chen, X. Hu, and J. Zhu. Minimum data aggregation time problem in wireless sensor networks. *Mobile Ad-Hoc and Sensor Networks*, 3794: 133–142, 2005.
7. P.J. Wan, S.C.H. Huang, L. Wang, Z. Wan, and X. Jia. Minimum-latency aggregation scheduling in multihop wireless networks. In *Proceedings of the 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. New Orleans, Louisiana, ACM, 2009, pp. 185–194.
8. S. Ji, R. Beyah, and Y. Li. Continuous data collection capacity of wireless sensor networks under physical interference model. In *IEEE Eighth International Conference on Mobile Ad Hoc and Sensor Systems*, Valencia, Spain, 2011, pp. 222–231.
9. Y. Xue, Y. Cui, and K. Nahrstedt. Maximizing lifetime for data aggregation in wireless sensor networks. *Mobile Networks and Applications*, 10(6):853–864, 2005.
10. H.C. Lin, F.J. Li, and K.Y. Wang. Constructing maximum-lifetime data gathering trees in sensor networks with data aggregation. In *2010 IEEE International Conference on Communications* (*ICC*), Cape Town, South Africa, 2010, pp. 1–6.
11. S. Ji, Y. Li, and X. Jia. Capacity of dual-radio multi-channel wireless sensor networks for continuous data collection. In *Proceedings of IEEE INFOCOM*, Shanghai, China, 2011, pp. 1062–1070.
12. A. Cerpa, J.L. Wong, L. Kuang, M. Potkonjak, and D. Estrin. Statistical model of lossy links in wireless sensor networks. In *Fourth International Symposium on Information Processing in Sensor Networks, 2005* (*IPSN'05*). Los Angeles, California, IEEE, 2005, pp. 81–88.
13. G. Zhou, T. He, S. Krishnamurthy, and J.A. Stankovic. Impact of radio irregularity on wireless sensor networks. In *Proceedings of the Second International Conference on Mobile Systems, Applications, and Services*. Boston, Massachusetts, ACM, 2004, pp. 125–138.
14. Y. Liu, Q. Zhang, and L.M. Ni. Opportunity-based topology control in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 21(3):405–416, 2010.
15. Y. Wu, S. Fahmy, and N.B. Shroff. On the construction of a maximum-lifetime data gathering tree in sensor networks: Np-completeness and approximation algorithm. In *27th Conference on Computer Communications.* Phoenix, AZ, IEEE, 2008, pp. 356–360.

16. S.C.H. Huang, P.J. Wan, C.T. Vu, Y. Li, and F. Yao. Nearly constant approximation for data aggregation scheduling in wireless sensor networks. In *INFOCOM 2007: 26th IEEE International Conference on Computer Communications*. Anchorage, Alaska, IEEE, 2007, pp. 366–372.
17. X. Xu, X.Y. Li, X. Mao, S. Tang, and S. Wang. A delay-efficient algorithm for data aggregation in multihop wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 22(1):163–175, 2011.
18. Y. Li, L. Guo, and S.K. Prasad. An energy-efficient distributed algorithm for minimum-latency aggregation scheduling in wireless sensor networks. In *2010 IEEE 30th International Conference on Distributed Computing Systems* (*ICDCS*). Genoa, Italy, IEEE, 2010, pp. 827–836.
19. K. Kalpakis and S. Tang. A combinatorial algorithm for the maximum lifetime data gathering with aggregation problem in sensor networks. *Computer Communications*, 32(15):1655–1665, 2009.
20. D. Virmani and S. Jain. Construction of decentralized lifetime maximizing tree for data aggregation in wireless sensor networks. *World Academy of Science, Engineering and Technology*, 52:54–63, 2009.
21. D. Luo, X. Zhu, X. Wu, and G. Chen. Maximizing lifetime for the shortest path aggregation tree in wireless sensor networks. In *2011 Proceedings of IEEE INFOCOM*, Shanghai, China, 2011, pp. 1566–1574.
22. S. Lin, J. Zhang, G. Zhou, L. Gu, J.A. Stankovic, and T. He. ATPC: Adaptive transmission power control for wireless sensor networks. In *Proceedings of the Fourth International Conference on Embedded Networked Sensor Systems*. Boulder, CO, ACM, 2006, pp. 223–236.
23. D. Son, B. Krishnamachari, and J. Heidemann. Experimental study of concurrent transmission in wireless sensor networks. In *Proceedings of the Fourth International Conference on Embedded Networked Sensor Systems*. Boulder, CO, ACM, 2006, pp. 237–250.
24. S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, 2004.
25. M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman & Co., New York, 1990.
26. S. Gueye and P. Michelon. A linearization framework for unconstrained quadratic (0–1) problems. *Discrete Applied Mathematics*, 157(6):1255–1266, 2009.

8

Biometric Sensors and How They Work

8.1 Introduction **8**-1
8.2 Biometric Sensors and Authentication **8**-2
8.3 Different Types of Biometric Sensors **8**-4
Fingerprint Sensors • Face Biometric Sensors • Other Biometric Sensors • Multimodal Biometric Sensors
8.4 Biometric Sensors in Consumer Electronics and Mobile Devices **8**-8
Fingerprint Sensors • Other Biometric Sensors
8.5 Challenges and Issues with Biometric Sensors **8**-11
Interoperability of Biometric Sensors • Quality of Biometric Sensors • Cost
8.6 Future of Biometric Sensor Technology **8**-13
8.7 Summary **8**-14
References **8**-14

Sinjini Mitra
California State University

8.1 Introduction

Biometrics-based techniques have emerged as the most promising options for recognizing individuals in recent years [1]. Instead of authenticating people and granting them access to physical and virtual domains based on passwords, PINs, smart cards, keys, tokens, and so forth, these methods examine an individual's physiological and/or behavioral characteristics in order to determine and/or ascertain his identity. Passwords and PINs are hard to remember, easy to forget, and can be guessed or stolen; cards and tokens can be misplaced, forgotten, or duplicated; cards with magnetic strips can become corrupted and unreadable. However, an individual's biological traits cannot be misplaced, forgotten, stolen, or forged, and thus, they provide a more secure and reliable means of authentication than the traditional methods [2]. Commonly used biometrics include face, fingerprints, iris, hand geometry, voiceprint (physical traits), gait, signature, and keystroke dynamics (behavioral traits), some sample images of which are being shown in Figure 8.1.

Biometrics are typically collected using a device called a *sensor*. These sensors are used to acquire the data needed for recognition and to convert the data to a digital form, which is in turn processed using a biometric identification system. Given the utmost importance of ensuring security in different parts of the society, biometric sensors are widely deployed today. The modern world has seen a rapid evolution of the technology of biometric authentication prompted by an increasing urgency for security following the 9/11 attacks. Biometrics are used today extensively in immigration and border control, and the US-VISIT program currently uses face and fingerprint information to validate travelers entering the country. Their use in law enforcement is well known as well. Fingerprints collected at crime scenes and facial images captured using surveillance cameras are commonly used to identify criminals by matching

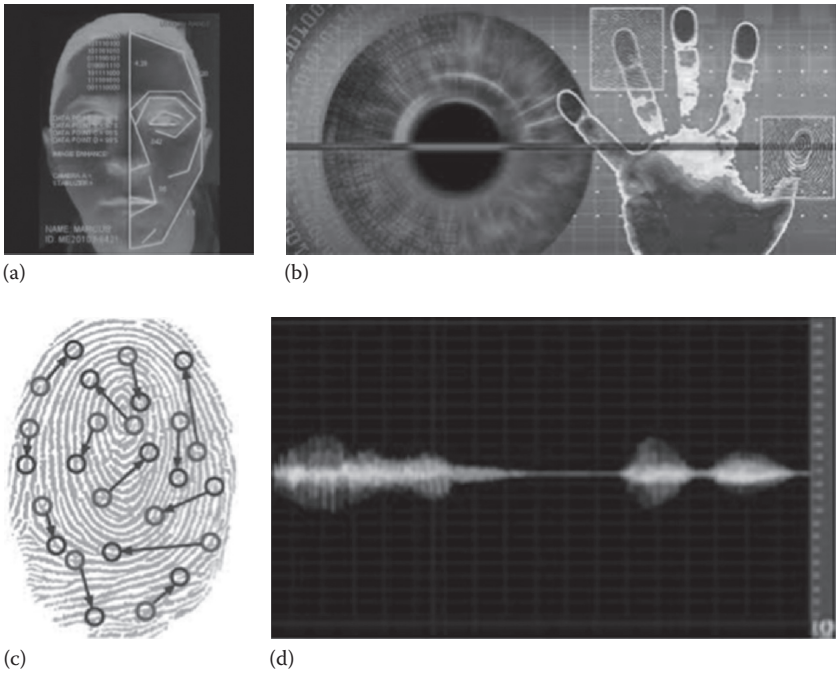


FIGURE 8.1 Commonly used biometrics: (a) face, (b) iris and hand geometry, (c) fingerprints, and (d) voiceprint.

them to existing records in police databases. Moreover, the use of biometrics is increasingly being recognized in the banking and finance industry as a viable means to prevent identity fraud, safeguard data, improve ease of use, and streamline operational processes. For instance, biometric solutions can provide a secure log-in to a bank's network by replacing username/passwords with something that nobody can lose or forget—your face! Apart from banking applications, biometrics are increasingly becoming popular in providing secure access control to consumer electronics like mobile phones, tablets, and laptops.

The rest of this chapter is organized as follows. Section 8.2 contains an overview of biometric sensors along with the authentication process based on those. Section 8.3 presents a detailed study of the three most popular biometric sensors, namely, face, fingerprints, and iris. Biometric sensors embedded in consumer electronics and mobile devices are discussed in Section 8.4. In Section 8.5, we present issues and challenges associated with biometric sensors, and in Section 8.6, we include a snapshot of the upcoming and future trends in biometric sensory technology. The chapter is finally concluded in Section 8.7.

8.2 Biometric Sensors and Authentication

Biometric sensors/detectors are semiconductors (meaning its electric conductivity is in between a conductor and an insulator), which use algorithms to process images from the user's physical characteristics [3]. Such sensors work by producing electrical currents when they scan a user's physical characteristic (the *biometric*), which in most cases is stored as an image of the relevant characteristic. Many physical characteristics may be scanned by a biometric sensor including face, eyes, fingerprints, or DNA. Sensors contain an analog to digital converter enabling it to digitize the image and store the digital information in memory so that it can verify the user next time he or she needs to authenticate their identity.

There is a wide array of companies offering several different types of biometric sensors that may be used for several applications. Many biometric technology companies specialize in fingerprint sensors for embedded fingerprint applications in USB devices, enrollment stations, MOC technologies, and networking ID verification. Sensors can be touchless, using thermal and optical stripe line sensors.

Other biometric technology companies are marketing their biometric sensors/detectors for use in applications such as locks and safes. The added security that comes with biometric sensor locks and safes (with fingerprint sensors typically) is the primary reason why many homes, institutions, and corporate facilities choose them for their security systems. It is nearly impossible to forge fingerprints and bypass sensory security systems; therefore, sensors provide an upgrade from conventional locks. Biometric sensors and detectors for locks and safes eliminate some of the inevitable nuisances and weaknesses of the conventional lock and key. Access time is reduced because the user simply scans his fingerprints or iris instead of finding one of many keys on a key chain. Security is increased because thieves have a much harder time accessing a biometric sensory lock than a conventional lock.

Sensors are also beginning to revolutionize computer and network security in several different ways. For example, biometric fingerprint sensors are becoming more effective, accurate, and useful in the information technology industry and provide convenience and secure user authentication at low cost. Biometric sensor products are being developed and marketed at a greater pace than ever before including USB flash drives and PC peripherals. Success is driven by the need for high reliability, security, and a market for alternatives to passwords, keys, and PINs. Biometric sensors and detectors are perhaps most useful and important in financial security applications. Personal financial information is often at risk of infiltration by hackers and frauds especially when transactions are made over the Internet. Using biometric sensors to control access to financial information has great benefits for both the customer and the financial institution. When private financial data are kept more secure, the customer benefits from the peace of mind that comes from knowing your money is as safe as possible, and the bank or financial company using the technology benefits from a greater reputation for security.

New innovations are constantly improving biometric sensors/detectors and increasing the range of their possible applications. An example of a recent innovation in biometric fingerprint sensors is the new identification flats, which requires the user to scan all 10 fingerprints. It uses live scan hardware devices that employ advanced image processing technology to perform ultrafast capture and quality control of fingerprint flats resulting in high-security and high-accuracy performance. Another important application today consists of consumer electronics such as mobile devices. Face and fingerprint sensors are prevalent in several of commonly used smartphones today, like Apple's iPhone (5 and 5S), Motorola Atrix, and Samsung Galaxy, to name a few.

There are two types of authentication problems based on images obtained from biometric sensors of all types: (1) Identification: who am I? (2) Verification: am I whom I claim to be? A typical biometric system has three components. In the first stage, called *enrollment*, the images are captured by some device, preprocessed for feature extraction, and enrolled in the system. The second stage, called *matching or testing*, consists of matching the enrolled image with existing images in a database to determine if a match occurs or not. The last stage is the *decision-making* stage where a decision is made about

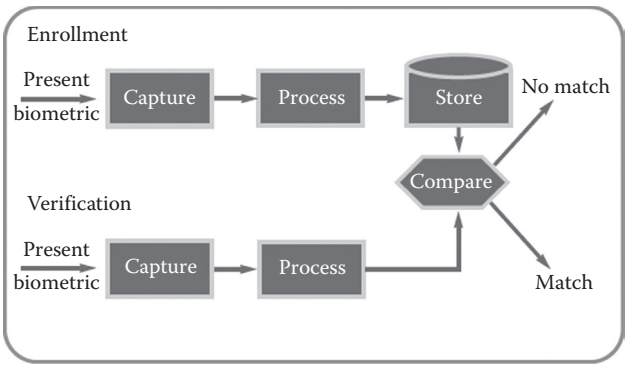


FIGURE 8.2 Schematic illustration of the process of biometric authentication. (https://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS3/handout/.)

whether the person is genuine or impostor. Figure 8.2 shows a schematic diagram illustrating how the process of biometric authentication works.

Because of the decision-theoretic framework involved, there is the scope for errors in any biometric authentication system. In particular, there are two types of errors in the context of the verification task, namely, (1) false acceptance rate (FAR) and (2) false rejection rate (FRR). The first one arises when an impostor is declared genuine by the system (i.e., the system finds a match when it should not have), and the second one arises when a genuine person is declared an impostor (when the system does not find a match when it actually exists). These are typically determined with respect to thresholds that are set on the match scores, so that a set of FARs and FRRs are generated for a system with varying thresholds. The most commonly used metric for performance evaluation of a biometric system is called the equal error rate (ERR), which is that value where the FAR and FRR approximately coincide.

8.3 Different Types of Biometric Sensors

In this section, we present an overview of sensors used for capturing two most commonly used biometrics today, namely, (1) fingerprints and (2) face. We conclude this section with a brief outline of some other types of biometric sensors in use today, namely, iris, ear, and some of those for behavioral biometrics like voiceprint and keystroke dynamics.

8.3.1 Fingerprint Sensors

Fingerprints are by far the most popular biometric choice for a number of reasons. They are the oldest and most accepted biometric modality and tend to be one of the easiest characteristics to capture and verify. According to the FBI, the odds of two people sharing the same fingerprints are one in 64,000,000,000. Fingerprints differ even for 10 fingers of the same person [4].

The analysis of fingerprints for matching purposes in order to perform identification or verification generally requires the comparison of several features of the print pattern. These include patterns, which are aggregate characteristics of ridges, called *minutiae*, which are unique to an individual [5]. In fact, [6] shows identical twins who have the same face structure, and DNA, which can be identified separately with the help of minutiae-based features of their thumb prints. The three basic patterns of fingerprint ridges are the arch, loop, and whorl:

- *Arch*: The ridges enter from one side of the finger, rise in the center forming an arc, and then exit the other side of the finger.
- *Loop*: The ridges enter from one side of a finger, form a curve, and then exit on that same side.
- *Whorl*: Ridges form circularly around a central point on the finger.

A fingerprint sensor is an electronic device used to capture a digital image of the fingerprint pattern. The captured image is called a live scan. This live scan is digitally processed to create a biometric template (a collection of extracted features), which is stored and used for matching. Fingerprint sensors or readers can be broadly classified into six different kinds: (1) capacitive sensors, (2) optical sensors, (3) thermal sensors, (4) pressure sensors, (5) radio frequency (RF) sensors, and (6) ultrasonic sensors.

Capacitive sensors use an array of capacitor plates to image the fingerprint [7]. Capacitive sensors are constructed from many different media, such as copper, indium tin oxide (ITO), and printed ink. Skin is conductive enough to provide a capacitive coupling with an individual capacitive element on the array. Ridges, being closer to the detector, have a higher capacitance, and valleys have a lower capacitance. Some capacitive sensors apply a small voltage to the finger to enhance the signal and create better image contrast. Capacitive sensors can be sensitive to electrostatic discharge (shock), but they are insensitive to ambient lighting and are more resistant to contamination issues than some optical designs.

Optical sensors work like digital cameras to capture a digital image of a fingerprint. The top layer of the sensor, where the finger is placed, is known as the touch surface. Beneath this layer is a light-emitting

FIGURE 8.3 Optical fingerprint sensor. (http://www.morpho.com/identification/secure-biometric-access/fingerprint-sensors/morphosmart-tm-mso-300-series/?lang=en.)

phosphor layer, which illuminates the surface of the finger. The light reflected from the finger passes through the phosphor layer to an array of solid-state pixels, which captures a visual image of the fingerprint. A scratched or dirty touch surface can cause a bad image of the fingerprint. A disadvantage of this type of sensor is the fact that the imaging capabilities are affected by the quality of skin on the finger. For instance, a dirty or marked finger is difficult to image properly. Also, it is possible for an individual to erode the outer layer of skin on the fingertips to the point where the fingerprint is no longer visible. It can also be easily fooled by an image of a fingerprint if not coupled with a *live finger* detector. However, unlike capacitive sensors, this sensor technology is not susceptible to electrostatic discharge damage. Optical sensors thus make it possible to read fingerprints from a distance. Figure 8.3 shows an optical sensor currently available in the market.

Thermal sensors use the same pyroelectric material that is used in infrared cameras. When a finger is presented to the sensor, the fingerprint ridges make contact with the sensor surface and the contact temperature is measured, and the valleys do not make contact and are not measured. A fingerprint image is created by the skin-temperature ridges and the ambient temperature measure for valleys. The biggest drawback of this technique is that the temperature change is dynamic and it only takes about a 10th of a second for the sensor surface touching ridges and valleys to come to the same temperature, erasing the fingerprint image. Additionally, this technology has many of the same contamination and wear issues as other sensors. While it can operate over a wide range of temperatures, if the ambient temperature is close to the finger surface temperature, the sensor requires heating to create a temperature difference of at least 1°C. Figure 8.4 shows a currently used thermal fingerprint sensor available in the market today.

Pressure sensing scanners can be made very thin and are often used in electronic devices. Early pressure sensing scanners had to make a trade-off between durability and quality because any protective layer on the detector surface would diminish the contrast of the impression. There are two types of pressure sensing detectors available: conductive film detectors and microelectromechanical (MEMS) devices. Conductive film sensors use a double-layer electrode on flexible films. MEMS is a newer technology that uses extremely tiny silicon switches on a silicon chip. When a fingerprint ridge touches a switch, it closes and is detected electronically.

In RF type of sensors, a low RF signal is applied to the user's finger and then read by the detector array, with each pixel operating like a tiny antenna. The advantage of this detector is that it reads the fingerprint from the dermal layer underneath the surface making it less susceptible to damaged or dry fingertips.

Ultrasonic scanners have an advantage of being able to see beneath the skin. This provides not only verification of a live finger but also more information as a biometric measure. But this technology is slow, expensive, bulky, and too data intensive for most access control applications; hence, they are seldom used in large practical situations.

FIGURE 8.4 Thermal fingerprint sensor. (http://diuf.unifr.ch/diva/biometrics/MyIdea/en/technology/protocol.html.)

Using any of the aforementioned sensors, a fingerprint sensor can be either a *static* one or a *swipe* one. A static fingerprint reader/sensor requires the user to place the finger motionless on a surface. A swipe fingerprint reader/sensor, on the other hand, requires the user to gently move the upper part of the finger across the sensor. The complete image is pieced together by accumulating the partial images obtained as the finger moves across the sensor surface. One primary advantage of a swipe sensor over a static one is that they can be much smaller in size and hence also cheaper. Moreover, the swiping motion reduces contamination (e.g., grease) buildup on the sensor, which is particularly a problem for optical sensors. On the other hand, matching using images from swipe fingerprint sensors tends to be less accurate due to increased distortion of the finger surface as it is pulled across the sensor.

8.3.2 Face Biometric Sensors

As one of the most successful applications of image analysis and understanding, face recognition has recently received significant attention, especially during the past few years. Face recognition can occur from both still and video images and also from both 2-D and 3-D images. Facial images are typically captured using digital cameras, from stand-alone high-resolution cameras to cameras installed in computers and mobile devices. One commercially used facial biometric sensor is the *PreFace with Camera API* manufactured by a company called *Aware* [8]. This device automatically captures and analyzes biometric facial images in order to maximize their quality, thus improving operational efficiency of the facial recognition process. PreFace integrates with the camera to perform analysis of the live facial image. Once basic quality criteria are met, PreFace triggers the camera to take a full-resolution image. Following capture, PreFace performs a thorough image analysis, which reports image geometry and noncompliant features. Scaling, rotation, and cropping of the image are performed to meet highly configurable targets and thresholds. These thresholds are derived from the ISO/IEC 19794-5 standard for biometric facial image quality. Results are reported to the user. Details about cameras in mobile devices are discussed in Section 8.4.

The problem of automatic face recognition involves three key steps/subtasks: (1) detection and normalization of faces, (2) feature extraction, and (3) identification and/or verification. Sometimes, different subtasks are not totally separated. For example, face detection and feature extraction can be achieved simultaneously using facial features like eyes and nose. Depending on the nature of the application, the sizes of the training and test databases, clutter and variability of the background, noise, occlusion, and speed requirements, some of the subtasks can be very challenging. Though fully automatic face recognition systems must perform all three subtasks, research on each subtask is critical. The NSTC [9] provides a highlighted summary of research on face segmentation/detection and feature extraction methods.

8.3.3 Other Biometric Sensors

Among the other popular biometric modalities, iris, voiceprint, hand geometry, and palm print are the most popular ones. Iris recognition methods use a noninvasive method for acquiring images, and an iris scan is similar to taking a photograph and can be performed from about 10 cm to a few meters away. There is no need for the person being identified to touch any equipment that has recently been touched by a stranger, thereby eliminating an objection that has been raised in some cultures against fingerprint scanners, where a finger has to touch a surface, or retinal scanning, where the eye must be brought very close to an eyepiece (like looking into a microscope), or recite a specific phonemic sequence for voice recognition. This is definitely an advantage over other biometrics-based methods where the operator is required to make physical contact with a sensing device or otherwise take some special action.

Voiceprint biometric is typically captured using a microphone. It is noninvasive and contactless and easy to collect; however, it is difficult to control sensor variances (caused by background noise, quality of the microphone, etc.) that significantly impact authentication accuracies. Voice samples are sometimes collected over the telephone as well, for example, for teleshopping and telebanking applications where the person has to authenticate himself or herself for conducting certain transactions.

In hand geometry, flatbed scanners are used at low resolution starting at 45 dpi to capture the images [10]. Since identification based on this biometric relies on reliable extraction of hand contours, typically hand geometry sensors use fixed pegs at specific positions between fingers to ensure correct alignment [11]. More advanced peg-free systems are also gradually becoming available in the market as discussed in [10]. Palm print sensors use the same technique as fingerprint sensors [12] like minutiae for matching purposes, and hence the classification principles are the same; the only difference being in the size. Since palm print sensors need to capture the entire palm of the hand, they are bigger in size than fingerprint sensors and hence more expensive [13]. The first online palm print–capturing device was invented at Hong Kong Polytechnic University in 1999 and acquired real-time palm print images using a white light source and a CCD camera capturing the reflected image of a hand placed on a glass plate [14]. While law enforcement and forensics agencies have always collected fingerprints, it is only in recent years that large palm print databases are becoming available.

Apart from the conventional biometrics like face, fingerprint, and iris, work is increasingly being done on other biometric traits, such as keystroke dynamics (for computers and other electronics) and ECG (for telemedicine). Keystroke dynamics is the process of analyzing the way a user types at a terminal by monitoring the keyboard inputs thousands of times per second and attempts to identify them based on habitual rhythm patterns in the way they type. According to some people, the use of keystroke rhythm is a natural choice for computer security. This argument stems from observations that similar neurophysiological factors that make written signatures unique are also exhibited in a user's typing pattern [15]. When a person types, the latencies between successive keystrokes, keystroke durations, finger placement, and applied pressure on the keys can be used to construct a unique signature (i.e., profile) for that individual. For well-known, regularly typed strings, such signatures can be quite consistent. Furthermore, recognition based on typing rhythm is not intrusive, making it quite applicable to computer access security, as users will be typing at the keyboard anyway. Moreover, unlike other biometric systems, which may be expensive to implement, keystroke dynamics is almost free—the only hardware required is the keyboard, which is attached to all computers.

Another physical trait that is recently gaining momentum for authentication applications is the human ear. Although it has been known that a person's ear is a promising candidate for biometric identification, it has not been explored much until recent times when several ear recognition systems have emerged [16,17] including methods using 3-D images that exploit the shape of the ear [18,19]. There are at least three methods for ear identification: (1) taking a photo of an ear, (2) taking *earmarks* by pushing ear against a flat glass, and (3) taking *thermogram* pictures of the ear [20]. Taking photo of the ear is the most commonly used method in research. The photo is taken and it is combined with previous taken

photos for identifying a person. The earmarks are used mainly in crime solving. Even though some judgments are made based on the earmarks, currently they are not accepted in courts. The thermogram pictures could be one solution for solving the problem when the ears are hidden, say, for instance, under a hat or due to hairstyle (long hair, especially for women).

8.3.4 Multimodal Biometric Sensors

Most biometric systems deployed in real-world applications are unimodal, that is, they rely on the evidence of a single source of information for authentication (e.g., single fingerprint or face). These systems have to contend with a variety of problems such as the following:

- *Noise in the sensed data*: A fingerprint image with a scar and a voice sample altered by cold are examples of noisy data. Noisy data could also result from defective or improperly maintained sensors (e.g., accumulation of dirt on a fingerprint sensor) or unfavorable ambient conditions (e.g., poor illumination of a user's face in a face recognition system).
- *Intraclass variations*: These variations are typically caused by a user who is incorrectly interacting with the sensor (e.g., incorrect facial pose).
- *Interclass similarities*: In biometric systems comprising a large number of users (say, at airports), there may be interclass similarities (overlap) in the feature space of multiple users.
- *Nonuniversality*: The biometric system may not be able to acquire meaningful biometric data from a subset of users. A fingerprint biometric system, for instance, may extract incorrect minutiae features from the fingerprints of certain individuals due to the poor quality of ridges (as may occur to aging, illness, etc.)
- *Spoof attacks*: This type of attack is especially relevant when behavioral traits such as signature or voice are used. However, some physical traits such as fingerprints are also susceptible to spoof attacks occasionally.

Some of the limitations imposed by unimodal biometric systems can be overcome by including multiple sources of information for establishing identity [21]. Such systems, known as *multimodal biometric systems*, are expected to be more reliable due to the presence of multiple, (fairly) independent pieces of evidence [22]. These systems are able to meet the stringent performance requirements imposed by various applications. They address the problem of nonuniversality, since multiple traits ensure sufficient population coverage. They also deter spoofing since it would be difficult for an impostor to spoof multiple traits of a genuine user simultaneously. More importantly, using more than one trait ensures greater reliability of the results, which is expected to maximize performance accuracy (minimize false alarm rates).

In case of multimodal biometric systems, there are multiple sensors associated with the multiple modalities. For example, a multimodal system using face and fingerprints will consist of a fingerprint sensor as well as a camera to capture fingerprint and facial images. Similar sensor technologies, as in the case of the unimodal biometric systems, are used in such scenarios for the individual biometric traits.

8.4 Biometric Sensors in Consumer Electronics and Mobile Devices

Given the current trend in deploying biometrics-based access control in consumer electronics, particularly in mobile phones, tablets, and laptops, this section summarizes the recent advances in biometric sensors in various consumer electronics. Incorporated with advanced sensing platforms that could detect physiological and behavioral signals of various kinds, many types of biometric methods could be implemented on cell phones and other mobile devices. This offers a wide range of possible applications such as personal privacy protection, mobile bank transaction and e-commerce transaction service security, telemedicine monitoring, and data storage protection. With the advent of smartphones and

tablets, people are constantly connected to the Internet, performing transactions of personal and sensitive nature on a continuous basis, so it is imperative to enhance the security of these devices. Thus, this is an emerging frontier in security research that is being increasingly explored in the current decade.

An Internet blog on biometric technology (December 2010) states that the main driver of biometric authentication for smartphones/tablets is their position to overtake personal computers as the preferred mode of communication and transactions. Since extremely sensitive transactions are performed on these devices and they possess multifaceted functionality, there will be a very pressing need to develop airtight security. According to a report in the American Banker (September 26, 2012), researchers at Intel, for example, have developed new mobile technology that combines software with a biometric sensor embedded in the computing device that recognizes the vein patterns on a person's palm, allowing access to banking sites, social networks, and other account-based services. This product is also *contactless*, while older biometric sensors require the finger to come into contact with the reader.

8.4.1 Fingerprint Sensors

The advent of small-sized solid-state fingerprint sensors permits these devices to be easily embedded in various applications such as laptops, computer peripherals, cell phones, PDAs, and tablets. The ease of interacting with fingerprint sensors (compared to, say, iris cameras) has contributed to their increase in popularity. This has resulted in a proliferation of these devices and their subsequent inclusion in a variety of applications (Figure 8.5).

The first computer manufacturer to embed a fingerprint scanner in laptops is IBM. The ThinkPad T42 laptop that came out in 2004 was the first ever computer to be equipped with a fingerprint reader [23]. A finger scan was required to gain log-on access to this portable computer and its contents, delivering simplified access to password-protected personal and financial information, websites, documents, and e-mail. The new software and the embedded security chip were seamlessly integrated with the fingerprint reader, protecting vital security information, such as encryption keys, electronic credentials,

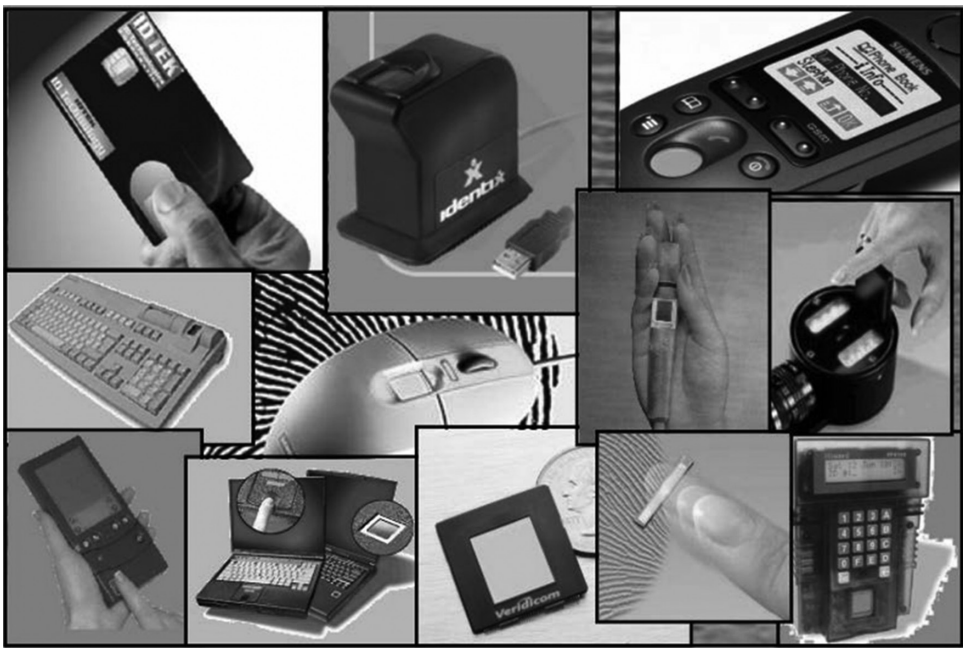


FIGURE 8.5 Some commonly used fingerprint sensors that are embedded in computer peripherals and other devices to facilitate user authentication.

and passphrases, and guarding against unauthorized user access. This provided an additional layer of security to consumers. Moreover, the swiping and recognition process could take place in under a minute, thus streamlining access to a considerable extent. Since then, many of the IBM ThinkPads (followed by Lenovo ThinkPads) came out with more advanced fingerprint scanner that provided advanced authentication features, thus contributing to the growth of the mobile biometric market. Today, most of the frontline laptops like Sony Vaio, Dell Latitude, Acer Aspire, HP Compaq, and Toshiba feature fingerprint sensors. With such rapid proliferation, these additional hardware costs are diminishing, making these fingerprint readers more and more affordable.

A recently released report [24] projects that the market for biometric security devices on mobile phones will reach $161 million by 2015, an increase of $30 million; the analysis found that embedded fingerprint sensors in mobile phones will primarily drive growth in this market as more firms seek additional security measures for mobile phones; voice biometrics is also expected to be another large driver of growth, as they are likely to be used as part of a multifactor authentication system. The report mentions Motorola's efforts as the biggest evidence—its latest smartphone, the Atrix, uses biometrics (fingerprint sensors) for security purposes. One existing fingerprint recognition system on mobile phones can be found in [25], which utilizes a sensor for acquisition of the fingerprint images and implements an algorithm on internal hardware to perform verification of users. Experimental results report good performance; however, one major inconvenience is that it requires an external attachment as a scanner for the images.

Another report published in 2012 [26] prompts people to deduce with a great degree of certainty that biometric security might be coming to Apple's gadgets like the iPhone and the iPad very soon. Indeed, the iPhone 5s came out in 2013 with a fingerprint scanner—owners of this phone can use a fingerprint to physically unlock their devices instead of using a numeric passcode. According to a recently published report [27], Apple's decision to include a fingerprint scanner in its new iPhone 5S is an important step toward bringing biometrics-based authentication into the mainstream. Recent reports have indicated some issues and difficulties with using the fingerprint reader on iPhones, and a much-enhanced technology is expected on future iPhone. In fact, Forbes reports that "iPhone 5S Touch Id fingerprint scanner is a fail for 20% of users" [28]. Figure 8.6 shows the fingerprint reader in iPhone 5S.

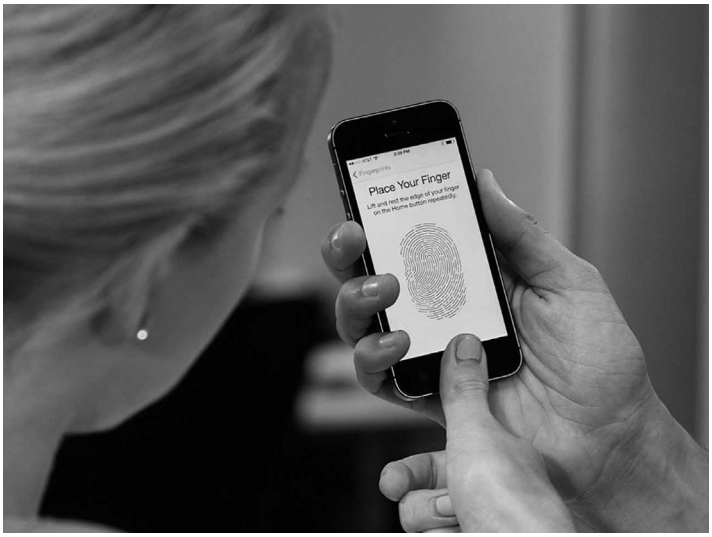


FIGURE 8.6 The fingerprint reader in iPhone 5S. (Business Insider, http://www.businessinsider.com/iphone-5s-fingerprint-sensor-setup-2013-9, September 21, 2013.)

FIGURE 8.7 iPhone's RecognizeMe app for facial recognition. (http://www.iphone4jailbreak.org/.)

8.4.2 Other Biometric Sensors

Although the use of fingerprint sensors is the most common in the mobile platform, other biometrics like iris and face are also gaining momentum in this market. Dave et al. [29] discuss a face recognition algorithm that was implemented on *Droid* phones. The authors used face detection first and then eigenfaces [30] and fisherfaces [31] algorithms, the two most popular face recognition methods today, to perform the recognition task. Based on a training set of 45 images containing 9 persons and a test set of 134 face images, they report a maximum verification accuracy of 84.3% for eigenfaces and 94% for fisherfaces.

The Samsung Galaxy SIII phone features both face- and voice recognition–based authentication mechanisms. The Galaxy S4 that released in 2013 was equipped with an iris tracker that helped the users scroll through their phone in a contactless fashion. Recent reports mention that the upcoming Samsung Galaxy S5 phone will feature a fingerprint reader as well. Today, there is also an application called *RegconizeMe* on the iPhone that could help the user unlock the device using facial recognition. This application is invented by iClarified. The original RecognizeMe 1.0 was updated to the more current RecognizeMe 2.0 version that works with all iPhone 4 and 5 devices that have a front facing camera. All these conclusively prove the explosive growth in the domain of mobile biometric sensors within the last 2–3 years. Figure 8.7 shows a snapshot of the RecognizeMe app on an iPhone.

8.5 Challenges and Issues with Biometric Sensors

Biometric sensors are widely used today, yet they suffer from several challenges. The accuracy of authentication based on biometric modalities is, to a significant extent, dependent on the functionality of the relevant sensors. Of all challenges they face, the three primary ones affecting sensor performance are (1) interoperability, (2) quality, and (3) cost. We discuss them in more detail in this section.

8.5.1 Interoperability of Biometric Sensors

Sensor interoperability refers to the ability of a biometric system to adapt to the raw data obtained from a variety of sensors [32]. Most biometric systems are designed to compare data originating from the same sensor. In some cases, the classifiers are trained on data obtained using a single sensor alone, thereby restricting their ability to act on data from other sensors. This limitation prevents the use of multiple sensors with different characteristics in a single biometric system.

The recently launched US-VISIT6 program, for example, obtains fingerprint (and face) information of certain travelers arriving in airports and seaports. An optical fingerprint sensor is currently being used during the enrollment phase to procure fingerprint images. However, it is not guaranteed that a similar type of sensor will be used at a later time when verifying the same individual. The cost of reenrolling individuals every time the sensor is changed will be tremendous and will, in fact, defeat the purpose of enrolling individuals at the port of entry in the first place. In cases such as these, the need for sensor interoperability is paramount and will significantly impact the usability of the system.

Martin et al. [7] make the following observation about the effect of the handset type (sensor) on the performance of a speaker recognition system:

> Microphone differences are one of the most serious problems facing speaker recognition, especially when dealing with the telephone, where Edison's old nonlinear carbon-button microphone still represents a significant fraction of all transducers.

Phillips et al. [23] state the following about the sensitivity of face verification algorithms to camera type:

> Many face verification applications make it mandatory to acquire images with the same camera. However, some applications, particularly those used in law enforcement, allow image acquisition with many camera types. This variation has the potential to affect algorithm performance as severely as changing illumination. But, unlike the effects of changing illumination, the effects on performance of using multiple camera types has not been quantified.

The International Biometric Group (IBG) recently conducted a battery of tests to evaluate the performance of various sensors (including fingerprint) under different test conditions. BIO-key International, Inc., demonstrated that its fingerprint system could enroll and verify fingerprint images obtained using different sensors. However, this kind of test scenarios is extremely rare as is borne out by the following statement by IBG [35]:

> Today, there is really very little interoperability among templates and the algorithms that the systems are using for matching. Those are proprietary technologies. So if you as an organization are considering deploying biometric technologies you should be concerned [whether] the vendor you are working with now will be around in 5 years and supporting the product. The cost to you of re-enrolling all your subjects could be significant.

This underscores the need for developing algorithms that are able to seamlessly operate on feature sets originating from different sensors. As a final note, the need for biometric sensor interoperability is pronounced due to the widespread deployment of biometric systems in various applications and the proliferation of vendors with proprietary algorithms that operate on a specific kind of sensor.

8.5.2 Quality of Biometric Sensors

It is well known that the quality of the image used for authentication purposes plays a vital role in the identification process for all biometrics, be it face or fingerprint or iris or fingerprint. Considerable research has been done and is also ongoing to investigate how the quality of an image affects identification accuracy.

Needless to say, the role of quality in an authentication protocol varies from biometric to biometric. Moreover, there are several factors that affect the quality of the captured image, the primary one being the quality of the sensor itself that captures the image. For instance, if the camera capturing a face or iris image is of poor resolution, that will produce low-quality images leading to inaccurate authentication results due to the difficulty of extracting all the important features of the specific biometric. Similarly, the quality of fingerprint sensor is crucial to sensing a high-quality image. With advent of technology, sensors of high quality, like those that are made of silicon, are being produced today by several companies, such as DigitalPersona [36]. Fingerprint sensors, especially those in public places like airports and other border control areas where millions of people are passing by on a daily basis, are likely to become unclean due to constant use and weather conditions. It is therefore imperative to keep them clean of dust particles, oil, and grease and prevent cracks and other damages because that directly affect their quality and hence that of the images they capture. Moreover, for biometrics-based authentication on the mobile platform, sometimes the built-in sensors like camera and microphone (for voice or speech recognition) are not of the greatest quality, thus affecting accuracies. However, latest mobile devices like most smartphones and tablets today come with very high-quality equipment that can obviate such concerns.

8.5.3 Cost

An important consideration in deploying a biometric system is its cost. The latter is estimated using both direct and indirect components that comprise the whole system [37]. The direct component cost is related to hardware, which includes the sensor along with processor, memory, and software modules for matching. It is thus essential for the sensor to be of low cost to keep the overall cost of the system low. Moreover, it should be easy to embed it in the existing security infrastructure of the application for ensuring smooth functioning. Moreover, there are additional costs in terms of system installation and maintenance. As discussed earlier, certain sensors that require human contact (e.g., fingerprint) get dirty very quickly that affects image quality, thus necessitating regular cleaning. However, the good news is that with the technology evolving at a superfast rate today, more and more sensors are being produced in the market, as a result of which the costs are going down at a very rapid pace making them more and more affordable. Besides, with biometrics-based authentication becoming increasingly more prevalent in consumer electronics and mobile devices, such biometric sensors are being embedded in the hardware of those devices also (as described previously). This makes biometric sensor and biometric authentication technology widely available to consumers at little or no extra cost today.

8.6 Future of Biometric Sensor Technology

Given such rapid advancement and wide deployment of biometric sensors and biometrics-based security control, a natural and imminent question is, "What is the future of biometrics and biometric sensors?" Indeed, the use of biometrics is expanding to several areas now, a major one being health care. The health-care industry in the United States is undergoing massive changes owing to recent health-care reforms and regulations coupled with technology advancement. Online tools, particularly with the advent of mobile apps, not only have the potential to seamlessly improve health-care delivery but also have day-to-day communications by connecting all stakeholders such as patients, providers, and insurers. Everywhere you look these days, there seems to be yet another sensor-enabled device or mobile app that wants to monitor everything from your heart rate and posture to your brain waves and breathing patterns. FDA-backed wrist monitors are beginning to help doctors track patients' blood pressure, heart rate, and other vital signs no matter where they might be. People are increasingly resorting to app-based fitness routines, diet, and exercise patterns in order to maintain healthy lifestyles. For example, consumer devices like Fitbits, Nike FuelBands, and Jawbone's UP are motivating people to pay more attention to their activity level and weight. Furthermore, patients suffering from some chronic diseases such as diabetes, high blood pressure, and cholesterol are adopting online and social media tools to connect with each other and share

experiences via support groups. Since people's personal health and wellness information is very sensitive, it is necessary to protect their privacy and confidentiality. Thus, biometrics has a tremendous role to play in not only enabling health monitoring but also ensuring the security of such transactions.

Various forms of biometric data technology are also gradually making inroads among athletes and in the entertainment industry [38]. For example, wearable ECG and Bluetooth sensors help NFL teams gauge players' reaction times, recovery speeds, and physical preparedness. Artists are on the way to combine biometric data with musical performance. A start-up company called BioBeats aims to develop apps and content that use biometric data to shape a media experience. Its experimental smartphone app, Pulse, generates music based on the activity of your heartbeat, and future plans include movies and music that adapt to your mood, as well as clinical tools that gauge a patient's stress level and then deliver appropriate sounds and messages. Additionally, University of Southern California's Center for Body Computing [39] is pioneering research on technology that literally puts biometric tracking technology into the driver's seat. With BMW, they are developing sensor-embedded steering wheels that could check a driver's heart rate, blood sugar, and other vital signs as soon as they turn the car on. Other researchers and carmakers, including Nissan and Ford, are using biometric data to detect drowsy or drunk drivers. All these demonstrate the vast potential of the technology of biometrics and biometric sensors—they clearly go beyond their primary functionality of ensuring security. Overall, this is indeed an exciting time to witness the growth of this field of research and development in new directions.

8.7 Summary

Biometric sensors are ubiquitous today. From face to fingerprints to iris, they provide the means to ensure security in our society from border control to law enforcement to access control. Following the attacks of 9/11, their supreme importance has led to explosive growth in technology producing state-of-the-art sophisticated authentication systems. Biometric sensor products are being developed and marketed at a greater pace than ever before driven by the need for greater accuracy and reliability. Particularly with the integration into consumer electronic products such as mobile phones and tablet computers, biometrics is now the most viable security measure in all walks of the society. This chapter has provided an in-depth presentation of what different types of biometric sensors exist in the world today, how they work, and what are some of the principal underlying issues. With the advancement of technology, such sensors are becoming more powerful, less costly, and hence more widely deployed in different applications. Moreover, there are several industries, from health care to auto to music, that are inventing novel ways to utilize biometric data for improving user experiences in innovative and creative ways. We thus hope that with more cutting edge research in this field in both academia and industry, the world will become more secure with better biometric technology in the next few decades.

References

1. Jain, A.K., Ross, A., Nandakumar, K. 2011. *Introduction to Biometrics*. Springer. New York.
2. Jain, A.K., Ross, A., Prabhakar, S. 2004. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, New York, 14: 4–20.
3. Rashid, F.Y. 2013. iPhone 5S: A biometrics turning point? *In Bank Info Security*, Published on September 16, 2013. http://www.bankinfosecurity.com/iphone-5s-biometrics-turning-point-a-6065/op-1. Accessed March 12, 2014.
4. Kosner, A.W. 2013. iPhone 5S touch ID fingerprint scanner is a fail for 20% of users, Here's what to do. *Forbes Magazine*, Published on October 15, 2013. http://www.forbes.com/sites/anthonykosner/2013/10/15/iphone-5s-touch-id-fingerprint-scanner-is-a-fail-for-20-of-users-heres-what-to-do/. Accessed March 12, 2014.

5. DigitalPersona. https://www.digitalpersona.com/. Accessed February 7, 2014.
6. Chen, A., Tian, J., Su, Q., Yang, X., Wang, F. 2005. A secured mobile phone based on embedded fingerprint recognition. In *Proceedings of Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks*, Towson, MD, pp. 254–259.
7. Martin, A., Przybocki, M., Doddington, G., Reynolds, D. 2000. The NIST speaker recognition evaluation—Overview, methodology, systems, results, perspectives. *Speech Communications*, 31: 225–254.
8. Joyce, R., Gupta, G. 1990. Identity authorization based on keystroke latencies. *Communications of the ACM*, 33(2): 168–176.
9. NSTC Palm Print Recognition. 2006. http://www.biometrics.gov/docs/palmprintrec.pdf. Accessed February 7, 2014.
10. Zhang, D. 2004. *Palmprint Authentication*. Kluwer Academic Publishers, Dordrecht, the Netherlands.
11. Chen, K., Bhanu, B. 2005. Contour matching for 3D ear recognition. In *Proceedings of the IEEE Workshop on Application of Computer Vision (WACV)*, Breckenridge, CO, pp. 123–128.
12. Jain, A.K., Kumar, A. 2010. Biometrics of next generation: An overview. In *Second Generation Biometrics*, E. Mordini and D. Tzovaras (Eds.). Springer.
13. Wikipedia page on "Capacitive Sensing." http://en.wikipedia.org/wiki/Capacitive_sensing.
14. Aware. http://www.aware.com/biometrics/preface.html. Accessed February 7, 2014.
15. Hjelmas, E., Low, B.K. 2001. Face detection: A survey. *Computer Vision and Image Understanding*, 83: 236–274.
16. Germain, J.M. 2004. IBM introducing fingerprint reader into laptop. *TechNewsWorld*, Published on October 4, 2004. http://www.technewsworld.com/story/37017.html. Accessed March 12, 2014.
17. Jain, A.K., Prabhakar, S., Pankanti, S. 2002. On the similarity of identical twin fingerprints. *Pattern Recognition Letters*, 35: 2653–2663.
18. Victor, B., Bowyer, K., Sarkar, S. 2002. An evaluation of face and ear biometrics. In *Proceedings of the 16th International Conference on Pattern Recognition (ICPR)*, Quebec City, Quebec, Canada, pp. 429–432.
19. Chang, K., Bowyer, K., Sarkar, S., Victor, B. 2003. Comparison and combination of ear and face images in appearance-based biometrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25: 1160–1165.
20. Yan, P., Bowyer, K. 2007. Biometric recognition using 3D ear shape. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 29(8): 1297–1308.
21. Ross, A., Jain, A.K. 2004. Biometric sensor interoperability: A case study in fingerprints. In *Proceedings of International ICCV Workshop on Biometric authentication (BioAW)*, Prague, Czech Republic, May 2004, Lecture Notes in Computer Science, 3087: 134–145.
22. Moore, S. 2004. Latest tests of biometric systems show wide range of abilities. *IEEE Spectrum Online*.
23. Phillips, P.J., Martin, A., Wilson, C.L., Przybocki, M. 2000. An introduction to evaluating biometric systems. *IEEE Computer*, 33: 56–63.
24. Phonearena.com. 2012. Apple buys biometric security company AuthenTec for $356 million, James Bond iPhone to follow, Published on July 27, 2012. http://www.phonearena.com/news/Apple-buys-biometric-security-company-AuthenTec-for-356-million-James-Bond-iPhone-to-follow_id32725. Accessed March 12, 2014.
25. Yörük, E., Konukoglu, E., Sankur, B., Darbon, J. 2006. Shape-based hand recognition. *IEEE Transactions on Image Processing*, 15: 1803–1815.
26. Jain, A.K., Ross, A., Pankanti, S. 1999. A prototype hand geometry-based verification system. In *Proceedings of the Second International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, March 22–23, 1999, Washington DC, USA, pp. 166–171.
27. Ross, A., Jain, A.K. 2004. Multimodal biometrics: An overview. In *Proceedings of 12th European Signal Processing Conference (EUSIPCO)*, Vienna, Austria, pp. 1221–1224.

28. Moayer, B., Fu, K.S. 1976. A tree system approach for fingerprint pattern recognition. *IEEE Transactions on Computers*, 25(3): 262–274.
29. Dave, G., Chao, X., Sriadibhatla, K. 2010. Face recognition in mobile phones. Department of Electrical Engineering, Stanford University, Stanford, CA.
30. Turk, M., Pentland, A. 1991. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3: 72–86.
31. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J. 1997. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19: 711–720.
32. Homeland Security News Wire. 2011. Biometrics for mobile phone market to grow 500 percent by 2015, July 5, 2011. http://www.homelandsecuritynewswire.com/biometrics-mobile-phone-market-grow-500-percent-2015. Accessed February 15, 2014.
33. Ross, A., Jain, A.K. 2003. Information fusion in biometrics. *Pattern Recognition Letters*, 24: 2115–2125.
34. Heussner, K.M. 2013. The future of biometric data tracking isn't about devices, it's about experiences, GIGAOM, Published on September 20, 2013. http://gigaom.com/2013/09/20/the-future-of-biometric-data-tracking-isnt-about-devices-its-about-experiences/. Accessed February 15, 2014.
35. Lammi, H.K. 2004. Ear biometrics. Technical Report, Lappeenranta University of Technology, Lappeenranta, Finland.
36. University of Southern California (USC)'s Center for Body Computing. http://www.uscbodycomputing.org/. Accessed February 7, 2014.
37. findBiometrics Website. http://findbiometrics.com/solutions/biometric-sensors-detectors/. Accessed February 7, 2014.
38. Federal Bureau of Investigation (FBI). 1984. The science of fingerprints: Classification and uses, US Government Printing Office, Washington, DC.
39. Bolle, R.M., Connell, J.H., Pankanti, S., Ratha, N.K., Senior, A.W. 2004. *Guide to Biometrics*. Springer, New York.

9

Radio Frequency Identification Device Readers and Tags

9.1 Introduction **9**-1
9.2 RFID Tags **9**-2
Active Tags • Semiactive Tags • Passive Tags • Classes of Tags • RFID Tag Cryptography
9.3 RFID Readers **9**-4
RFID Systems • RFID Reader Components
9.4 RFID Standards **9**-5
9.5 RFID Implementations **9**-6
9.6 RFID Concerns **9**-8
9.7 Summary **9**-9
References **9**-9

C.J. Wiemer
kCura Corporation

9.1 Introduction

Ever wonder how you are able to put a little plastic box on the dashboard of your car and have it pay your highway tolls for you? Or how about what sets off the alarms at stores when people try to steal merchandise? Or how from a distance you are able to unlock your car by pressing a tiny button? Figure 9.1 shows a type of radio frequency identification device (RFID) tag that makes all of these things and more possible.

RFID is a type of technology that allows for data to be stored on small devices that can then be read and written to wirelessly. This is accomplished by having a microchip placed in what is referred to as a tag, which is the device that stores the data. These tags can usually store up to 2 kB of data. The other component is the RFID reader (also known as the interrogator). This is the device that is able to connect to the tag and read (or write) the data that are stored on the tag's microchip. The reader then usually connects back to some sort of database. This is where it can use the data from the tag to look up further information about the item that the tag is placed on.

RFID tags are somewhat similar to those of bar codes (or UPC codes). There are a few differences however. For example, whereas bar codes are just a design on a sticker or piece of paper, RFID tags are actual microchips that require electricity. Also, because RFID leverages wireless technology, the tag does not have to be in the line of sight of the reader in order for it to be able to connect to the tag and read the data. It just has to be in the range of the reader. This depends on the type of the tag. Some tags though are known to be able to be read from 300 ft away! That's 300 ft compared to the bar code's mere couple of feet, if you have the bar code in a clear line of sight. A bar code's sole purpose is to be able to

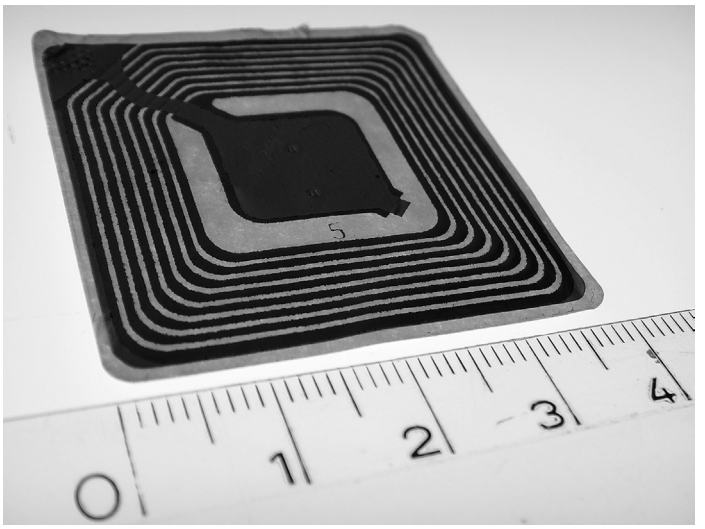


FIGURE 9.1 An RFID tag in the form of a sticker measured in inches.

read data, which are usually in relation to the item it has been placed on. RFID tags however can be written to as well. So, not only can you find out information about the item that the RFID tag has been placed on, but you can add to that data as well using the reader. So in reality, the term RFID reader is a little misleading since you can use it to write data as well.

As the name implies, RFID technology uses radio frequency. The way RFID and other wireless technologies are able to communicate with different devices is through electromagnetic waves. To be able to produce electromagnetic waves, you need a power source. One thing to keep in mind with all of the different tags and readers that are out there is which component is going to be doing most of the work. In other words, which component is going to act as the power source the most? Is the tag going to do almost all of the work? Will the tag and reader make a sort of compromise and both contribute to the power? Or will the reader do all of the work? The term radio frequency is just identifying the range of frequencies these electromagnetic waves fall into.

For example, there is the radio frequency that you are able to hear. Frequency is measured in a unit called hertz (Hz). The FM radio frequency that you hear when you turn on the radio uses frequencies ranging from 88 MHz (88,000,000 Hz) to 108 MHz (108,000,000 Hz). These frequencies fall into the very-high-frequency (VHF) designation. RFID signals tend to fall in the high-frequency (HF) designation, which is usually between 3 and 30 MHz, though they can go up to the extremely high-frequency range. Radio frequency can be anywhere from 3 kHz (3000 Hz) all the way up to 300 GHz. This is up to the manufacturer of the RFID components to decide at what frequency the devices should communicate at, taking into account the frequencies that other devices operate at.

So, how are these frequency signals being sent and received? Well, this is similar to the way your car is able to receive the audible radio that you listen to, by use of antennas. Both the tag and the reader are equipped with some sort of antenna to send and receive radio frequency signals.

9.2 RFID Tags

The greater the distance a tag can communicate with the reader, the more expensive it usually is. This is because it requires more power from the tag to be able to produce a strong enough signal to reach the reader. Knowing these aspects of an RFID tag, we can break it into its own categories: the cost of the tag, what kind of power supply it has, and the range that it can communicate with the reader. Table 9.1 shows this for us.

TABLE 9.1 RFID Tag Aspects

	Passive Tags	Semiactive Tags	Active Tags
Cost	Low	Medium	High
Power supply on tag	No	Yes	Yes
Range	Low	Medium	High

9.2.1 Active Tags

Remember earlier when it was mentioned that the tag could do most of the "work" instead of the reader? Tags that have their own power supply and do most of the work as far as communicating are what are called active tags. These are tags that have their own power source on the tag itself, which is usually a battery. Since it has its own power source, it is able to transmit farther and without assistance from the reader. This means that the tag's signal can be transmitted even when there is no reader in its presence. If you want the tag to go farther, you have to increase the power supply that the tag contains. Having an onboard power supply though is what increases the cost of these tags. They are usually the most reliable however.

9.2.2 Semiactive Tags

Now that we know that active tags are considered to be ones that contain a power supply on the tag itself, we can discuss semiactive tags. You might be thinking, "Well what do you mean by semiactive?" Either it does have an onboard power supply or it does not. Semiactive tags are tags that contain batteries on the tag; however, they still need help from the reader. This is where the tag and the reader are making a sort of compromise for work. The battery on the tag helps it power the microchip, but the reader is necessary for it to actually transmit the signal. Since its power supply is not that powerful and it still needs help from the reader, we can see that it is in between active and passive in Table 9.1.

9.2.3 Passive Tags

Lastly, when defining tags based on their power supply, we have the tags that do not have a power supply. These are passive tags. They require the reader to do all of the work. Since they do not have a power supply, they are unable to power themselves in order to initiate any type of communication. Passive tags can only work with the help of the reader and thus have to be in the reader's presence. They are receiving their power source from the reader, which is sending radio signals to the tag. As we see with the other tags, it is usually the battery making the device more expensive, which is why passive tags are generally the cheapest. But since they are unable to power themselves, they also have the shortest transmission range.

9.2.4 Classes of Tags

So far, we have defined tags by their power supply, or the lack thereof. Another way to define tags is by class.

Class 0 tags are tags that are read only. The tag is written to at the creation of the microchip that it contains. These tags contain nonvolatile memory, which means that power is not required to keep the memory. They are passive tags that do not necessarily notify a device of any data, but just the fact that the tag is there. It announces its presence to other devices. Think of these tags as the ones that are placed on merchandise in stores. They are the tags that notify the devices near the door that they have not been deactivated and are most likely attached to merchandise that has not been paid for yet.

Class 1 tags are the same as Class 0 tags, except that they operate at different protocols. This is why they are separated into two different classes. Class 0 tags, for instance, receive at a different frequency than which they transmit. Also, you are able to write once to Class 1 tags.

Class 2 tags are where we start to see tags that we are able to write to, as well as read from. These are tags where we can usually store around 65 kB of data. These tags are still passive, however.

TABLE 9.2 Classes of Tags

Class	Active or Passive	Memory
Class 0	Passive	Read only
Class 1	Passive	Read only, write once
Class 2	Passive	Read–write
Class 3	Semiactive	Read–write
Class 4	Active	Read–write
Class 5	Active	Read–write

Class 3 tags are very similar to Class 2 tags in that they are also able to store around 65 kB of data and that they have read–write memory. The difference though is that these tags are now semiactive. Now remember that semiactive means they now have a battery on the tag; however, it is not a very big battery. But this addition of an onboard power supply to the tag helps increase its transmission or read range.

Class 4 tags are where we start to see active tags. These are the tags that have a battery that is strong enough to help power the circuitry of the microchip on its own in order to be able to broadcast a signal from the antenna to the reader. These tags are rewritable as well, just like Class 2 and 3 tags.

Class 5 tags are just like Class 4 tags except that not only can they just communicate with the reader and other tags, they are able to communicate with other devices completely by containing other wireless components. These networking components can allow them to create their own ad hoc network. Table 9.2 shows us a summary of the different classes and their components.

9.2.5 RFID Tag Cryptography

Certain tags are able to perform cryptographic functions. Some tags are able to perform symmetric-key cryptography. Symmetric-key cryptography is when both of the involved parties share the same key, the key is used to both encrypt and decrypt. With this method, each tag has a different key. But remember the reader needs to have the same key in order to encrypt and decrypt the data. That means that for each tag that the reader intends to communicate with, it must have a key for each tag. That is the downside of this method because the reader has to contain many keys, one for each tag. The reader has to search through all of its stored keys to find the correct one for that particular tag.

Other tags are able to perform asymmetric-key cryptography, otherwise known as public-key cryptography. This involves a public key and a private key. The public key, as the name implies, is shared with the public. The private key, however, is only held by that individual party. The private key is used to encrypt, and the public key is used to decrypt. They can also be used vice versa. For example, if the tag wanted to communicate the data it had stored on its microchip securely to a reader, the reader would have to share its public key with the tag. Since it is the public key, it would gladly share this with anyone since it is public. Remember though that only the private key can decrypt data that have been encrypted with the public key. So the tag, now having the reader's public key, would use that to encrypt the data it has stored on its microchip. Now that the data have been encrypted with the reader's public key, the tag can send them back to the reader. That private key is the only key that is able to decrypt anything encrypted with the public key. Knowing that only the reader will be able to decrypt it since the reader is the only one that possesses the private key ensures that the data are secured and are unable to be read by any other parties besides the reader. Tags that are not able to perform any of these cryptographic functions are what are referred to as basic tags [1].

9.3 RFID Readers

Now that we have discussed what the tags in an RFID system consist of, let us talk about the interrogator or the RFID reader. This is the device that communicates with the tag and has the ability to both read and write to the tag (i.e., if the tag is readable–writeable). Since we have discussed the different types of tags, we can talk about the different RFID systems or the different combinations of the tags and readers.

9.3.1 RFID Systems

There is the passive reader active tag (PRAT) system. As we learned from the section about RFID tags, passive means that it is unable to send or receive signals on its own. It needs assistance from another device. In this case, the passive reader is only able to receive signals that are from an active tag. The reception range for this type of system depends on how powerful the active tag is, since in this system, it is the tag that is doing all of the work.

Also, we have the active reader passive tag (ARPT) system. This is the opposite of the PRAT system. In this system, the reader has to do all of the work since it is the active one and the tag is passive. Since the tag is passive, it is unable to transmit its signal without being in the presence of the reader.

Lastly, we have the active reader active tag (ARAT) system. In this system, both of the devices are active so they are able to transmit and receive on their own.

9.3.2 RFID Reader Components

RFID readers contain a component that is known as the control unit. This control unit is necessary to perform a couple of different functions. For starters, as the name implies, it controls the communication. The communication between the reader and the tag is controlled by this control unit. Another function that the control unit is in charge of is that if the reader is hooked up to a back-end database (as mentioned before that it sometimes is), it is what helps with the communication between the reader and the database. It sends commands to the database/server to be executed. The control unit is also what encodes and decodes the signal between the reader and the tag. Depending on the RFID system, it is also able to accomplish more complex tasks. For example, in how it was mentioned before that certain tags can support different cryptographies, the control unit on the reader is the one that would encrypt the data being sent from the reader or decrypt the data it received from a tag. It is also the one performing authentication between the reader and the tag.

The other main component of an RFID reader is its frequency interface. This is the unit that is mainly concerned with the actual signal. This is the component that actually generates the power to communicate with the tag. For example, if this was an ARPT system, the frequency interface is what is sending the signal in order to activate that passive tag that cannot do it on its own. Also, it is in charge of demodulating the signals that it receives from the tag. With that, it also modulates the signal that it sends to the tag.

9.4 RFID Standards

As mentioned before, there are different protocols that tags can use. These protocols differ between countries as well, since not all countries have the same standards for different radio frequencies. Some of these radio bands are not even available in some countries. There are some organizations out there that are trying to set standards however. One of the main organizations in charge of this is the International Organization for Standardization (ISO).

The ISO has some standards already out there for radio frequency technology. One of these that is more specific to RFID is ISO 1800 that is a standard used for item management. There are several parts in this standard that break down the standard for different frequencies since, as mentioned earlier, tags and readers can communicate on different frequencies that are usually dependent on their manufacturer. Since RFID can use different protocols which involve using different protocols, a lot of the standards are focused on a particular frequency rather than just RFID in general. They also focus more on just the general term of smart cards. There are specific standards for item management specifically like the ISO 1800 mentioned as well as ISO 15961, 15962, and one 15963 [1]. There are also specific standards for using RFID to identify animals. These standards include, but are not limited to, ISO 11784, 11785, and 14223.

Besides ISO, there is the Electronic Product Code (EPC) standard. Think of EPCs, sort of like the UPCs (or bar codes) that were mentioned earlier. Except since EPCs are wireless and submitting signals,

there is no need for a scanner to be up close. Also, line of sight is not required, unlike UPC where the scanner has to be able to see and read the code. Since tags are able to submit their information wireless and from further distances, it makes it easier to handle item management. You can place a reader in one central location and have it be able to read all of the tags nearby, rather than having to take a handheld UPC scanner and go up to every single item to where you can see the bar code. Also, since it is wireless, you could place the RFID tags within the item itself or just in its container. All that is required is that the item be somewhat near the reader, it does not have to be up close, and the tag does not have to be visible to the reader since it is wireless and able to go through most common materials. Using EPCs is how, as I mentioned at the beginning of this chapter, you are able to pay your highway toll just by having a small plastic box in or on your car. The bridge that you go under has readers hanging from it, which are able to read the EPC off of your device.

This EPC standard was first created by a group at MIT. The EPC system currently however is operated by the EPCglobal. There is a specific format of an EPC tag that the EPCglobal standard uses. The tag's data are split into four different parts. First, there is the header that identifies the type, structure version, length, and generation of the EPC and takes up 8 bits. Following that is the EPC manager number that maintains the last partitions and is 28 bits long. After that is the object class that, as the name implies, identifies a certain class of objects and is 24 bits. Last is the serial number that takes up the most space being 36 bits long and identifies the instance. EPCglobal assigns out the first two parts: the header and the EPC manager number. The last half of the format, the object class and serial number, is assigned out by the EPC manager owner. This format of the EPC that I have just described is the 96 bit version; however, there are 64 bit types as well with each part varying in length [2].

9.5 RFID Implementations

As mentioned with some of the standards, there are several different real-world implementations of RFID systems including inventory management, animal tracking, personal identification, access control, and assembly line management.

RFID systems can be utilized in inventory management as we briefly mentioned with the EPC standard. RFID is an easy way to keep track of products or items by attaching tags to them to identify the item. Writing some simple uniquely identifying information onto the memory of the RFID tag is a simple way to keep track of items. All you have to do then is place some readers within the range of the inventory. This way the readers can quickly scan an entire warehouse of items and give you a list of all the inventory.

In relation to inventory management, this is what a lot of retail stores use to make sure no unpaid merchandise is leaving their store. Figure 9.2 shows an example of this. Readers are placed at the exit and the merchandise contains readers on them. That way if a tag is marked as unpaid, the readers will be able to detect this as the customer leaves, notifying the employees of the store that they are leaving the store with unpaid merchandise.

Animal tracking is another use of RFID systems where the tags are placed on the animals themselves. Then you can place a reader around entrances/exits of fences, barns, or cages to know when an animal has entered or exited a certain location.

Personal identification has started to become a possibility with RFID systems with the implementation of RFID tags into passports. The tag contains all of the information that the passport does. The airport employees then just place the passport near a reader and have all of the information about how you appear on a screen right in front of them. Some people have even gone as far as to implant themselves with RFID tags as shown in Figure 9.3.

In relation to personal identification, RFID systems can be used for controlling access. A person or entity has a tag with information about them or it placed on it. The reader is then coupled with a door or some means of access to something. This reader is then tied back to a database. The database contains a

FIGURE 9.2 An example of a retail store using RFID to keep track of unpaid merchandise.

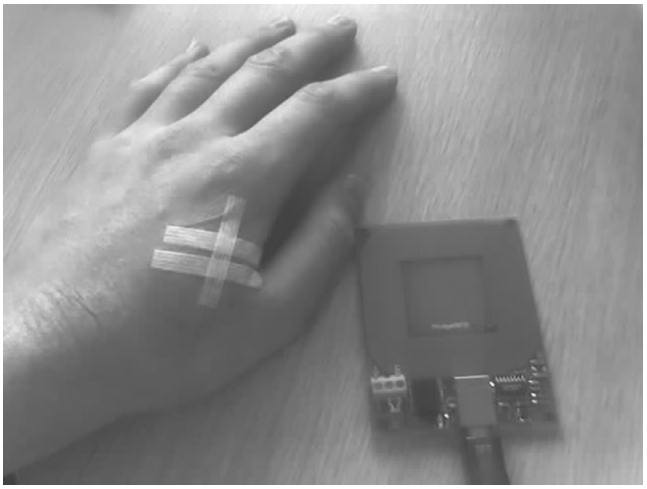


FIGURE 9.3 An RFID tag implanted into a person's hand next to an RFID reader.

list of who/what should be able to access that certain door. The person would then scan their tag at the reader; the reader then checks the database to see if they are on the list of people who should have access to this door, and if they are, it then allows the door to open. This is a way to use RFID systems for access control.

Another way RFID systems can be used is to keep track of products while they are being manufactured. You can place a tag on one of the first parts of the product. You would then place readers at various points of its construction or at various points in the assembly line. Then as the product is being assembled, it communicates with all of those readers as it makes its way down the line. You are then able to see, judging by which reader is reporting it has communicated with the tag, how far down the assembly line the product is. This helps you judge how close it is to being finished [3].

9.6 RFID Concerns

As with any form of communication, wireless or otherwise, there are security and privacy concerns as depicted by the logo in Figure 9.4. How does one measure how secure an RFID system really is? How do you make sure that only the proper reader is receiving the transmission from the tag? In other words, how can you ensure confidentiality? Just like other wireless systems, there are several different threats that one should be concerned with when implementing an RFID system.

An example of one of these threats is a denial-of-service (DoS) attack. A DoS attack is one that interrupts the service to the system's users. DoS attacks are threatening because they are usually fairly hard to protect against. It can do this through several different means of interruption:

- Tags referred to as blocker tags are able to simulate many tags all at once. This would overload the reader with too many signals to respond to, thus interrupting normal service to real tags.
- It can overload the system by sending one radio frequency signal that is more powerful than that of the tag. This now interrupts the reader from reading the data that it is actually after.
- Certain tags are programmed with what is referred to as a kill command. This command, when sent to the tag, renders the tag useless. If the attacker were to find out a tag's kill command, it could program a reader to constantly send out this command to tags, thus disrupting the system by requiring that each tag of those tags be replaced [2].

Eavesdropping is a concern with RFID. Eavesdropping occurs when someone reads the RFID tag's signals as it is transmitting to a reader. It is eavesdropping on a radio frequency conversation between the tag and a reader.

Skimming is another concern. Skimming refers to using a reader and scanning a tag without the tag owner's knowledge. This is a big concern when RFID tags that are implemented into passports. If you have your passport or something else containing an RFID tag in your pocket, someone can walk by with a handheld reader and scan your passport.

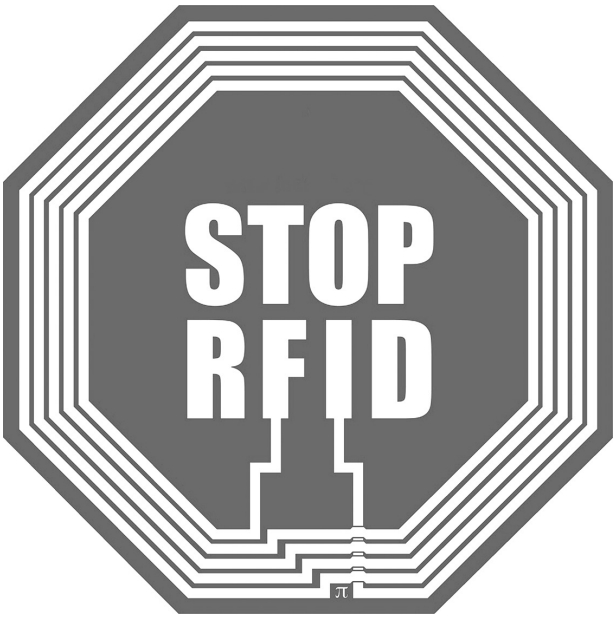


FIGURE 9.4 Petition logo to stop using RFID.

Spoofing is another threat within wireless communication in general. It is when the attacker is able to pose as an authorized user. The attacker can program a tag to be authorized by finding out an authorized tag's information, which could be done from eavesdropping or skimming as mentioned earlier.

With all forms of wireless communication, these threats and others should be taken into great consideration before implementing the system.

9.7 Summary

RFID technology involves the wireless communication between a tag and a reader. The tags and readers are able to accomplish all of this by using wireless technology, specifically that of radio frequency. They submit signals from their antennas on their respective devices while information is read and/or written to on the microchip of the tag.

A tag is usually providing information about whatever entity it is placed on. A tag can be either passive where it does not contain a battery, semiactive where it contains a small battery, or active where it contains a powerful battery. Tags are broken into different classes to define their capabilities. Readers are the devices that communicate with the tags to read and/or write to the tag.

RFID is all around us these days. It is an easy way to identify something, whether that be merchandise, inventory, a person or animal, or a way to track something, like a product as it is being assembled and can be used as a means of access control.

References

1. Rong, C., Cayirci, E., Zhao, G., and Yan, L. 2010. Radio frequency identification security. In J.R. Vacca (ed.), *Network and System Security*, Elsevier Inc., Burlington, USA, pp. 333–360.
2. Chaduhry, N., Thompson, D.R., and Thompson, C.W. 2005. RFID technical tutorial and threat modeling. Paper created by the Department of Computer Science and Computer Engineering, University of Arkansas, Fayetteville, AR.
3. 2014. Frequently asked questions. *RFID Journal*. http://www.rfidjournal.com/faq/.

IV

Sensor Networking Signal Processing

10 **Multimedia/Audio Sensor Networking Signal Processing** *Juan R. Aguilar*..... **10**-1
Introduction • Microphone Arrays • Signal Processing for Microphone Arrays: Microphone Array Beamforming • Summary • References

11 **Data Mining in Sensor Networks** *Sinjini Mitra and Pramod Pandya* **11**-1
Introduction • Data Mining: An Overview • Data Mining in Sensor Networks • Time Series Sensor Streams • Social Sensing • Distributed Algorithms • Nonlinear Regression Using Choquet Integral • Challenges in Sensor Data Mining • Summary • References

10

Multimedia/Audio Sensor Networking Signal Processing

10.1 Introduction **10**-1
10.2 Microphone Arrays **10**-2
10.3 Signal Processing for Microphone Arrays: Microphone Array Beamforming **10**-3
Acoustic Source Localization • Blind Source Separation
10.4 Summary **10**-11
References **10**-11

Juan R. Aguilar
Laboratorio de Electro Acústica & Procesamiento de Señal E.I.R.L.

10.1 Introduction

The microphone was invented in 1876 by Emile Berliner for the Bell telephone transmitter, and its transduction principle was the loss of contact between a plate and a pin made of metal or carbon. Afterwards, a number of other technological approaches for the transduction process were introduced including the moving coil microphone, the condenser microphone, the ribbon microphone, the prepolarized condenser or electret microphone, the radio frequency condenser microphone, and recently the microelectromechanical microphone. While the transduction principles mainly determine the sensitivity, the frequency response, and the self-noise of the transducer, its acoustical operating principle determines the directional characteristic of the microphone. Understanding the way in which the acoustical operating principle will affect the polar pattern of the transducer requires a review to acoustic wave propagation.

A number of alterations are introduced in the media as the waves propagate across it. Most representatives in a transducer operation, are the variations in the air pressure by the above and below atmospheric pressure; and, the particle velocity associated to the relative back-and-forth movement of the air particles, imposed by the propagating compressions and rarefactions. In this way, the transducers are designed to be sensitive to either one of these two parameters or to a given combination of them. As a result, pressure-sensitive transducers are inherently omnidirectional since the acoustic pressure is a scalar quantity that only has magnitude but no directional information. By contrast, transducer sensitive to particle velocity has directional pattern because it contains directional information. The combination of these two operating principles gives origin to several directivity characteristic microphone pickup patterns. Table 10.1 resumes microphone classification according to the acoustical principle and its associated polar responses.

TABLE 10.1 General Classification of Microphones according to Acoustic Operating Principle

Acoustical Principle		Directional Characteristic	
Pressure	Order 0	Omnidirectional	$R(\theta) = 1$
Particle velocity	First order	Bidirectional	$R(\theta) = \cos(\theta)$
Combination of pressure and pressure gradient		Subcardioid	$R(\theta) = 0.75 + 0.25\cos(\theta)$
		Cardioid	$R(\theta) = 0.5 + 0.5\cos(\theta)$
		Supercardioid	$R(\theta) = 0.37 + 0.63\cos(\theta)$
		Hypercardioid	$R(\theta) = 0.25 + 0.75\cos(\theta)$
Wave device	High order	Superdirectional	$R(\theta) = \sin(Nk\Delta r)/N\sin(k\Delta r)$
Boundary layer	Order 0	Hemispherical	$R(\theta) = 1$

In addition, boundary layer microphone corresponds to a pressure transducer that is used at a reflecting surface and consequently has hemispherical pickup pattern. High-order directional microphones employ a wave device such as interference tubes, parabolic reflectors, or acoustic lenses to obtain significant increases in the directivity of the microphone [1]. There are further improvements if the directivity of the microphone can be obtained using an assembly or array of microphones.

10.2 Microphone Arrays

Microphone array can be thought as a concentrated arrangement of acoustic sensor that is designed to operate together in order to obtain some specific functionality that cannot be achieved using individual transducer only. In this sense, there are a variety of functions that can be performed by microphone arrays; these include the control of directivity pattern or beamforming, sound source localization, and blind source separation (BSS). Other applications such as speech enhancement and noise cancellation are achieved by combining the previously mentioned techniques.

Linear array is perhaps the most commonly used because it has relatively simple formulation. The directivity pattern of a linear microphone array of N equidistant element is given by

$$R(\theta) = \frac{\sin(Nk\Delta r)}{\sin(k\Delta r)} \tag{10.1}$$

where

k is the wavenumber

$\Delta r = d\sin(\theta)$ [2]

Figure 10.1 shows the plot of Equation 10.1 given $N = 20$ and $d = 0.1$ m, $f = 680$ Hz. Notice the directivity pattern is symmetric with respect to the axis of the array along ±90°.

When controlling the directivity pattern of a microphone array, intersensor spacing d also determines the upper frequency limit of the array. Restriction of d being less than a half wavelength of the highest frequency is imposed to avoid spatial aliasing. Spatial aliasing corresponds to duplicates of the main lobe, also known as grating lobes, which appear in the directional characteristic of the array, as shown in (10.2). When the main lobe of the array directivity is on the axis of the array, the array is also termed end-fire array.

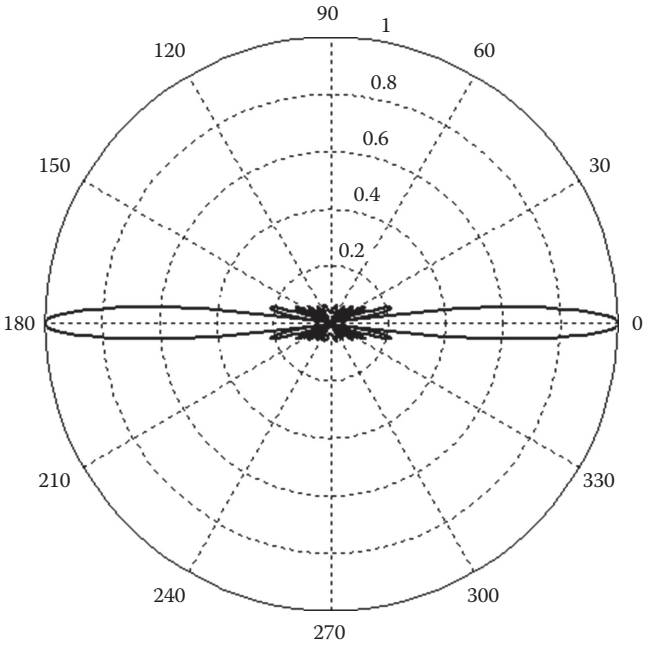


FIGURE 10.1 Directivity pattern of a linear array of 20 pressure microphones ($d = 0.1$ m, $f = 680$ Hz).

10.3 Signal Processing for Microphone Arrays: Microphone Array Beamforming

Beamforming constitutes a family of array signal processing, which are designed to control in a desired way the directivity pattern of an array of sensors using electronic software means. The technique is widely used in antenna design, radio astronomy, geolocalization, sonar and radar, and multimedia. Beamforming consists in to steer the directivity pattern of an array of microphones so that their main directivity lobe points out at a desired direction, normally over a sound source of particular interest. Beam steering also called spatial filtering is produced by applying convenient amounts of delay to the signals coming from the microphones. The objective is to compensate the natural delays occurred when a wavefront impacts the array from a given angle, thus making the signals from the sensors become time aligned. The effect of steering the directivity pattern of the array into 45° is shown in Figure 10.3; in this case, the array is the same as that used in Figure 10.2.

When the microphone signals are added together, this time synchronization introduces two main facts: an increase in the amplitude of the coherent signal and a decrease of the amplitude of the background uncorrelated noise. Consequently, beamforming leads to increased signal-to-noise ratio of the incoming sound waves. Performance of the beamformer is usually expressed as the array gain G, which quantifies the increase in the signal-to-noise ratio of the beamformer to a single sensor and is given by

$$G = \frac{SNR_{array}}{SNR_{sensor}} \tag{10.2}$$

Delay-and-sum beamforming can achieve a gain of only 3 dB per doubling of the number of sensors.

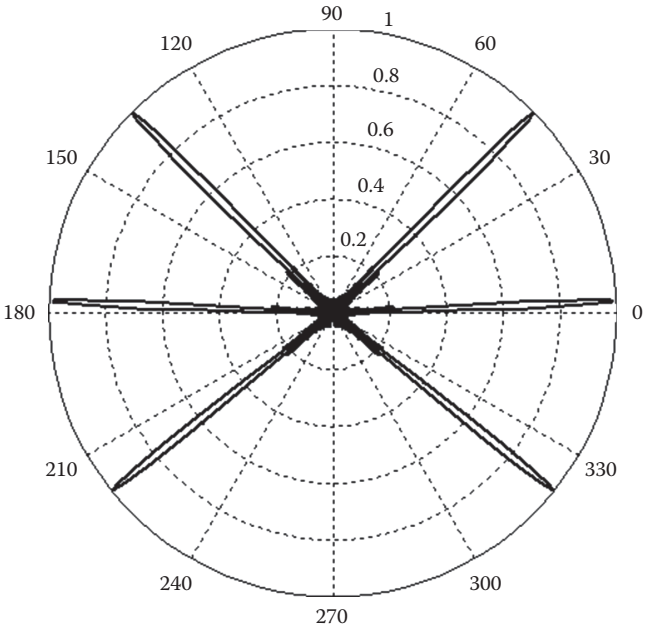


FIGURE 10.2 Directivity pattern of a linear array of 20 pressure microphones showing spatial aliasing or grating lobes ($d = 0.1$ m, $f = 5160$ Hz).

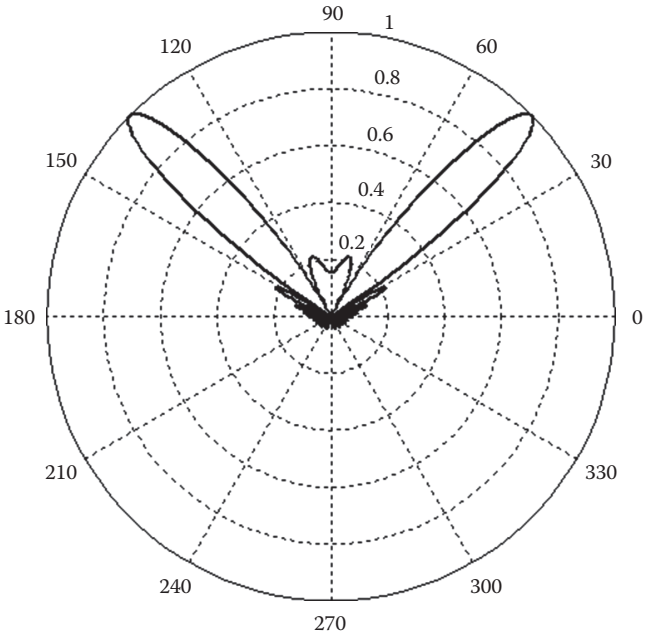


FIGURE 10.3 Directivity pattern of a linear array of 20 pressure microphones with the main lobe steered to 45° ($d = 0.1$ m, $f = 680$ Hz).

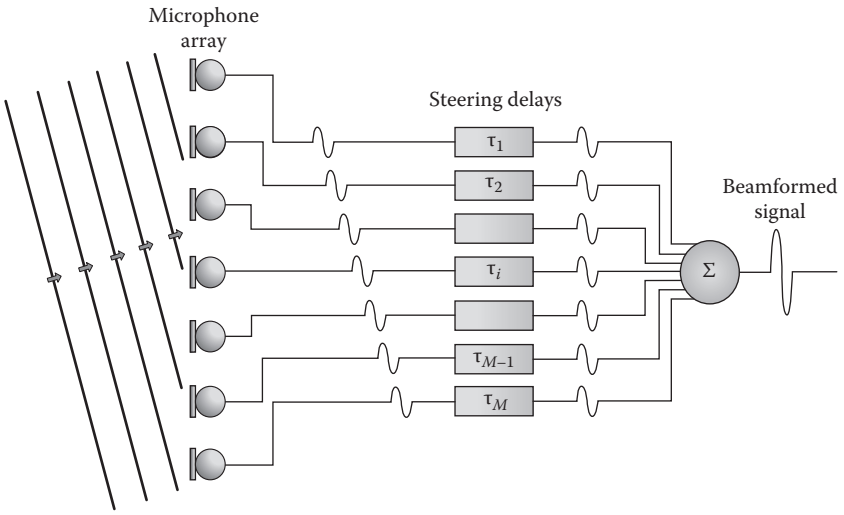


FIGURE 10.4 Schematic diagram of a delay-and-sum beamformer.

Schematic diagram of Figure 10.4 depicts a delay-and-sum beamformer, the signals that arrive to the microphones have a delay due to their inclination with respect to the axis of the array, the beamformer compensates these delays to align in the time domain the array signals and then sums the signals to obtain the maximum output signal. The output of a delay-and-sum beamformer is given in the following equation. Here, x_i is the microphone signal, t_i is the steering delays, and w_i is the beamformer weightings:

$$y = \sum_{i=1}^{m} w_i x_i(t - \tau_i) \tag{10.3}$$

Delay-and-sum beamformer constitutes the simplest approach to beamforming and has shown to be suitable for narrowband sources.

Finite impulse response (FIR) filters have been introduced as weighting functions for each channel in the array during the summation process, as described in the block diagram of Figure 10.5. Variations on the FIR weights can be introduced in order to meet some specific performance criteria, such as constant bandwidth or increased directivity. In this way, the increased signal-to-noise ratio is then dependent on the number of microphones in the array and their respective weightings.

Beamformer weights can be further updated in an adaptive way to minimize the power from localized acoustic sources. This is made based on assumptions that signals of interest are semistationary, while the noise is a nonstationary process across all sensors of the array. Beamformer weight adaptation is performed in a least square sense. Examples of adaptive beamformers are the Frost beamformer [3] and the Howells–Applebaum adaptive loop [4].

Superdirective beamformers constitute another approach to beamforming, which can improve the directional selectivity by optimizing gain *G*. Among these approaches, we can found the generalized sidelobe cancelling (GSC) and the minimum variance distortionless response (MVDR) beamformers [5].

10.3.1 Acoustic Source Localization

Acoustic source localization is the process of estimating the spatial coordinates of the location sound source by analyzing the acoustical emission that it produces. In human beings, this is performed using

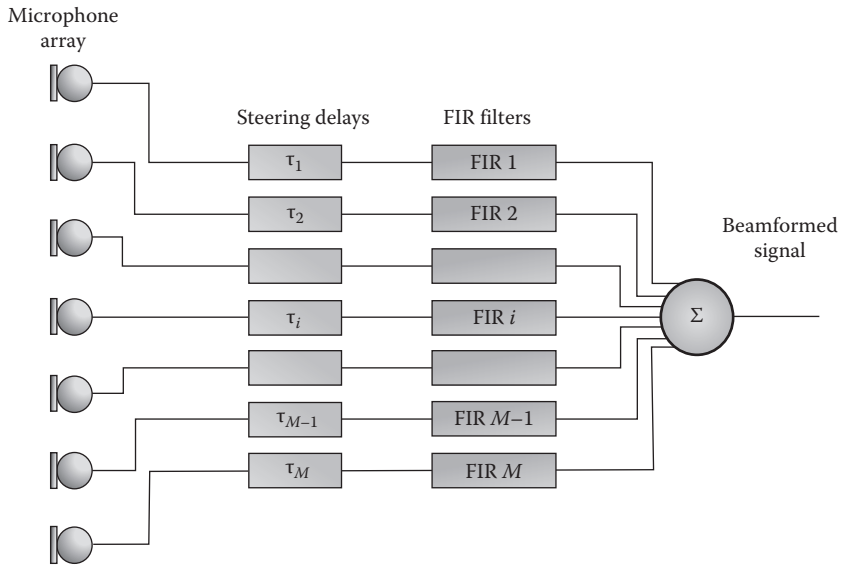


FIGURE 10.5 Schematic diagram of a delay-and-sum beamformer with FIR filters.

our binaural hearing. By measuring interaural time differences and spectral cues, our ear–brain system can detect the direction of arrival (DOA) of a wavefront with a resolution of about 2°. First technological approaches to acoustics source localization were introduced during World War I to detect the location of airplanes and Zeppelins but were replaced by the radar during World War II. These early systems used arrays of large horns, like the T3 sound locator of 1927, connected to the ears of the operator to increase directivity and sensitivity to interaural time differences. The increase on angular resolution provided by the arrangement of separate horns connected to the ear was first observed by Alfred M. Mayer who invented the topophone in 1880 [6]. The most advanced device of the war period was the German Ringtrichterrichtungshoerer that used an arrangement of four horns and could detect targets up to 12 km with angular accuracy of about 2°.

The advent of electroacoustic technology leads to the use of microphone arrays and signal processing techniques for acoustic source localization. These developments have found significant application in the field of multimedia, particularly in talker tracking for automatic videoconferencing. In these applications, microphones are used to estimate the DOA of the acoustic wavefronts to the array. Further implementation of appropriate triangulation algorithm is required to estimate the location of the talker. Estimations of DOA are usually performed by measuring the differences in the time of arrival TDOA of the acoustic wavefronts to different microphones in the array. These are obtained computing the cross correlation between two microphone signals:

$$\mathrm{TDOA} = \arg\max\left(R_{ij}(t)\right) \tag{10.4}$$

TDOA can also be computed in the frequency domain using the generalized cross correlation (GCC) between the two microphone signals. GCC or cross power spectrum can be computed using

$$\mathrm{GCC} = \mathrm{FFT}^{-1}\left\{H_i(\omega)H_j(\omega)\right\} \tag{10.5}$$

in which H_i and H_j are the complex Fourier transforms of the two microphone signals [7]. The DOA is then calculated from

$$\theta_{\text{arrival}} = \sin^{-1}\left\{\frac{c \cdot \text{TDOA}}{d}\right\} \tag{10.6}$$

where

d corresponds to the microphone separation distance

c is the speed of sound

A single τ_{delay} produces a hyperbola pointing to the estimated DOA, as shown in Figure 10.6. However, an ambiguity is introduced due to parity of inverse cosine function around the array axis. To resolve such ambiguity, the hyperbolic multilateration technique is used, sketched in Figure 10.7. This algorithm introduces a third sensor to conform a second pair of microphones and a two-hyperbola intersection solution, which is given by the following system of equations:

$$\begin{aligned} R_i - R_j &= \sqrt{(x - x_i)^2 + (y - y_i)^2} - \sqrt{(x - x_j)^2 + (y - y_j)^2} = c \cdot \text{TDOA}_{ij} \\ R_j - R_k &= \sqrt{(x - x_j)^2 + (y - y_j)^2} - \sqrt{(x - x_k)^2 + (y - y_k)^2} = c \cdot \text{TDOA}_{ik} \end{aligned} \tag{10.7}$$

where R_i, R_j, and R_k denote the distances from the corresponding microphone to the source.

Choosing an origin for the system at the microphone i location, the locations for microphones j and k become relative to i that simplifies the problem.

Other approaches for acoustic source localization include the steered beamformer methods and the methods based on high-resolution spectral estimation. In the steered beamformer approaches,

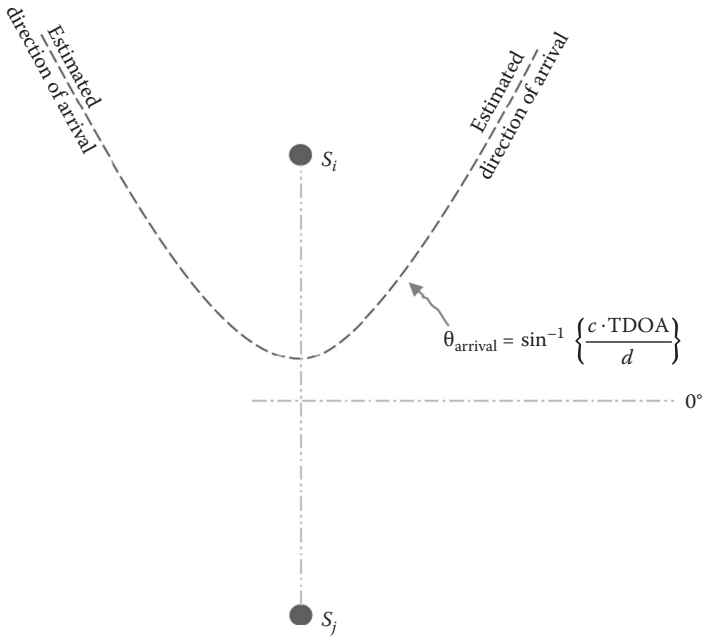


FIGURE 10.6 When estimating the DOA with a single pair of sensors, the analysis leads to a hyperbola pointing out to the source location.

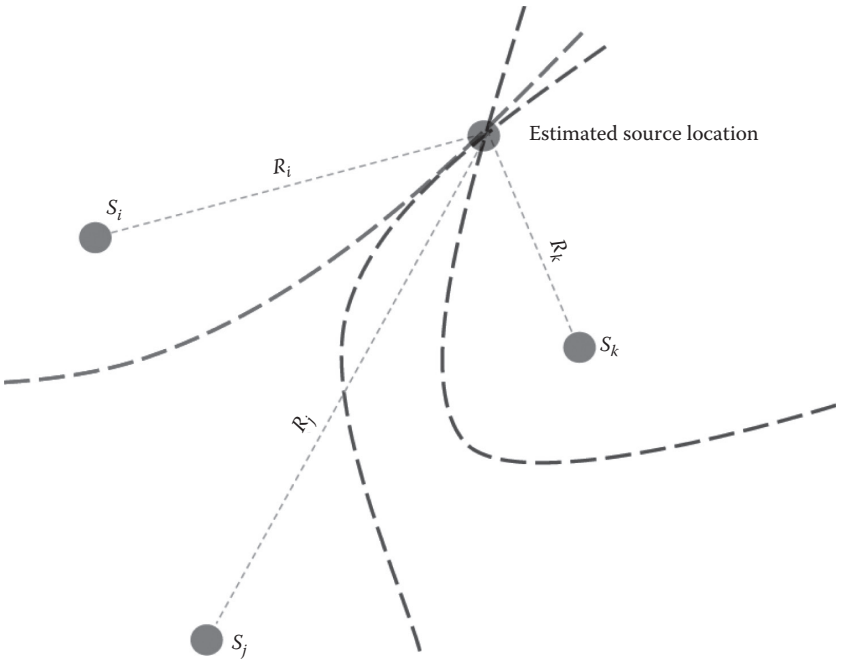


FIGURE 10.7 Explanation of the multilateration algorithm using three pairs of microphones.

a delay-and-sum beamforming is used together with a maximum likelihood estimator algorithm that steers the directivity pattern of the array to various locations and searches for the maximum in the output power [5]. Furthermore, acoustic source locators based on high-resolution spectral estimation methods are designed for narrowband signals and considers those beamformers based on eigenvalue analysis, autoregressive analysis, or minimum variance spectral estimation. Examples of this category are the modern eigenspace beamformer and the MUSIC algorithm [5].

10.3.2 Blind Source Separation

BSS is a statistical signal processing technique, which is based on the independent component analysis (ICA) method. This permit to identify latent data structures in a set of observations made to a process of particular interest. Statistical models of ICA attempt to describe without a priori information how the observations $x = (x_1, x_2, \ldots, x_m)$ are generated by latent data structures $s = (s_1, s_2, \ldots, s_n)$. In the context of microphone arrays' signal processing for multimedia applications, the observations are obtained by the microphones in the array, and the data structures represent the superposition process of multiple acoustic sources at the microphone's locations.

Data structures are present in the observations as an unknown linear combination of a set of also unknown components. The linear combination that mixed the signals is described by means of a matrix A of dimensions $m \times n$:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{10.8}$$

The BSS technique assumes the components s_j are non-Gaussian and statistically independent between them. In this way, BSS becomes the problem of estimating matrix A. Latent data structures $s = (s_1, s_2, \ldots, s_n)$ can be further estimated from the inverse of A:

$$\hat{\mathbf{s}} = \mathbf{A}^{-1}\mathbf{x} \tag{10.9}$$

Independent components are estimated through an iterative process in which a linear combination of x_i is described as

$$\hat{\mathbf{s}} = \mathbf{w}^{\mathrm{T}}\mathbf{x} = \sum_{\mathrm{i}} \mathbf{w}_{\mathrm{i}}\mathbf{x}_{\mathrm{i}} \tag{10.10}$$

where w is the random vector to be determined. Applying the following change of the variable $z = A^T w$ yields

$$\hat{\mathbf{s}} = \mathbf{w}^{\mathrm{T}}\mathbf{x} = \mathbf{w}^{\mathrm{T}}\mathbf{A}\mathbf{s} = \mathbf{z}^{\mathrm{T}}\mathbf{s} \tag{10.11}$$

which is in turn a linear combination of $\hat{\mathbf{s}}_i$, weighted by z_i.

The property of the central limit theorem establishes that the distribution of a sum of independent random variables tends, under some conditions, to be a Gaussian distribution. This means the sum of independent non-Gaussian random variables is closer to be Gaussian than any of the random variables, that is, $z^T s$ is more Gaussian than any s_i [8]. Regarding w as the vector that maximizes non-Gaussianity of $w^T x$, results are evident that w corresponds to vector z.

Maximizing non-Gaussianity of $w^T As = z^T s$, the independent components are obtained, so measures of non-Gaussianity are required. In this concern, fourth-order statistical moment kurtosis could be employed. Kurtosis is zero for Gaussian random variables, but for the most of non-Gaussian random variables, it is not zero [8]. Other measures of non-Gaussianity are the negentropy and their approximations.

As an example, the plot in Figure 10.8 shows the time series corresponding to three different mixes of three original audio signals: a dog bark, a telephone ring, and a woman scream. Assume the signals have been obtained at the microphones of a three-element array. After processing with ICA algorithm the observations (mixes), the original signals are retrieved and separated, as shown in Figure 10.9. Furthermore, Figure 10.10 shows the ICA estimation error for each component.

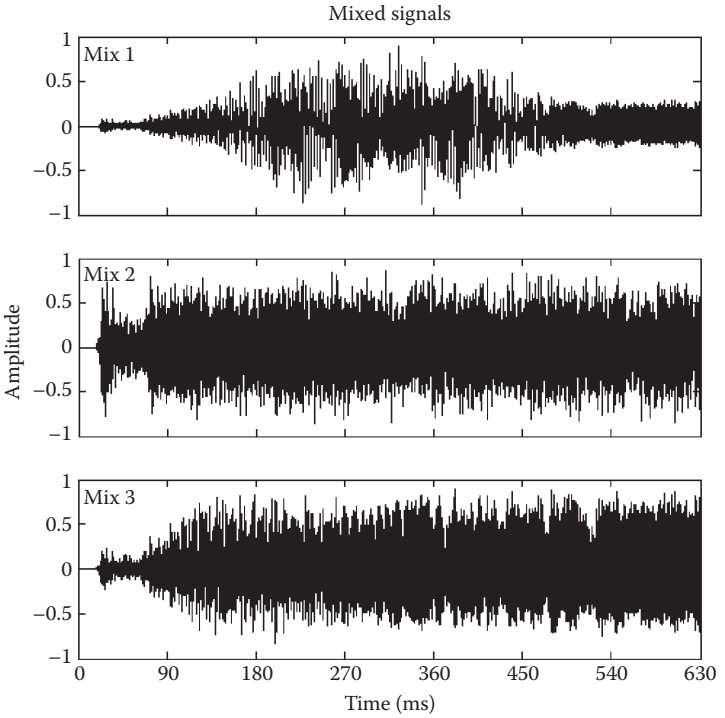


FIGURE 10.8 Time series that are linear combinations of three unknown sources.

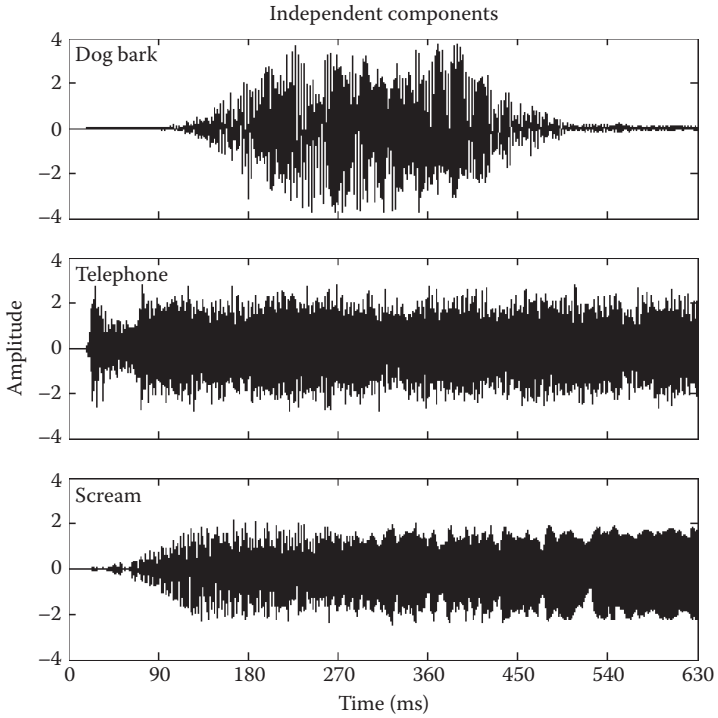


FIGURE 10.9 The three independent components estimated using ICA algorithm.

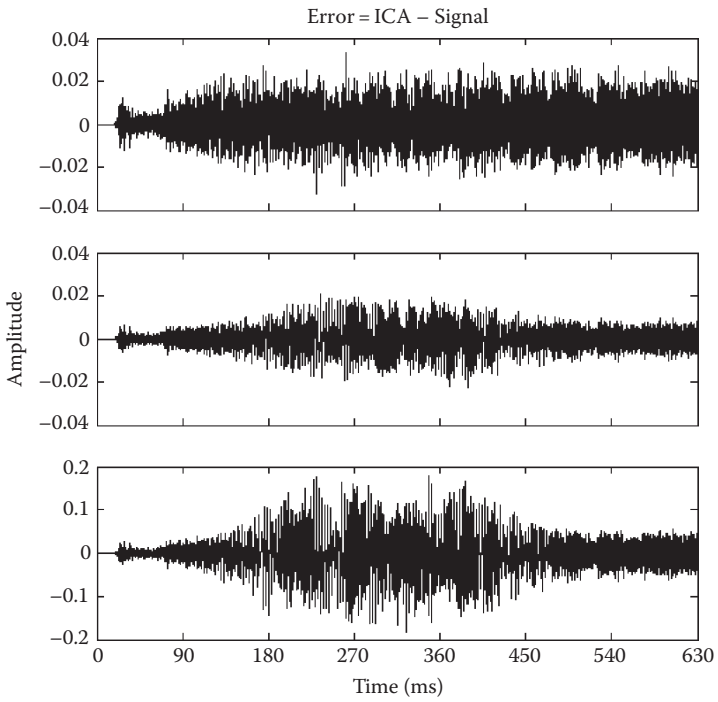


FIGURE 10.10 Independent component estimation error for each signal.

BSS analysis has found application also in hearing research for the development of cocktail party effect models and processors. Cocktail party effect describes the psychoacoustic phenomenon of selective attention or selective listening and has relevance in speech enhancement applications.

10.4 Summary

This chapter looked at the number of alterations that are introduced in the media, as the waves propagate across it. Most representatives in a transducer operation, are the variations in the air pressure by the atmospheric pressure above and below. This also includes the particle velocity associated to the relative back-and-forth movement of the air particles imposed by the propagating compressions and rarefactions. This way, the transducers are designed to be sensitive either to one of these two parameters or to a given combination of them. As a result, pressure-sensitive transducers are inherently omnidirectional, since the acoustic pressure is a scalar quantity, which only has the magnitude, but with no directional information. By contrast, transducers that are sensitive to particle velocity have a directional pattern, because they contain directional information. The combination of these two operating principles gives origin to several directivity characteristics of microphone pickup patterns.

References

1. H. Olson. 1967. Directional microphones. *J. Audio Eng. Soc*. 15:420–430.
2. L. Kinsler, A. Frey, A. Coppens, and J. Sanders. 1982. *Fundamentals of Acoustics*, Third Edition. John Wiley & Sons, Inc., p. 176.
3. O. Frost. 1972. An algorithm for linearly constrained adaptive array processing. *Proc. IEEE* 60:8.
4. B. van Veen et al. 1999. *Beamforming Techniques for Spatial Filtering*, Chapter 61 in *The Digital Signal Processing Handbook*, V. Madisetti and D. Williams Editors, CRC Press, pp. 61–17.
5. J. DiBiase, H. Silverman and M. Brandstein. 2001. *Robust Localization in Reverberant Rooms*, Chapter 8 in *Microphone Array: Signal Processing Techniques and Applications*. M. Brandstein and D. Ward Editors. Springer-Verlag, pp. 157–178.
6. A. Mayer, *Topophone*, U.S. Patent No. 224199A, February 3, 1880.
7. S. Stearns. 2003. *Digital Signal Processing with Examples in MATLAB,* CRC Press, pp. 190–193.
8. A. Hyvärinen, J. Karhunen, and E. Oja. 2001. *Independent Component Analysis.* Wiley-Interscience, pp. 3–5, 147–164.

11

Data Mining in Sensor Networks

11.1 Introduction **11**-1
11.2 Data Mining: An Overview **11**-2
Core Ideas in Data Mining • Steps in Data Mining • Challenges in Data Mining
11.3 Data Mining in Sensor Networks **11**-4
Data Stream Clustering • Data Stream Classification • Frequent Pattern Mining
11.4 Time Series Sensor Streams **11**-6
Dimensionality Reduction • Compressions and Filtering • Forecasting
11.5 Social Sensing **11**-8
11.6 Distributed Algorithms **11**-9
11.7 Nonlinear Regression Using Choquet Integral **11**-9
11.8 Challenges in Sensor Data Mining **11**-11
11.9 Summary **11**-11
References **11**-12

Sinjini Mitra
California State University

Pramod Pandya
California State University

11.1 Introduction

Sensor network is a collection of addressable nodes of a data network, capable of capturing data in real time. In this sense, sensor network is a distributed computing and communication system or a resource. To our minds, the first and most obvious such a system would be the Internet, with a vast number of nodes designed for data processing, resource sharing, and communications. Hardware engineering advancement in last few years has given rise to devices with smaller and smaller footprint capable of collecting many different kinds of data. Of course, the sensor network that we all possess, and we cannot function without it, is our own brain, albeit not quite similar to the silicone-based distributed computing and communication system. In very recent years, devices with built-in global positioning satellite (GPS) have entered the arena of sensor network. We have now on our hand very smart and immensely capable sensor network with the ability to collect vast amounts of data, ready for processing, and guiding us in decision making.

Sensor data have become pervasive in recent years because of the popularization and wider availability of sensor technology that is cheap and easy to use. Sensors produce large volumes of data continuously over time, and this leads to numerous computational challenges in terms of data storage and data manipulation and analyses and retrieval. One major issue is *scalability.* The scalability challenges of sensor data have reached extraordinary proportions, with the increasing proliferation of ubiquitous and embedded sensors and mobile devices, each of which can potentially generate large streams of data. Coupled with the fact that many of these sensors are connected to the Internet, it is foreseeable that

in the near future, machine-generated data will dominate human-generated data by several orders of magnitude and this gap is only likely to increase with time [1]. In this context, the challenges associated with scalable and real-time management and mining of sensor data will potentially become even more significant in the coming years.

In the last couple years, we have a birth of a new discipline—data analytics. The pace of advancement in software technologies has not kept in pace with hardware technologies so far. So we have to address the following challenges:

- *Data management*: How do we store the data? What media do we use to store the data? Do we store all the data? How long should we store the collected data? What are the legal consequences of storing the data?
- *Sensor data mining and processing*: We have to address the need for new algorithms to process the data—of course, this requirement demands faster processors and memories. Do we process the sensor data in network or out of network?

Sensor data mining is relatively a new area of research. It involves collection, modeling, and processing of sensor data arising from several different types of sources. Often these are collectively referred to as *sensor data analytics*. The deluge of available data makes it possible to apply data mining techniques for obtaining a variety of useful analytical insights. The next section provides a brief outline of some basic data mining tools, and in the subsequent sections, we provide the application of these techniques to various types of sensor data. We conclude with a discussion of the current challenges to mining sensor networks and the emerging areas of research in this domain.

11.2 Data Mining: An Overview

The science of extracting useful information from large datasets or databases is known as *data mining*. A more elaborate definition of data mining according to Hand [2] is as follows:

> Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner.

Data mining is used in a variety of fields and applications, from the military and intelligence agencies to health-care and other business organizations. It is a relatively new discipline and is in a constant state of evolution, lying at the intersection of statistics, machine learning, data management, pattern recognition, artificial intelligence, and other areas. Several techniques for exploring and analyzing data have been around for a long time in the world of statistics such as regression methods (one of the most widely used analytical approaches even today), discriminant analysis, and analysis of variance, but many of these tools could not be implemented on today's huge datasets without adequate computational power and resources. Perhaps the most pertinent factor propelling the growth of data mining in recent times is the explosion of data, declining cost of massive data storage facilities and increasing availability of automatic data-capturing mechanisms such as sensors. Scannable bar codes, point-of-sale (POS) devices, mouse click trails, patient's health records, banking transactions, and GPS data are just a few examples. The mass retailer WalMart in 2003 captured 20 million transactions per day that was stored in a 1 TB database.

11.2.1 Core Ideas in Data Mining

In this section, we briefly outline some of the main data mining tools used today.

11.2.1.1 Classification

Classification is perhaps the most widely used data mining tool in most applications, used for categorical variables. A credit card transaction can be legitimate or fraudulent. A packet of data traveling on

a network can be benign or threatening. A common task in data mining is to examine data where the classification is unknown (say, we do not know whether a particular credit card transaction is fraudulent or not), with the goal of predicting what that classification is or will be. Similar data where the classification is known are used to develop rules, which are then applied to the data with the unknown classification. Some well-known techniques of classification include logistic regression, classification and regression trees (CARTs), *k*-nearest neighbors (*k*-NNs), naïve Bayes, neural networks, and linear discriminant analysis (LDA).

11.2.1.2 Prediction

Prediction is similar to classification, except here the goal is to predict the value of a numerical variable (e.g., selling price of a house) rather than a class (e.g., fraudulent transaction or not). Most of the aforementioned classification methods can also be suitably adapted to perform prediction although the most popular prediction tool is linear regression.

11.2.1.3 Clustering

The goal of clustering or cluster analysis methods is to segment a given set of data records into a set of homogeneous groups (called *clusters*) based on several measurements made on those records for the purpose of generating insight. It is very popular in business applications such as customized or targeted marketing and industry analysis. Common clustering approaches include hierarchical clustering and *k*-means clustering.

11.2.1.4 Association Rules

Large databases on customer transactions lend themselves naturally to the analysis of associations among items purchased or *what goes with what*. *Association rules* or *affinity analysis* can be used in a variety of ways. For example, grocery stores can use such information after a customer's purchases have all been scanned to print discount coupons and to help arrange and organize items on shelves for better chances at selling together. Online vendors such as Amazon.com and Netflix use these methods in their recommender systems that suggest new purchases to customers based on past ones.

11.2.1.5 Data Exploration and Reduction

A critical component of data mining and often the initial step is preprocessing and cleaning the data at hand. This involves reviewing and examining the data to identify important and relevant variables, detect outliers or missing and inaccurate data, and transform data (if necessary). Sensible data analysis often required distillation of complex data into simpler data. Rather than dealing with thousands of product types, a market researcher might want to aggregate them into a smaller number of groups. This process of consolidating a large number of variables (or cases) into a smaller set is termed *dimension reduction*. Such data processing and exploration also often provide insights into the types of data mining tasks required to answer the specific questions of interest.

11.2.1.6 Data Visualization

Another technique for exploring data is through graphical analysis or visualization methods. Such methods are very useful in not only understanding the behavior of the variables included in the study but forming an initial idea about relationships between several variables. Some common traditional visualization tools are histograms, bar and pie charts, scatterplots, boxplots, and line plots (for time series data). Some more novel methods include heat maps; interactive plots involving zooming, panning, and filtering; treemaps; network plots; and map charts, among others.

Data mining techniques fall broadly in two categories: (1) *supervised learning algorithms* and (2) *unsupervised learning algorithms*. In supervised methods, an outcome or response variable is available and learning occurs based on training data (where the outcome values are known). Once the algorithm is trained, it is applied to another set of data, called validation data, where the outcome is

unknown and needs to be determined. Classification and prediction methods fall in the domain of supervised learning tools. Unsupervised learning algorithms, on the other hand, are those used when there is no outcome variable to predict or classify. Hence, there is no learning or training from cases where such outcomes are known; instead, the task here is to unravel and study the underlying patterns in the dataset. Association rules, cluster analysis, visualization methods, and dimension reduction techniques are all unsupervised learning methods.

11.2.2 Steps in Data Mining

Here is a list of steps to be followed in a typical data mining effort:

1. Develop an understanding of the purpose of the data mining project.
2. Obtain the dataset to be used in the analysis (often sampling techniques and database retrieval methods are employed for this).
3. Explore, clean, and preprocess the data (includes data visualization as well).
4. Reduce the data (if necessary).
5. Determine the data mining task.
6. Choose the data mining techniques to be used.
7. Use algorithms to perform the task.
8. Interpret the results.
9. Deploy the model.

11.2.3 Challenges in Data Mining

The main challenges in the area of data mining include data management and computational efficiency. With the availability of more and more data, it is imperative to have adequate data management and storage facilities. Often real data are unstructured and complex in nature (e.g., health-care data), and unless these data can be processed in such a way that they are amenable to the existing data mining algorithms, the task at hand cannot be accomplished. Therefore, the rapid and continuing improvement in computing capacity and data handling mechanisms are essential enablers of the growth of the field of data mining in the current years. Another challenge is the choice of the correct method or ensemble of methods to use in a particular scenario. There are numerous data mining methods available today, so an in-depth understanding of the problem at hand and correct formulation of the research questions are extremely important to help select the appropriate tool. Often multiple techniques are employed and then results are compared to assess the best one suitable for the given data. Caution should also be exercised in accurately interpreting the results so that people with no statistical and computing background are also able to develop an understanding of the insights gained from the data mining activity.

Some of the emerging research areas of application in data mining include data from GPS devices, social networks, clickstreams (on the Internet), and surveillance videos (at airports, say), among others. More and more algorithms and software packages are also emerging in the market that help carry out data mining tasks, such as SAS, SPSS (owned by IBM now), Tableau (visualization software), and XLMiner (Microsoft Excel Add-In). All of the data mining tools outlined in this section are elaborated in great detail in [2].

11.3 Data Mining in Sensor Networks

In recent years, there has been an explosive growth in the amount of data generated by sensor networks in different arenas. Hence, data mining and analytical tools are also constantly evolving to be able to deal with these massive datasets. One area of growth has been in the area of mathematical and statistical

model-based techniques, such as time series models and Markov models. Particularly, when the volume of data is very large, it leads to a number of computational and mining challenges:

- As the volume of the data increases, it becomes increasingly more difficult to process them efficiently with multiple passes. Thus, one data item has to be processed at one time. This leads to implantation problems for the existing algorithms, and they need to be redesigned.
- In most cases, there is an inherent temporal component in mining data streams arising from sensor networks, which tend to evolve over time. Data mining techniques thus need to be designed carefully so as to be able to handle the temporal variations in the underlying data.
- Data collected from sensors are often noisy and error prone; hence, it calls for tools to reduce the degree of uncertainty in the mining tasks. Much of the errors occur in the transmission stage and can often have incompleteness (e.g., battery of the GPS system runs out).
- Data from sensor networks often need to be analyzed in a distributed fashion; hence, data mining methods such as clustering and classification also need to be adapted to meet these requirements.

In the next sections, we outline some data mining tools for sensor data and the associated issues and challenges.

11.3.1 Data Stream Clustering

Clustering is a popular data mining technique that helps in learning patterns in a dataset in an unsupervised manner. However, it is difficult to adapt these traditional clustering methods to data streams from sensor data because of one-pass constraints discussed earlier.

An interesting adaptation of the *k*-means algorithm was discussed in [3], which uses a partitioning-based approach to create clusters over the entire data stream. However, in certain applications in practice, it might be necessary to be able to examine clusters over specified time intervals. For example, an analyst may wish to study the behavior of clusters in the data stream over the past week or the past month in order to fully understand underlying data behavior and perform comparative analysis. One such technique is microclustering [4] in which first-order and second-order moments of the data are tracked via feature vectors. These in turn help in calculating important cluster characteristics such as centroids in real time. Other authors [4] clearly demonstrated that this method is more effective than the partitioning-based approach. A couple of examples of data applications for this technique are (1) high-dimensional data [5], (2) data with uncertainty [6], (3) text data [7], and (4) categorical data [7]. For both text- and categorical-type data, counts of frequencies of discrete attributes as well as correlations are stored instead of the moments as in the case of quantitative data. A number of density-based clustering approaches are also available for stream clustering [8,9].

Another type of sensor data arises in a distributed setting where large volumes of data are collected separately at different sensors. In such a case, the natural approach is to transmit all the data to a centralized server, a phenomenon that significantly raises costs. Moreover, computation becomes harder too. A method proposed in [10] performs local clustering at each node and then merges these different clusters into one single big cluster. A second method [10] is also mentioned for distributed clustering, which is called the *parallel guessing algorithm*. Another method for distributed sensor stream clustering that reduces the dimensionality and cost by maintaining an online discretization may be found in [11].

11.3.2 Data Stream Classification

Classification is one of the most widely applied data mining techniques applied in case of sensor data streams. Owing to the temporal component of stream data, some adjustments are required

for traditional classification methods. The concept of stream evolution is sometimes referred to as *concept drift* in the stream classification literature [12]. Of all available methods, the following are most popular:

1. *Very fast decision tree (VFDT) method*: This method is adapted from the classic decision tree method with the use of sampling-based approximations. These are so designed as to be able to handle evolving data streams by using sliding windows to update the classifier at each step. Furthermore, the VFDT algorithm [13] has been extended to process numerical attributes and reduce the sample size.
2. *On-demand classification*: This method focuses on the case when both the training and the test streams evolve over time and work by creating class-specific microclusters from the underlying data [14]. For an incoming record in the test stream, the class label of the closest microcluster is used to determine the class label of the test case.
3. *Ensemble-based classification*: This technique [15] uses an ensemble or combination of classification methods such as C4.5 (decision trees) and naïve Bayes in order to enhance the classification accuracy. This method works best if the data behave differently over time and a different method produces the optimal results in each instance. Thus, the use of multiple methods together provides the required robustness as well as increased accuracy.
4. *Compression-based methods*: Such methods typically work by the use of compression techniques applied to real-time classification of streaming sensor data, wherein time series bitmaps are updated in constant time [16]. This makes these classifiers very efficient in practice.

11.3.3 Frequent Pattern Mining

The problem of frequent pattern mining in data streams consists of finding the frequent item sets either over a sliding window or the entire data stream [17]. For the *entire data stream model*, the frequent patterns are mined over the entire data stream. The main difference from a conventional pattern mining algorithm is that the patterns need to be mined in one pass over the entire data stream. In the *sliding window model*, on the other hand, the evolution of data over time is accounted for by determining frequent patterns over a particular sliding window. A method for determining the frequent patterns over a sliding window is included in [18]. The primary focus of the algorithm is to detect the closed frequent item sets over the entire data stream. The proposed algorithm is called *moment*, and the primary underlying idea of this technique is based on the fact that the boundary between the closed frequent item sets and frequent item sets moves slowly. The reader is referred to [19] for a more in-depth review of the various clustering, classification, and pattern mining techniques for mining sensor data.

11.4 Time Series Sensor Streams

The main component of time series sensor streams is correlation; hence, analytical techniques for this purpose have gained a lot of popularity. The problem consists of capturing correlations both across multiple streams (e.g., prices in the same market) and in single streams across time (autocorrelations). The latter help capture periodic trends in time series streams as well since values at different times are usually not independent. Both these problems are inherently related although the results are interpreted slightly differently in each case.

The main data mining techniques applicable in case of time series sensor streams are dimensionality reduction, filtering, and forecasting. These are all closely related, and all three areas are too extensive to cover within this chapter. So we discuss each of these tools in brief details in the next few subsections and provide appropriate references.

11.4.1 Dimensionality Reduction

Since data arising from sensor networks are typically of large dimensions, it is necessary to perform some form of dimension reduction in order to consolidate the set of variables to be analyzed. Dimension reduction is thus an important aspect of data mining for all applications and is often the first step in the process following data cleaning and preprocessing. There are several methods for dimension reduction such as subset selection and floating search, but the most popular one is *principal component analysis* (PCA). PCA works particularly well in case we have subsets of measurements (predictors in case of predictive models) that are highly correlated. In that case, it provides few variables that are weighted linear combinations of the original variables that retain the explanatory power of the full set. One drawback of this tool is that it is applicable only for numerical or quantitative variables and does not work with categorical or qualitative variables. For a detailed overview of the PCA technique, the reader is referred to [2].

11.4.2 Compressions and Filtering

Initial work on time series representation used Fourier transforms [20,21]. More recent work focuses on fixed, predetermined bases or approximating functions. A PCA [22] approximates the time series with piecewise constant or linear functions. DAWA [23] combines the discrete cosine transform (DCT) and discrete wavelet transform (DWT). However, all these techniques enable compression of the time series for indexing purposes and not for knowledge discovery. The seminal work found in [24], for rule discovery in time series, is based on sequential patterns extracted after discretization. More recently, vector quantization has been used for time series compression [25]. The author in [26] has presented ways to efficiently store time series as well as facilitate easy computation of correlations and application of other diagnostic tools (like graphical techniques, for instance) after compression. This is accomplished via multiscale analysis that yields sparse time and frequency representations of the series.

The recently developed theory of compressed sensing [27,28] studies the problem of signal summarization and reconstruction based on a subset of observed values. Specifically, the framework helps to estimate projections of a signal into a set of given basis functions from a small sample of its values.

11.4.3 Forecasting

Forecasting is probably the most important component of time series analysis, whereby the future value is estimated based on past ones from historical data. Probably the most popular and widely used forecasting method is based on *autoregressive* (*AR*) models. The information found in [29] contains details about all these time models. The main idea is to express the observation at time t, namely, x_t as a function of its previous values plus noise:

$$\mathbf{x}_t = \phi_1 \mathbf{x}_{t-1} + \phi_2 \mathbf{x}_{t-2} + \cdots + \phi_W \mathbf{x}_{t-W} + \epsilon_t,$$

where

W is the forecasting window size
ϕ_i's are parameters of the model
ϵ represents the noise

The parameters are estimated from the data, and then past values of the time series sensor are used to forecast a future value using this equation. Some common estimation methods include ordinary least squares, method of moments (Yule–Walker equations), or Markov chain Monte Carlo (MCMC) methods.

Another variant of the AR model is the *autoregressive moving average* (ARMA) model that is also used widely for forecasting in time series data. The model consists of two parts: an AR part and a moving average (MA) part. The model is usually then referred to as the ARMA(p, q) model where p is the order of the AR part and q is the order of the MA part. MAs are used for smoothing a time series and work

by calculating the average of successive values of the series using a moving window. ARMA models in general can, after choosing p and q, be fitted by least squares regression to find the values of the parameters, which minimize the error term. It is generally considered good practice to find the smallest values of p and q that provide an acceptable fit to the data. For a pure AR model, the Yule–Walker equations may be used to provide a fit. Finding appropriate values of p and q in the ARMA(p, q) model can be facilitated by plotting the partial autocorrelation functions for an estimate of p and likewise using the autocorrelation functions for an estimate of q. Further information can be gleaned by considering the same functions for the residuals of a model fitted with an initial selection of p and q. Many statistical software packages like R, MATLAB®, SAS, and STATA have built-in functions for fitting these models.

The reader is referred to [29] for details about the implementation of these forecasting models. Furthermore, the information found in [30] contains a detailed overview of all the data mining tools applicable to time series sensor streams along with associated issues and challenges.

11.5 Social Sensing

A number of sensor applications today collect data that can be directly associated with human interactions. This is driven in part by the explosive growth of social networking sites in the recent years, such as Facebook, Twitter, LinkedIn, and Google+ to name a few. Facebook has 1.2 billion monthly users as of January 2014 [31], Twitter reports 200 million active users as of February 2013 [32], and this has given users the ability to easily share information online by connecting individuals and groups. Moreover, all these technologies are available on mobile devices (smartphones, tablets, etc.) today in the form of mobile apps, which makes them easily accessible on the go. These networks are data rich and contain a lot of structure that can be mined effectively to gain valuable insights into behavioral patterns [33,34]. A natural way to enhance the power of such social applications is to embed sensors within these platforms to continuously collect large amounts of data for prediction and other mining purposes. A few applications are as follows:

1. *Vehicle tracking applications*: Much data are available from real-time tracking of locations of automobiles today via GPS. This can provide other drivers with important information as to how to avoid points of traffic congestion in the city. This is also important for military vehicles, which often require location coordination. One example of vehicular application can be found in [35], which proposed the *GreenGPS* system.
2. *Health-care applications*: In recent years, numerous medical sensor devices are being used to track the personal health of individuals or make recommendations regarding their lifestyle. This is highly useful for deploying emergency responses; for making long-term predictions about chronic diseases such as high blood pressure, cholesterol, asthma, and diabetes; and for providing suggestions about healthy living habits like diet control and exercises [36].

Social sensing applications provide numerous research challenges from the perspective of analysis, some of which are listed below:

- The main concern in social networks is regarding privacy since a lot of personal data are shared over these. Hence, suitable privacy-control mechanisms need to be embedded in these systems that gather the data.
- Most sensors operate on batteries, which often have limited life. Certain types of sensor data collection can drain the battery life more quickly than others (e.g., GPS, mobile phones). Therefore, it is critical to design the applications with the underlying trade-offs, so that the battery life is maximized without significantly compromising the goals of the application.
- The volume of data can be very large, especially those that arise from real-time continuous tracking such as GPS and social media sites. Moreover, these data are often unstructured and do not conform to any known standards that can be analyzed using traditional software. Thus, appropriate techniques and software packages are required to store and process these efficiently. There are many advances made in this area today with the advent of cloud computing capabilities

and platforms for handling *big data* (e.g., HADOOP). Besides, sometimes it is necessary to mine sensor data in a dynamic fashion, providing real-time output (e.g., credit monitoring to trigger alerts in the event of fraudulent activities), which increases associated challenges significantly.

- Sensor data are often error prone and hence there are several challenges about dealing with *trustworthiness* of the collected data.

Once the data are collected and stored properly in databases and data warehouses, common data mining techniques can be applied, such as clustering, association rules, prediction, and classification techniques to make inferences. If predictions are required over time, time series forecasting methods described in the earlier section can be employed. One tool that is increasingly becoming popular is the use of *text analytics* and *sentiment analysis* [37] for mining all the social interactions and opinions expressed in social media sites to understand consumer behavior and preference about specific brand products—an extremely important application of these outcomes being in marketing.

11.6 Distributed Algorithms

As we have stated earlier, the sensor network is an assembly of processing nodes with small footprint. The primary reason for small footprint is so that the demand for power requirement is low enough. There is also need for power requirement for transmission of data from the sensor nodes to central data storage. Do we require sensor nodes to transmit the data as they collect them, or some processing should take place at the node, so as to eliminate and discard the data that would be redundant due to being corrupt as a consequence of presence of noise? Of course, if no processing takes place at nodes, then large bandwidth would be required to transmit all the collected data in real time. Apart from this, there too would be duplication of data collected from each of the sensor as well. Hence, we need some sort of filtering mechanism to eliminate duplicated and redundant data. We could summarize the sensor data network as distributed communication system—which necessitates a distributed algorithm [38]. Sensor nodes can be configured to act either in a synchronous or an asynchronous mode. In a synchronous mode, sensor nodes would have a global view; therefore, nodes would exchange messages with one another as they collect data and route the data after some local processing. So we need a routing algorithm to this end. In an asynchronous mode, each sensor node would operate independently of one another, collect data, and route the data to a central depository.

We could model the sensor data network using graph theory, $G = (V, E)$ with vertices $V = \{v_1, v_2, v_3, \ldots\}$ and edges $E = \{e_1, e_2, e_3, \ldots\}$, where each of the vertex of the graph is the sensor node and the edges of the graph are the communication links. We could break up the sensor data network, into groups of smaller data networks, or we could build larger sensor data network from groups of smaller sensor data networks. Then we could apply techniques from sheaf theory to build consistency between pieces of local information (i.e., groups of smaller sensor data networks) to arrive at global inference (corresponding to larger sensor data network in question) [39].

11.7 Nonlinear Regression Using Choquet Integral

The Choquet integral (CI) is a tool for the information fusion that is very effective in the case where fuzzy measures associated with it are well chosen [40,41]. A new approach for calculating fuzzy measures associated with the CI was proposed in a context of data fusion in multimodal biometrics. CI is a tool for the information fusion, which can generalize many operators such as the ordered weighted averaging, the arithmetic sum, the minimum, and the maximum.

Information collected from diverse sources is aggregated in a standard way by

$$y = w_1 f(x_1) + w_2 f(x_2) + \cdots + w_n f(x_n),$$

where y represents the weighted sum and that $\sum_{i=1}^{i=n} w_i = 1$.

In databases, the information sources $x_1, x_2, \ldots, x_n$ are to be regarded as attributes and $f(x_1), f(x_2), \ldots, f(x_n)$ are their observed values. Thus, the weighted sum represents the Lebesgue integral on the set of information sources and represents a linear aggregation model. Such a linear model is applied in multiobjective decision modeling and classification. In using the linear model, a fundamental assumption is made that there is no interaction among the contributions from individual attributes towards a certain target.

Regression is one of the most often used tool in statistical data analysis. Regression helps to determine if a relationship exists between observational data (predictive attributes) and a target attribute. This relationship could be either linear or nonlinear, defined by a set of unknown parameters. Once this relationship is determined, then one predicts the value of the target variable if a new set of predictive variables has been collected.

Given a set of $n + 1$ attributes $x_1, x_2, x_3, \ldots, x_n$ and y is the target attribute, then we want to determine the relationship between the x's and the y. We regard y as a random variable and express a linear relationship as noted in the following:

$$y = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + \cdots + a_nx_n + N\left(0,\sigma^2\right),$$

where

$a_0, a_1, a_2, a_3, \ldots, a_n$ are the unknown regression coefficients
$N(0, \sigma^2)$ is a random variable with mean zero and unknown variance σ^2

In such a model, the underlying assumption is that there are no interactions among the $x_1, x_2, x_3, \ldots, x_n$ towards the target y. The unknown parameters $a_0, a_1, a_2, a_3, \ldots, a_n$ are determined through the usual least square method that minimizes the total squared error.

In the earlier discussion, the symbol x_i is used to denote an attribute; if it were to represent an observation, then we may use $f(x_i)$. In such a case, the multiregression model can be written as

$$a_1f\left(x_1\right) + a_2f\left(x_2\right) + a_3f\left(x_3\right) + \cdots + a_nf\left(x_n\right) = \int f d\mu,$$

which represents a weighted sum of values of function f on a set X and μ represents a classical measure. Hence, the multiregression model is expressed as

$$y = \int f d\mu + N\left(0,\sigma^2\right).$$

If interactions among the attributes towards a certain target are to be included, then the weighted sum cannot be the correct representation. Instead we need to use the nonadditive set function, such as the CI, which is a generalization of Lebesgue integral and coincides with the Lebesgue integral when the nonadditive measure is replaced by the additive measure. The CI is considered to be a type of nonlinear integral.

Definition 11.1

Let f be a nonnegative measurable function on $(X, \mathcal{F})$ and $E \in \mathcal{F}$. The CI of f on E with respect to monotone measure μ, denoted by $(C)\int_E f d\mu$, is defined as

$$(C)\int_E f d\mu = \int_0^\infty \mu\left(F_\alpha \cap E\right) d\alpha,$$

where $F_\alpha = \{x | f(x) \geq \alpha\}$, called the α-level set of f, for $\alpha \in [0, \infty)$.

If the set function is б-additive, the preceding definition is equivalent to the definition of Lebesgue integral of f with respect to μ. For a nonlinear multiregression model:

$$y = c + (C)\int_E (a + bf)d\mu + N\left(0, \sigma^2\right),$$

where

c represents a constant
a and b represent a real-valued function on X
f represents an observation of $x_1, x_2, x_3, \ldots, x_n$
μ is a measure on X

11.8 Challenges in Sensor Data Mining

As clearly evident in the earlier sections, data mining in sensor network has several technical challenges in the form of data processing, communication, and sensor management. First of all, there are huge volumes of data available from these networks, most often on a continuous basis. Secondly, there is significant noise frequently present in such data. Although much advance has occurred today in data storage and computational efficiency, there are still challenges in the form of handling so much data in a dynamic fashion that is required in case of several sensor network applications such as social networks, GPS systems and surveillance cameras at airports, and ATMs that are used for security purposes. This is sometimes referred to as the *big data problem* in the context of analytic applications. Because of potentially harsh uncertain and dynamic environments, along with energy and bandwidth constraints, wireless ad hoc networks pose additional technical challenges in network discovery, network control and routing, collaborative information processing, querying, and tasking [42].

Another daunting challenge facing sensor data mining is regarding where the processing of the gathered data takes place. If sensor data have to be processed out of network, then a fast communication link is required to transmit the data from the sensor network to a central storage media. Of course, in such a scenario, duplicate and even corrupt data could get transmitted, leading to a waste in bandwidth usage. Solution to this problem would be a local processing at the sight of the network—filter out duplicate and corrupt data and transmit well-behaved data. We need efficient and fast algorithms to this end.

Moreover, the application of data mining to sensor data requires choice of the appropriate methods, choice of appropriate parameters, and interpretation of results that often are not straightforward and require the expert skills. It is imperative to understand the importance of applying the right technique to each type of data in order to make the correct business decisions. Ad hoc and heuristic methods should be avoided and caution should be exercised in mining sensor network data. Furthermore, cost and privacy issues, as mentioned in context of some of the aforementioned specific mining tasks, are present as well and need to be taken into account.

11.9 Summary

Sensor networks and hence sensor data are ubiquitous now, and so are data analytics or data mining methods. The marriage of these two leads to empowering insights about the phenomenon at hand. Sensor data are varied and often arrive dynamically in large volumes, so an understanding of the underlying issues and challenges is crucial to processing and analyzing them. As technology become more and more advanced, with several platforms available today for handing complex data forms, application of data mining tools to sensor data is becoming more and more popular and widespread in different areas. The future thus holds tremendous promise in terms of the power of information gained from

analyzing such data. Google has sold a limited number of wearable computers with an optical head-mounted display. Users can interact with the device through a touch pad or voice-activated mechanism, to access the e-mail, browse the web, do navigation, record the video, and use the webcam. Google glass has catapulted sensor data technology to a new dimension. The next generation of information technology would be carbon-based sensor data network, processing and transmitting sensor data in real time! The presently available information technologies sooner would be antiquated, and we would have to continue to spin the wheels of technologies endlessly, and such will be our fate for eternity!

References

1. Lohr, S. (2012). The age of Big Data. *The New York Times*, Sunday Review, Appeared on February 12, 2012. http://www.nytimes.com/2012/02/12/sunday-review/big-datas-impact-in-the-world.html?pagewanted=all. Accessed February 10, 2014.
2. Shmueli, G., Patel, N.R., Bruce, P.C. (2010). *Data Mining for Business Intelligence: Concepts, Techniques, and Applications in Microsoft Office Excel with XLMiner*. Wiley, Hoboken, New Jersey.
3. Guha, S., Mishra, N., Motwani, R., O'Callaghan, L. (2000). Clustering data streams. *IEEE FOCS Conference*, Redondo Beach, CA.
4. Aggarwal, C.C., Han, J., Wang, J., Yu, P. (2003). A framework for clustering evolving data streams. *VLDB Conference*, Berlin, Germany.
5. Aggarwal, C.C., Han, J., Wang, J., Yu, P. (2004). A framework for high dimensional projected clustering of data streams. *VLDB Conference*, Toronto, Canada.
6. Aggarwal, C.C., Yu, P. (2008). A framework for clustering uncertain data streams. *ICDE Conference*, Cancun, Mexico.
7. Aggarwal, C.C., Yi, P. (2006). A framework for clustering massive text and categorical data streams. *SIAM Data Mining Conference*, Bethesda, MD.
8. Cao, F., Ester, M, Qian, W., Zhou, W. (2006). Density-based clustering of an evolving data stream with noise. *SIAM Data Mining Conference*, Bethesda, MD.
9. Chen, Y., Tu, L. (2008). Density-based clustering for real-time stream data. *ACM KDD Conference*, Las Vegas, NV.
10. Cormode, G., Muthukrishnan, S., Zhuang, W. (2007). Conquering the divide: Continuous clustering of distributed data streams. *ICDE Conference*, Istanbul, Turkey.
11. Rodrigues, P., Gama, J., Lopes, L. (2008). Clustering distributed sensor data streams. *PKDD Conference*, Antwerp, Belgium.
12. Domingos, P., Hulten, G. (2000). Mining high-speed data streams. *Proceedings of ACM KDD Conference*, Boston, MA.
13. Jin, R., Aggarwal, C.C. (2003). Efficient decision tree construction on streaming data. *ACM KDD Conference*, Washington, DC.
14. Aggarwal, C.C., Han, J., Wang, J., Yu, P. (2004). On-demand classification of data streams. *ACM KDD Conference*, Seattle, WA.
15. Wang, H., Fan, W., Yu, P., Han, J. (2003). Mining concept-drifting data streams using ensemble classifiers. *ACM KDD Conference*, Washington, DC.
16. Kasetty, S., Stafford, C., Walker, G., Wang, X., Keogh, E. (2008). Real-time classification of streaming sensor data. *ICTAI Conference*, Sacramento, CA.
17. Giannella, C., Han, J., Pei, J., Yan, X., Yu, P. (2002). Mining frequent patterns in data streams at multiple time granularities. *Proceedings of NSF Workshop on Next Generation Data Mining*, Baltimore, MD.
18. Chi, Y., Wang, H., Yu, P., Muntz, R. (2004). Moment: Maintaining closed frequent item sets over a stream sliding window. *ICDM Conference*, Brighton, UK.
19. Aggarwal, C.C. (2013). Mining sensor data streams. In C. Aggarwal (ed.), *Managing and Mining Sensor Data*. Springer, New York.

20. Agarwal, R., Faloutsos, C., Swami, N.A. (1993). Efficient similarity search in sequence databases. *FODO*, Chicago, IL.
21. Faloutsos, C., Raghunathan, M., Manolopoulos, Y. (1994). Fast subsequence matching in time-series databases. *SIGMOD*, Minneapolis, MN.
22. Chakrabarti, K., Keogh, E., Mehrotra, S., Pazzani, M. (2002). Locally adaptive dimensionality reduction for indexing large time series databases. *ACM Transactions on Database Systems (TODS)*, 27(2), 188–228.
23. Hsieh, M.J., Chen, M.S., Yu, P.S. (2005). Integrating DCT and DWT for approximating cube streams. *CIKM*, Bremen, Germany.
24. Das, G., Lin, K.I., Mannila, H., Raghunathan, G., Smyth, P. (1998). Rule discovery from time series. *KDD*, New York, NY.
25. Lin, S., Gunopulos, D., Kalogeraki, V., Lonardi, S. (2005). A data compression technique for sensor networks with dynamic bandwidth allocation. *TIME*, Burlington, VT.
26. Reeves, G., Liu, J., Nath, S., Zhao, F. (2009). Managing massive time series streams with multi-scale compressed trickles. *VLDB Conference*, Lyon, France.
27. Donoho, D. (2006) Compressed sensing. *IEEE TOIT*, 52, 1289–1306.
28. Haupt, J., Nowak, R. (2006). Signal reconstruction from noisy random projections. *IEEE TOIT*, 26, 4036–4048.
29. Borckwell, P.J., Davis, R.A. (1991). *Time Series: Theory and Methods*. Springer Series in Statistics, 2nd edn. Springer-Verlag, New York.
30. Papadimitriou, S., Sun, J. (2013). Dimensionality reduction and filtering on time series sensor streams. In C. Aggarwal (ed.), *Managing and Mining Sensor Data*. Springer, New York.
31. Wikipedia page on Facebook. http://en.wikipedia.org/wiki/Facebook.
32. Wikipedia page on Twitter. http://en.wikipedia.org/wiki/Twitter.
33. Clauset, A., Newman, M.E.J., Moore, C. (2004). Finding community structure in very large networks. *Physical Review E*, 70, 066111.
34. Kumar, R., Raghavan, P., Rajagopalan, S., Tomkins, A. (1999). Trawling the web for emerging cybercommunities. *WWW Conference*, Toronto, Canada.
35. Ganti, R.K., Pham, N., Ahmadi, H., Nangia, S., Abdelzaher, T. (2010). *GreenGPS: A Participatory Sensing Fuel-Efficient Maps Application*, Mobisys, San Francisco, CA, June 2010.
36. Ganti, R.K., Srinivasan, S., Gacic, A. (2010). Multi-sensor fusion in smartphones for lifestyle monitoring. *International Conference on Body Sensor Networks*, Singapore, June 7–9, 2010.
37. Barker, M., Barker, D.I., Bormann, N.F., Neher, K.E. (2008). *Social Media Marketing: A Strategic Approach*. Cengage, South Western, Cengage Learning (Publisher), Mason OH.
38. Lenzen, C., Wattenhofer, R. (2008). Leveraging linial's locality limit. In *Proceedings of the 22nd Symposium on Distributed Computing* (*DISC*), Arcachon, France, pp. 394–407.
39. Kashiwara, M., Schapira, P. (1990). *Sheaves on Manifolds*. Springer-Verlag, Berlin, Germany.
40. Khalifa, A.B., Gazzah, S., Benamara, N.E. (2013). Multimodal biometric authentication using Choquet integral and genetic algorithm. World Academy of Science, Engineering and Technology, *International Journal of Computer, Information Science and Engineering*, 7(3), 27–36.
41. Su, K.-L., Jau, Y.-M., Jeng, J.-T. (2011). Modeling of nonlinear aggregation for information fusion systems with outliers based on the Choquet integral. *Sensors*, 11, 2426–2446.
42. Chong, C., Kumar, S.P. (2003). Sensor networks: Evolution, opportunities and challenges. *Proceedings of the IEEE*, 9(8), 1247–1256.

V

Obstacles in Sensor Networks

12 Wireless Sensor Network Security: PHY-Layer Security *Andrea Bartoli, Juan Hernandez-Serrano, Miquel Soriano, Mischa Dohler, Apostolos Kountouris, and Dominique Barthel* **12**-1
Introduction • PHY-Layer Security Solutions • Authentication Preamble for Out-of-Sync • Summary • References

13 Impact of Correlated Failures on Wireless Sensor Network Reliability *Jorge E. Pezoa and Silvia Elena Restrepo* **13**-1
Introduction • Graph Theory and Random Field Theory: Graph Theory and the WSN Topology • Modeling Correlated Failures Using Markov Random Field Theory • Sampling Correlated Failures from the Global Gibbs Distribution • Results • Summary • Acknowledgment • References

14 Acoustic Sensors and Algorithms for Urban Security *Juan R. Aguilar* **14**-1
Introduction • Acoustic Signature of Small Firearm • Electroacoustic Sensor Networks for Gunshot Detection • Signature Analysis for Gunshot Detection • Summary • References

12

Wireless Sensor Network Security: PHY-Layer Security

Andrea Bartoli
Universitat Politecnica de Catalunya

Juan Hernandez-Serrano
Universitat Politecnica de Catalunya

Miquel Soriano
Universitat Politecnica de Catalunya

Mischa Dohler
King's College London and Worldsensing

Apostolos Kountouris
Orange Labs

Dominique Barthel
Orange Labs

12.1 Introduction **12**-1
DoS Exhaustion Threats • State of the Art
12.2 PHY-Layer Security Solutions **12**-4
Authentication Preamble for Normal Traffic Flow • Frame Format Position • Authentication Preamble Window • AP Protocol Uses • Complete Security Suite • Comparative Table
12.3 Authentication Preamble for Out-of-Sync **12**-10
Authentication Preamble during Recovery • Out-of-Sync Handshake Model • Out-of-Sync Overview
12.4 Summary **12**-16
References **12**-17

12.1 Introduction

The wireless sensor network (WSN) technology is a novel communication paradigm that is rapidly gaining an importance in the vision of telecom applications and services. WSN devices are typically required to be small, inexpensive, and able to operate in a self-organized mode and to communicate through the wireless medium. An important constituent of this technology is its capacity to interconnect machines, which are often referred to as devices, to exchange data using energy-efficiency communication protocols.

Although WSNs can allow the emergence of new opportunities, several security challenges must be addressed in order to take advantage of all the benefits offered by this technology. Consider, for example, an unattended WSN in charge of monitoring a given utility, such as energy consumption in future Smart Grid communication system; if the network's nodes can be easily forged or compromised, the supply service is no longer reliable for both the customer and the provider.

Moreover, reliability of the service relies on the resilience of the nodes and the communication links to denial-of-service (DoS) attacks, that is to say, in the service availability.

Analyzing availability, the specific nature of low-power networks that are often made up of hundreds to thousands of nodes, which are left unattended for years of operation without human intervention, makes the guarantee of this security requirement especially challenging in WSNs. While security, understood as protecting the communication data, has been widely studied [1–6], providing the service availability in WSN communication systems still remains as an open field with a lot of work to be done. In this context, novel Physical(PHY)-layer methods can provide new opportunities to develop more efficient and secure WSNs.

Generally speaking, availability relies on the proper operation of the network nodes and their communication links. Therefore, the said availability is mainly jeopardized by DoS attacks, which can be classified into (1) attacks damaging network nodes; (2) attacks disturbing the communication links, for example, by means of jamming techniques; and (3) attacks exhausting the network nodes, for example, by engaging them into meaningless packet exchanges that consume their precious batteries and thus significantly shorten their lifetime. Protection of the network nodes is mainly related with physical security that should allow keeping nodes out of reach of attackers. Regarding communications, several security proposals have been proposed in order to prevent or mitigate jamming-based attacks, most of them based on frequency-hopping and channel-surfing techniques [7].

However, to the best of our knowledge, there is only one energy-efficient solution able to mitigate the effects of exhaustion attacks exploiting PHY-layer benefits [8]. Exhaustion attacks often exploit the lack of authentication at link or network layer in two ways: (1) injecting fake packets in the network, which is related to network exhaustion attack, and (2) forcing link-layer dialogs that makes a given node continuously transmit and receive messages, which is related to node exhaustion attack. In the former case, the malicious packet is often detected at the application layer, and thus precious network resources are wasted relaying the packet to its destination. In the latter case, an attacker can exhaust the node's resources by repeatedly sending useless fake messages that are completely received before being discarded.

Focusing on node exhaustion attacks, one can argue that even the normal operation of a group of nodes can be in some manner considered as a node exhaustion attack. Indeed, besides all the packets intended to a given node, due to the broadcast nature of wireless media, that node receives all packets sent by its in-range neighbors. Typically, energy is spent in receiving the entire packets, performing the security checks, and checking on intended destinations; only then the packets are discarded. Since non-intended legitimate packets arrive at a fairly regular frequency that depends on the neighborhood cardinality and the neighboring nodes' transmission rates, the energy spent on these to-be-rejected packets is not negligible.

In order to provide a solution for the presented issues (protection against node exhaustion attacks and avoiding of unnecessary waste of resources due to complete reception of non-intended packets), in [8], we have proposed an authentication/verification method at the PHY layer that is able to challenge exhaustion problems and attacks in embedded networks, such as WSNs. The presented method is able to reject non-intended nonmalicious and/or malicious packets after the reception of just an authentication preamble (AP) at PHY layer, saving energy and therefore increasing the system lifetime, that is to say, guarantying its long-term availability. However, as pointed out in [8], the use of the proposed AP could lead to out-of-synchronization states at the physical layer, and thus, it may deadlock the communication link.

Consequently, in this chapter, we present the following contributions: (1) the proposed AP solution comparing it with the state of the art (Section 12.2), (2) a recovery protocol for the out-of-sync state (Section 12.3), and (3) we fit both contributions under an example scenario based on the IEEE 802.15.4e amendment to the IEEE 802.15.4-2006 standard (Sections 12.2 and 12.3).

Taking into account the IEEE 802.15.4e amendment, it inherits the physical layer defined in the IEEE 802.15.4-2006 standard while largely modifies the link layer. This amendment is intended to add functionalities to the traditional 15.4 link layer to (1) better support the industrial markets and (2) permit compatibility with modifications being proposed within the Chinese wireless personal area network (WPAN). In particular, it aims to specifically advocate time–frequency scheduling approaches between transmitter and receiver with the twofold benefit of significantly reducing outage and also facilitating

routing. For such a reason, several telecommunication companies, such as Orange/France Telecom and Siemens [9], have decided to further work in this specification, and this is also the reason why we have selected it for our implementation case.

12.1.1 DoS Exhaustion Threats

Malicious attacks targeting network availability in WSNs can be considered as DoS attacks, which attempt to delay, exhaust, block, or corrupt information transmission in order to make network resources unavailable to nodes that need information exchange. DoS attacks against Transmission Control Protocol (TCP)/Internet protocol(IP) have been well studied in the literature regarding attacking types, prevention, and response [10,11]; therefore, in the following, we will discuss potential attacks that specifically target power network availability by exhausting the network and the devices' limited resources. These typical attacks in low-power embedded networks aim to drain the devices' battery and can be launched against the network or against a specific node:

- *Network exhaustion attack*: The attacker can fake a message asking the nodes to continuously retransmit messages to exhaust its energy. As WSNs are typically composed of multihop nodes, if only the destination node is able to check the authenticity of a message, the intermediate nodes could waste energy to send and receive fake packets.
- *Node exhaustion attack*: Every fake message recited from a valid node involves energy consumption; thus, an attacker can exhaust the receiver energy by only sending invalid messages.

A specific example of node exhaustion exploits the two-way request-to-send/clear-to-send (RTS/CTS) handshake that many link-layer protocols use to mitigate the hidden node problem. An attacker can exhaust a node's resources by repeatedly sending RTS messages to elicit CTS responses from a targeted neighbor node; strong link-layer authentication can mitigate these attacks; however, a targeted node receiving the bogus RTS messages still consumes energy and network bandwidth. To counter this, access security mechanisms are typically deployed at link layer. Energy is thus spent in receiving the entire packet and performing the security checks on the entire packet; only then the packet is discarded. Our aim is to improve these techniques to better meet WSN requirements.

12.1.2 State of the Art

Exploring the state of the art regarding exhaustion attacks, the following results are emerged: packets being received can either be intended or not intended for the specific receiver. The non-intended packets can be of nonmalicious but as well as of malicious nature. Non-intended packets are typically received fully, just to be rejected at higher layers due to nonmatching link-layer address, IP address, or security primitives. The energy spent on these to-be-rejected packets is not negligible. Non-intended nonmalicious packets arrive at a fairly regular frequency that depends on the neighborhood cardinality and the neighboring nodes' transmission rates. Non-intended malicious packets arrive rarely but consistently in the case of a DoS attack. Non-intended nonmalicious packets typically originate from transmitting nodes in the one-hop neighborhood of the receiving node. These can be data packets as well as control packets. Typically, energy is spent in receiving the entire packet, performing the security checks on the entire packet, checking on intended destination; only then the packet is discarded. No specific security solutions are known to provide energy savings against these packets. Non-intended malicious packets yield DoS attacks with the aim to jeopardize device and/or link availability. Typical attacks in low-power embedded networks are exhaustion attacks with the aim to drain a device's battery. To counter this, access security mechanisms are typically deployed. Energy is thus spent in receiving the entire packet and performing the security checks on the entire packet; only then the packet is discarded.

Commercial and industrial standards [12] for embedded networks, such as Zigbee, WirelessHART, and ISA 100.11a, are based on the physical and link layers of the IEEE 802.15.4 standard, and they provide a

simple solution against exhaustion attacks: authentication/integrity mechanism at link layer. This mechanism permits to identify an invalid message only after the reception of the whole packet by verifying the message authentication code presented in the last bytes of the packet. With these authentication mechanisms at link layer, the intention of such attacks is not solved because the exhaustion of the victim's limited energy resources is still possible due to the high reception costs. In [13], an alternative technique is presented, which is based on defining specific network topology-based patterns to model normal network traffic flow and to facilitate differentiation between legitimate traffic packets and anomalous attack traffic packets. In this paper, the performance of the proposed attack detection scheme is evaluated in terms of the size of the sensor resource set required for participating in the detection process for achieving a desired level of attack detection accuracy. The results signify the need for distributed pattern recognition for detecting distributed node exhaustion attacks in a timely and accurate manner. This solution seems to be interesting, but numerous drawbacks are identifying comparing it with the proposed PHY-layer protocol in the following presented in this chapter; this solution does not depend on patterns or in detecting the attack by a huge participation of all the nodes, and also it is simple and very fast just because it should not receive the whole packet. In [14], a possible solution to exhaustion is to apply rate limits to the link-layer admission control such that the network ignores excessive requests, thus preventing the energy drain caused by repeated transmissions [15]. A second technique presented in the same paper consists of using time-division multiplexing where each node is allotted a time slot in which it can transmit [15]. This eliminates the need of arbitration for each frame and can solve the indefinite postponement problem in a back-off algorithm. All these solutions do not decrease the reception costs; in others words, they mitigate the problem but do not solve it integrally. In the following, we propose a new and innovative mechanism to protect networks and the nodes that compose them, from node and network exhaustion threats, which does not involve a large additional cost in terms of energy. The goal of this solution is to ensure that the energy necessary to provide the defense against exhaustion attacks is very low compared with the solutions discussed in the preceding.

12.2 PHY-Layer Security Solutions

The proposed PHY-layer security solution relies on the use of an AP of 32 bits that allows discarding a non-intended or a malicious packet just after the AP's reception. The earlier the AP is placed in the packet, the more energy can be saved due to early discards. With such a purpose, the AP must be placed in the packet's PHY header, preferably right after the synchronization preamble [8]. In the following, we detail how the AP is generated and how it is used for lightweight authentication of the received packets.

12.2.1 Authentication Preamble for Normal Traffic Flow

Figure 12.1 depicts the process of generating a valid AP for packets transmitted from node A to B during normal operation, being such nodes one-hop neighbors. With this purpose, A and B use their respective identification tags or IDs and the pairwise keys, shared between them as input data, to compute a hash-based message authentication code (HMAC). The first 32 bits of the output are the initial AP field to be used in the first packet transmitted from A to B. This AP, included at the physical layer of the packet,

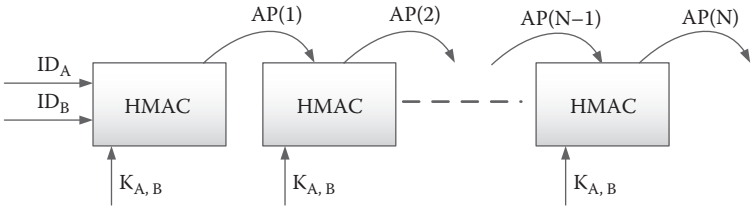


FIGURE 12.1 The AP method used during normal operation.

provides a lightweight authentication or verification, confirming the appropriate origin and destination of the packet. In order to avoid replay attacks, once an AP value is transmitted, it cannot be reused. Recursively, computing the HMAC of the previous AP with the shared pairwise key as in Figure 12.1 guarantees AP's freshness over the time.

The AP verification process is done at the destination and consists in a deterministic method where the receiver checks if the received AP is a potential AP from any of its neighbors. Obviously, aiming at minimizing additional delays due to this authentication, all the potential future APs should be previously computed and stored in a list. If the received AP is in the list, the whole packet is received and processed; otherwise, it is rejected just after the AP reception.

12.2.2 Frame Format Position

In this section, we first briefly present the main specifications of the IEEE 802.15.4e version of the 15.4 standard, and then we settle the proposed AP field in this amendment. In general, the physical layer defined in the 15.4e is the same of the traditional IEEE 802.15.4-2006 standard while the link layer is slightly modified. This amendment thus intends to add functionalities to the traditional 15.4 link layer to (1) better support the industrial markets and (2) permit compatibility with modifications being proposed within the Chinese WPAN. In particular, it aims to specifically advocate time–frequency scheduling approaches between transmitter and receiver with the twofold benefit of significantly reducing outage and also facilitating routing. Taking into account the proposed AP solution, as this mechanism is the short result of a hash function (32 bits or even 20 bits are enough to reach our intent), it can be defined in several standard specifications, included the aforementioned 15.4e, without modifying their functionalities.

As shown in Figure 12.2, we propose to fix the AP after the synchronization preamble symbols because it is the first modifiable place where we can define it without restrict the flexibility of the

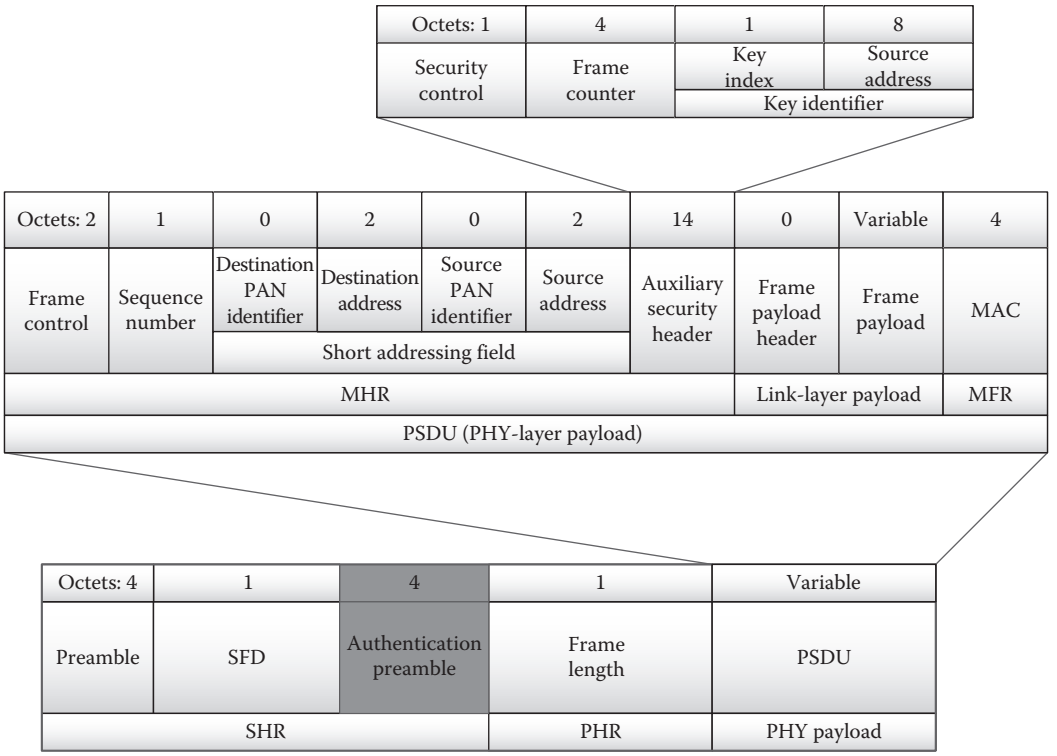


FIGURE 12.2 IEEE 802.15.4e standard packet format adapted to include the AP mechanism at PHY layer.

network; with the synchronization preamble field defined before the AP method, also no-security frames can be implemented. In addition, as the AP mechanism is designed to save as much energy as possible with a fast "on-air" authentication verification test, the numbers of bits that are not received, when this test failed, are directly related to the energy savings. In a few words, the proposed AP method should be received as soon as possible to completely take advantage of its characteristics and thus to save as much energy as possible. However, studies are necessary to fix the proposed method also for the 15.4 k and the 15.4 g standard versions. Our adaptation also considers security suite at link and physical layer.

12.2.3 Authentication Preamble Window

The implementation of the AP solution can allow out-of-sync situations when couples of nodes expect different values for the authenticator at PHY layer: that is, after packet losses. For such a reason, maintaining a window of W future APs for each node will improve the performance of this solution. Notice that without a window of potential future APs, if a packet is lost, the receiver will remain expecting the lost AP value and thus implying the rejection of all future packets sent by this transmitter; that is to say that both ends get in an out-of-sync state.

With an AP window of W, besides potential packet losses, there are W valid APs for a given emitter. Once a valid AP packet from an emitter is received, the receiver updates the window to keep W potential future AP values for this emitter. As a result, synchronization and thus communication between two nodes would be compromised only if W or more consecutive packets are lost. That is to say that, assuming independent packet losses, the probability of "out of sync" between pairs of nodes decreases from p, where p is the probability of losing a packet, to p^W. The optimum AP window size that is able to minimize the probability of reaching an out-of-sync state while minimizing the energy consumption is determinate in [8].

12.2.4 AP Protocol Uses

For the sake of clarity, in this subsection, we describe the AP communication protocol uses, considering the implementation of AP window for typical WSN applications:

1. Every node precalculates the values of the AP window for each neighbor. These specific values are the output results of a special hash function that has the ID addresses and secret keys as input (Figure 12.3).
2. Node "A" builds a packet and sends it with the first AP value contented in the AP window, for the relative emitter–destination nodes (Figure 12.4).
3. The sender updates the AP window (Figure 12.5).
4. Once node "B" has identified the synchronization symbols, it starts receiving the AP field. The AP field should be placed immediately after the synchronization field just to save as much energy as possible.
5. When the node completes the reception of the AP field, it starts the comparison between it and the results stored in all the AP windows.
6. If the result is identified as valid, the node continues the reception; otherwise, the packet is discarded.
7. If the result is correct and the packet also passes the media access control (MAC) layer test, the receiver sends the encrypted acknowledgement (ACK) data packet to the sender, if it is necessary, and updates the physical authentication window with a new value (Figure 12.6).

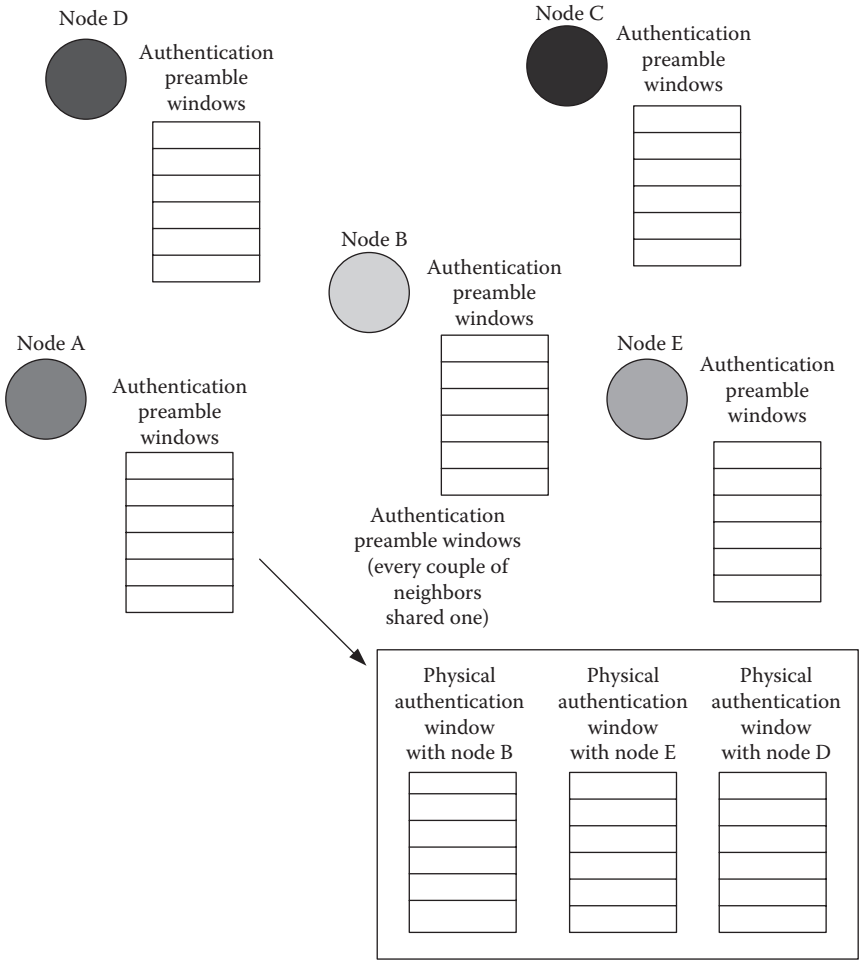


FIGURE 12.3 The initial scenario with all the AP windows.

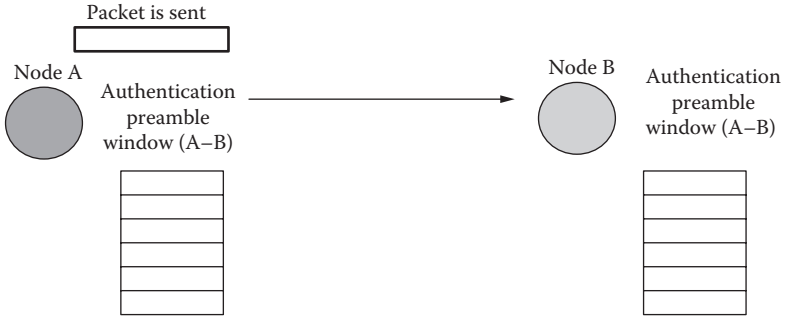


FIGURE 12.4 Node "A" sends a packet to node "B."

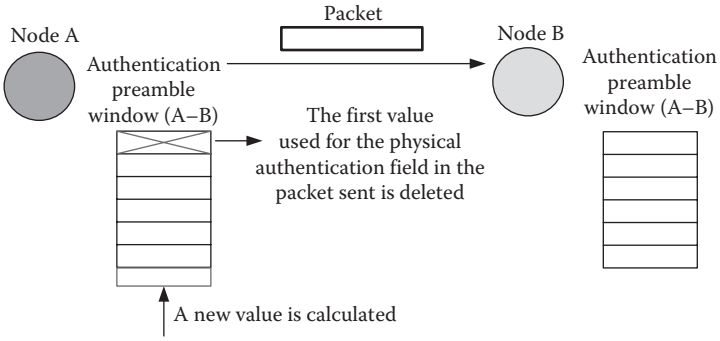


FIGURE 12.5 Updates of the AP window of node "A."

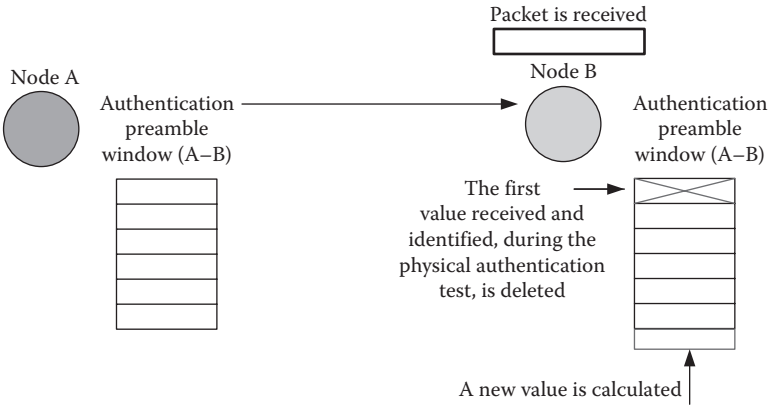


FIGURE 12.6 Updates of the AP window of node "B."

12.2.5 Complete Security Suite

The use of the proposed PHY-layer solution is recommended in harmony with additional security techniques. Indeed, it has to be designed as an adding scheme to improve the traditional security solutions at link layer; it is not proposed to replace them. Considering all the security services recommended for typical WSN security suite, it has to say the AP at physical layer and the MAC at link layer, when a node receives a packet, three different situations are possible:

1. The reception fails after the AP test (Figure 12.7).
2. The reception fails after the MAC test (Figure 12.8).
3. The received packet is accepted (Figure 12.9).

12.2.6 Comparative Table

After a first presentation of the AP, in Table 12.1, we compare it with the literature in order to emphasize which are main differences and benefits. The proposed protocol offers a high level of security with an authentication test already at physical layer. It is perfect for the scenario where availability is a top-priority security service, and efficient communication protocols are important due to constrained devices. Commercial and industrial standards [12], for example, use message authentication code protection at link layer that only mitigate the DoS exhaustion attacks; it is a high-protection process but with low efficiency. All prior schemes have some problems of different nature: security, energy

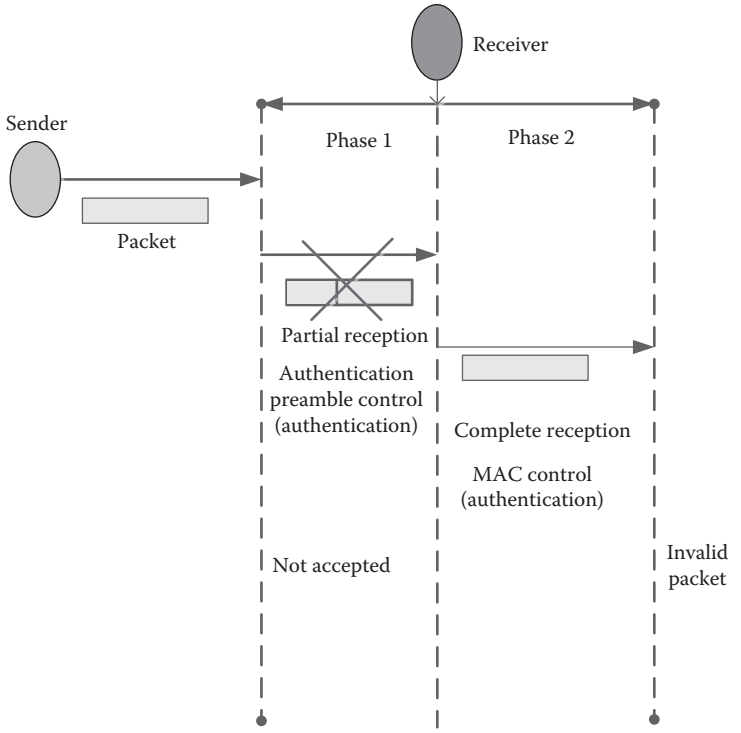


FIGURE 12.7 Invalid packet identified at physical layer.

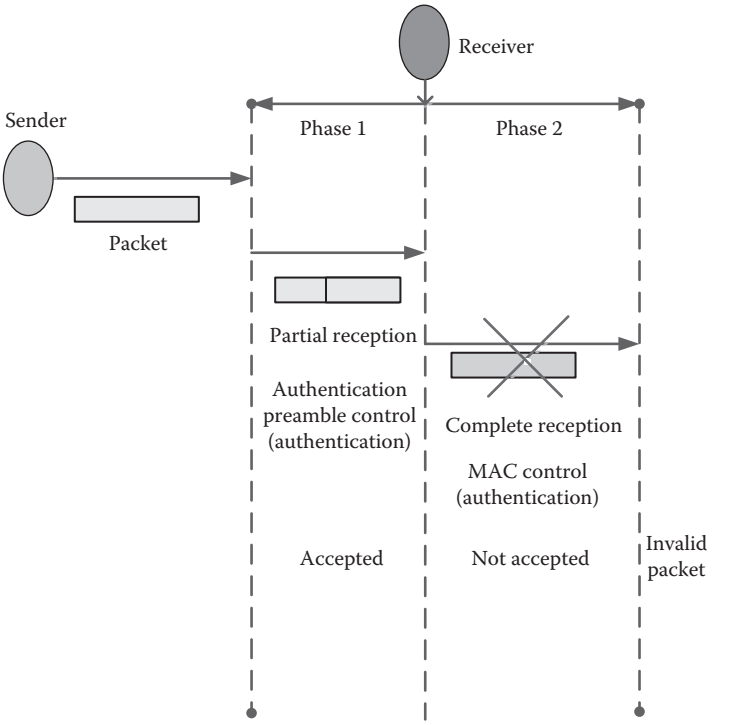


FIGURE 12.8 Invalid packet identified when the packet is completely received.

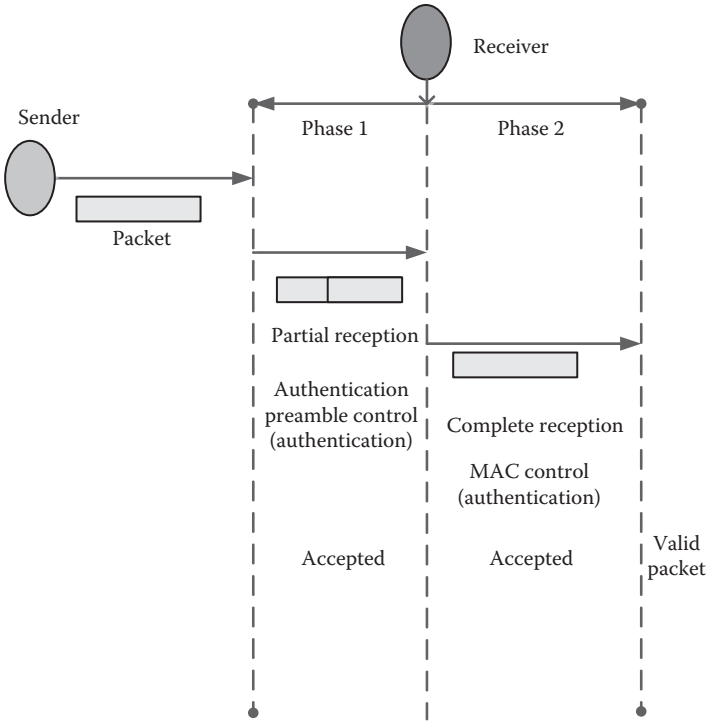


FIGURE 12.9 Valid packet.

TABLE 12.1 Comparing of Previous Secure Availability Schemes with the Proposed One

Scheme	Protection	Efficiency	Concerns
Industrial solution (i.e., Zigbee)	High	Low	Only mitigation
Network topology model	High	Medium	No flexibility
Limiting link-layer rate	Medium	Medium	Limited number of messages
AP	High	High	Nodes synchronization

consumption, flexibility, etc. Our solution is a good trade-off saving energy with a simple lightweight mechanism of authentication, and also it is not jeopardized by traditional attacks. AP limitation is represented by synchronization concerns: nodes have to be constantly synchronized to allow traffic flow and communication control.

12.3 Authentication Preamble for Out-of-Sync

The proposed authentication window solution is able to mitigate the out-of-sync vulnerability. However, since there is still a probability that all packets over this window get corrupted, especially in the presence of low signal-to-noise ratios, nodes can still reach an out-of-sync state. In this situation, a synchronization recovery protocol is to be started by the desynchronized nodes. The synchronization protocol requires the nodes to periodically send at least a keep-alive packet to their neighbors. That is to say that the receiver expects at least a legitimate packet from every neighbor every Δt. Therefore, if after $(W + 1)\Delta t$, a given receiver has not received any legitimate packet from a certain neighbor, it starts the synchronization recovery protocol with that neighbor. In the following, we describe the message protocol and how to physically authenticate these messages given that both ends are in an out-of-sync state.

12.3.1 Authentication Preamble during Recovery

In order to propose a secure and efficient mechanism to secure availability for low-power networks during all the network's lifetime, we focus our efforts to identify which is the best resynchronization method when an "out-of-sync" process is needed.

The first solution proposed consists in defining an out-of-synchronization message with specific synchronization symbols and "in clear" (without security mechanisms). As this solution could lead a possible hole in the security system, it is not recommended for scenario where DoS attacks are possible with reasonable probability. The proposed AP protocol was designed as secure mechanism to provide availability and siding as defense against DoS attacks at PHY layer. If a no-secure message is implemented as mechanism against out-of-sync situation, malicious parties can take advantage of this message and sending it to realize exhaustion attacks. In this case, the valid parties are further deceived, and the valid emitter will start the out-of-synchronization process without a real need. The second solution is more expensive in terms of energy consumption, but it provides a secure method. The potential receiver, besides the AP window described in Section 12.2, should keep and compute a potential out-of-sync AP every Δt.

As shown in Figure 12.10, the generation of this out-of-sync AP between nodes A and B is similar to that described in Section 12.2 for normal traffic AP. This verification field is the output result of a hash function that uses confidential information as input data. In this case, the input data are related to a counter field or time stamp of the current period and a specific pairwise key shared between couples of nodes, $K_{A,B}$.

Since this mechanism is proposed for low-rate networks, the receiver expects a legitimate message every Δt, so it is able to precompute a different and valid out-of-sync AP every fixed period of time. This value will be used as a valid AP field at the physical layer only when a receiver realizes that it is out of sync with respect to a specific transmitter. With the previous reasoning, every network node has to precompute not only an AP window during normal operation (Figure 12.1) but an AP field for the potential packets needed during the synchronization recovery process as well. Regarding the two AP methods, the difference between them is really important for our intent.

The AP method for transmission/reception of traffic flow is a chain of results, and for this reason, infinite different values are possible. The limitation here is that the Nth value of the chain strictly depends on the (N − 1)th one; thus, an out-of-sync situation is possible. However, an attacker would not be able to foresee the valid AP sequence even if AP results are repeated. This is because for every communication, the attacker is only able to sniff the first 32 bits of every hash function output, that is, the transmitted AP field, while the remaining bits are safely stored in the node's memory. When the next AP field has to be computed, the node uses the total amount of bits of the previous output as new input. Summarizing, the correspondence between consecutive values of the AP fields for transmission/reception is very difficult to predict for an attacker due to our method's definition and the hash function characteristics. As for the AP for out-of-sync packets, it is again the result of a hash function, but the values are independent from each other's; they depend on the network time rate, the Δt, and the secret key. The limitation here is represented by the maximum number of possible secure values for this field: that is, with the Δt field

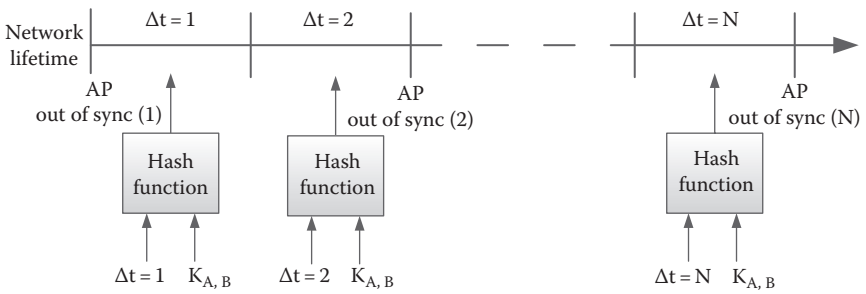


FIGURE 12.10 The AP method used to recover synchronization between couples of devices.

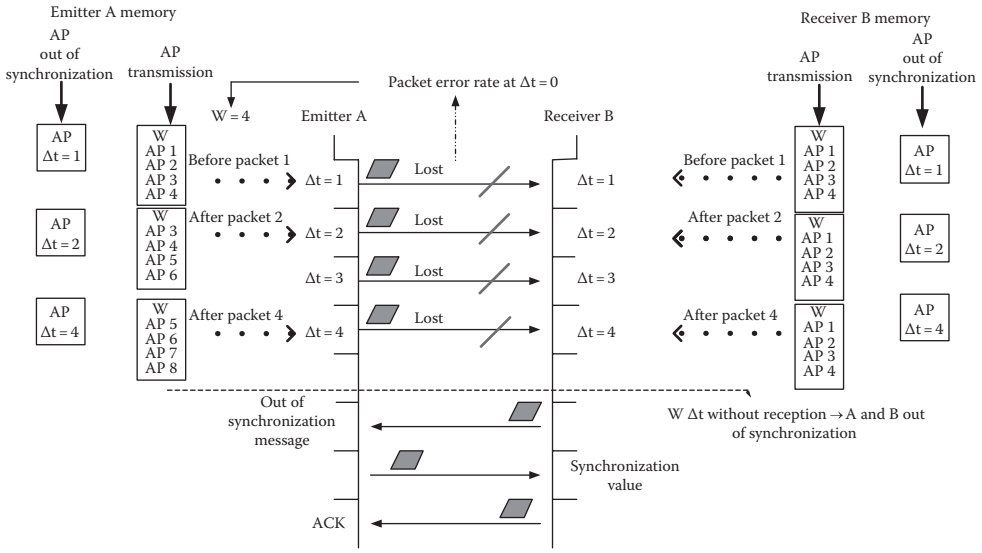


FIGURE 12.11 "Out-of-sync" situation.

of 32 bits, the maximum number of possible secure values of the AP for out-of-sync messages is 2^{32}. Assuming that the out-of-sync packets are not used with high frequency, this drawback does not really compromise the network operations.

Summarizing, the AP for out-of-sync messages, it is both an interesting security scheme and a lightweight method because (1) the attacker is not able to guess the valid key with simple sniffing methods and brute-force attacks and (2) its implementation does not influence the energy and memory consumption of the nodes. Figure 12.11 shows the memory statement during an out-of-sync process. Here, it is possible to observe that while the APs for transmission/reception are disconnected, the APs for out of sync are always synchronized between couples of nodes at every Δt with minimum extra costs in terms of energy and memory.

12.3.2 Out-of-Sync Handshake Model

The out-of-sync three-way handshake is the specific method proposed to recover synchronization between pair of devices that are no longer able to exchange valid packets when the proposed AP protocol is implemented. Similar to the TCP-handshake socket connection, this method is referred to the "Out-of-SYN," "SYN," and "ACK" messages that are defined as follows.

This mechanism is designed so that two devices are able to communicate and to exchange the parameters to recover the first valid AP value of the chain before restarting the typical traffic flow. This handshaking is also designed so that both ends are able to initiate the process whenever they need to. For our study case, we adapt the IEEE 802.15.4e standard packet format to include the AP field at the PHY layer; we like to underline that this insertion does not compromise the normal network functionalities:

- *Out-of-SYN message*: When node A realizes that it is in an out-of-sync state with node B, it sends to B an "Out-of-SYN" packet authenticated with the nondesynchronized AP for the recovery process (Figure 12.12). This frame is a control frame with an empty payload, and thus, it needs just authentication mode at link layer.
- *SYN message*: When B receives the "Out-of-SYN" message, it sends to A the "SYN" message that contains the encrypted synchronization value (Figure 12.13). This value is the first valid element of the AP window for normal operation. Since it provides sensitive data, this frame must be both authenticated and encrypted.

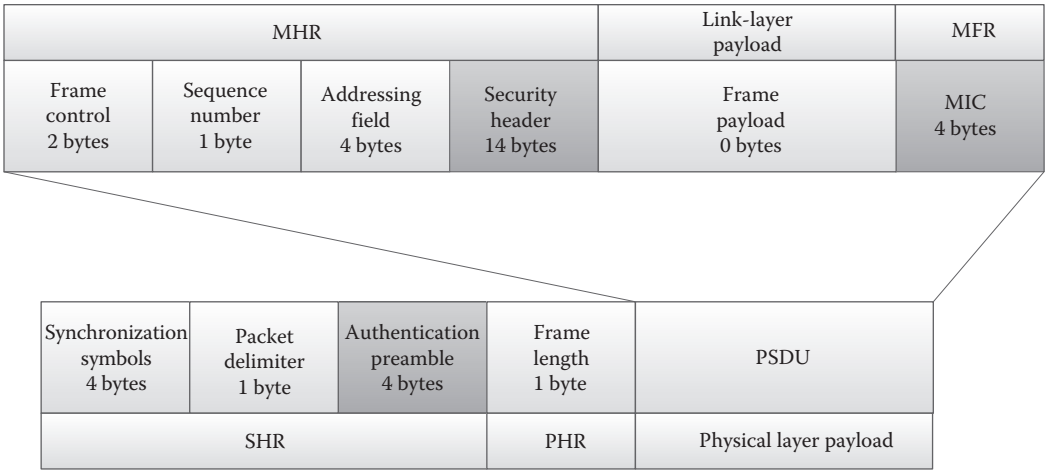


FIGURE 12.12 The out-of-sync message from the receiver to the emitter.

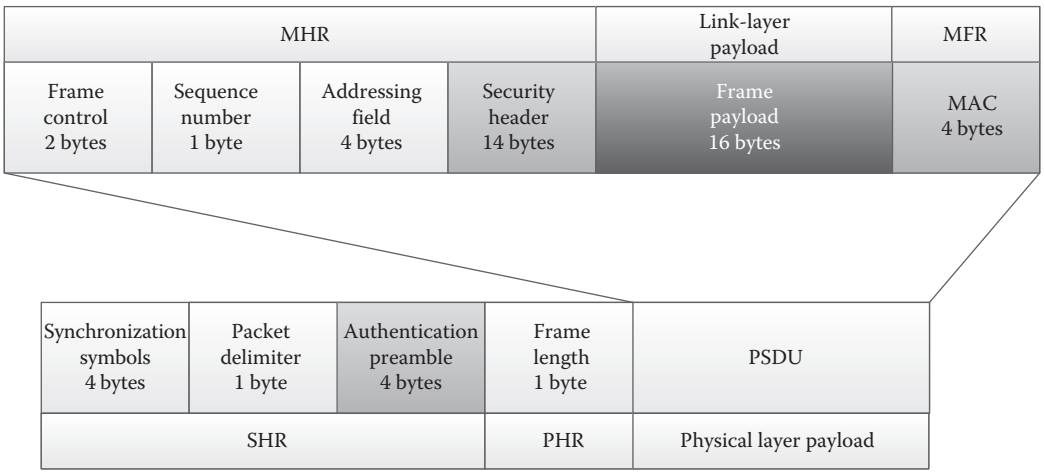


FIGURE 12.13 The sync message from the emitter to the receiver.

- *ACK message*: After the "SYN" message reception, A is able to resynchronize with B. However, as this state needs a confirmation, A has to report such state to B by sending an "ACK" message, in which only authentication mode is required (Figure 12.14). Once B has received the "ACK," it will start again the normal traffic flow from the next Δt.

12.3.3 Out-of-Sync Overview

The main characteristics of the out-of-sync AP are its capacity (1) to be synchronized at every Δt and (2) to be difficult to predict. This is very important to guarantee a secure mechanism able to synchronize a couple of nodes in every situation. In the following, we will analyze the out-of-sync protocol, emphasizing how the system will react to any packets lost:

- *Situation A*: N out-of-sync messages are lost. Though N consecutive "out-of-sync" messages sent from the receiver "B" to the emitter "A" are lost, the same receiver will send new different messages with the out-of-sync AP computed with the corresponding Δt until the reception of a valid "SYN" value. Figure 12.15 shows this situation.

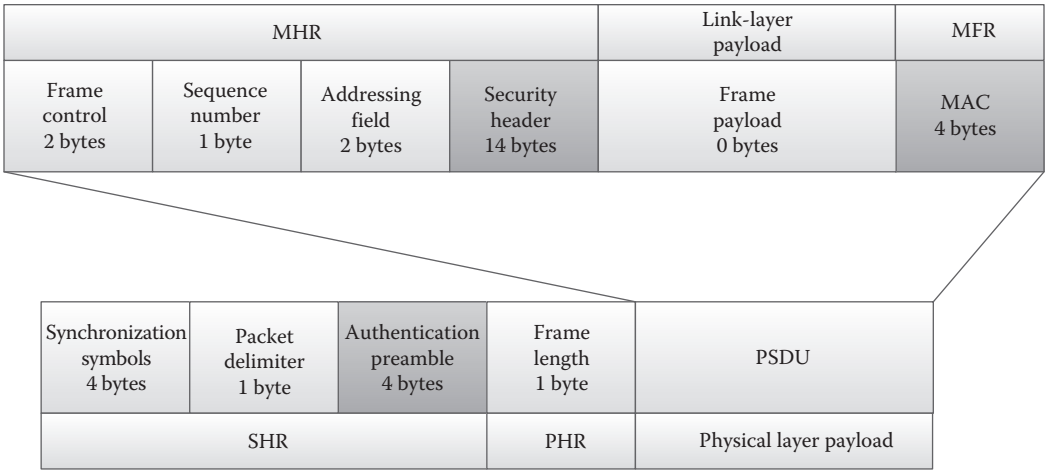


FIGURE 12.14 The confirmation message from the receiver to the emitter.

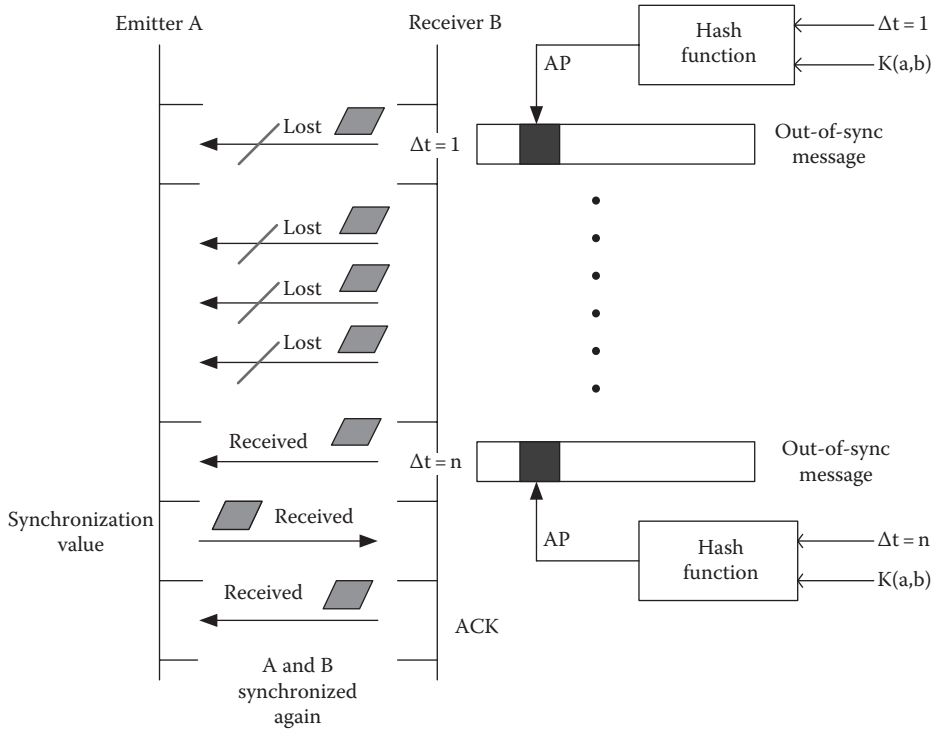


FIGURE 12.15 The transmission of the out-of-sync message from the receiver "B" is concluded only when a valid "SYN" value, emitted from "A," is received.

- *Situation B*: N synchronization messages are lost. Though N "SYN" messages sent from "A" to "B" are lost, the emitter "A" will continue the transmission of this value (with the correspondent out-of-sync AP) until the reception of the ACK from "B." Figure 12.16 shows this situation.
- *Situation C*: N ACKs messages are lost. The receiver "B" is now able to receive a traffic message from the emitter "A," but the same emitter needs a confirmation of this state. In this case, the maximum number of possible ACKs sent from "B" to "A" is limited. Figure 12.17 shows this situation.

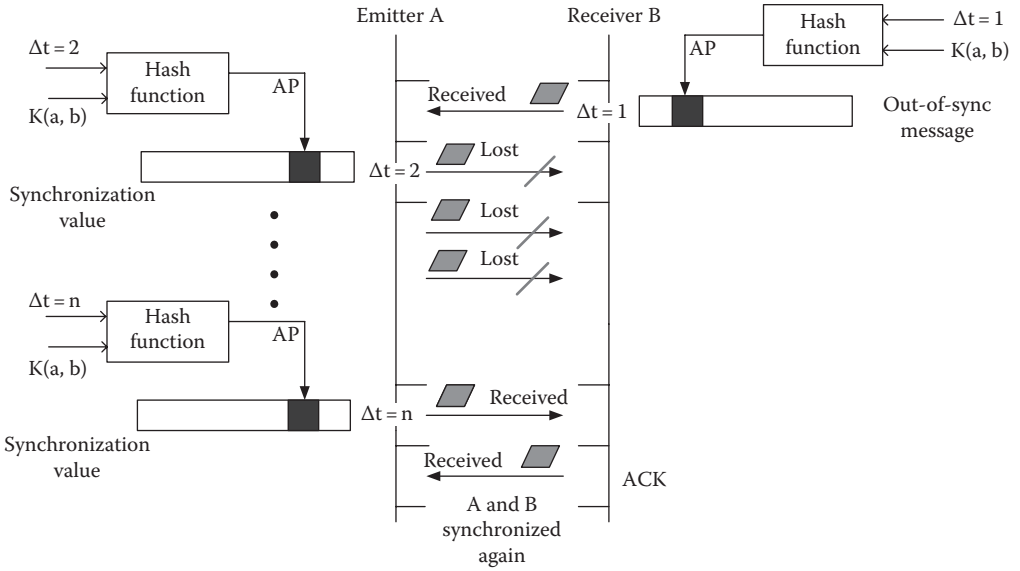


FIGURE 12.16 The emitter "A" receives an out-of-sync packet from the receiver "B." The same emitter sent the "SYN" value until the ACK reception from "B."

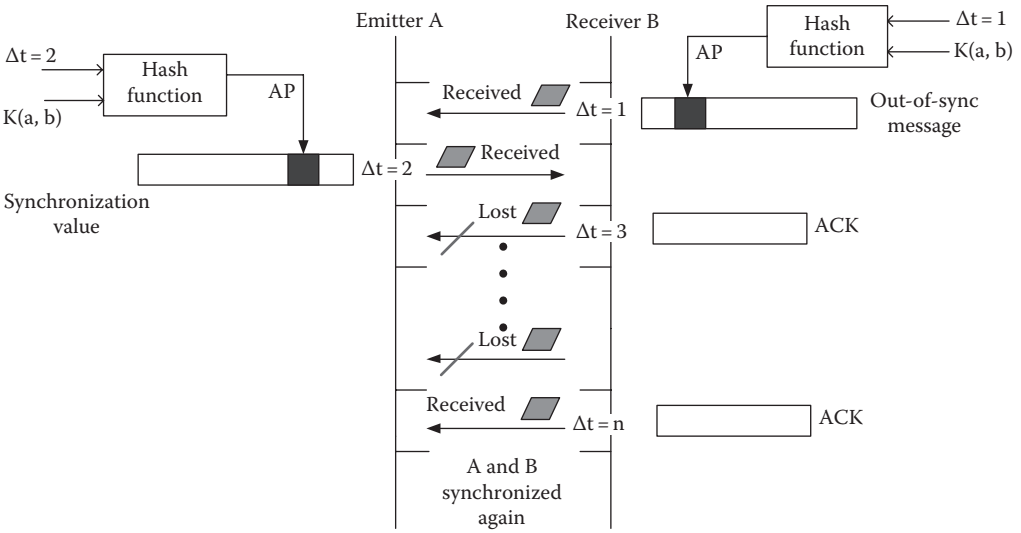


FIGURE 12.17 The receiver "B" receives the "SYN" value from the emitter "A." The same receiver sent the ACK until the reception of a valid traffic packet from the emitter "A."

- *Situation D*: W consecutive normal traffic packets are lost. The maximum possible ACKs sent form "B" to "A" is correspondent to the AP window length (strictly dependent on the packet error rate of the channel). If after WΔt rates the receiver "B" does not receive any valid messages from the emitter "A," it will start again the "out-of-synchronization" process. Instead, if a valid message sent from "A" is received from "B," this means that the meters are synchronized again. This restriction is mandatory to avoid energy wasting when the emitter "A" receives the ACK sent from "B" but all the following W traffic messages sent from "A" to "B" are lost consecutively. Figure 12.18 shows this situation.

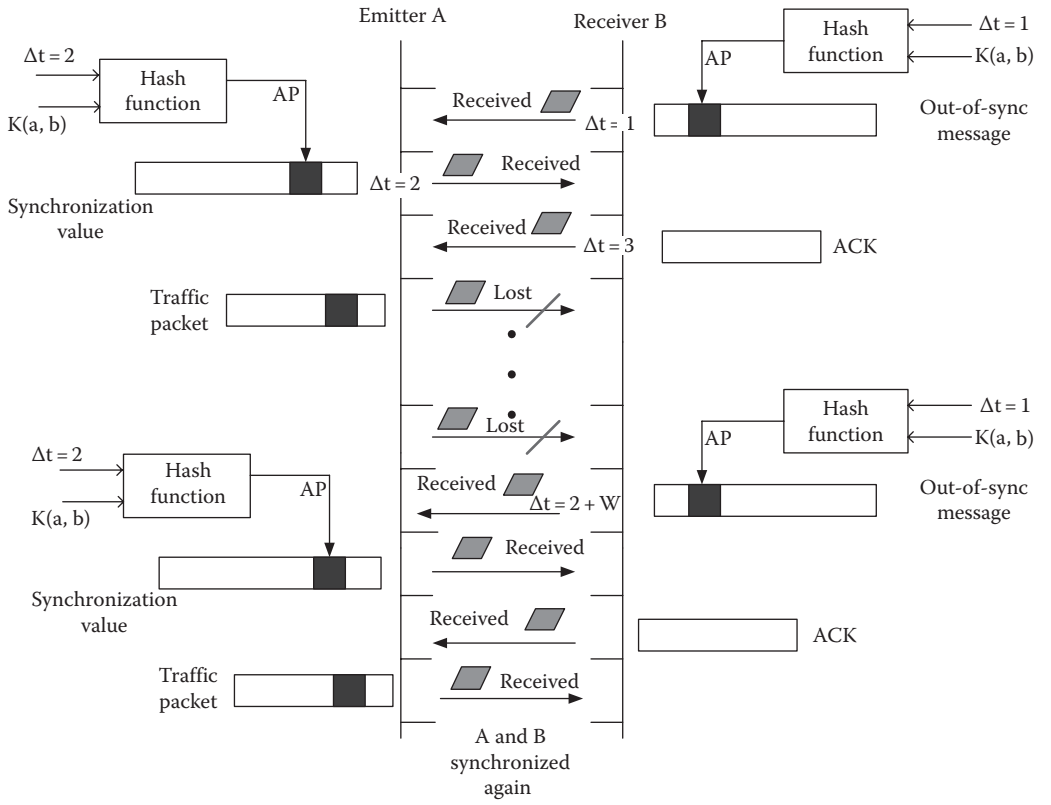


FIGURE 12.18 If "B" is not able to receive a normal traffic packet after WDt, it has to start again the out-of-sync process.

12.4 Summary

In this chapter, we have addressed the problem of availability in WSNs by discussing a new PHY-layer security solution. Indeed, we had presented an authentication verification test at PHY layer to challenge exhaustion DoS attacks. This method is able to reject non-intended packets without the need to their total reception, an ability that is a great value in networks where energy-efficient wireless communication mechanisms must be used. Since the proposed AP method can suffer synchronization problems, we have proposed a synchronization window to improve the protocol performance, and we have introduced a novel synchronization process to deal with poor channel conditions, where the synchronization window might be too short. Finally, since many companies such as Orange or Siemens have shown special interest in the IEEE 802.15.4e amendment to the IEEE 802.15.4-2006 standard, we have described how to adapt the proposed authentication mechanism into such technology, which constitutes another main contribution of this work.

In other words, WSNs have become an integral part of industry monitoring and control applications. Seen as a facilitator of the emerging Internet of Things (IoT), they act as a connectivity paradigm in consumer and industrial applications. Due to the limited capabilities of WSN nodes and the wireless communications settings, DoS attacks constitute one of the major threats and still require further attention from the scientific community. In addition, due to the wireless channel nature, a specific receiver can waste its energy by trying to receive a non-intended to-be-rejected packet sent from neighboring devices. In this chapter, we have detailed a recently introduced security scheme at PHY layer that is shown to yield enormous benefits with regard to such attacks and non-intended exhaustion messages

by making use of an AP. However, since this proposal can suffer from some synchronization issues, we present a novel synchronization protocol that addresses this problem. Moreover, we have shown how to fit the proposed AP mechanism into the IEEE 802.15.4e amendment to the IEEE 802.15.4-2006 standard, which is considered a more energy-efficient protocol than the previous embodiments.

References

1. Hwang, M.-S., Chong, S.-K., and Ou, H.-H. (2011). On the security of an enhanced UMTS authentication and key agreement protocol. *European Transactions on Telecommunications*, 22(3):99–112.
2. Zahariadis, T., Leligou, H.C., Trakadas, P., and Voliotis, S. (2010). Trust management in wireless sensor networks. *European Transactions on Telecommunications*, 21(4):386–395.
3. Lv, C., Li, H., Ma, J., and Zhang, Y. (2012). Vulnerability analysis of elliptic curve cryptography-based RFID authentication protocols. *Transactions on Emerging Telecommunications Technologies*, 23(7):618–624.
4. Labraoui, N., Gueroui, M., and Aliouat, M. (June 2012). Secure DVHop localization scheme against wormhole attacks in wireless sensor networks. *European Transactions on Telecommunications*, 23(4):303–316.
5. Sodagari, S., Attar, A., Leung, V.C., and Bilen, S.G. (August 2012). Combating channel eviction triggering denial-of-service attacks in cognitive radio networks. *Transactions on Emerging Telecommunications Technologies*, 23(5), 454–465.
6. Ishmanov, F., Malik, A.S., and Kim, S.W. (2011). Energy consumption balancing (ECB) issues and mechanisms in wireless sensor networks (WSNs): A comprehensive overview. *European Transactions on Telecommunications*, 22(4):151–167.
7. Wang, L. and Wyglinski, A. (August 23–26, 2011). A combined approach for distinguishing different types of jamming attacks against wireless networks. In *the Proceedings of the Conference on Communications, Computers and Signal Processing Pacific Rim(PacRim) in Victoria (BC), IEEE,* pp. 809–814. doi: 10.1109/PACRIM.2011.6032998.
8. Bartoli, A., Hernandez-Serrano, J., Soriano, M., Dohler, M., Kountouris, A., and Barthel, D. (2011). Secure lossless aggregation over fading and shadowing channels for smart grid M2M networks. *IEEE Transactions on Smart Grid*, 2(4):844–864.
9. Chen, F., German, R., and Dressler, F. (March 29-April 2, 2010). Towards IEEE 802.15.4e: A study of performance aspects. In *2010 Eighth IEEE International Conference on Pervasive Computing and Communications Workshops* (*PERCOM Workshops*), *IEEE* pp. 68–73. doi: 10.1109/PERCOMW.2010.5470605.
10. Schuba, C., Krsul, I., Kuhn, M., Spafford, E., Sundaram, A., and Zamboni, D. (1997). Analysis of a denial of service attack on TCP. In *IEEE Symposium on Security and Privacy, 1997*, Oakland, CA, pp. 208–223.
11. Mirkovic, J. and Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. *SIGCOMM Computer Communication Review*, 34:39–53.
12. Lopez, J., Roman, R., and Alcaraz, C. (2009). Analysis of security threats, requirements, technologies and standards in wireless sensor networks. In Aldini, A., Barthe, G., and Gorrieri, R., eds., *Foundations of Security Analysis and Design V*, *Lecture Notes in Computer Science*, Vol. 5705. Springer, Berlin, Germany, pp. 289–338.
13. Baig, Z. (2011). Pattern recognition for detecting distributed node exhaustion attacks in wireless sensor networks. *Computer Communications*, 34(3):468–484 (Special Issue of *Computer Communications on Information and Future Communication Security*).
14. Wang, Y., Attebury, G., and Ramamurthy, B. (2006). A survey of security issues in wireless sensor networks. *IEEE Communications Surveys Tutorials*, 8(2):2–23.
15. Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114.

13

Impact of Correlated Failures on Wireless Sensor Network Reliability

13.1 Introduction **13**-1
13.2 Graph Theory and Random Field Theory: Graph Theory and the WSN Topology **13**-2
Node Neighborhood and the Neighborhood System • Markov Random Fields • Gibbs Random Fields
13.3 Modeling Correlated Failures Using Markov Random Field Theory **13**-5
MRF-Based Model for Correlated Failures • On the Level of Interaction between Neighboring Nodes • Multilayer Correlated Failures in WSNs
13.4 Sampling Correlated Failures from the Global Gibbs Distribution **13**-7
Gibbs Sampler • Metropolis Sampler
13.5 Results **13**-8
13.6 Summary **13**-14
Acknowledgment **13**-14
References **13**-14

Jorge E. Pezoa
Universidad de Concepción

Silvia Elena Restrepo
Universidad de Concepción

13.1 Introduction

Wireless sensor networks (WSNs) are extensively used in applications such as environmental monitoring and control, military and civilian surveillance, traffic control, health monitoring, and lighting control in commercial spaces (Raghavendra et al., 2004). Nodes forming a WSN have a small size, reduced computing capabilities, small storage capabilities, scarce energy resources, and heterogeneous sensing devices. Nodes exchange data by means of an ad hoc network, whose communication channels are wireless, unreliable, highly dynamic, and delay infested (Raghavendra et al., 2004).

Under normal operating conditions, nodes must form a network and coordinate their actions in order to execute a prescribed mission. To do so, they must remain functioning; however, it is a fact that sensor nodes will fail unavoidably. Node failures yield random topological changes in the network, and depending on the extent of failures, the WSN may get fractioned in disconnected segments. Furthermore, if the WSN happens to be large, both the likelihood of a failure and the number of failures increase due to the complexity of both the topology and the communication protocol (Xu and Wang, 2010).

Unfortunately, node failures appear not only by the normal operation of the network. When a WSN is deployed in open areas or harsh environments, failures may be inflicted by natural disasters, unintended

actions, or malicious attacks. Remarkably, in these WSN environments, there is evidence that node failures are likely to be correlated (Kong and Yeh, 2007; Xu and Wang, 2010). One example of correlated failures in WSNs is the incorrect execution of a power-saving routing strategy. If such strategy is not properly coordinated, communications may be disrupted if all the bridge nodes turn themselves off in order to save batteries. Other examples of correlated failures are battery depletion at several nodes due to the excessive exchange of data, coordinated attacks generated by viruses, and natural disasters, such as floodings and earthquakes, which may physically damage the nodes. Correlated failures are indeed very challenging because the failure at a node may trigger a cascade of failures that percolate among the rest of the network (Xu and Wang, 2010; Rahnamay-Naeini et al., 2011).

Node failures have been extensively studied under the assumptions that they occur at random and independently (Qian et al., 2007; Buttyan et al., 2010; Karkvandi et al., 2011; Petridou et al., 2013). The study of correlated failures in WSNs has been tackled recently by reviewing geographically and logically correlated failures. Sen (2009) analyzed the effect of localized topological failures on WSNs. Sen proposed a new metric, termed as region-based connectivity, to assess the damage inflicted by localized failures. Next, he extended the metric to capture multiple region failures. These ideas introduced the novel concepts of region-disjoint paths and region cuts to the WSN research community. Under the assumption that nodes fail following explicit geometrical shapes, Azimi et al. (2010) modeled geographically correlated failures on WSNs. This type of failures affects simultaneously all the nodes in circle- or rectangle-shaped areas. Liu et al. developed also an algorithm for evaluating the capacity degradation of a WSN, when failures occur at random within a region (Liu et al., 2011). Xu and Wang (2010) characterized theoretically the spread of correlated failures in WSNs. They used percolation theory to derive a model for correlated failures and evaluated the resilience of networks. Their model represents a wide range of failure scenarios, and their results show the spread of correlated failures in large WSNs. Azni et al. (2012) modeled the correlated behavior of nodes based on a semi-Markov process and epidemic theory. They focused on predicting the damage propagation induced by malicious attackers, and their goal was developing countermeasures to secure the network. Finally, in an earlier work, the model for geographically and logically correlated failures to be reviewed in the next pages was presented (Pezoa, 2012). Such model was successfully employed to derive a mission allocation algorithm for WSNs, which maximizes the likelihood of completing a prescribed mission in the presence of correlated failures (Pezoa, 2013). The major conclusion drawn in all these works is that geographically and logically correlated failures impact severely the connectivity and reliability of WSNs.

13.2 Graph Theory and Random Field Theory: Graph Theory and the WSN Topology

Network topologies are commonly abstracted by means of a graph, where nodes are represented by a set of vertices and relationships among the nodes are represented by edges linking pairs of nodes. Here, the topology of an n-node network is described by the connected, undirected graph $G = (V, E)$, where $V = \{1, 2, \ldots, n\}$ is the set of nodes and E is the set of point-to-point wireless communication links between pairs of nodes. Since the set of edges describes only the logical connections between the nodes, the weight function $w{:}E \to (0, \infty)$ has been introduced for associating to each edge e in the graph a positive weight. This weight function allows the modeling of other relations between the nodes, such as geographical distances between them.

For $v \neq u$, a trajectory from the node v to the node u on the graph $G = (V, E)$ is the nonempty, acyclic subgraph of G denoted by $\mathbf{T}_{v,u} = (\mathbf{X}, \mathbf{Y})$ that links nodes $\mathbf{v} = \mathbf{v}_0$ and $\mathbf{u} = \mathbf{v}_k$. Formally, a trajectory is defined as $\mathbf{T}_{v,u} = (\mathbf{X}, \mathbf{Y}) \subset \mathbf{G}$ where $\mathbf{X} = \{\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_k\}$ is the set containing all those nodes visited along the trajectory from node $\mathbf{v}_0$ to node $\mathbf{v}_k$, while $\mathbf{Y} = \{(\mathbf{v}_0, \mathbf{v}_1), (\mathbf{v}_1, \mathbf{v}_2), \ldots, (\mathbf{v}_{k-1}, \mathbf{v}_k)\}$ is the set with the sequence of edges traveled in the trajectory. For convenience, the empty trajectory is defined as $\mathbf{T}_{v,v} = \varnothing$. Let $\mathbf{T}_{v,u}$ denote the collection of all possible trajectories between nodes v and u. With all this mathematical setting at

hand, the logical distance, $d_L(v,u)$, as well as the weighted distance, $d_W(v,u)$, between nodes v and u in G can be defined as follows:

$$d_L(v,u) \triangleq \min_{(X,Y)\in T_{v,u}} |Y|, \tag{13.1}$$

$$d_W(v,u) \triangleq \min_{(X,Y)\in T_{v,u}} \sum_{e\in Y} w(e), \tag{13.2}$$

where $|\cdot|$ is the cardinality of a set.

13.2.1 Node Neighborhood and the Neighborhood System

Nodes can be related to some group of nodes introducing the notion of *neighborhood*. Let $\mathcal{N}_v$ denote the neighborhood of node v. A *neighborhood* is a relationship between nodes satisfying the following properties:

1. A node is not a neighbor of itself; that is, for all $v \in V$, $v \notin \mathcal{N}_v$.
2. A neighboring relationship is reflexive; that is, for all $v, u \in V$, $v \in \mathcal{N}_u \Leftrightarrow u \in \mathcal{N}_v$.

In this chapter, the neighborhood of node v is defined as follows:

$$\mathcal{N}_v \triangleq \{u : d_W(v,u) \le D_{Max} \quad \text{with } (v,u)\in E, v,w \in V\}, \tag{13.3}$$

where the positive number D_{Max} is a parameter that can extend or reduce the size of the neighborhood. Note that if $D_{Max} = 1$ and $d_W(v,u)$ is replaced by $d_L(v,u)$ in (13.3), then the definition of neighborhood reduces to the traditional definition of the nearest neighbor on a graph; that is, $\mathcal{N}_v = \{u : (v,u)\in E \text{ with } v,u\in V\}$. For mathematical consistency, the notion of closed neighborhood is introduced as $\bar{\mathcal{N}}_v \triangleq \mathcal{N}_v \cup \{v\}$. Also, the *neighborhood system*, $\mathbf{N}$, induced by the graph G is defined as the collection of all neighborhoods, that is, $\mathbf{N} = \{\mathcal{N}_v, v\in V\}$.

Note that more sophisticated definitions for the neighborhood can be presented. For instance, if the distance between nodes v and u is defined as

$$d(v,u) = \begin{cases} 0, & v = u \\ d_L(v,u)d_W(v,u), & d_W(v,u) \le d_{Max} \\ \rho d_{Max}, & d_W(v,u) > d_{Max} \wedge d_L(v,u) = 1 \\ (\rho+1)d_{Max}, & \text{otherwise} \end{cases}, \tag{13.4}$$

where ρ is the (logical) diameter* of the graph G, then the neighborhood defined as

$$\mathcal{N}_v \triangleq \{u : d(v,u) \le \rho d_{Max} \quad \text{with } (v,u)\in E, v,w \in V\} \tag{13.5}$$

defines that nodes v and u are neighbors if they are within a distance d_{Max} between each other or if they are separated by more than d_{Max} units of distance, but there is an edge linking them.

* The diameter of a graph is formally defined as $\rho \triangleq \max_{(v,u)\in V\times V} d_L(v,u)$, that is, ρ is the longest of the shortest paths between any pair of nodes.

13.2.2 Markov Random Fields

The failed or working state of a sensor node in the WSN is denoted by a value from the binary set $\Lambda = \{0, 1\}$. More precisely, the value "1" indicates that the node has failed, while the value "0" indicates otherwise. The *configuration space*, Λ^V, is the set of all possible states for the nodes in the WSN. In addition, a positive probability measure has been assigned to every configuration in the configuration space. With these definitions at hand, a random field on the set of nodes V, with phases in Λ, is defined as the collection of random variables $\mathbf{X} = \{X(v), v \in V\}$, where $X(v)$ takes values on Λ. Thus, a random field can be regarded as a random variable taking values on the configuration space, where a particular value of a configuration will be denoted as $\mathbf{x} = \{x(v), v \in V\}$ and also as $\mathbf{x} = (x(1)x(2)\ldots x(n))$, with $x(v) \in \Lambda$ for all $v \in V$. Further, the restriction of x to the set A, for $A \subset V$, is defined as $\mathbf{x}(A) = \{x(v), v \in A\}$. Next, a Markov random field (MRF) is defined on the neighborhood system induced by the topology of the WSN.

Definition 13.1

X is an MRF with respect to the neighborhood system $\mathbf{N}$, induced by the graph $G = (V, E)$, if for all $v \in V$, the random variables $X(v)$ and $\mathbf{X}(V \setminus \bar{\mathcal{N}}_v)$ are independent given $\mathbf{X}(\mathcal{N}_v)$. More precisely, for all $v \in V$ and $\mathbf{x} \in \Lambda^V$,

$$P\{X(v), X(V \setminus \bar{\mathcal{N}}_v) \mid X(\mathcal{N}_v)\} = P\{X(v), \mid X(\mathcal{N}_v)\} P\{X(V \setminus \bar{\mathcal{N}}_v) \mid X(\mathcal{N}_v)\}. \tag{13.6}$$

This equation is equivalent to

$$P\{X(v) = x(v) \mid X(V \setminus \mathcal{N}_v) = x(V \setminus \mathcal{N}_v)\} = P\{X(v) = x(v) \mid X(\mathcal{N}_v) = x(\mathcal{N}_v)\}, \tag{13.7}$$

which clearly shows the Markovian property of the field.

In other words, this definition of an MRF postulates that the likelihood of node v of being failed or not depends on what occurs (i.e., if nodes have failed or not) on the neighborhood of v, but not on anything beyond such local description. In addition, a local characterization, $\mathbf{X}$, of the MRF at the node v is a function defined from Λ^V to [0,1] as follows:

$$\pi\{x(v) \mid x(\mathcal{N}_v)\} = P\{X(v) = x(v) \mid X(\mathcal{N}_v) = x(\mathcal{N}_v)\}. \tag{13.8}$$

From this, the local specification for the MRF is the collection $\{\pi^V, v \in V\}$.

13.2.3 Gibbs Random Fields

According to the Hammersley–Clifford theorem, there is an equivalence between MRFs and the Gibbs random fields. This equivalence occurs only for those Gibbs random fields that can be expressed in terms of the so-called Gibbs potentials (Kinderman and Snell, 1980; Bremaud, 2001). To exploit this equivalence, first, two definitions must be introduced.

Definition 13.2

A Gibbs field is random field on the configuration space Λ^V following the Gibbs distribution defined as follows:

$$\pi_X(x) = \frac{1}{Z_T} \exp\left(-\frac{\mathcal{E}(x)}{T}\right), \tag{13.9}$$

where Z_T is a normalizing constant (also known as the partition function), T is a constant termed as the temperature, and $\mathcal{E}(\mathbf{x})$ is termed as the energy function.

Definition 13.3

A Gibbs potential defined on Λ^V with respect to the neighborhood system N, induced by the graph $G = (V, E)$, is a function $V_C : \Lambda^V \to \mathbb{R} \cup \{\infty\}$ such that

1. $V_C = 0$ if C is not a clique*
2. For all $\mathbf{x}, \mathbf{x}' \in \Lambda^V$ and $v \in V$, $\mathbf{x}(v) = \mathbf{x}'(v) \Rightarrow V_C(\mathbf{x}) = V_C(\mathbf{x}')$

The Hammersley–Clifford theorem states that if the energy function can be written as the sum of Gibbs potentials, then an MRF is equivalent to a Gibbs field (Kinderman and Snell, 1980; Bremaud, 2001). In symbols, if $\mathcal{E}(\mathbf{x}) = \sum_{C \in \mathcal{C}} V_C(\mathbf{x})$, where C denotes the collection of all cliques. From this equivalence relation, it can be concluded that the global distribution of an MRF takes the simple form in (13.9).

13.3 Modeling Correlated Failures Using Markov Random Field Theory

The model for spatially correlated failures reviewed in this chapter exploits the aforementioned equivalence between MRFs and Gibbs fields (Pezoa, 2012). By defining two types of cliques, an energy function that describes local interactions between failed and working sensor nodes can be introduced.

13.3.1 MRF-Based Model for Correlated Failures

The following types of cliques have been defined: (1) cliques with one element, $\{v\}$ with $v \in V$, and (2) cliques with two elements $\{(v, u)\} = \{v, u : (v, u) \in E\}$. In addition, for convenience, consider the following energy function:

$$\mathcal{E}(\mathbf{x}) = \sum_{v \in V} r_v x(v) - \sum_{v \in V} \sum_{u \in \mathcal{N}_v} s_{v,u} x(v) x(u), \tag{13.10}$$

where

r_v is a nonnegative parameter modeling the reliability of node v to a single failure
$s_{v,u}$ is a nonnegative parameter modeling the level of interaction between nodes u and v (when a failure affects node v)
$\mathcal{N}_v$ is defined as in (13.3) with $d_W(v,u) = d_L(v,u)$ and $d_{Max} = 1$

Note that the closer r_v to zero, the more vulnerable is the node v to a single failure.

Equation 13.10 can be written using matrix–vector notation as $\mathcal{E}(\mathbf{x}) = \mathbf{x}^T \mathbf{r} - \mathbf{x}^T \mathbf{S} \mathbf{x}$, where $\mathbf{x} = (x(1) \ldots x(n))^T$, $\mathbf{r} = (r_1, \ldots, r_n)^T$, and $\mathbf{S} = ((s_{v,u}))$ is an n-by-n, not necessarily symmetric matrix with the convention that $s_{v,u} = 0$ if nodes u and v are not neighbors. Thus, the Gibbs distribution associated to this potential function can be written as follows:

$$\pi_X(\mathbf{x}) = \frac{1}{Z_T} \exp\left(-\frac{\mathbf{x}^T \mathbf{r} - \mathbf{x}^T \mathbf{S} \mathbf{x}}{T} \right). \tag{13.11}$$

* Formally, a subset C of V with one or more elements is called a clique of the graph $G = (V, E)$ if and only if any two distinct members of C are mutual neighbors.

In addition, from the equivalence between MRFs and Gibbs fields, the local specification for a node v can be obtained from (13.11) (Kinderman and Snell, 1980; Bremaud, 2001). In particular, conditional on the neighborhood of the node v, the local specifications for such node, under the working condition $x(v) = 0$ and under the failed condition $x(v) = 1$, are the following:

$$p_0 = \pi\left(0 \mid x(\mathcal{N}_v)\right) = \frac{1}{1 + \exp\left(-T^{-1}\left(r_v - \sum_{u \in \mathcal{N}_v} s_{v,u} x(u)\right)\right)}, \tag{13.12}$$

$$p_1 = \pi\left(1 \mid x(\mathcal{N}_v)\right) = \frac{\exp\left(-T^{-1}\left(r_v - \sum_{u \in \mathcal{N}_v} s_{v,u} x(u)\right)\right)}{1 + \exp\left(-T^{-1}\left(r_v - \sum_{u \in \mathcal{N}_v} s_{v,u} x(u)\right)\right)} = 1 - p_0. \tag{13.13}$$

13.3.2 On the Level of Interaction between Neighboring Nodes

To understand the idea behind the parameter associated to the level of interaction between neighboring nodes, consider the following requirements. First, note the aim of the model for correlated failures is to capture the effect induced by a local failure on the rest of the network. Second, note that different node failures may inflict different amounts of damage to the nodes depending on the geographical distance between the nodes and the center of event inducing the failure. Third, note also the failure of a node in the WSN will induce failures in other functioning nodes as a result of the inability of the latter nodes to exchange data and information with a failed node. In summary, it is of interest to model situations where the geographic and logical proximity of a node to a failure event increases the likelihood of failure on such node and its neighboring nodes.

The aforementioned effects can be modeled using the level of interaction parameter. For the Gibbs distribution (13.11) dictating the probability law of the MRF, the parameter $s_{v,u}$ is defined as

$$s_{v,u} = \begin{cases} \dfrac{D_{Max}}{d_W(v,u)} + s^L_{v,u}, & u \in \mathcal{N}_v \wedge d_W(v,u) \le D_{Max} \\ s^L_{v,u}, & u \in \mathcal{N}_v \wedge d_W(v,u) > D_{Max} \\ 0, & u \notin \mathcal{N}_v \end{cases}, \tag{13.14}$$

where $s^L_{v,u}$ is a nonnegative, inhomogeneous parameter modeling the logical level of interaction between nodes v and u.

It must be commented that (1) the level of interaction parameter $s_{v,u}$ is composed of two terms; the first one is related to the geographical connection between nodes (the ratio $D_{Max}/d_W(v,u)$), while the second is related to the logical connection between nodes (the term $s^L_{v,u}$); (2) the closer the nodes v and u, the stronger their interaction; (3) for all those neighboring nodes located more than D_{Max} units away from v, the strength of interaction is dictated solely by the logical connection between v and u and is independent of the geographic distance separating the nodes; and (4) the D_{Max} parameter can be thought of as the maximum radius of damage that a particular type of failure event may inflict on the network, when such event is centered at node v. Finally, note that the case of independent (totally random) failures is modeled when the level of interaction parameters is all set to zero.

13.3.3 Multilayer Correlated Failures in WSNs

Suppose now that it is of interest to assess the effect of correlated failures in multilayer WSNs as those in Liu et al. (2010), Shwe et al. (2010), and Yoon et al. (2012), where different environmental

variables are sensed by subsets of nodes. The model presented in the previous section can be extended to regard such cases as follows.

Let $\tilde{G}_{i,i-1}$ be the undirected graph modeling the interlayer topology for every two consecutive layers i and $i-1$ in a multilayer WSN. Then the graph $\tilde{G}_{i,i-1}$ can be used in the MRF model presented in the previous section to represent the neighborhood system of the interlayer topology. More precisely, consider the ith and $(i-1)$th layers of a multilayer topology and assume that their topologies are modeled, respectively, by the undirected graphs $G_i = (V_i, E_i)$ and $G_{i-1} = (V_{i-1}, E_{i-1})$, which induce the neighborhood systems N_i and N_{i-1}. The graph $G_{i,i-1} = (V_i \cup V_{i-1}, E_{i,i-1})$ models the interlayer topology consisting in all the nodes at both layers, and $E_{i,i-1} \subseteq V_i \times V_{i-1}$ represents cross-links connecting nodes in layers i and $i-1$. Thus, the graph $\tilde{G}_{i,i-1}$ is defined as $\tilde{G}_{i,i-1} = G_i \cup G_{i-1} \cup G_{i,i-1}$ and models the multilayer topology. Let $\tilde{X}_{i,i-1}$ be an MRF associated to the graph $\tilde{G}_{i,i-1}$ that induces the neighborhood system $N_{i,i-1}$. For a particular configuration, say $\tilde{x}_{i,i-1} \in \{0,1\}^{V_i \cup V_{i-1}}$, let the notation $\tilde{x}_{i,i-1} = (\tilde{x}_i \tilde{x}_{i-1})$ represent the configurations of the layers i and $i-1$, respectively, while the symbols $\tilde{x}_i \in \{0,1\}^{V_i}$ and $\tilde{x}_{i-1} \in \{0,1\}^{V_{i-1}}$. Hence, the Gibbs distribution for the graph $\tilde{G}_{i,i-1}$ is

$$\pi_{\tilde{X}_{i,i-1}|X_{i-1}}(\tilde{x}_{i,i-1} \,|\, x_{i-1}) = \frac{1}{Z_{T_{i,i-1}}} \exp\left(-\frac{\mathcal{E}(\tilde{x}_{i,i-1})}{T_{i,i-1}}\right), \tag{13.15}$$

where $\mathcal{E}(\tilde{x}_{i,i-1})$ is the energy function and $T_{i,i-1}$ is the temperature assigned to the graph $\tilde{G}_{i,i-1}$.

13.4 Sampling Correlated Failures from the Global Gibbs Distribution

To simulate correlated failures in WSNs, one should draw samples directly from (13.11) using traditional sampling methods, such as the inverse cumulative distribution function (CDF) or the acceptance/rejection methods. In practice, however, the normalizing constant Z_T can be impossible to compute, making the use of such methods unfeasible. Moreover, since in the MRF-based model for correlated failures there are 2^n different configurations for an n-node WSN, the aforementioned sampling methods are computationally expensive due to the large number of comparisons to be performed.

To overcome these practical problems, the Gibbs and Metropolis samplers are presented here. These two samplers can be efficiently used to generate realizations of correlated failures following (13.11). These sampling algorithms rely on the construction of a field-valued, homogeneous Markov chain (MC), whose stationary distribution is the Gibbs distribution proposed in (13.12). The general idea of such algorithms is to generate a realization for an MC that, after a large number of iterations, approximates the desired global distribution. The convergence of the MC to the Gibbs distribution is guaranteed by the fundamental result of convergence to a steady state in MCs (Bremaud, 2001). In a more practical matter, it is interesting to note that the MC required to construct the sampler is obtained directly from the local specifications (13.12) and (13.13) of the Gibbs field (Bremaud, 2001).

13.4.1 Gibbs Sampler

The operations to be conducted by the Gibbs sampler are the following:

1. *Initialization*: set x to any random value in Λ^V.
2. Select a node, say v, at random.
3. Draw a random number, say r, following a uniform distribution in [0, 1].
4. Compute p_0. If $r < p_0$, set $x(v) = 0$, otherwise set $x(v) = 1$.
5. Repeat from step 2 a large number of times to obtain in x a sample of correlated failures following (13.11).

13.4.2 Metropolis Sampler

The operations to be conducted by the Metropolis sampler are the following:

1. *Initialization*: set x to any random value in Λ^V.
2. Generate $\mathbf{x}'$ by randomly changing a single or a few values from x.
3. Draw a random number, say r, following a uniform distribution in [0, 1].
4. Compute $p = \min\left\{1, \dfrac{\pi(\mathbf{x}')}{\pi(\mathbf{x})}\right\}$. If $r < p$, set $\mathbf{x} = \mathbf{x}'$, otherwise do not update x.
5. Repeat from step 2 a large number of times to obtain in xa sample of correlated failures following (13.11).

It has been observed that both algorithms produce proper samples of the global Gibbs distribution but, in most cases, the Metropolis sampler turns out to be faster than the Gibbs sampler (Li, 2009).

13.5 Results

In this section, results showing the ability of the MRF-based model for generating correlated node failures in a WSN are presented. To do so, the WSN deployed in the Intel Berkeley Research lab was used as an example (Madden, 2004). This WSN is composed of 54 Mica2Dot sensor nodes (Willow Technologies, 2013) and a sink node; however, 2 out of the 55 nodes were eliminated from the network. The sink node was eliminated due to the lack of information about of its location in the laboratory. The sensor node labeled as "5" was eliminated because, from the aggregate connectivity data* available at (Madden, 2004), it remained disconnected from the rest of the WSN. Thus, after relabeling all the nodes, the WSN with 53 sensor nodes depicted in Figure 13.1 was used in the examples presented here. For illustration purposes, a correlated failure sample is shown in Figure 13.1. In the figure, failed sensor nodes appear in black and functioning nodes appear in gray.

In the statistical analysis conducted here, two different topologies, derived from the 53-node WSN in Figure 13.1, were defined. The first topology was defined assuming that wireless channels (or links) are available to the sensor nodes only when their estimated probability of successfully sending a message is greater than or equal to 0.1. The resulting topology is shown in Figure 13.2 and is termed here as *Topology 1*. The second topology was defined assuming that a wireless channel is available only when such probability is larger than or equal to 0.3. The resulting topology is shown in Figure 13.3 and is termed as *Topology 2*. Note that Topology 1 resulted in a more connected topology than Topology 2. In fact, the average degree of Topology 1 is 19.73 [nodes] and its Fiedler eigenvalue† is 3.01, while the average degree of Topology 2 is 8.79 [nodes] with a Fiedler eigenvalue equal to 9.79.

The Gibbs sampler described in Section 13.4 was used to draw samples of correlated node failures. Unless indicated otherwise, the sampler employs the following default set of parameters: (1) the robustness vector is $\mathbf{r} = (4, 4, \ldots, 4)^T$, (2) the logical level of interaction parameter $s_{v,u}^L$ is equal to the estimated probability of sending a message from the node v to the node u, (3) the temperature parameter is $T = 2$, (4) the maximal separation distance between any pair of nodes is $D_{max} = 5$ (m), (5) the neighborhood system is the nearest neighbor, and (6) the Gibbs sampler iterates 5000 times before drawing a sample of correlated node failures. To assess the correlation in the failure patterns, covariance matrices were constructed for each topology using 1000 failure patterns. Next, t-tests for the hypothesis of no correlation, with a confidence interval of 99%, were conducted for each off-diagonal element of the covariance matrices.

* The aggregate connectivity data are the estimated probabilities of successfully sending a message from one sensor node to any other node in the WSN.

† The Fiedler eigenvalue (or algebraic connectivity) is a measure of graph connectivity associated with the second smallest eigenvalue of graph's Laplacian matrix.

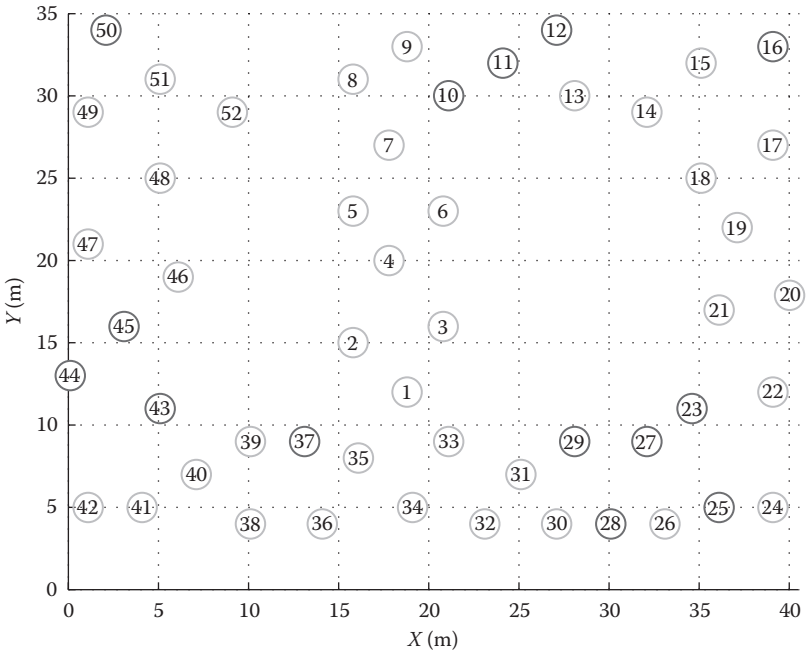


FIGURE 13.1 The 53-node WSN topology used as a test network on the *X*–*Y* plane. This topology was derived from the Intel Berkeley Research lab WSN (Madden, 2004). Sensor nodes in black have failed, depicting a pattern of correlated failures.

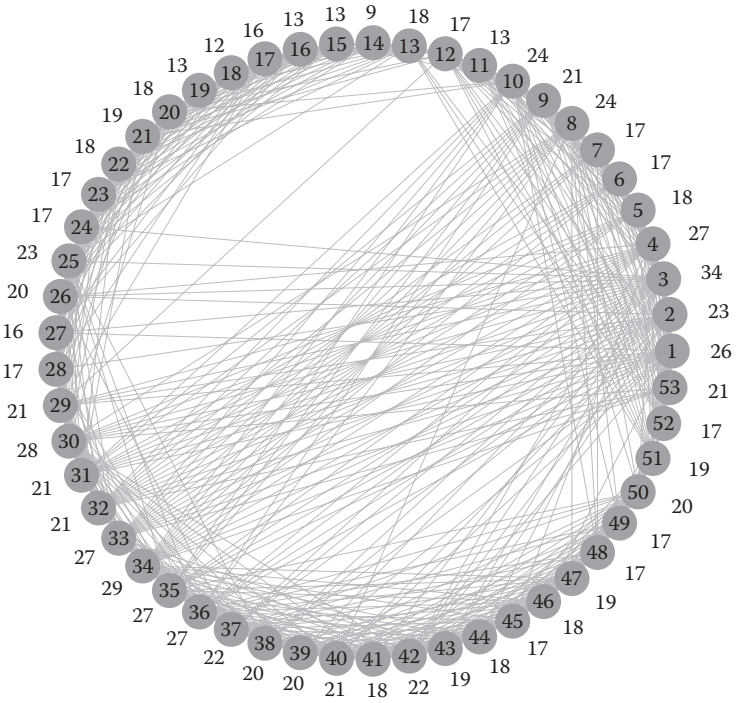


FIGURE 13.2 The WSN topology obtained when the probability of sending a message is larger than or equal to 0.1. The values outside of the circle indicate the degree of each sensor node.

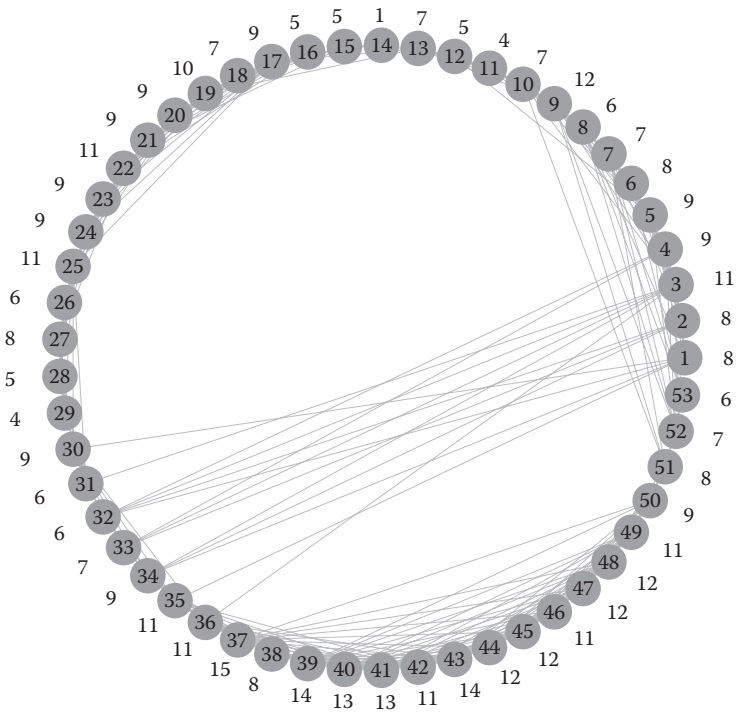


FIGURE 13.3 The WSN topology obtained when the probability of sending a message is larger than or equal to 0.3. The values outside of the circle indicate the degree of each sensor node.

The effect of network connectivity on correlated node failures is analyzed first. To do so, Topologies 1 and 2 were tested, specifying the same parameters in the MRF-based model for correlated failures. The results of the correlation tests, for Topologies 1 and 2, are shown in Figures 13.4 and 13.5, respectively. In the figures, the evidence of a correlated failure between nodes v and u is denoted as either a light gray or a dark gray square at the location. In particular, light gray squares indicate evidence of correlation and also mean that there is a link connecting nodes v and u. Also, blue squares denote the existence of links between nodes and no evidence of correlated failures. Results in Figures 13.4 and 13.5 clearly show that more connected topologies exhibit larger degrees of correlated failures. Moreover, from the number and location of the light gray squares, it can be concluded that the topology in fact induces patterns of correlated failures among the sensor nodes.

Next, a comparison showing the effect of independent and correlated node failures on Topologies 1 and 2 is presented. To perform the comparison, the robustness vector was varied homogeneously for all the sensor nodes in the range [0, 10]. Results are shown in Figure 13.6, and three major conclusions can be drawn. First, for a fixed value in the robustness parameter, the average number of failed nodes is smaller in the case of independent failures as compared to the case of correlated failures. Second, the model for independent failures yields the same average number of failures, regardless of the topology and its connectivity. Third, Figure 13.6 shows also that the average number of correlated failures is larger in Topology 1 than in Topology 2. This effect is expected because more connected topologies produce more failed nodes due to the percolation effect induced by the MRF-based model for correlated failures.

The third type of analysis conducted is assessing the effect of model parameters on the average number of failed nodes. For brevity, the analysis is presented only for Topology 1. Figure 13.7 depicts the average number of failed nodes as a function of both the homogeneous robustness parameter and the temperature. From the figure, it can be observed that the average number of failed nodes decreases as the robustness and temperature parameters increase. This behavior is expected due to the Gibbs

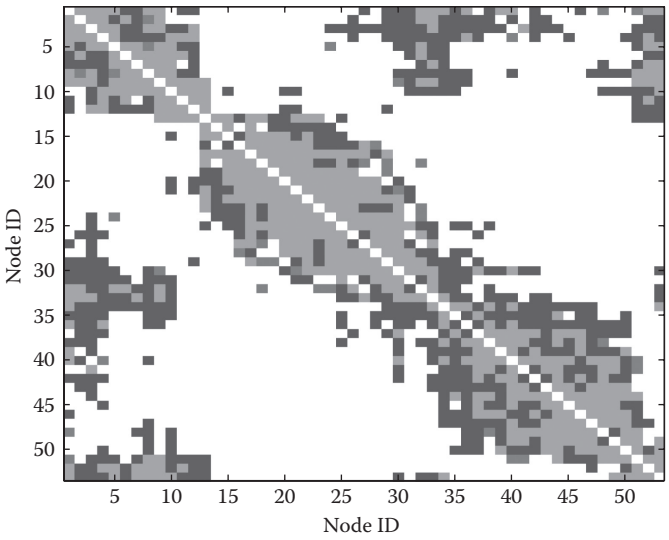


FIGURE 13.4 *T*-test conducted on the correlation matrix of Topology 1. Dark gray squares show evidence of correlation between nodes. Black squares indicate the existence of a link between nodes. Light gray squares indicate both evidence of correlation and the existence of a link between nodes.

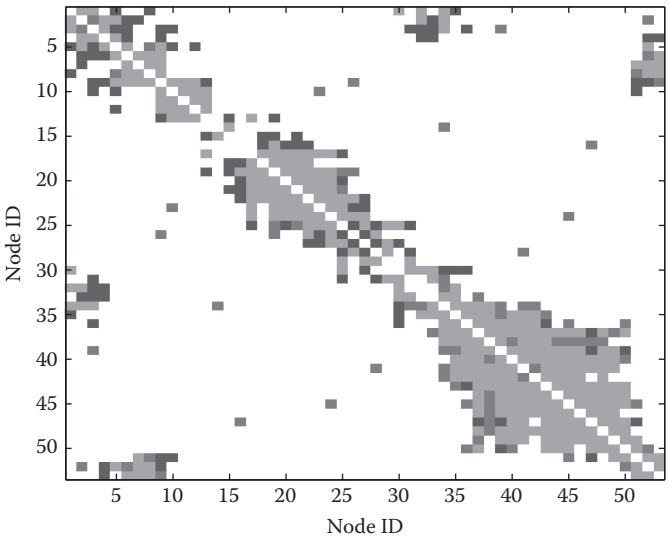


FIGURE 13.5 *T*-test conducted on the correlation matrix of Topology 2. Dark gray squares show evidence of correlation between nodes. Black squares indicate the existence of a link between nodes. Light gray indicate both evidence of correlation and the existence of a link between nodes.

distribution decreases as both parameters increase, hence decreasing the likelihood of observing failed nodes and consequently decreasing the average number of failures. In addition, Figure 13.8 shows the average number of failed nodes as a function of both the homogeneous robustness parameter and D_{Max} From the figure, it can be concluded that the effect of increasing the maximum distance between nodes is to increase both the likelihood of observing failed nodes and the average number of failures.

As the last example, the likelihood of observing certain failure patterns in the WSN is calculated. It is worth mentioning that the actual failure probabilities cannot be determined, due to the normalizing

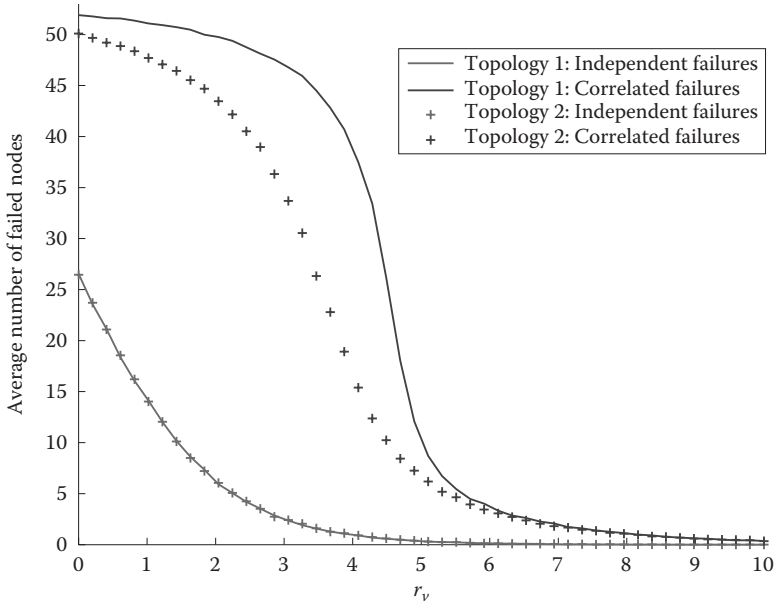


FIGURE 13.6 Average number of failed nodes as a function of the homogeneous robustness vector for Topologies 1 and 2.

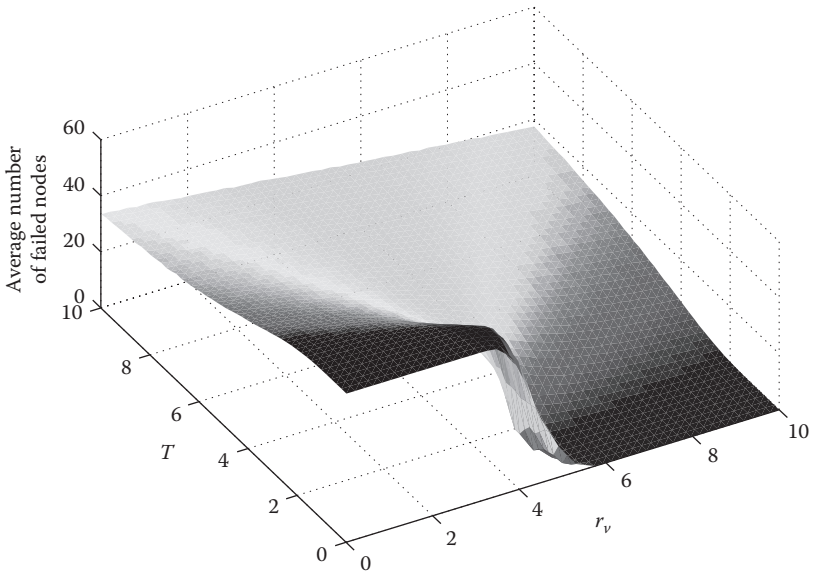


FIGURE 13.7 Average number of failed nodes as a function of the homogeneous robustness vector and the T parameter for Topology 1.

constant Z_t is unknown and its calculation poses an enormous computational burden. Thus, the failure likelihoods presented here are computed using the expressions $\exp\left(-\frac{\mathbf{x}^T\mathbf{r}}{\mathbf{T}}\right)$ for the independent case and $\exp\left(-\frac{\mathbf{x}^T\mathbf{r}-\mathbf{x}^T\mathbf{S}\mathbf{x}}{\mathbf{T}}\right)$ for the correlated case. Note that these likelihoods are clearly proportional to the actual probabilities. Note also that the likelihoods of correlated and independent cases computed here cannot be directly compared as their normalizing constants are different and unknown.

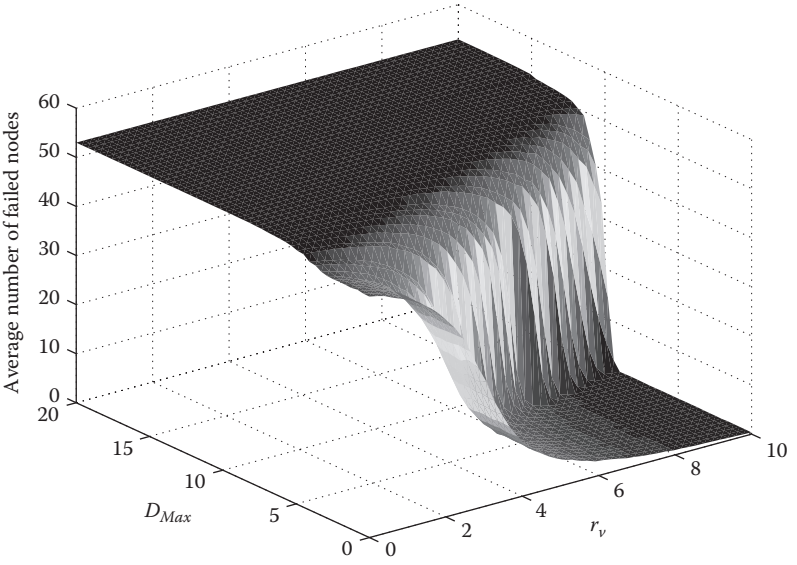


FIGURE 13.8 Average number of failed nodes as a function of the homogeneous robustness vector and maximum distance for Topology 1.

For brevity, representative cases of correlated failures occurring at 13 and 14 different sensor nodes were considered, for the Topologies 1 and 2. The failure patterns are presented in Table 13.1 and were labeled as *clustered*, *unclustered*, and *independent*. The *clustered* case represents a failure at a cluster of well-interconnected sensor nodes, which corresponds to the sensor nodes labeled as 1–13 or 1–14. The *unclustered* case represents a failure at 13 or 14 not well-connected nodes. The *independent* case represents a totally random failure at 13 or 14 sensor nodes. Note that in Table 13.1, some likelihoods are larger than 1 because these are not normalized or actual probabilities. From Table 13.1, it can be first noted that *clustered* correlated failures are about 10 orders of magnitude more likely to occur than *unclustered* correlated failures. Second, note also that WSN connectivity increases the likelihood of *clustered* and *unclustered* correlated failures. Third, for either the *clustered* or the *unclustered*

TABLE 13.1 Likelihood of Clustered, Unclustered, and Independent Failure Patterns in the WSN

Topology	Failure Likelihood	Failed Nodes	Failure Type
1	6.1×10^{5}	1–13	Clustered
	3.5×10^{-8}	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, and 49	Unclustered
	5.1×10^{-12}	1–13 or 1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, and 49	Independent
	3.4×10^{5}	1–14	Clustered
	1.1×10^{-8}	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, and 53	Unclustered
	6.9×10^{-13}	1–14 or 1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, and 53	Independent
2	3.7×10^{3}	1–13	Clustered
	2.8×10^{-9}	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, and 49	Unclustered
	5.1×10^{-12}	1–13 or 1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, and 49	Independent
	4.4×10^{2}	1–14	Clustered
	3.7×10^{-10}	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, and 53	Unclustered

correlated failures, adding one more failed node does not significantly increase the likelihood of failure; however, it is clear that such likelihood does significantly increase when adding sensor nodes connected with those that have already failed. Lastly, note that for the *independent* case, the failure likelihood decreases with the number of failed nodes and is independent of which particular node has failed.

13.6 Summary

This chapter reviews an MRF-based model for correlated failures in WSNs. The model abstracts the topology of the WSN using graph theory, and by exploiting the Markovian property of random fields, it captures the logical and geographical correlation between sensor nodes. Moreover, the model is also capable of inducing percolation effects on the remaining sensor nodes upon the failure at one or more nodes. The MRF-based model for correlated failures is defined in terms of the so-called local specifications, which are related to geographical and communication parameters imposed by the topology of the WSN.

The examples presented here show that a single-node failure percolates to other working nodes, and the degree of percolation depends on the connectivity of the WSN and the so-called level of interaction parameters. The analysis shows also that the average number of failed nodes increases with the number of connections among the sensor nodes. In addition, the average number of failed nodes is larger in the case of correlated failures as compared to the independent case. Finally, the examples show also that the likelihood of observing clustered patterns of correlated failures is considerably higher than observing independent or unclustered failures.

In other words, WSNs must operate unattended in all sorts of environments. Nodes forming a network are inevitably prone to fail, and in very harsh environments, nodes may get physically damaged or even attacked, resulting in single- and multiple-node failures. In these harsh environments, one can conjecture that multiple-node failures may be triggered in a correlated manner as a result of an event, a natural disaster, or an intended attack. In this chapter, a probabilistic model for correlated node failures in WSNs is reviewed. The model is based on the theory of MRF and uses simple yet meaningful interactions among neighboring nodes to induce a global distribution for correlated node failures. The statistical analysis conducted on an actual WSN topology has shown the ability of the model to produce correlated node failures. Results show that, as compared to independent failures, correlated failures increase both the average number of failed nodes and the probability of observing a large fraction of failed nodes. Moreover, results also show that more connected topologies induce more correlated failures among the nodes.

Acknowledgment

FONDECYT Project Iniciación Folio 11110078, PIA-CONICYT PFB0824, and grant CONICYT-PCHA/Doctorado Nacional/2014-63140151.

References

Azimi, N.H., H. Gupta, X. Hou, and J. Gao. Data preservation under spatial failures. In *Proceedings of the 11th ACM Symposium on Mobile ad hoc Networking and Computing*. Chicago, IL: ACM, 2010, pp. 171–180.

Azni, A.H., R. Ahmad, and Z.A. Mohamad Noh. Correlated node behavior in wireless ad hoc networks: An epidemic model. In *Proceedings of Internet Technology and Secured Transactions 2012*. London, U.K.: IEEE, 2012, pp. 403–410.

Bremaud, P. *Markov Chains, Gibbs Fields, Monte Carlo Simulation, and Queues*. New York: Springer-Verlag, 2001.

Buttyan, L., D. Gessner, A. Hessler, and P. Langendoerfer. Application of wireless sensor networks in critical infrastructure protection: Challenges and design options [security and privacy in emerging wireless networks]. *IEEE Wireless Communications*, 2010, 17: 44–49.

Karkvandi, H.R., E. Pecht, and O. Yadid-Pecht. Effective lifetime-aware routing in wireless sensor networks. *IEEE Sensors Journal*, 2011, 11: 3359–3367.

Kinderman, R. and J.L. Snell. *Markov Random Fields and Their Applications*. Providence, RI: AMS, 1980.

Kong, Z. and E.M. Yeh. Distributed energy management algorithm for large-scale wireless sensor networks. In *Proceedings of ACM MobiHoc*. Montreal, Quebec, Canada: ACM, 2007, pp. 209–218.

Li, S.Z. *Markov Random Field Modeling in Image Analysis*. London, U.K.: Springer, 2009.

Liu, J., X. Jiang, H. Nishiyama, and N. Kato. Reliability assessment for wireless mesh networks under probabilistic region failure model. *IEEE Transactions on Vehicular Technology*, 2011, 60: 2253–2264.

Liu, Y., N. Xiong, Y. Zhao, A.V. Vasilakos, J. Gao, and Y. Jia. Multi-layer clustering routing algorithm for wireless vehicular sensor networks. *IET Communications*, 2010, 4: 810–816.

Madden, S. Intel lab data, June 2, 2004. http://db.csail.mit.edu/labdata/labdata.html (accessed April 6, 2014).

Petridou, S., S. Basagiannis, and M. Roumeliotis. Survivability analysis using probabilistic model checking: A study on wireless sensor networks. *IEEE Systems Journal*, 2013, 7: 4–12.

Pezoa, J.E. A Markov random field approach for modeling correlated failures in distributed systems. In *Proceedings of International Conference on Advances in System Simulation (SIMUL 2012)*. Lisbon, Portugal: IARIA, 2012, pp. 121–137.

Pezoa, J.E. Optimizing mission allocation in wireless sensor networks under geographically correlated failures. In *Proceedings of 11th ACM Conference on Embedded Networked Sensor Systems*. Rome, Italy: ACM, 2013, pp. 57:1–57:2.

Qian, Y., K. Lu, and D. Tipper. A design for secure and survivable wireless sensor networks. *IEEE Wireless Communications*, 2007, 14: 30–37.

Raghavendra, C.S., K. M. Sivalingam, and T. Znati. *Wireless Sensor Networks*. Berlin, Germany: Springer, 2004.

Rahnamay-Naeini, M., J.E. Pezoa, G. Azary, N. Ghani, and M.M. Hayat. Modeling stochastic correlated network failures and assessing their effects on reliability. In *Proceedings of IEEE ICCCN 2011*. Maui, HI: IEEE, 2011.

Sen, A. Region-based connectivity—A new paradigm for design of fault-tolerant networks. In *Proceedings of High Performance Switching and Routing 2009*. Paris, France: IEEE, 2009, pp. 1–7.

Shwe, H.Y., H. Gacanin, and F. Adachi. Multi-layer WSN with power efficient buffer management policy. In *Proceedings of Communication Systems (ICCS) 2010*. Singapore: IEEE, 2010.

Willow Technologies. MOTE-KIT 5X4XMICA2/DOT KIT, 2013. http://www.willow.co.uk/html/mote-kit_5x4x_mica2_dot_kit_.php (accessed April 6, 2014).

Xu, Y. and W. Wang. Characterizing the spread of correlated failures in large wireless networks. In *Proceedings of IEEE INFOCOM 2010*. San Diego, CA: IEEE, 2010, pp. 1–9.

Yoon, I., D.K. Noh, and H. Shin. Multi-layer topology control for long-term wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2012, 2012: 1–9.

14

Acoustic Sensors and Algorithms for Urban Security

14.1 Introduction **14**-1
14.2 Acoustic Signature of Small Firearm **14**-3
Muzzle Blast Wave • Directivity of the Muzzle Blast • Outdoor Propagation of Muzzle Blasts
14.3 Electroacoustic Sensor Networks for Gunshot Detection **14**-7
Electroacoustic Transducer Specifications • Coverage of the Sensor Network • Transmission Channels for Sensor Data
14.4 Signature Analysis for Gunshot Detection **14**-8
Gunshot Detection • Estimation of Gunshot Origin
14.5 Summary **14**-12
References **14**-12

Juan R. Aguilar
Laboratorio de Electro Acústica & Procesamiento de Señal E.I.R.L.

14.1 Introduction

The gunshot detection technology for urban security formally appeared as the result of the transfer of the sniper detection technology from military to civilian context. Motivated to provide a solution to sniper fire events in the Bosnian War, the US Army starts a joint research program in gunshot and sniper localization together with the Defense Advanced Research Projects Agency (DARPA) and several defense technology industries [1]. Among the several technological approaches developed as part of the DARPA initiative, an acoustics-based gunfire location system was introduced. This approach was capable to be deployed in an urban warfare scenario. The System for the Effective Control of Urban Environment Security (SECURES) developed by Alliant Techsystems, Inc. was first installed at the Fort Benning Olympic Village during the games of Atlanta in 1996 [2]. Although it was originally conceived for military purposes, its introduction as an instrument for surveillance of public spaces constitutes the first successful application of a gunshot detection system in the context of urban security. After the successful results of SECURES in Fort Benning, the system was implemented in Dallas and in a number of locations around the United States.

In general, a system of acoustic sensors is regarded as a high-technology instrument, which is capable of performing surveillance and target acquisition functions as well. Surveillance embodies to the actions and resources used to make observations of environmental noise in a systematic and continuous way, regardless of weather condition. Target acquisition signifies to characterize ambient noise by performing detection, spatial localization, reconnaissance, and identification of one or more previously defined targets. A particular kind of systems of acoustic sensors is those conceived to detect, localize, and identify gunshot noise in urban terrains, which are generically termed as gunshot detection or gunshot localization systems.

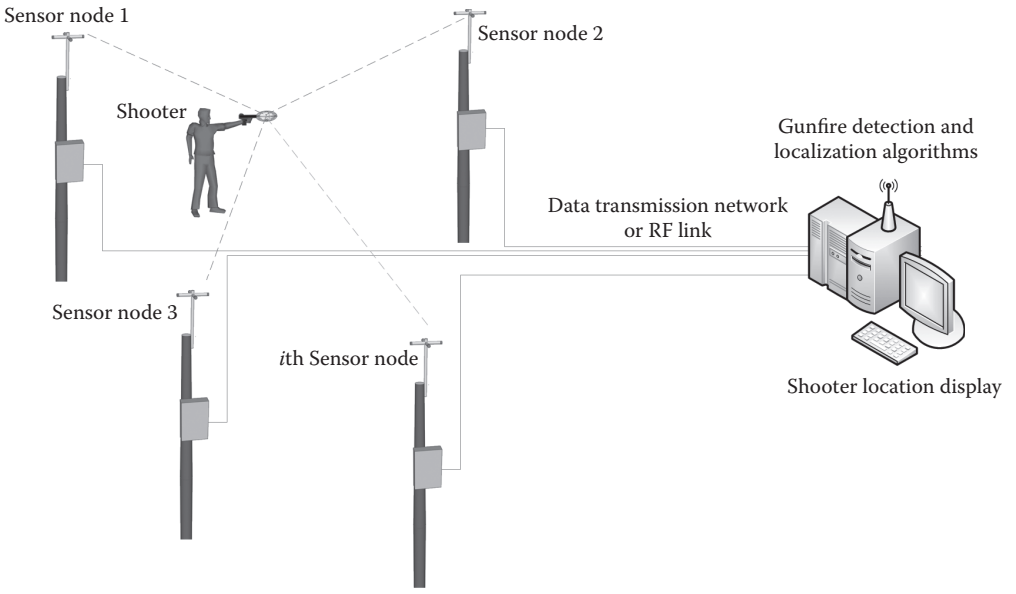


FIGURE 14.1 Schematic diagram of a gunshot detection system.

The solution to gunfire detection problem usually involves a two-stage procedure: The first corresponds to discrimination of gunfire blast from other sources, which could also be components of the background noise. Signal processing features are required to distinguish gunfire acoustic signature from other high-level impulsive sounds like door slams, hammer strikes, cars backfires, or fireworks. The second stage consists in the triangulation of the spatial coordinates of gunfire blast origin, which finally produces the shooter location estimates.

Gunshot detection system implementation comprises the installation of several microphones, which acts as a distributed sensor network over the urban area under observation. Figure 14.1 shows a simplified layout of a modern gunshot detection system. The microphone signals are then transmitted to a computer where signal processing for gunfire detection and shooter localization takes place. Gunfire alert and shooter location result are finally displayed to the users a few seconds after a shot by means of a geographical information platform. Under some circumstances, the system can also provide information about the type and caliber of the fired weapons.

The first endeavor to develop a system for locating the origin of gunfire in urban terrains was performed in 1992 at the US Geological Survey (USGS) in Menlo Park [3]. Motivated by the high level of gunfire-related crime in the city, Lahr and Fischer demonstrate the feasibility to locate high-intensity impulsive sound sources using acoustic sensors together with traditional seismic techniques and software. The technique only uses seismic software for locating the origin of an impulse sound, but does not implement any gunfire detection methodology for discrimination of gunfire noise against background noise. As it can be expected, this approach cannot discriminate gunfire noise from other high-level impulsive sounds and therefore does not solve the problem at all.

In 1995, a company named Trilon Technologies entered the market, and a new gunshot detection system called ShotSpotter appears, first installed in Redwood City since early 1996 [4]. Later in 2004, US-based company Safety Dynamics introduced another approach named the Sensor Enabled Neural Threat Recognition and Identification (SENTRI) system [5]. This approach introduced interesting capabilities for integration with video surveillance, by allowing guidance of cameras based on acoustic sensor information. In 2009, SECURES technology was acquired by ShotSpotter, Inc., thus becoming the dominant of market. In 2013, more than 70 cities around the world had adopted gunshot detection technology against firearm-related violence and crime in urban spaces.

System performance benchmarks indicate that this technology can detect up to 95% of true positive incidents in the area being monitored with a mean localization error of about 20 m. If the accuracy is augmented to give a localization error of less than 10 m, the true positive detection reduces down to 80%–90%. Regarding the impact produced by the use of this technology in public security, the use of gunshot detection technology is assumed to be responsible of reductions by 30% in the number of gun-related violent crime and of about 80% in public and random gunfire rates [6]. Gunshot detection technology had also revealed the hidden numbers of nonreported fire events, showing that only the 25% of gunshots are reported to police by citizens.

14.2 Acoustic Signature of Small Firearm

The acoustic wave processes originated by the gunfire are consequences of fluid dynamics phenomena originated by the cartridge detonation and by the projectile in flight. A roadmap that attempts to explain the origins of different gunfire signatures is shown in Figure 14.2. Acoustic signature of gunfire has at least three main components. The shock-induced chemical reaction of the ammunition propellant inside the barrel generates impulsive pressure depletion in the form of a muzzle blast wave. The aerodynamics of the projectile released into air from the open end of the weapon's barrel can originate ballistic shock waves in the case the projectile has supersonic speed. The projectile in flight also produces vortex shedding along the trajectory, which could eventually originate Aeolian tones.

Given the assumption that the problem of urban violence and crime is mostly associated to the illegal use of small firearms, namely, guns, revolvers, and shotguns, current acoustic gunshot detection technology

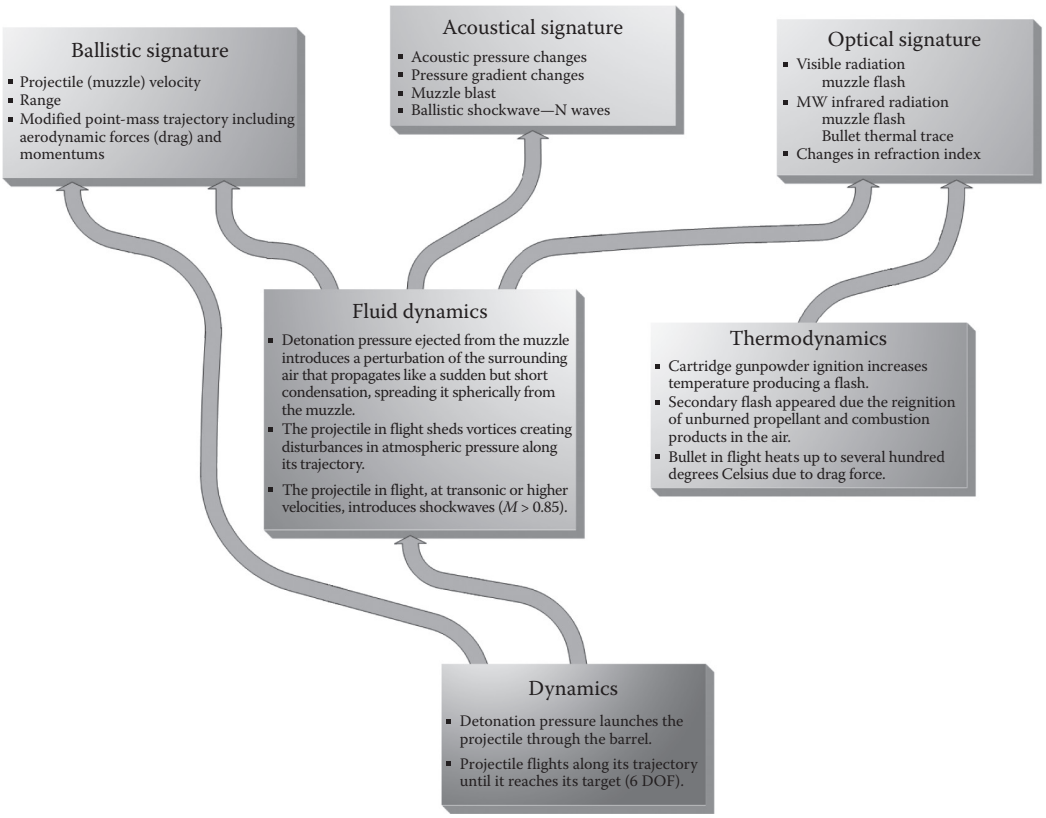


FIGURE 14.2 Roadmap to acoustic signature of a gunshot.

TABLE 14.1 Muzzle Velocities of Several Small Arms

Weapon	Caliber	Muzzle Velocity (m/s)
M16 rifle	5.56 mm	1000
Remington 30-06 rifle	30.06	960
AK-47 submachine gun	5.45 mm	700
Revolver Magnum	0.500	550
Revolver Magnum	0.44	390–490
Beretta 96D model	40	445
Gun	9 mm	280–350
Heckler & Koch USP tactical pistol	0.45	260
Revolver	0.22	200

uses only the muzzle blast wave information, for performing shooter location estimations. By contrast to sniper detection technology for military purposes, gunshot detection technology does not perform ballistic shock wave analysis. Table 14.1 contains the values of projectile velocity at the muzzle, or muzzle velocity, of several small arms of different calibers. Notice that most guns and revolvers have muzzle velocities that are transonic or subsonic. Supersonic muzzle velocities are present in submachine guns and deer rifles.

14.2.1 Muzzle Blast Wave

Muzzle blast wave of gunfire is derived from fluid dynamics processes associated to the axisymmetric muzzle blast flow of the exuding high-pressure and high-temperature plasma from the barrel. In turn, the muzzle blast flow is originated by the two-phase explosion of the cartridge propellant. The first phase corresponds to the ignition of cartridge and the following turbulent heat flow along the gun barrel, and the second comprises the re-ignition of both unburned propellant and combustion products outside the muzzle. Blast wave is the acoustic impulse of finite amplitude that diverges spherically from the weapon muzzle. In order to model acoustic impulses, Friedlander waves are used [7]. Equation 14.1 corresponds to the analytic expression of an ideal Friedlander wave with finite rise time. Here, P_s is the blast peak pressure, b represents the blast rise time, c is the impulse duration measured from overpressure to the first zero crossing, and T is the overall blast duration:

$$p_F(t) = \begin{cases} 0 & -\infty < t < 0 \\ P_s t/b & 0 \le t < b \\ P_s(1-(t-b)/c)e^{-(t-b)/c} & b \le t < \infty \end{cases} \tag{14.1}$$

Plot of Figure 14.3 represents a Friedlander wave form according to Equation 14.1. Figure 14.4 shows the frequency spectrum of a theoretical Friedlander wave; notice most of the frequency content is below 10 kHz.

14.2.2 Directivity of the Muzzle Blast

In-depth analysis of the muzzle blasts reveals that their directional characteristic contains information quite significant for gunshot detection purposes. First, the frequency content of the muzzle blast waves is highly directive, being more directional in the low-frequency region than in the high frequencies. So most of the low-frequency energy of the muzzle blast is radiated only in front of the gun and not to the sides or the rear (see Figure 14.5). The second directional feature of gunfire blast noise is more associated to variations in the waveform of the muzzle blast as the function of sound radiation angle [8]. First, the overall sound pressure level measured in front of the weapon is about 20 dB high than the level measured at the rear of the weapon. Besides, the gunshot acoustic signatures measured in front of the

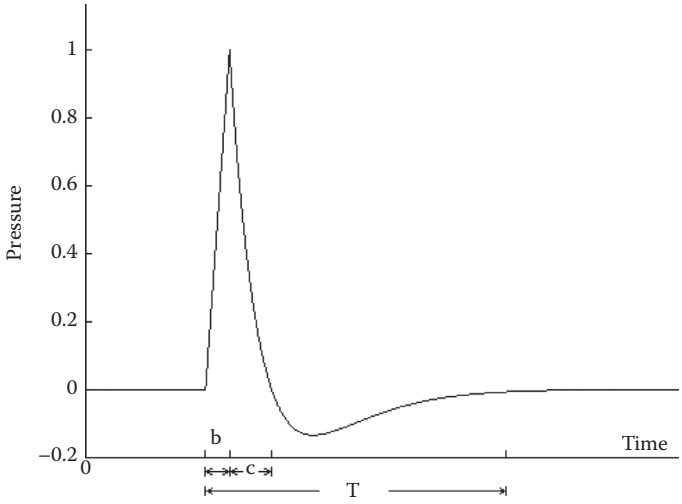


FIGURE 14.3 Theoretical Friedlander wave with finite rise time.

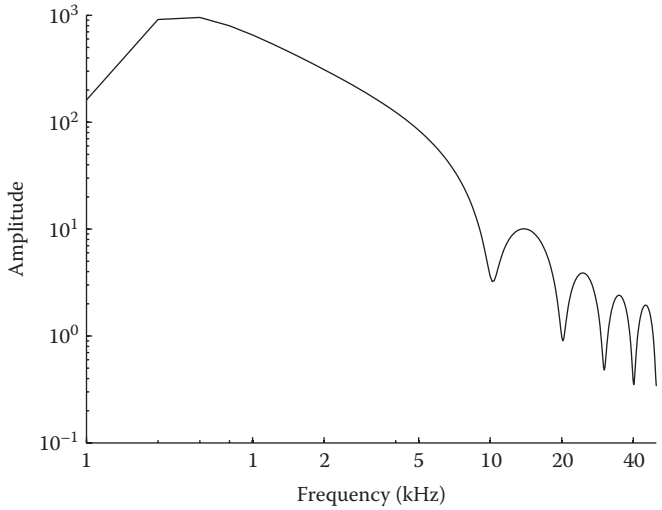


FIGURE 14.4 Frequency spectrum of a theoretical Friedlander wave.

weapon compared to those measured at the rear of the weapon display wave form patterns of peaks and oscillations that are quite different (see Figure 14.6). These two directional cues of the gunshot noise are capable to be used to discriminate a gunshot from other impulsive sounds, to obtain indications about firearm orientation, and to distinguish weapon barrel length [9].

14.2.3 Outdoor Propagation of Muzzle Blasts

There are several phenomena that intervene in outdoor propagation of sound; most relevant to gunshot detection are geometrical divergence, atmospheric absorption, and multipath distortion. Geometrical divergence of the muzzle blast wave front progressively reduces overpressure amplitude according to inverse square law. This means that the sound pressure level of the muzzle blast will decay 6 dB per doubling of the distance to the muzzle. For example, a level of 145 dB will decay to approximately 85 dB at 1 km only because of this phenomenon.

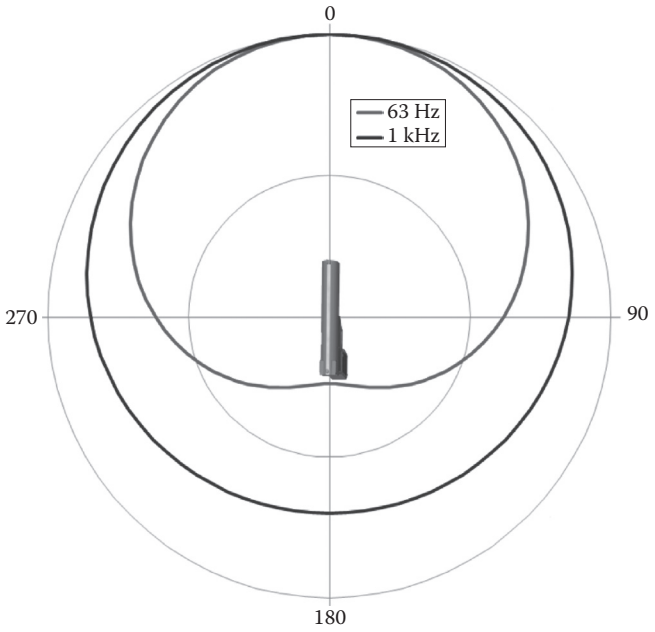


FIGURE 14.5 Simplification of directional attributes of muzzle blast at low and high frequencies.

FIGURE 14.6 Directional attributes of muzzle blast wave form.

Atmospheric absorption of finite amplitude acoustic waves is a dissipative process, which comprises the effects of viscosity, thermal conductivity, molecular relaxation, and dispersion. The atmospheric absorption, including thermoviscous and relaxation effects, is expressed in its differential form as

$$\frac{\partial p}{\partial r} = \frac{\delta}{2c_o^3}\frac{\partial^2 p}{\partial t^2} + \frac{m}{2c_o}\frac{\partial^2 p}{\partial t^2}\sum_{j=1,2}\left[2\pi f_{ri} + \frac{\partial p}{\partial t}\right]^{-1} \tag{14.2}$$

where

δ is the diffusivity of sound
$m = (c_{inf}/c_o)^2 - 1$
c_o is the low-frequency speed of sound
c_{inf} is the high-frequency speed of sound
f_{ri} corresponds to the ith relaxation frequency ($i = 1$ for oxygen; $i = 2$ for nitrogen) [10]

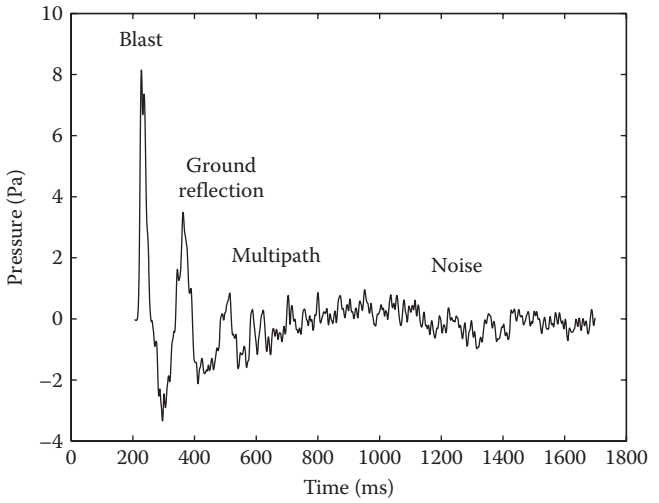


FIGURE 14.7 Muzzle blast wave in the presence of multipath and background noise.

Multipath distortion, in our case called acoustic reverberation, is originated by the multiple and successive reflections of the muzzle blast wave front in the surrounding surfaces, including ground and buildings. Multipath distorts the blast wave form and could affect gunshot localization estimators. Time series of Figure 14.7 shows the effects of multipath distortion and background noise in a muzzle blast wave produced by a 9 mm blank cartridge. In this recording, the microphone height was 1.5 m.

Although phenomena such as wind and temperature gradients could also affect significantly muzzle blast propagation, its introduction in the gunshot detection technology requires additional implementations of both physical models and appropriate sensors, which have not been made in current systems. Moreover, given the altitude at which sensor is frequently located, the stratification of environmental variables can be regarded as nonsignificant in muzzle blast wave propagation.

14.3 Electroacoustic Sensor Networks for Gunshot Detection

Sensor network architecture consists of sensor nodes that are distributed inside the area under surveillance. The sensor nodes are mainly located at street intersections and at high places. Cellular telephony antennas, lighting poles, or high buildings can be regarded in this concern. Sensor nodes use electroacoustic transducers, namely, microphones, to pick up the gunshot noise, and each node can contain one or more microphones. Most approaches use only one transducer in each node [3,9,11]. In this way, signal processing between nodes is required for gunshot localization. The main impact of this approach in the gunshot localization performance is that it only permits a 2D localization in the horizontal plane. In recent times, multiple-transducer approaches have been introduced in the design of sensor nodes by using tetrahedral microphone arrays that permit localized gunshots in 3D [5].

14.3.1 Electroacoustic Transducer Specifications

The type of transducer plays very important role in the sensor network performance. First attempts at USGS were done using dynamic moving coil microphones, but in the present time, low-cost condenser microphones are preferred. Dynamic microphones have the well-known advantages that are capable to accept very high sound pressure levels and that are resistant to weather conditions. However, the condenser microphones have prevailed because of their high sensitivity, short impulse response, and flat

frequency response. The microphones should be pressure transducers having omnidirectional pickup pattern and should be capable to accept sound pressure levels up to 140 dB (re 20 uPa). Small diaphragm microphones are appropriated because they accept higher sound pressure levels and also have omnidirectional pattern of higher frequencies than large diaphragm transducers. Microphone frequency response must appropriately match the muzzle blast frequency content, which is mostly in the range below 10 kHz.

Since the transducer will be continuously exposed to weather condition, waterproof microphones need to be used. In this regard, there are several lavalier-type microphones that have waterproof characteristics, and also waterproof windscreens can be found. Current low-cost condenser microphones are capable to operate at air relative humidity of 95% and at temperatures between –20°C and 50°C. Furthermore, condenser microphones operating with radio frequency (RF) polarization technology should be considered because of their recognized low sensitivity to humidity and better low-frequency response.

14.3.2 Coverage of the Sensor Network

The number of sensor nodes per unit area, or sensor node density, required to detect gunshots depends on the background noise level and on the amount of multipath distortion or acoustic reverberation present. In outdoors and in the absence of high-level background noise, gunshot muzzle blast wave can be perceived up to very large distances from the shooter. However, in urban environments, acoustic multipath introduced by buildings and the presence of higher levels of background noise could reduce the sensitivity to pick up gunfire noise to about 1.5 km. As a general rule, the higher the levels of background noise and of reverberation, the higher the density of sensor nodes required. The sensor node density also varies from one approach to another. In SECURES system, sensor nodes are spaced less than 100 m, which in turn results in about 30 acoustic sensor nodes per square kilometer of coverage, but in ShotSpotter, the distance between sensor nodes could be 200 m or more, and consequently the system requires less than 6 sensors per km^2 [5].

14.3.3 Transmission Channels for Sensor Data

Microphone signal is further transmitted to remote computer for digital signal processing. In this concern, two technological approaches have been tried. First is the RF link used by Lahr and Fischer and later in the SECURES and SENTRI systems. This technology has the advantage of low operation costs, but this susceptibility to RF interference could affect significantly the system performance. The RF transmission is usually in the 900 MHz range. Transmitter RF power of will depends both on network dimensions and on the distance to the processing unit; at least 100 mW with directional antenna arrays could be required [9,11].

The second approach corresponds to data transmission through telephone lines, as used in ShotSpotter. Being obviously more robust than wireless RF link, this approach requires dedicated telephone line for each microphone or sensor, which significantly increases their operation and maintenance costs. The use of telephone lines also imposes a reduction in bandwidth available for gunshot signal quantization, and further processing, to only 3.5 kHz [11].

14.4 Signature Analysis for Gunshot Detection

Modern gunshot detection systems are based on the observation and analysis of the acoustic signature produced by gunfires. Signature analysis for gunshot detection must be capable to discriminate a fire event from the background noise and reverberation and to estimate the spatial coordinates of the

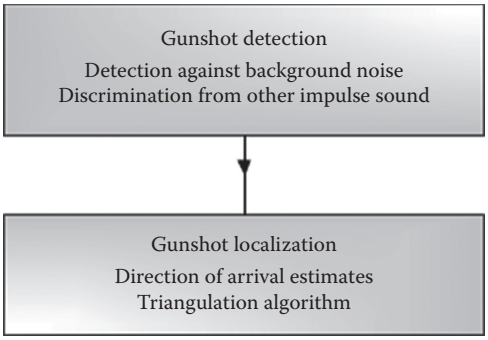


FIGURE 14.8 Basic flow of gunshot detection process.

shooter location, as sketched in block diagram of Figure 14.8. In the case when the acquired gunshot signals carry appropriate information, signature analysis would also lead to other important ballistic features of the fire event, like the muzzle orientation and the weapon type.

14.4.1 Gunshot Detection

Discriminating gunfire noise from the urban background noise requires complex signal processing procedures. Such techniques may include energy estimation, transient shape analysis, rise-time derivation, spectral analysis, time-domain analysis, and correlation analysis. Effective and robust detection of gunfire noise is only achieved after a multiple-stage discrimination process, which comprises at least two steps. The first step is to detect the shooting noise from the overall background noise, and the second comprises discriminating the gunshot noise from other sounds of similar nature.

The objective to distinguish gunshot noise from the background noise is performed, observing the amplitude of the gunshot signals at the microphones. A possible gunshot is detected when the amplitude of the signal exceeds a previously defined threshold over the background noise. Threshold is determined by the background noise conditions, which can vary between 10 and 20 dB SPL from day to night. Either the instantaneous amplitude or the signal-to-noise ratio is used in this regard. The detection threshold may be either fixed or variable, the latter obtained by a dynamic averaging of peak background noise levels [9].

Furthermore, several signal processing approaches have been devised in order to discriminate gunfire from other high-level impulsive sounds. These include measurements and analysis of features like amplitude, temporal duration, frequency content, periodicity, and directional characteristic of signals, which are candidates to be identified as gunshot. Gunshot noises have amplitudes in the range of 140–160 dB SPL when measured at 1 m from the muzzle. In urban environments, gunshot signal can be detected against background noise up to distances of more than 1 km from the shooter, and consequently the gunfire sound should be perceived by several sensors in the network. By contrast, weaker impulsive sounds of nonexplosive nature can have levels about 20 dB SPL below the gunshot noise, and therefore, they could be detected only by a reduced number of sensors [11].

The temporal duration of the impulse is also used as descriptor to classify a gunfire from other similar noise sources. Gunshot detection consists of classifying the impulse duration within a previously defined range, usually more than 0.2 ms and less than 6 ms. This enables discriminating between short impulses (<0.2 ms), which can correspond to small explosive source like balloon pops, small firecrackers, or blank cartridges; midterm impulses (<0.2 and >6 ms), which can be associated to large explosive source like gunshot or firecracker; and long impulses (>6 ms), which can be related to impulse sound derived from mechanical impact that has associated with a resonant process, like hammer strikes,

door slams, and backfires. In fact, impulsive sounds originated by mechanical impacts, although these sounds can have rise times similar to gunshots, resonate for longer periods than gunshot, and have harmonic structures not present in gunfire. Moreover, resonant processes derived from mechanical impact produce frequency content and periodicity attributes of these signals. The analysis of these features involves measurements of the frequency content in specific ranges and the detection of resonant frequency and harmonic content and permits to distinguish gunfire from other resonant noise sources like metal clanks, hood slams, and backfires [9]. Measurements of propagating muzzle blast waves have shown that the blast impulse duration is usually less than 5 ms up to propagation distances of 600 m from the weapon muzzle. Atmospheric absorption processes usually introduce a rapid growth of impulse duration at propagation distances larger than 300 m.

Based on the observation that gunshot has an onset considerably more abrupt than other high-level sounds, some implementations for gunshot detection consider measures of the sharpness or abruptness of the candidate signal. In this procedure, root mean square (rms) profiles of the background noise are successively time-windowed and analyzed in search of peaks in the ratio between the rms values of the signal before and after the point of maximum rms within the window [11].

Gunshot detection is ultimately achieved by measuring directional attributes of candidate signals. Directional attributes of the propagating signals are obtained as the sound waves pass across the sensor in the network, and its analysis enables discrimination of gunshot from large firecracker. In addition, directional information can provide information about the weapon type [9].

The last advances in gunshot detection algorithms have been introduced by implementing computer intelligence algorithms for automatic pattern recognition of muzzle blast wave forms in the background noise. In this approach, muzzle blast wave form has been parameterized in a small set of wave form descriptors, such as the peak overpressure amplitude, impulse duration time, rise time, ratio of the positive to the negative peaks of the impulse, and slope of the impulse. Furthermore, neural networks of radial basis function previously trained to recognize a set of reference muzzle blast wave forms are used for the detection of gunshots [5].

On the other hand, the presence of acoustic multipath and reverberation in the propagation of muzzle blast waves can introduce some shortcomings in the performance of the time-of-arrival estimators and consequently to the gunshot localization algorithms. Gunshot detection against multipath distortion, or acoustic reverberation, is based on the analysis of the first 50 ms since the arrival of direct sound from the weapon to the microphones. Notice that the sequence of echoes of a gunshot produced at the surrounding buildings generates repetitive impulse signal, which has gaps between consecutive reflections. This is a unique feature associated to impulsive sound in the presence of multipath distortion and is not present on other steady or continuous noise sources. Statistical measures of the number and duration of these gaps provide information for gunshot detection in the presence of reverberation [9]. Furthermore, signal processing techniques, for example, cepstral dereverberation or blind deconvolution, have been proposed for reducing the effect of reverberation in the performance of the time-of-arrival estimators.

Autocorrelation analysis computed to each sensor signal permits to discriminate multiple-gunshot events that occurred at a particular location. If the peak's structure in the autocorrelation function is similar in two or more sensors, multiple gunshots can be verified. Furthermore, detection of a multiple-gunshot event can be achieved by shifting in the time domain the signals of various sensors in order to align the gunshot sequence and compare their time alignment.

14.4.2 Estimation of Gunshot Origin

There are two general approaches to get estimations of the spatial coordinates of the location of a sound source. These can be obtained either by comparing the amplitudes of the peak overpressure of the muzzle blast at the sensor nodes to a spherical divergence model or by measuring the direction of arrival of

the muzzle blast waves at microphone pairs in the sensor network and then using a triangulation algorithm. The most recurrent approach in gunshot localization algorithms is the triangulation approach.

The direction of arrival of the muzzle blast wave to microphone pair is estimated from the time delay of arrival (TDOA) of the blast wave to each transducer. This parameter is calculated using the cross-power spectrum between the two microphone signals. Estimations of TDOA also have been done using Bayesian estimation and using genetic algorithms. Cross-power spectrum, also called generalized cross-correlation, is a frequency-domain measure of the correlation between two signals and can be computed using the following equation:

$$\mathrm{GCC} = \mathrm{FFT}^{-1}\left\{H_1(\omega)H_2(\omega)\right\} \tag{14.3}$$

Here, H_1 and H_2 are the complex Fourier transform of the microphone signals 1 and 2, respectively [5]. The angle of arrival of the muzzle blast to the microphone pair is then obtained by calculating the inverse sine function of the product between the speed of sound and the TDOA divided by the distance in m between the microphones. This in fact yields two hyperbolas, but just one of them points out to the source location. Then, to avoid ambiguities in the direction of arrival estimates, at least two pairs of sensor nodes are required for localization in a 2D plane. Furthermore, at least the intersection of three hyperbolas, that is, three pairs of sensors, is required for an appropriate triangulation of the gunshot origin, as sketched in Figure 14.9.

When multiple transducers are incorporated into the sensor node, each node can independently detect the direction of arrival of the incoming muzzle blast. In this way, the gunshot origin could be obtained intersecting at least three DOAs as is depicted in Figure 14.10. Collaborative work with video surveillance cameras has become an important capability of this technique.

Furthermore, the triangulation can be performed in either two or three dimensions as permitted by the sensor network architecture. However, the use of more than three sensor nodes to triangulate the source location not always implies improvements in the localization accuracy. In the case that more than three sensors detect the same event, a selection criteria should be used to choose the best three signals for triangulation. Matrix inversion method can be used when more than three sensors are considered in order to achieve better gunshot location estimates [11].

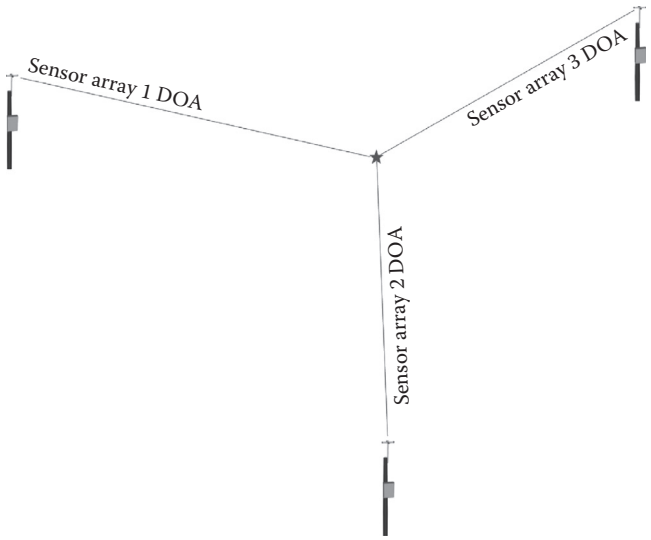


FIGURE 14.9 Triangulation using single transducer at sensor nodes.

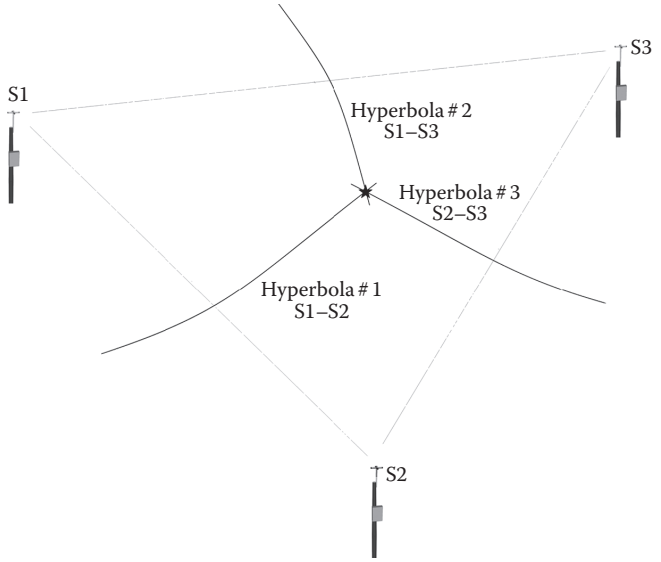


FIGURE 14.10 Triangulation using multiple-transducer arrays at sensor nodes.

14.5 Summary

The electroacoustic technology has found significant application in the field of urban security since the introduction of the gunshot detection systems, devised for acoustic surveillance of urban areas that are exposed to high levels of firearm-related violence and crime. Since then, this technology has permitted that institutions in charge of law enforcement have an instrument for real-time detection of gunfire noises generated by those who undermine law and order and respond to them in a more informed, fast, and efficient way. The gunshot detection technology appropriately circumscribes within the intelligence-led policing paradigm, by providing systematic acquisition and collection of gunfire data, key for the intelligence process and for criminalistics research. By discouraging criminal action, this technology also provides situational awareness that enables reductions in the fear of crime on individuals and communities.

References

1. E. Carapezza, D. Law, C. Csanadi. 1997. DARPA counter-sniper program: Phase 1 acoustic systems demonstration results. *Proceedings of SPIE* 2938:299–310.
2. E. Page, B. Sharkey. 1995. SECURES: System for reporting gunshots in urban environments. *Proc. SPIE* 2497:160–172.
3. J. Lahr, F. Fischer. 1993. Location of acoustic sources using seismological techniques and software—Open-file report 93-221. U.S. Geological Survey Menlo Park. http://pubs.usgs.gov/of/1993/0221/report.pdf (accessed April 21, 2014).
4. R. Showen. 1997. An Operational Gunshot Location System. *Proc. of SPIE*, 2935:130–139.
5. T. Berger. 2007. Real time acoustic event location and classification system with camera display. U.S. Pat. No. 7,203,132 B2.
6. L. Mazerolle, C. Watkins, D. Rogan, J. Frank, C. Kadleck. 1999. Random gunfire problems and gunshot detection systems. U.S. Department of Justice, National Institute of Justice. Research in Brief. https://www.ncjrs.gov/pdffiles1/nij/179274.pdf (accessed April 21, 2014).

7. R. Hamernik, K. Shueh. 1991. Impulse sound: Some definitions, physical acoustics and other considerations. *J. Acoust. Soc. Am*. 90:189–196.
8. G. Settle, T. Grumstrup, J. Miller, M. Hargather, L. Dodson, J. Gatto. 2005. Full-scale high-speed "Edgerton" retroreflective shadowgraphy of explosions and gunshots. In *Proceedings of Fifth Pacific Symposium on Flow Visualisation and Image Processing*. http://www.mne.psu.edu/psgdl/Pubs/2007-Biss-ISSW.pdf (accessed April 21, 2014).
9. U.S. Patent No. 5,504,717.
10. R. Cleveland, M. Hamilton, D. Blackstock. 1996. Time-domain modeling of finite-amplitude sound in relaxing fluids. *J. Acoust. Soc. Am.* 99:3312–3318.
11. U.S. Patent 5,973,998 A.

VI

Sensor Networking Applications and Uses

15 Smart Buildings and Smart Cities *Rim Boujrad and Thomas M. Chen* **15**-1
Introduction • Smart Buildings • Smart Cities • Summary • References

16 Imaging Technology *Hua Lee* **16**-1
Introduction • Scanning Tomographic Acoustic Microscopy • Synthetic-Aperture Sonar Imaging • Ground-Penetrating Radar Imaging • Terahertz Imaging • Optical Composite Imaging • Summary • References

17 Reconfigurable MIMO FMCW Imaging Technique and Applications in Acoustical and Microwave Imaging *Hua Lee and Michael Lee* **17**-1
Introduction • Image Reconstruction Algorithm • Resolution Analysis • FMCW Medical Ultrasound and Microwave Imaging • Summary • References

18 Recent Advances in EO/IR Imaging Detector and Sensor Applications *Ashok K. Sood, Yash R. Puri, Nibir K. Dhar, and Dennis L. Polla* **18**-1
Introduction • UV, Visible, and IR Spectrum and Bands of Interest • UV Nanostructured Detector Array Development: Applications of UV Imaging Technology • Development of GaN UVAPD for UV Sensor Applications • Visible–NIR Detector Technologies • SWIR Detector Technologies • Nanostructured Detector Technology for MWIR and LWIR Bands • Summary • Acknowledgment • References

19 Vehicular Sensor Networks *Juan Pan and Cristian Borcea* **19**-1
Introduction • Vehicular Sensor Networks for Congestion Avoidance • Driver Rerouting System for Congestion Avoidance • Rerouting Strategies • Evaluation of the Rerouting System • Summary • References

20 Acoustic Sensing System for Underwater Geolocation and Navigation *Hua Lee*..... **20**-1
Introduction • Acoustic Sensor Unit • Double-Integration Method • Normalization • Reversed System • Extension to Active Systems • Summary • References

21 Load-Balanced Virtual Backbones in Wireless Sensor Networks *Jing (Selena) He*..... **21**-1
Introduction • Related Work • LBVB under DNM • LBVB under PNM • Summary • References

15

Smart Buildings and Smart Cities

15.1 Introduction 15-1
15.2 Smart Buildings 15-2
Motivations for Change • Smart Building Characteristics and Technologies • Example of Smart Building Projects • Challenges
15.3 Smart Cities 15-9
Urban Population Trends • Smart City Vision • Example Projects
15.4 Summary 15-13
References 15-14

Rim Boujrad
Swansea University

Thomas M. Chen
City University London

15.1 Introduction

Smart buildings and smart cities have become popular terms encompassing a broad array of projects but lack universally accepted definitions. For example, a survey of definitions of smart buildings showed a variety of opinions (Harris 2012). Generally, the term "smart" refers to a combination of environment sensing, data communications, computation, and automated control leading to benefits in energy efficiency, sustainability, operational efficiency, and ultimately cost savings.

"Intelligent building" and "smart building" are sometimes used interchangeably (Martins 2012). The term "intelligent" may have originated in the 1980s with the installation of information and communication technologies (ICT) in buildings and the popularization of the concept of automated living environments. In the 1990s, ICT was seen as a means to increase efficiency and minimize operational costs by integrating the management of all building systems from lighting to fire detection. Over time, the notion of smart or intelligent buildings has evolved to mean more than reactive to environment conditions. A smart building is also proactive in learning and adapting to environmental and occupancy trends.

The US Intelligent Building Institute states that intelligent buildings "provide a productive and cost-effective environment through optimization of its four basic elements including structures, systems, services, and management, and the interrelationships between them." The European Intelligent Building Group describes that intelligent buildings "create an environment which maximizes the effectiveness of the building occupants, while at the same time enabling efficient management of resources with minimum lifetime costs of hardware and facilities." The US definition emphasizes technological aspects, whereas the European one focuses on user requirements. The combination demonstrates that smart or intelligent buildings encompass multiple aspects and not only technology.

The term "smart city" also lacks a universal definition and is sometimes confused with other terms such as digital city, intelligent city, or eco-city. Many smart city projects are being undertaken around the world without a clear definition of smartness (Allwinkle and Cruickshank 2011). At the simplest, a smart city takes advantage of ICT to coordinate and manage all aspects of city operations more efficiently

(Celino and Kotoulas 2013). Most importantly, ICT is applied to energy technologies to improve the effectiveness of real-time energy management (Bach et al. 2010). This simple definition sounds similar to the idea of smart buildings, but a smart city is much more complex than a collection of smart buildings. The operations of a smart city include resource (energy, water) management, traffic management, public safety, governance, and emergency response. In addition to tangible economic and environmental benefits, there should be a positive societal and cultural transformation. Citizens, businesses, and authorities are encouraged to see their city as efficient, interactive, engaging, adaptive, and responsive (Komninos 2002).

Interest in smart buildings and smart cities has been stimulated by government and industry promotion of the smart grid. The smart grid is a separate but related concept. It is commonly assumed that smart buildings and smart cities will play a central role in the smart grid. The smart grid is a vision to modernize the existing electrical power grid, which has not changed for several decades. Consequently, it is increasingly inefficient and vulnerable to failures. The use of ICT in the smart grid enables two-way information exchange between consumers and utility companies (Martins 2012). Utilities receive detailed information about power consumption, and consumers can manage their appliances more intelligently based on more information about dynamic pricing and past energy usage. The smart grid will also enable two-way transfer of power; consumers will be able to return energy to the smart grid that they have generated or stored.

The smart grid encompasses a multitude of national and international standards covering communications. US standards are being coordinated by the National Institute of Standards and Technology (NIST). The European Union created the Smart Grids European Technology Platform in 2005 to generate a common vision for the evolution of the European electric grid. Many pilot projects are proceeding around the world.

15.2 Smart Buildings

Several decades ago, the amount of technology in a typical building was minimal. It might have consisted of public telecommunications; a pneumatic control system for the heating, cooling, and ventilation system; and a word-processing system (Sinopoli 2010). Buildings began to be automated in the 1960s starting with simple actions such as turning off lights or reducing the temperature in unoccupied areas (Harris 2012). Significant savings in cost and energy use could be realized at little cost.

Before 1980, building automation was achieved at the level of the individual device (Wang 2010). In the 1980s, single-function systems were integrated and then multifunction systems. The 1990s saw building level–integrated systems and computer-integrated buildings. More sophisticated control was made possible by sensors embedded into walls and ceilings. Building performance began to be measured in real time. Software was developed to automatically optimize building systems without manual intervention. Over the years, an enormous amount of research has been carried out on building automation related to communication protocols and integration of systems including heating, ventilation, and air conditioning (HVAC); lighting; fire detection; elevators; and security (Martins 2012). However, integration of different mechanical and electrical subsystems is challenging because manufacturers have tended to use proprietary hardware and communication protocols (Penya et al. 2013).

15.2.1 Motivations for Change

The driving forces for smart buildings are economics, energy, and technology. Smart buildings take advantage of ICTs, which have obviously made enormous progress in the past few decades, to integrate systems at the building level and automate intelligent control of building services. Modern technology allows buildings to be constructed and operated more efficiently and safely. More self-sustainable and livable buildings will have better value and marketability.

Commercial buildings consume about 42% of all electricity worldwide, and by 2025, they will be the main sources of greenhouse gas emissions on the planet (Harris 2012). In the United States,

buildings consume almost 70% of all electricity, and commercial buildings account for more than half of this electricity usage (Weng and Agarwal 2012).

Commercial buildings are increasingly mixed use, that is, occupied by both humans and IT equipment (computers, monitors, printers, routers). A modern mixed-use building will typically have four major energy-consuming subsystems: HVAC, lighting, IT equipment, and miscellaneous plug-load devices. The amount of electricity consumed by lighting is 26%, heating is 13%, and cooling is 14%. Their energy consumption might be reduced, but drastic savings are unlikely. HVAC and plug-load devices are usually operated on static schedules, which results in energy waste. Potentially significant energy reductions are possible with fine-grained dynamic control over HVAC systems and plug-load devices with minimal impact on building occupants.

15.2.2 Smart Building Characteristics and Technologies

While a unique definition of smart buildings may be elusive, it is possible to enumerate certain characteristics that are cited frequently. While this chapter focuses on technologies, smart buildings are more than technology. Their ultimate purpose is to improve building performance and better serve the needs of their occupants.

15.2.2.1 Data Driven

Smart buildings are fundamentally driven by data, which involve collection from sensors, analysis, and application to control. It could be argued that "smartness" depends on the quantity and quality of the data. Certain building devices (e.g., fire sensors, HVAC) may be managed by a service provider. Data are sent to the service provider for billing, but data may be accessible to building occupants. In larger buildings, data are collected and analyzed by a building management system (BMS) for controlling devices and services in the building.

15.2.2.2 Automated Control

Smart buildings have a high degree of automated control of building operations according to specific policies. Significant energy consumers in a building are typically HVAC equipment, IT equipment, lighting, and miscellaneous plug loads. They require different treatments in terms of mechanisms for actuation:

- *HVAC system and lighting*: Modern buildings containing a BMS not only sense the environmental conditions but also control the per-zone environmental settings. A smart building has access to these controls in order to actuate the HVAC system on a fine-grained basis. Zones are actuated according to policies typically depending on occupancy and other factors.
- *Plug-load devices*: Plug-load devices are often left powered on even when the user is absent. Significant energy may be saved by actuating devices at the level of each outlet, which can be handled by several existing commercial products. Plug-load devices are controlled in a smart building according to policies dependent on factors such as occupancy (Weng and Agarwal 2012).
- *IT equipment*: Control of IT equipment such as computers is not handled in the best way through direct plug-load control. Instead, IT equipment is usually controlled by power management software installed on individual computers.

15.2.2.3 Integration of Systems

The traditional way to construct a building is to design, install, and operate each system separately (Sinopoli 2010). Systems include HVAC control, data networks, voice network, power management, video surveillance, fire alarm, lighting control, audiovisual, video distribution, and access control. Each system traditionally has its own management console. These systems may be integrated at different levels: physical and functional. Physical integration of systems refers to common cabling, space, power, environmental controls, and infrastructure support.

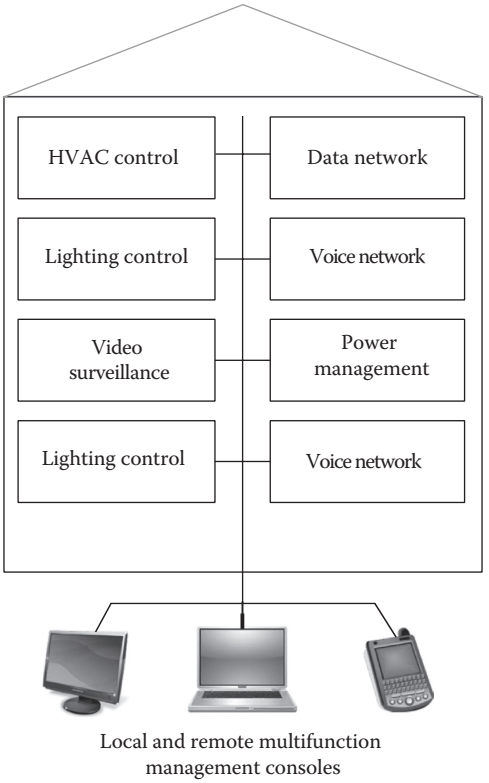


FIGURE 15.1 Integration of building automation systems in a smart building.

For functional integration, the systems should be networked together using some common communication protocols as shown in Figure 15.1. By exchanging state information among them, each system may affect the operational decisions from another system. In addition, the systems should interoperate through integrated applications that can offer capabilities beyond the functions of any single system. Integration at the application level should create a whole that is greater than the sum of its parts.

15.2.2.4 Sensors

Sensors are a critical component of smart buildings. Effective control depends on knowing the operational status of the building at fine temporal and spatial granularities. Three important variables that should be monitored are occupancy and user context, environmental conditions, and energy usage:

- *Occupancy and user context*: Passive infrared (IR) sensors are the most common type of sensors in modern buildings but suffer from accuracy problems because they are actually motion sensors. False positives (when the sensor detects a person who is not actually there) and false negatives (when the sensor fails to detect a person) are common (Weng and Agarwal 2012). In addition to occupancy, user context is useful information to measure if feasible. For example, knowing whether users are actively using devices can lead to better policies for plug-load device control.
- *Environmental sensing*: Most modern buildings have a BMS that monitors temperatures and other variables for the zones in a building. This information is important for optimizing control over HVAC settings.
- *Energy sensing*: Two broad classes of plug-load energy detection are direct sensing, which connects a meter directly in-line with the plug-load device, and indirect sensing, which attempts to measure energy usage without attaching a meter to every device. Direct methods are more

accurate than indirect methods, although indirect methods are generally easier to deploy. Nonintrusive load monitoring began with an attempt to disambiguate energy loads from a central energy meter (Hart 1992), while more recent approaches use learning algorithms. Indirect sensing approaches have used sensors to detect the magnetic field variations that occur near plug-load devices.

In addition to sensors, an important development is the smart meter, which was first introduced as a method to reduce billing costs for service providers by providing an electronic means to read meters remotely (compared to the traditional method of manual reading). Since their first introduction, smart meters have evolved in capabilities, which now include support for demand response. Dynamic pricing signals are sent to the smart meter allowing consumers to make more intelligent decisions about energy consumption, for example, shifting load to off-peak periods.

15.2.2.5 Communication Protocols

Machine-to-machine communications between devices (namely, sensors, actuators, appliances) are essential to smart buildings. Communications can be wired or wireless, and several protocols are available. Wireless communications are attractive for easier installation than wired communications but introduce issues of security and device management.

Ethernet and Internet protocol (IP): Virtually all local area networks (LANs) use Ethernet. For many years, the trend has been convergence of voice and data communications through Ethernet and IP. Building automation systems have specific industry protocols (e.g., BACnet [building automation and control networks], LonTalk), but they are also converting or interfacing the protocols to the universal IP. Video surveillance systems have moved to IP, and access control is moving in that direction as well.

BACnet: It is a data communication protocol developed by the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) in 1996. In 2003, it was adopted as an ISO (International Organization for Standardization) standard 16484-5. BACnet was designed specifically for building automation and control systems for applications such as HVAC, lighting, security, fire, access control, maintenance, and waste management. BACnet can communicate over several types of networks, including Ethernet/IP, master–slave/token passing (MS/TP), and point to point (PTP), and works over different media (phone lines, coaxial cable, fiber optics).

BACnet allows computer-based control equipment from different manufacturers to interoperate by providing a common control language. BACnet defines a standard set of "objects" to represent devices in the network. Each object has a standard set of "properties" that describe the object and its current status. An object may be controlled through its properties by other BACnet devices. BACnet defines 18 standard types of objects. One of the standard BACnet objects is the analog input object shown in Figure 15.2,

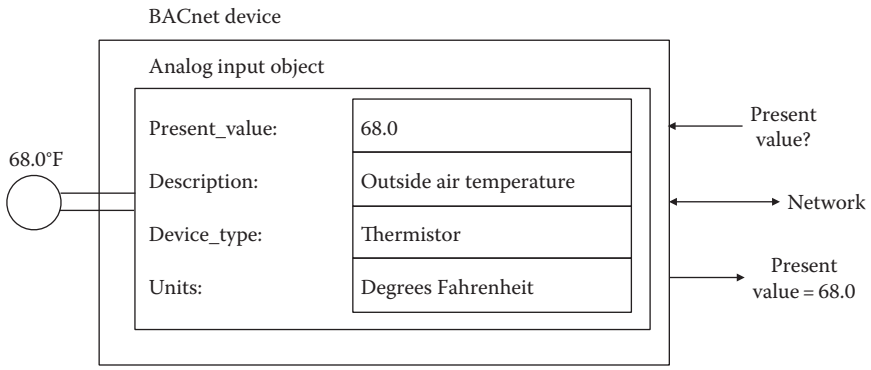


FIGURE 15.2 BACnet analog input object.

which represents a thermistor. Some of the properties are set during installation or time of manufacture, while others provide dynamic status about the sensor input. All properties may be read, for example, a query about the Present_Value property of this analog input object would get the reply "68.0."

LonWorks (local operating network), from Echelon Corp., has become a popular platform for networking embedded devices used in building and home automation (HVAC controls, elevator/escalator controls, lighting, security) and the transportation and utility industries. In 1999, LonTalk (the protocol part of LonWorks) was accepted as a US standard for control networking, ANSI/CEA-709.1-B. In 2009, LonWorks became an international standard ISO/IEC 14908. The protocol defines how devices communicate and the content and structure of the information that is exchanged. LonWorks is a completely peer–peer network; in contrast to master–slave protocols, any device can exchange data with any other LonWorks device on the network. The protocol works mainly over two physical layers: twisted-pair cable and a power line carrier, although LonWorks can also use radio frequency (RF), IR, coaxial cable, and optic fiber. The LonWorks platform uses an affiliated IP tunneling standard (ANSI/CEA-852) to connect devices on LonWorks-based networks to IP networks and applications.

LonWorks is primarily implemented using a device that contains an 8-bit processor called the Neuron chip, designed by Echelon and manufactured by various partners. The Neuron chip is the basic interface between each device being controlled and the central control system software. The Neuron chip is the heart of almost all LonWorks-based devices. The Neuron chip is a complete system on a chip containing the entire LonTalk protocol stack. Reportedly, about 30 million Neuron chips are deployed worldwide. LonWorks is compatible with BACnet (BACnet can let Neuron chips interact with building control devices made by other manufacturers).

The LonTalk protocol provides a common application framework that ensures interoperability using a concept called network variables. Network variables are lines of computer code that define inputs and outputs of devices, such as temperature. Communication between nodes on a network takes place using the network variables that are defined in each node. Some nodes may send a network variable, while others may receive. By only allowing links between inputs and outputs of the same type, network variables enable an object-oriented approach to product development. Whenever a node program writes a new value into one of its output variables, the new value is propagated across the network to all nodes with input network variables connected to that output network variable. This action is handled by the protocol programmed within the Neuron chip. Interoperability is made possible by the standardization of standard network variable types (SNVTs). Examples of SNVTs are shown in Table 15.1.

Modbus is an application layer messaging protocol for client–server communication published by Modicon in 1979. At that time, Modbus focused primarily on programmable logic controllers (PLCs) used in industrial automation. Modbus defines message formats, device addresses, and rules for exchanging messages between devices. Modbus works over different types of buses or networks such as Ethernet or token-passing network. The most common implementation of Modbus uses the serial RS-485 physical layer with either Modbus RTU (a binary representation of the data) or Modbus ASCII (human readable). The Ethernet implementation option uses Modbus/TCP.

ZigBee is a wireless communications standard based on IEEE 802.15.4 for low-rate wireless personal area networks (PANs) with maximum distance between 20 and 50 m. Figure 15.3 shows the ZigBee

TABLE 15.1 Examples of SNVTs in LonWorks

Variable Type	Units
Temperature	Degrees Celsius
Switch state	Boolean
Device state	Boolean
Real time	MM, DD, YYYY
Elapsed time	Days, hours, seconds
Energy	Kilowatt-hours
Power	Watts

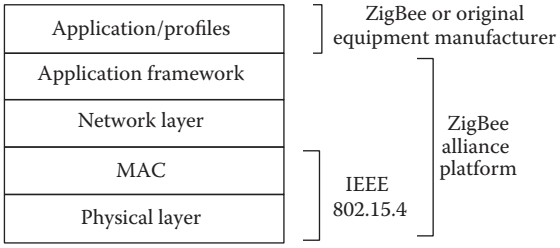


FIGURE 15.3 ZigBee protocol stack.

protocol stack. It operates in the unlicensed frequency bands at 900 MHz and 2.4 GHz. ZigBee is popular for wireless sensor networks (WSNs) because it is low power, widely supported by manufacturers, scalable to thousands of devices, and uses a flexible mesh topology. As a wireless technology, ZigBee is easier to deploy in smart buildings than wired technologies but offers less bandwidth than physical wired media. The technology is particularly useful for sensors and control devices in building automation systems where only small amounts of data need to be transmitted because the maximum speed of ZigBee devices varies between 192 and 250 Kbps.

ZigBee end devices are reduced-function devices (RFDs) or full-function devices (FFDs). FFDs are equipped with a full set of medium access control (MAC) layer functions, which enable them to act as a network coordinator or a network end device. FFDs acting as network coordinators offer synchronization, communication, and network join services. RFDs can only act as end devices and are equipped with sensors or actuators. RFDs may only interact with a single FFD. ZigBee devices can be networked in a star, tree, or mesh topology as shown in Figure 15.4.

The ZigBee Home Automation (ZHA) profile supports a variety of devices for the home including lighting, heating, and cooling, and even window blind control. It provides interoperability from different vendors that allow a greater range of control and integration of different devices in the home. It mainly deals with sporadic real-time control of devices.

The 6LoWPAN working group in the Internet Engineering Task Force (IETF) created a standard for transmitting IPv6 (Internet protocol version 6) over low-power and lossy networks (LLNs) based on IEEE 802.15.4. 6LoWPAN is described mainly in RFC 4919, RFC 4944, and RFC 6282. The main functionality of 6LoWPAN consists of the LoWPAN adaptation layer, which allows for the compression of IPv6 and related headers such as UDP, totaling 60 bytes, down to just 7 bytes, and optimizing mechanisms for wireless embedded networking (Shelby and Bormann 2009). The LoWPAN header consists of a dispatch value identifying the type of header, followed by an IPv6 header compression byte indicating which fields are compressed, and then any in-line IPv6 fields. An example of 6LoWPAN compression

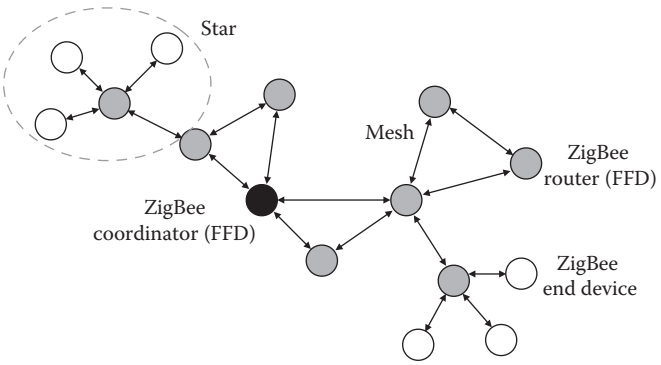


FIGURE 15.4 ZigBee network topologies.

IEEE 802.15.4 frame with full UDP/IPv6 (64-bit addressing)

Bytes	21	1	40	8	53	4
Field	MAC	L	IPv6	UDP	Payload	FCS

IEEE 802.15.4 frame with minimal UDP/6LoWPAN (16-bit addressing)

Bytes	9	2	4	108	4
Field	MAC	L	UDP	Payload	FCS

FIGURE 15.5 6LowPAN

is given in Figure 15.5. In the upper packet, a 1-byte LoWPAN dispatch value (L) is included to indicate full IPv6 over IEEE 802.15.4. The lower packet is an example of 6LoWPAN/UDP in its simplest form with a dispatch value and IPv6 header compression, all IPv6 fields compressed, then followed by a UDP next-header compression byte with compressed source and destination port fields and the UDP checksum (4 bytes).

IEEE 802.11 (Wi-Fi) is a popular wireless alternative to cabled Ethernet for LANs. IEEE 802.11a operates in the 5 GHz frequency band, while IEEE 802.11b and 802.11g operate in the 2.4 GHz frequency band. These standards can provide maximum throughputs of 11 and 54 Mbps. A more recent standard IEEE 802.11n offers a throughput of 110 Mbps.

The 2007 standard IEEE 802.11s enables 802.11 nodes to work in a mesh network as shown in Figure 15.6 (compared to the usual mode where all nodes communicate through a central access point). The coverage of a Wi-Fi network can be extended through multihop packet forwarding. The nodes in the mesh network automatically learn about one another and self-configure network traffic through multiple route configurations. The mesh network can balance traffic by routing packets around areas of congestion or interference. On the other hand, mesh networks create latency that can affect applications such as voice over IP.

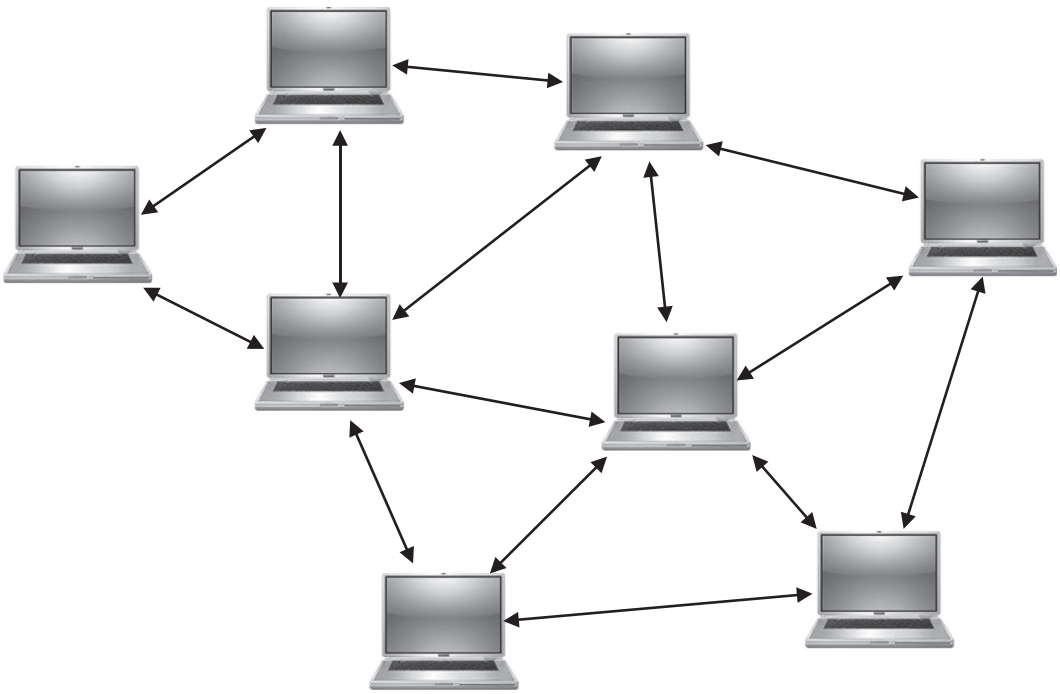


FIGURE 15.6 802.11 mesh network.

IEEE 802.11 is included in the Smart Energy Profile 2.0 (SEP2) approved in 2013, which allows interoperability between 802.11, ZigBee, and HomePlug power line technologies into an IP-capable home energy networking framework. The first version SEP1 consisted only of ZigBee. SEP2 is intended for applications that enable home energy management via both wireless and wired devices that support IP (Tariq et al. 2012).

15.2.3 Example of Smart Building Projects

In July 2013, Seattle launched a High-Performance Building program that involves real-time tracking of energy efficiency to help reduce energy costs and carbon emissions. The "smart buildings" partnership between the city, Microsoft, and the Seattle 2030 District (a public–private collaborative of downtown Seattle property owners) uses analytics software and cloud services to gain insight from data collected from equipment running heating, cooling, and lighting systems. The goal is to reduce 10%–25% of existing costs for both energy and maintenance expenditures and to help the Seattle 2030 District halve building energy usage across the entire downtown by the year 2030.

The Pearl River Tower in Guangzhou, China, is among the first of a new generation of super smart buildings and is intended to serve as a model for future super-tall building design. It will reportedly use less than half the energy of a conventional skyscraper and will harvest energy from natural forces. With a design specific to its environment, the Pearl River Tower combines a number of energy-efficient technologies in a single building structure.

In 2002, the Ave Maria Foundation partnered with the Barron Collier Companies to develop a permanent campus for Ave Maria University near Naples, Florida. The first phase of construction included 23 integrated building technology systems. The project has won global awards for building automation (Sinopoli 2010).

15.2.4 Challenges

Despite many pilot projects, corporate adoption of smart buildings has been relatively limited to date. A few reasons might explain this slow acceptance:

- A challenge lies in accessing the data from existing BMSs due to the disparity of systems, varying ages of the assets, and different communication protocols. For externally hosted smart building solutions, a secure connection may need to be established, which can complicate the data exchange. Large data volumes can strain bandwidth and storage resources.
- Another challenge is collecting data of sufficient granularity in quantity and quality. Contextual information should also be collected.
- Applications need to be usable by people who may not be expert in advanced analytics tools. Some applications may overwhelm users with too many features or confuse users with a nonintuitive user interface.
- Although the cost of implementing a smart building solution can be modest compared to typical building operating costs, it might be difficult to secure funds for a smart building. Developers face the challenge of demonstrating a return on investment in financial and sustainability terms.
- WSNs are not fully developed. Most WSN operating systems (e.g., TinyOS, Contiki, RIOT) only support a basic 802.15.4/6LoWPAN bootstrap.

15.3 Smart Cities

In principle, the same ideas for smart buildings can also apply to smart cities on a much larger scale. However, one major difference is that cities are far more complex with more resources to manage (such as transport). Cities also involve a broader range of issues including how ICT affects the interactions between citizens and government.

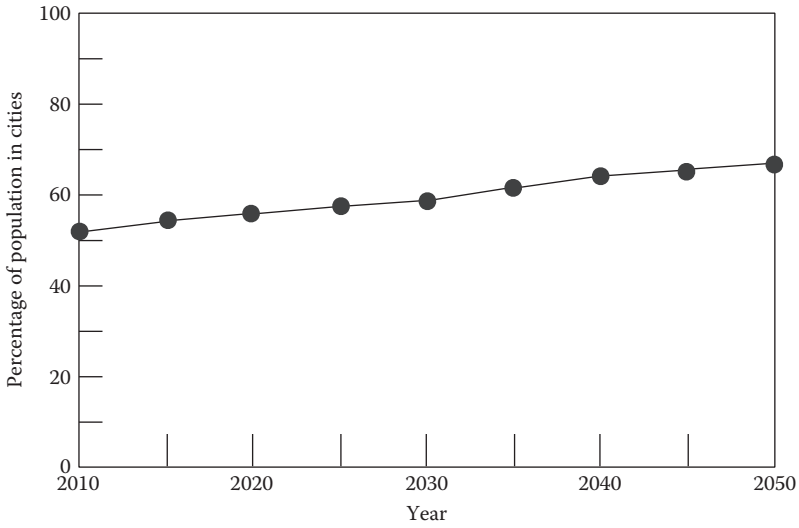


FIGURE 15.7 Percentage of world population living in cities (http://esa.un.org/unup/CD-ROM/Urban-Rural-Population.htm).

15.3.1 Urban Population Trends

An urgency for smart cities can be clearly seen from urban population trends. Half of the world's population lives in cities, and this population is growing at a rate of two people per second. The trend is evident in Figure 15.7. Seventy-five percent of energy is consumed in cities, and 80% of carbon dioxide emissions are produced in cities. The United Nations forecasts that the number living in cities will grow to nearly 70% by 2050 (an increase of 3.6–6.4 billion). Almost all of that growth will occur in developing countries.

Furthermore, the UN Population Fund predicts that there will be 27 megacities with more than 10 million people by 2030. Most of those megacities will be in Asia. China's urbanization has been particularly rapid in the recent decade. The 2010 City Blue Book forecasts China's overall urbanization level will reach 65% by 2030, including more than 70 cities with a population above one million. The strains of growing urbanization are evident in traffic jams, severe pollution, and energy shortage. For example, statistics show that 400 of the 655 cities in China are short of water, including about 200 of them suffering from severe water shortage. Over 400 cities have traffic jams at the peak hours, during which the average speed is only 10–12 km/h (compared to 20 km/h in other international cities).

As resources are strained in cities, an enormous amount will be invested in urban infrastructure. Thus, an opportunity exists to innovate urban design, technologies, and services to improve sustainability, competitiveness, and quality of life. The central idea of smart cities is to embed ICT deeply into all aspects of urban development to create a city responsive to the needs and demands of a knowledge-based citizenry (Zengjun 2013).

15.3.2 Smart City Vision

Smart city agenda are very broad and complex due to the large scale of urban environments. For example, smart cities must address issues of urban infrastructure, transport, quality of life, and citizen engagement with government. Smart cities are much more than ICT technologies. There are also dimensions of business (competitiveness), social life, economics, politics, and environmental sustainability. It involves all stages of city design, construction, and operations.

TABLE 15.2 Layers

User
Services
Infrastructure
Data

The European Union (www.smart-cities.eu) suggests that smart cities can be identified along six main dimensions: a smart economy, smart mobility, smart environment, smart people, smart living, and smart governance. These six dimensions connect with traditional theories of regional competitiveness, transport and ICT economics, natural resources, human and social capital, quality of life, and participation of citizens in the governance of cities. A city is smart when investments in human and social capital, as well as transport and ICT, support sustainable economic development and a high quality of life, with wise management of natural resources. Another view of smart cities has been proposed (Anthopoulos and Vakali 2012) consisting of four layers as shown in Table 15.2:

- *User layer*: Citizens who receive smart city services and authorities who supervise the smart city
- *Service layer*: E-services offered by the smart city
- *Infrastructure layer*: Networks and information systems
- *Data layer*: All the information that is collected, processed, and acted upon

15.3.2.1 Data

Like smart buildings, smart cities are fundamentally driven by data. It is not only important to collect, analyze, and act upon data. There is also a social dimension to smart cities where feedback of data to citizens is important to engage citizens and make the city responsive to their needs. In a smart city, citizens are better informed, and information influences their behavior for their collective benefit. This is analogous to demand response in the smart grid, where consumers are informed about energy pricing and past usage, and they can modify their energy consumption more intelligently.

15.3.2.2 Infrastructure

Along with wired broadband infrastructure, WSNs are an important technology for smart cities. A distributed network of intelligent sensor nodes will measure many parameters for a more efficient management of the city. Data will be communicated wirelessly for easy deployment. For example, sensors could monitor vehicle traffic to inform citizens about traffic jams and available parking.

Recently, a convergence of the concepts of smart cities and the Internet of Things (IoT) has been proposed (Srivastava and Vakali 2012, Suciu et al. 2013, Vlacheas et al. 2013, Jin et al. 2014, Zanella et al. 2014). The IoT recognizes the trend toward everyday objects equipped with microcontrollers and transceivers for digital communication. They will be able to communicate with one another and with users, becoming a predominant part of the Internet. New application services will be developed to take advantage. The IoT concept has obvious similarities to the functional concept of smart buildings and cities. IoT is an active area of current research. An example protocol stack is shown in Table 15.3.

TABLE 15.3 Prospective IoT Protocol Stack

Application
TCP/UDP
IPv6
6LoWPAN

15.3.2.3 Services

Networks enable Internet-based services that may transform city governance processes. Smart city services will be accessible through web services and mobile software applications (Al-Hader 2009). Examples of government e-services can be seen in AOL Cities, Digital City of Kyoto, and Digital City of Amsterdam.

15.3.2.4 Users

A smart city is ultimately about the user experience. Citizens benefit from a richer, more personalized experience through smart interfaces with the city. They experience more efficient and effective delivery of services. As a result, citizens are more engaged with the city in a more informed way.

15.3.3 Example Projects

The number of smart city projects around the world is a long list. This chapter highlights a few selected projects (in no particular order). Smart cities have attracted attention from many governments and technology companies (e.g., IBM claims involvement in some 2500 smarter cities projects and has even trademarked the term "smarter cities"). As an example, the European Union has allocated 956 billion euros to a smart city project fund for 2014–2020. In the United States, almost 400 cities competed for $9 million in awards for city innovation as part of the Mayors Challenge launched by Bloomberg Philanthropies. In the United Kingdom, Glasgow won a £24 million Future Cities Demonstrator competition out of 30 cities. The following are examples of some projects in various cities:

- The first technological steps toward smart cities might be traced back to the 1990s with AOL Cities, Digital City of Kyoto, Digital City of Amsterdam, and Digital City of Trikala (Da Silva et al. 2013).
- *New Songdo, South Korea*: Songdo (www.songdo.com) is an artificial island on reclaimed land about 65 km west of Seoul, South Korea, with 22,000 living there currently. The city is scheduled for completion in 2015 when it will be home to 65,000 people and 300,000 workers. Started in 2000 and estimated to cost US $35 billion, Songdo is the largest private real estate venture in history. Most of the investment came from US private real estate firm Gale International and investment bank Morgan Stanley with the intention to showcase technology and make Songdo a model for future smart cities. Cisco has embedded sensors into roads, streets, and buildings, which send data to a central control hub. Radio frequency identification (RFID) tags are attached to every vehicle's number plate. These report data to the control hub, providing a picture of the traffic situation, enabling the control hub to adjust traffic light timings, create diversions, and provide early warnings about congestion. In addition, every home has a telepresence system that people can use for video calls or control heating and locks. It also delivers education, health-care, and government services.
- *"Eco-city," Masdar City, UAE* (www.masdarcity.ae): This city in the middle of the desert of Abu Dhabi is designed to be one of the most sustainable cities by taking advantage of ICT. Sensors measure the environment and buildings' energy capacity. Smart solutions for renewable energy production are encouraged such as a solar power station at its heart and wind farms generating energy. It is entirely car free although there is an experimental network of driverless electric podcars.
- *Barcelona*: The first City Protocol workshop was held in Barcelona in 2012. The City Protocol (www.cityprotocol.org) is a collaboration between Barcelona's City Council, GDF SUEZ, Cisco, and numerous other companies, cities, and universities to develop a "more sustainable, efficient, cohesive, innovative, and smart city." It addresses the whole of the city ecosystem including water, waste matter, energy and utilities, mobility, goods, people, and information. Taking its inspiration from the way Internet and web standards have been delivered, it fosters a similar process of open, transparent, and robust collaboration on an international basis. The City Protocol Society was set up to manage it.

- *Glasgow*: Glasgow recently won a £24 million competition in the United Kingdom for pilot projects that show how mass deployment of sensors and real-time information can help local government run more efficiently while also boosting the quality of life for its 600,000 citizens. The plan is to look for ways to improve how the city uses energy, runs its transport, manages its public safety, and distributes health services. Smart technology will play a role in managing real-time transport data, helping residents communicate with their city government, and better monitoring of power and utility usage.
- *Lisbon*: This city is a test bed for Microsoft, Cisco, Massachusetts Institute of Technology, and McLaren Electronic Systems experimenting with the Living PlanIT Urban Operating System (UOS), a distributed real-time control platform. It aims to produce 150% of the energy needed, manage solid waste, and recycle all water consumed. The system will be powered by a vast sensor network. All data will be captured to assist in prediction and decision making.
- *Rio de Janeiro*: In 2010, the mayor invited IBM to create a city-wide operations center connecting multiple systems from all of the city's 30 agencies with real-time data to improve management of transportation, emergencies, and big events. A coordinated response can be rolled out in the event of a crisis, such as a collapsing building. Transport systems can be shut down, emergency services mobilized, and gas supplies can be cut off, while citizens can be informed of alternative routes via Twitter.
- *Amsterdam*: Amsterdam Smart City is a collaboration between citizens, businesses, and government to try to save energy by implementing smart technology and by changing people's behavior. The aim is to reduce carbon dioxide levels to 40% of their 1990 levels by 2025.
- *Dublin*: The city is a test bed for IBM, which is working with the council to collect and analyze traffic data in an effort to ease congestion. Journey information is updated every minute. Residents can look up information online to find the quickest route to their destination. A digital map of the city is being built, overlaid with the real-time position of buses.
- *Dubuque, Iowa*: Smarter Sustainable Dubuque (SSD) is an initiative launched in 2009 to help the city meet residents' environmental, economic, and social equity needs. It is a public/private partnership between the city, IBM Watson Research Center's Global Smarter Planet initiative, and others. One completed project is Smarter Water, in which the city used data analytics and cloud computing offerings from IBM to reduce residents' water usage. In 2010, the city created a cloud-based portal that helped 151 households conserve an average 7% in water consumption. Another pilot program called Smarter Travel uses a smartphone application developed by IBM to collect anonymous data using RFID technology on where people travel. The city and its partners will analyze movement patterns to improve mass transit within Dubuque.
- The Smart Santander project aims to be a laboratory for prototyping and developing technologies in a real environment (Sanchez et al. 2011). The city contains 20,000 experimental sensors that capture information from different services.
- *London*: In the mayor's office, a wall of iPads displays information about how the city is performing. Intel has funded smart city research at London's Imperial College and University College. Cisco has set up a Future Cities center in Shoreditch. Living PlanIT is using the Greenwich peninsula as a test bed for its urban operating system.

15.4 Summary

The smart city agenda has been criticized for paying too much attention on technology and neglecting the human dimension. For instance, New Songdo has been criticized for being more efficient but not very livable. Most smart city projects have been developed by engineers, technologists, and planners with insufficient attention to social dynamics, psychology, anthropology, and other insights into the social dimension of cities (Da Silva et al. 2013).

In the United States, adoption of smart cities is not a certainty. It will depend on appropriate standards, technologies, and market success. For instance, one of the crucial technologies, ZigBee, has been around but used a proprietary protocol incompatible with the ubiquitous TCP/IPs until the recent SEP2 in 2013.

The market can be fickle: This was demonstrated in a 2009 smart meters trial in Bakersfield by California utility Pacific Gas and Electric (PG&E). A combination of utility rate increases, and high temperatures and other factors caused utility bills to increase drastically, provoking distrust of smart meters by consumers and consumer advocates. As a result, the California legislature blocked future smart grid deployments temporarily.

References

Al-Hader, M. 2009. Smart city components architecture. In *Proceedings of International Conference on Computational Intelligence, Modelling and Simulation*, Brno, Czech Republic, pp. 93–97.

Allwinkle, S. and P. Cruickshank. 2011. Creating smart-er cities: An overview. *Journal of Urban Technology* 18:1–16.

Anthopoulos, L. and A. Vakali. 2012. Urban planning and smart cities: Interrelations and reciprocities. In *The Future Internet*, Lecture Notes in Computer Science, Vol. 7281, F. Álvarez et al. (eds.), pp. 178–189. New York: Springer.

Bach, B., D. Wilhelmer, and P. Palensky. 2010. Smart buildings, smart cities and governing innovation in the new millennium. In *Proceedings of Eighth IEEE International Conference on Industrial Informatics (INDIN)*, Osaka, Japan, pp. 8–14.

Celino, I. and S. Kotoulas. 2013. Smart cities. *IEEE Internet Computing* 17:8–11.

Chen, H. et al. 2009. The design and implementation of a smart building control system. In *Proceedings of IEEE International Conference on e-Business Engineering*, Macau, China, pp. 255–262.

Chourabi, H. et al. 2012. Understanding smart cities: An integrative framework. In *Proceedings of 45th Hawaii International Conference on System Sciences*, Maui, Hawaii, pp. 2289–2297.

City Protocol Society, City protocol, http://www.cityprotocol.org. Accessed September 20, 2014.

Da Silva, W. et al. 2013. Smart cities software architectures: A survey. In *Proceedings of 28th Annual ACM Symposium on Applied Computing (SAC'13)*, Coimbra, Portugal, pp. 1722–1727.

Deakin, M. and H. Al Waer. 2011. From intelligent to smart cities. *Intelligent Buildings International* 3:133–139.

Elmangoush, A. et al. 2013. Design aspects for a reference M2M communication platform for smart cities. In *Proceedings of Ninth International Conference on Innovations in Information Technology (IIT)*, Al-Ain, Abu-Dhabi, UAE, pp. 204–209.

Evangelatos, O., Samarasinghe, K., and J. Rolim. 2012. Evaluating design approaches for smart building systems. In *Proceedings of IEEE Ninth International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, Las Vegas, Nevada, pp. 1–7.

Fantacci, R. et al. 2013. Enabling technologies for smart building, what's missing? In *Proceedings of AEIT Annual Conference*, Mondello, Palermo, Italy, pp. 1–5.

Farias, C. et al. 2013. A control and decision system for smart buildings. In *Proceedings of 10th International Conference on Ubiquitous Intelligence and Computing*, Vietri sul Mare, Italy, pp. 254–261.

Fernandez, P., Jara, A., and A. Skarmeta. 2013. Evaluation framework for IEEE 802.15.4 and IEEE 802.11 for smart cities. In *Proceedings of Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Taichung, Taiwan, pp. 421–426.

Filipponi, L. et al. 2010. Smart city: An event driven architecture for monitoring public spaces with heterogeneous sensors. In *Proceedings of Fourth International Conference on Sensor Technologies and Applications*, Venice/Mestre, Italy, pp. 281–286.

Gale International, Songdo IDB, http://www.songdo.com, accessed September 20, 2014.

Gil-Castineira, F. et al. 2011. Experiences inside the ubiquitous Oulu smart city. *IEEE Computer* 44:48–55.

Harris, A. 2012. Smart buildings. *Engineering and Technology* 7:52–54.

Hart, G. 1992. Nonintrusive appliance load monitoring. *Proceedings of the IEEE* 80:1870–1891.

Jin, J. et al. 2014. An information framework of creating a smart city through internet of things. *IEEE Internet of Things Journal* 1(2):112–121.

Jung, M. et al. 2013. Building automation and smart cities: An integration approach based on a service-oriented architecture. In *Proceedings of 17th International Conference on Advanced Information Networking and Applications Workshop*, Barcelona, Spain, pp. 1361–1367.

Juraschek, F. et al. 2012. Toward smart Berlin—An experimental facility for heterogeneous smart city infrastructures. In *Proceedings of IEEE 37th Conference on Local Computer Networks Workshops (LCN Workshops)*, Clearwater, FL, pp. 886–892.

Komninos, N. 2002. *Intelligent Cities: Innovation, Knowledge Systems, and Digital Spaces*. London, U.K.: Spon Press.

Lee, E.-K., Chu, P., and R. Gadh. 2013. Fine-grained access to smart building energy resources. *IEEE Internet Computing* 17:48–56.

Liu, P. and Z. Peng. 2013. Smart cities in China. *IEEE Computer* 99:1.

Martins, J. 2012. Smart homes and smart buildings. In *Proceedings of 13th Biennial Baltic Electronics Conference*, Tallinn, Estonia, pp. 27–38.

Mascar, Masdar city, http://www.masdar.ae/en/#city/all. Accessed September 20, 2014.

Morvaj, B., L. Jugaric, and S. Krajcar. 2011. Demonstrating smart buildings and smart grid features in a smart energy city. In *Proceedings of Third International Youth Conference on Energetics (IYCE)*, Leiria, Portugal, pp. 1–8.

Mulligan, C. and M. Olsson. 2013. Architectural implications of smart city business models: An evolutionary perspective. *IEEE Communications Magazine* 51:80–85.

Nam, T. and T. Pardo. 2011. Conceptualizing smart city with dimensions of technology, people, and institutions. In *Proceedings of 12th Annual International Conference on Digital Government Research*, College Park, Maryland, pp. 282–291.

Naphade, M. et al. 2011. Smarter cities and their innovation challenges. *IEEE Computer* 44:32–39.

Pellicer, S. et al. 2013. A global perspective of smart cities: A survey. In *Proceedings of Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Taichung, Taiwan, pp. 439–444.

Pentikousis, K., D. Zhu, and H. Wang. 2011. Network infrastructure at the crossroads: The emergence of smart cities. In *Proceedings of 15th International Conference on Intelligence in Next Generation Networks (ICIN)*, Berlin, Germany, pp. 109–114.

Penya, Y. et al. 2013. Smart buildings and the smart grid. In *Proceedings of 39th Annual Conference of the IEEE Industrial Electronics Society (IECON 2013)*, Vienna, Austria, pp. 113–117.

Sanchez, L. et al. 2011. SmartSantander: The meeting point between future Internet research and experimentation and the smart cities. In *Proceedings of Future Network and Mobile Summit (FutureNetw)*, Warsaw, Poland, pp. 1–8.

Shelby, Z. and C. Bormann. 2009. *6LoWPAN: The Wireless Embedded Internet*. New York: Wiley.

Sinopoli, J. 2010. *Smart Building Systems for Architects, Owners, and Builders*. Burlington, MA: Butterworth-Heinemann.

Snoonian, D. 2003. Smart buildings. *IEEE Spectrum* 40:18–23.

Srivastava, L. and A. Vakali. 2012. Towards a narrative-aware design framework for smart urban environments. In *The Future Internet*, Lecture Notes in Computer Science, Vol. 7281, F. Álvarez et al. (eds.), pp. 166–177. New York: Springer.

Suciu, G. et al. 2013. Smart cities built on resilient cloud computing and secure Internet of things. In *Proceedings of 19th International Conference on Control Systems and Computer Science*, Bucharest, Romania, pp. 513–518.

Tariq, M. et al. 2012. Smart grid standards for home and building automation. In *Proceedings of 2012 IEEE International Conference on Power System Technology (POWERCON)*, Auckland, New Zealand, pp. 1–6.

Vienna University of Technology, European Smart Cities, http://www.smart-cities.eu. Accessed September 20, 2014.

Vlacheas, P. et al. 2013. Enabling smart cities through a cognitive management framework for the Internet of things. *IEEE Communications Magazine* 51:102–111.

Wang, S. 2010. *Intelligent Buildings and Building Automation*. Abingdon, U.K.: Spon Press.

Weng, T. and Y. Agarwal. 2012. From buildings to smart buildings—Sensing and actuation to improve energy efficiency. *IEEE Design and Test of Computers* 29:36–44.

Zanella, A. et al. 2014. Internet of things for smart cities. *IEEE Internet of Things Journal* 1:22–32.

Zengjun, Z. and C. Jin. 2013. Insigma's blue print of Chinese smart city with Cisco. In Technology Management for Emerging Technologies (PICMET 13), San Jose, CA, pp. 97–102.

16

Imaging Technology

16.1 Introduction **16**-1
16.2 Scanning Tomographic Acoustic Microscopy **16**-2
16.3 Synthetic-Aperture Sonar Imaging **16**-5
16.4 Ground-Penetrating Radar Imaging **16**-5
16.5 Terahertz Imaging **16**-7
16.6 Optical Composite Imaging **16**-8
Oil Well Imaging • Construction of 3D Terrains from Image Sequences • Three-Dimensional Endoscopy
16.7 Summary **16**-13
References **16**-13

Hua Lee
University of California

16.1 Introduction

The development of imaging systems has been an important branch in the advancement of sensors and sensing technology. It represents the full-scale integration of mathematical analysis, physical modeling, hardware and software implementation, image visualization and display, and computer–human interactions. Its direct applications play a vital role in the industry, and its basic research components have been critical to graduate engineering education. The purpose of this chapter is to provide a comprehensive overview of five imaging systems as examples of the development and optimization of the imaging technology.

Active imaging operations involve the use of probing waveforms. Corresponding to various applications, the probing formats can be ultrasound, acoustic, microwave, optical, x-ray, and many others. The formulation of the probing waveforms can be in the form of continuous wave (CW), wideband pulses, or frequency modulated continuous wave (FMCW) signals. The data acquisition modalities can be monostatic, bistatic, or multistatic, operating in transmission or reflection mode. In addition, the wavefield detection can be conducted with physical element arrays or synthetic apertures. Because of the varieties, the design and development of imaging systems can be in different forms. Yet, upon close examination, the commonalities are strikingly significant.

In this chapter, five tomographic imaging systems are selected to illustrate the profound relationships and similarities among the imaging modalities and applications. The systems include the scanning tomographic acoustic microscopy (STAM) [1–8], synthetic-aperture sonar (SAS) imaging [9–13,15], ground-penetrating radar (GPR) imaging [16–18], terahertz imaging, and multiangle composite optical imaging [14,19–22]. These five imaging systems cover illumination modalities of acoustic, microwave, and optical, with illumination waveforms of CW or wideband pulse, in either transmission or reflection mode, with physical element or synthetic apertures.

The applications illustrated with these systems cover the inspection of biomedical specimens, nondestructive evaluation (NDE) monitoring of civil structures, and underwater survey. These systems were designed and developed at the Imaging Systems Laboratory of the University of California, Santa Barbara. And subsequent to laboratory experiments, all five systems have been deployed and utilized in the direct industry.

16.2 Scanning Tomographic Acoustic Microscopy

Conventional acoustic microscopy has been limited to the imaging of thin planar specimens. The research objective of the tomographic acoustic microscopy program is to achieve 3D acoustic imaging at the microscopic scale and advance the resolving capability especially in the depth direction.

STAM's imaging modality is single-frequency illumination in the transmission mode with multiple observation angles. The operating frequency of the illumination acoustic plane waves is 100 MHz, and the data acquisition of the acoustic wavefield is performed by a focused scanning laser beam and subsequently through a knife-edge detector. Figure 16.1 shows the STAM prototype system, converted from a conventional scanning laser acoustic microscope (SLAM).

Subsequent to the knife-edge detector, the signal is down-converted from 100 to 32.4 MHz, followed by a quadrature receiver. This allows the detection of both the amplitude and phase of the waveforms. Because of the availability of the phase information, the detected waveforms can be backward propagated down toward the 3D subsurface region to form a holographic image. Figure 16.2a is the image of a subsurface range finder from the conventional SLAM system, and Figure 16.2b is the STAM image of the subsurface layer, illustrating the focused image of the range finder.

The subsurface 3D imaging capability can be further improved by elevating it to the tomographic imaging mode by operating in the format of multiprojection tomography. This can be achieved by performing rotational scans with an electronically controlled mechanical platform. Figure 16.3a is the

FIGURE 16.1 STAM.

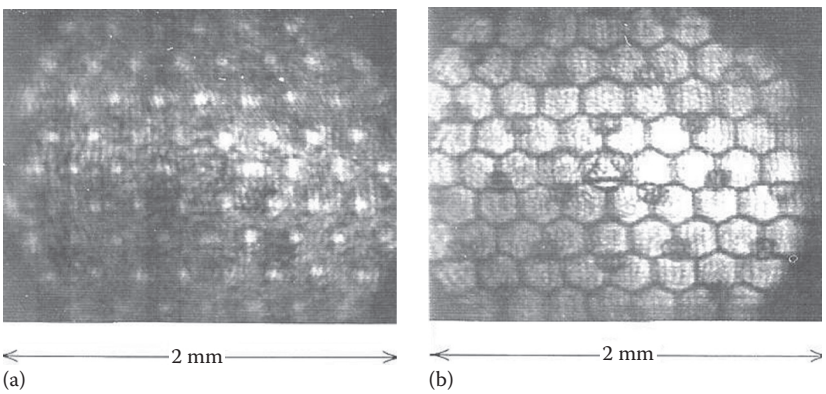


FIGURE 16.2 (a) SLAM image of a range finder. (b) STAM image.

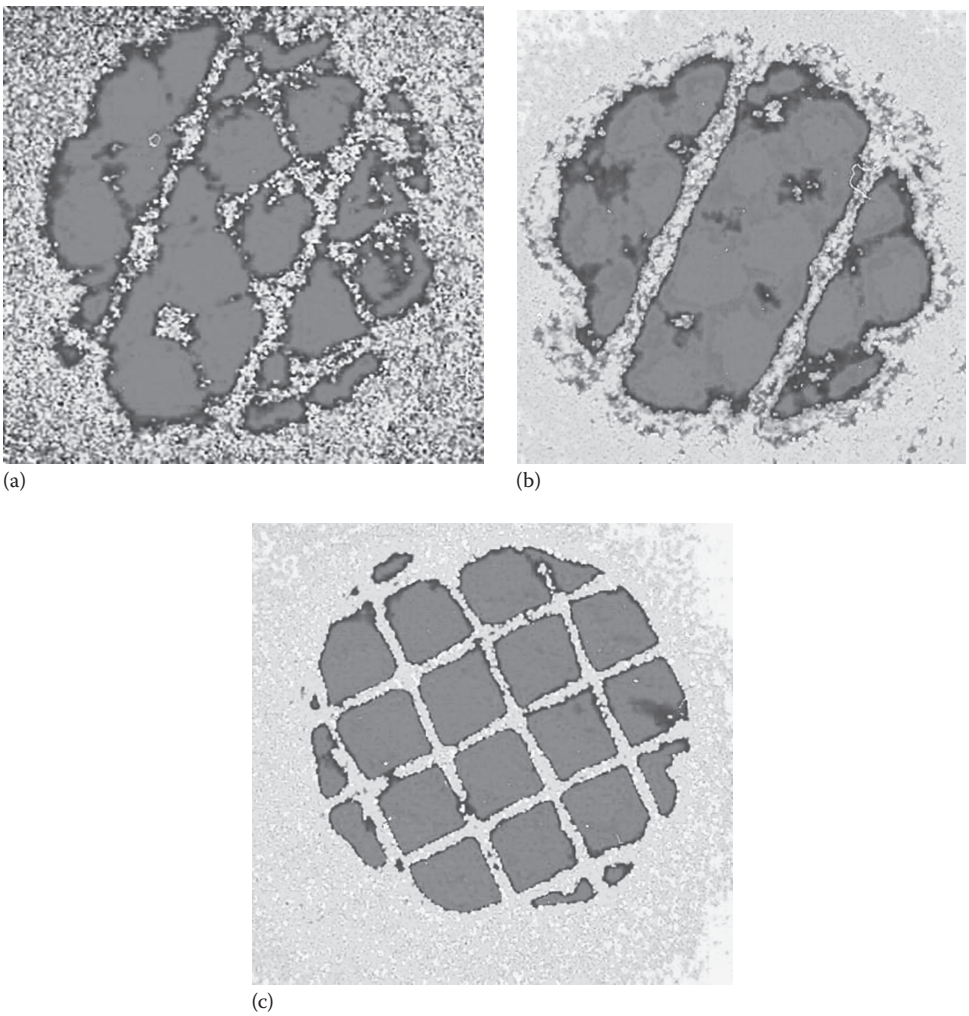
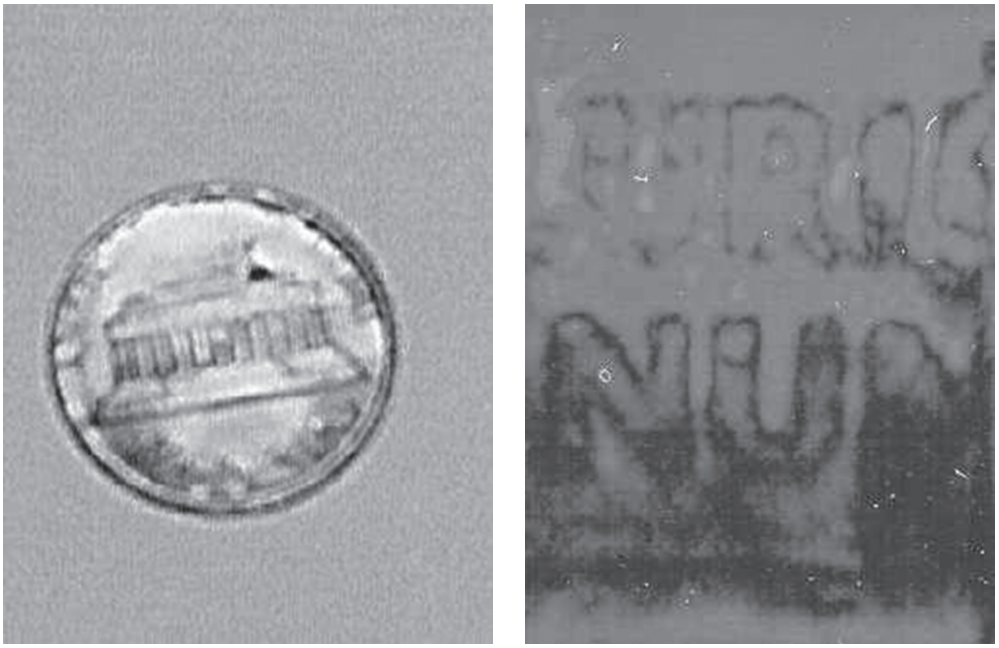


FIGURE 16.3 (a) The conventional SLAM image of two subsurface layers. (b) STAM image of the first layer. (c) Image of the second layer.

conventional SLAM image of two subsurface layers. Figure 16.3b is the STAM image of the first layer, and Figure 16.3c is the image of the second layer. This further illustrates the improvement of resolving capability in the depth direction.

Tomographic acoustic imaging can be enhanced with the multiple-frequency data. This is to operate the data acquisition process with a collection of CW frequencies. It enables us to further improve the resolution in the range direction. Figure 16.4 shows the experiment of the imaging of a penny. The transmission mode waveforms are detected on the opposite side of the penny by the STAM system. Then the detected complex waveforms are backward propagated to form the image of the surface profile of the penny at the opposite side.

Because the STAM system is capable of the detection of the complex wavefield, it allows us to reconstruct 3D images with complex amplitude. For the use of CW plane waves, the phase information of the image can be converted to the propagation time delay, which is directly related to the propagation velocity of the specimens. Since the propagation velocity of material is directly related to the density, the phase information can be mapped into the density distribution of the specimen. This technique

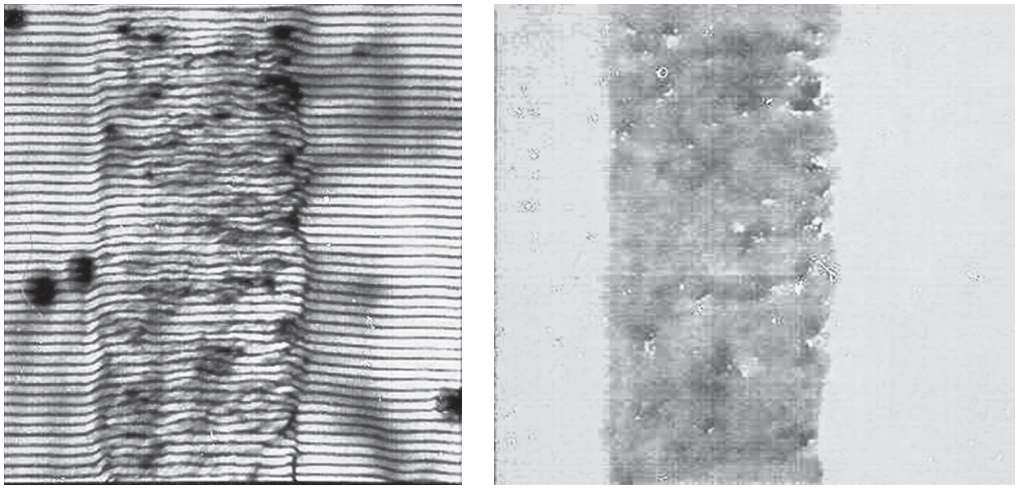


(a) (b)

FIGURE 16.4 (a) The penny and (b) STAM image of the penny formed from the waveforms at the opposite side.

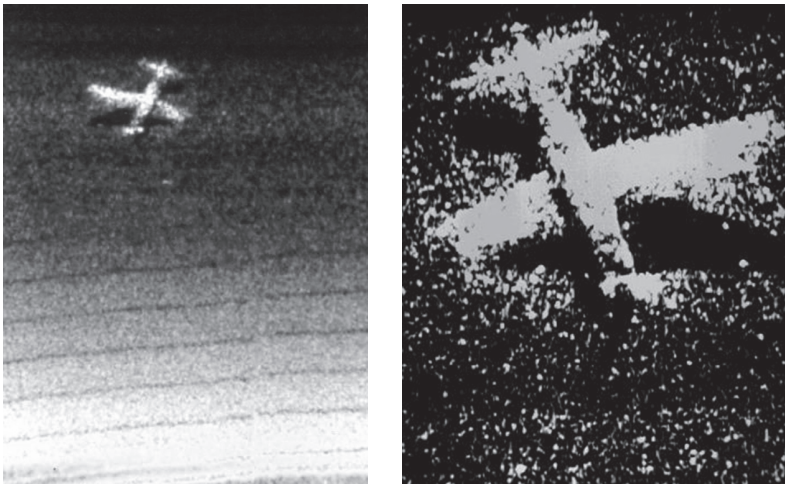
is applied to imaging of the hardening of biological tissues. Figure 16.5a is the interference pattern of a piece of liver specimen, which is the conventional approach to the estimation of tissue hardening. Figure 16.5b is the density profile computed from the phase profile of the STAM image. It shows clearly the hardening areas of the specimen.

This research program was supported by the National Science Foundation and 3M. The development of this research project has been deployed and implemented onto Sonoscan's SLAM systems.



(a) (b)

FIGURE 16.5 (a) Interference pattern of the liver specimen. (b) Density profile converted from the STAM image.



(a) (b)

FIGURE 16.6 (a) Reconstructed image of a sunken airplane. (b) Enlarged version of the image.

16.3 Synthetic-Aperture Sonar Imaging

SAS imaging is the acoustic equivalent of synthetic-aperture radar (SAR) imaging. The applications cover a wide range of oceanic search, surveys, and mapping. It normally functions in the reflection mode with a multielement array. The field experiment was conducted in the San Diego Bay with a linear 10-element sonar array operating in the side-looking linear-scan model. The use of a multielement array is to produce the redundancy for the estimation of platform motion. The platform motion consists of six parameters, of which three are associated with the translational vector and another three associated with the rotation matrix in three dimensions. The synthetic aperture is 1D, and thus, the final image is 2D. Figure 16.6a is the reconstructed image of a sunken airplane, and Figure 16.6b is the enlarged version.

This research program was supported by the UC MICRO Program in collaboration with Sonatech.

16.4 Ground-Penetrating Radar Imaging

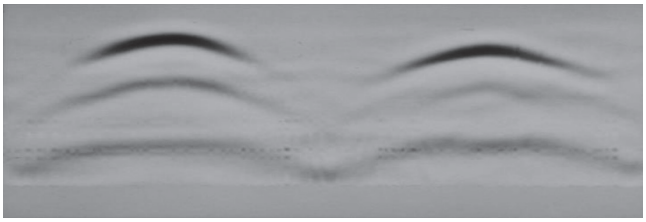
One important microwave subsurface sensing application is the GPR imaging. It is the reflection mode imaging in the microwave frequency range. Typically, it operates in the monostatic pulse-echo mode. Two-dimensional or 3D surveys can be performed through synthetic-aperture data acquisition. Applications include subsurface surveys and NDE of civil structures.

The experiments documented in this chapter are performed with a laboratory prototype. The illumination probing signal is a CW pulse at the operating frequency of 1 GHz. The useable bandwidth is approximately 100 MHz. The antenna is a bow-tie antenna.

Figure 16.7a shows the laboratory setup of the experiments. Figure 16.7b is the signature of the microwave echoes, and Figure 16.7c is the reconstructed cross-sectional image, showing the four rebars embedded in the concrete specimen. For this experiment, it was operated in the monostatic pulse-echo mode with a linear 1D synthetic aperture.

The synthetic-aperture scan can be expanded by repeating the linear scan, to form a 2D planar synthetic aperture. As a result, 3D tomographic subsurface images can be formed. Figure 16.8 is the 3D image of the internal structure, showing two layers of rebars. The image also shows the artifacts at the two corners, due to the reflections from the edges of the concrete specimen.

(a)



(b)

(c)

FIGURE 16.7 (a) Laboratory setup of the experiments. (b) Signature of the microwave echoes. (c) Reconstructed cross-sectional image of embedded rebars.

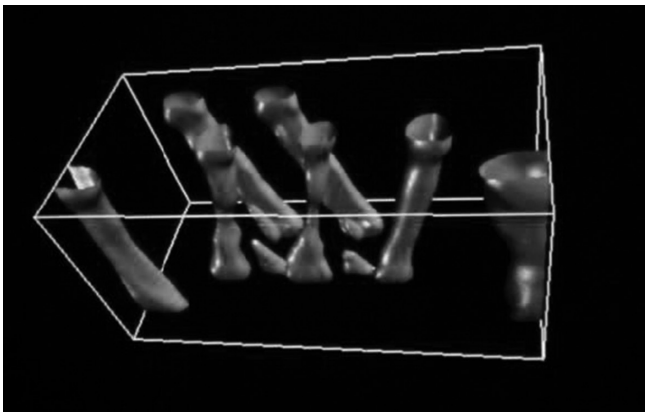
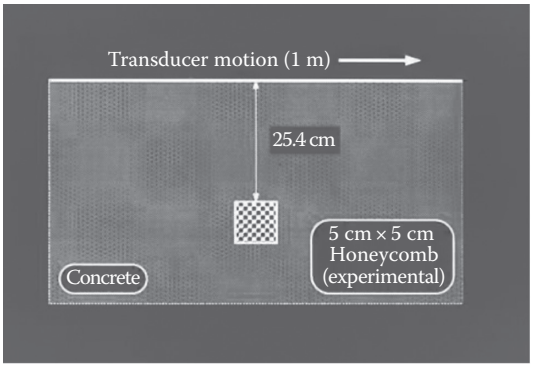
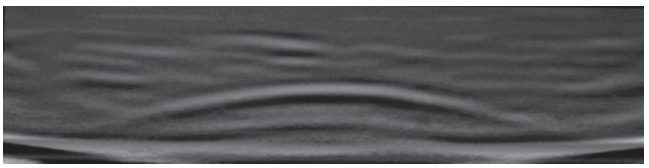


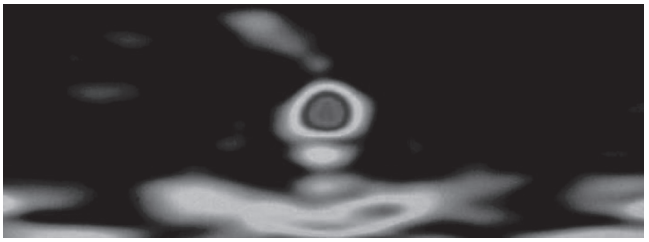
FIGURE 16.8 Three-dimensional image of the internal structure.



(a)



(b)



(c)

FIGURE 16.9 (a) Arrangement of the test specimen (honeycomb). (b) Signature of the microwave echoes. (c) Reconstructed image (honeycomb).

With the same laboratory setup, the experiment was repeated with a specimen containing a honeycomb region. The honeycomb structure represents the deterioration of the cement material. Figure 16.9a shows the arrangement of the test specimen. Figure 16.9b is the signature of the microwave echoes, and Figure 16.9c is the reconstructed cross-sectional image, clearly showing the honeycomb region.

The system was later deployed to conduct a full-scale field test at the New York Capitol building. Figure 16.10a shows the building structure of the test area. Figure 16.10b is the signature of the microwave echoes, and Figure 16.10c is the reconstructed image, showing the subsurface profiles.

This research program was supported by the US Army Research Office, National Science Foundation, the US Department of Transportation, and the Lawrence Livermore National Laboratory. The research results have been transferred to Akela and the Special Technology Laboratory of the US Department of Energy.

16.5 Terahertz Imaging

Terahertz imaging is a recent advancement, taking advantage of the fundamental properties of both microwave and optical sensing systems. Due to the complexity of the hardware, it operates in the form of raster scan with focused beams, in the reflection mode. It is capable of covering many traditional

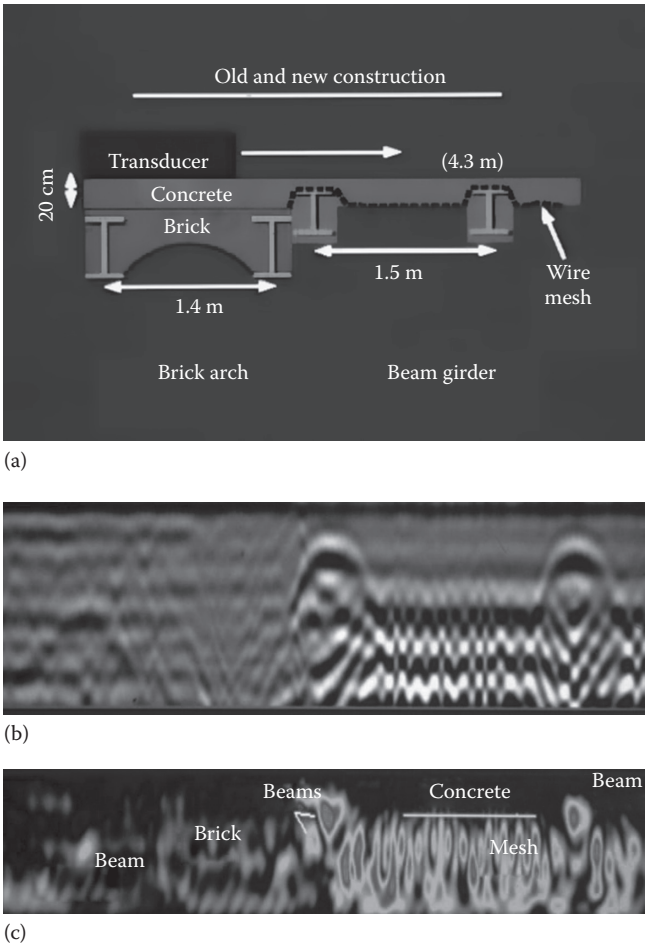


FIGURE 16.10 (a) Building structure of the test area. (b) Signature of the microwave echoes. (c) Reconstructed image of the subsurface profiles.

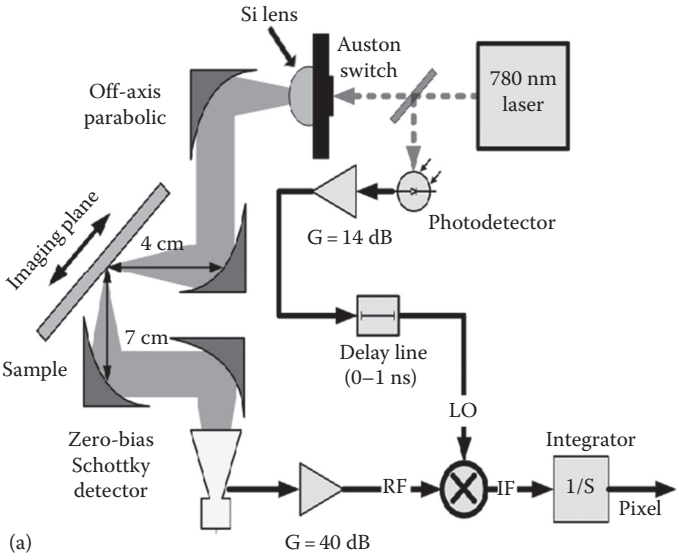
applications of microwave and optical applications. Figure 16.11a shows the system diagram of the terahertz imaging system, and Figure 16.11b shows the hardware setup of the laboratory experiments.

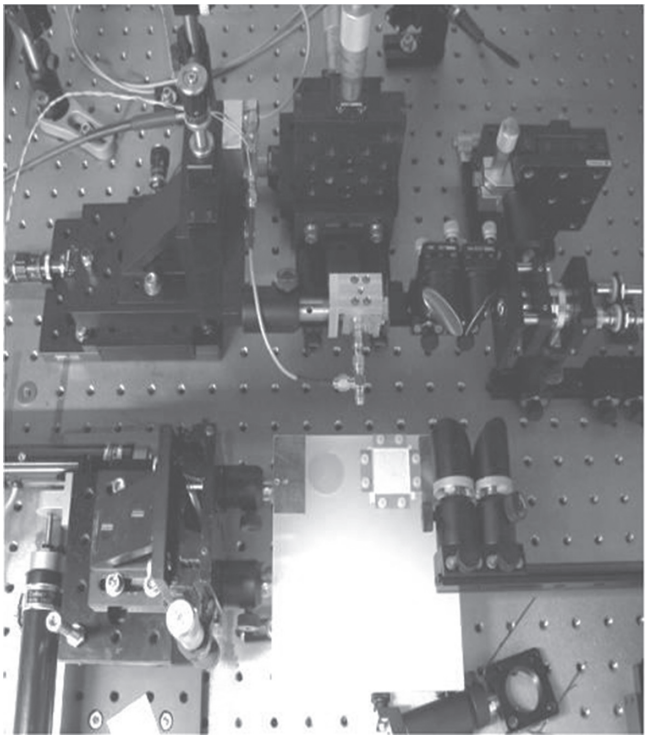
The resolution capability of the terahertz images can be formulated in a form similar to that of the optical systems. And, because the penetration capability is similar to the behavior in the microwave range, it is capable of imaging through material normally opaque to optical systems. Figure 16.12 is the terahertz subsurface image of the burned skin through layers of cloth cover.

This research program was supported by the US Army Medical Command and the National Science Foundation. The laboratory prototype has been deployed to UCLA School of Medicine for clinical trials.

16.6 Optical Composite Imaging

High-resolution tomographic microwave and acoustic imaging is typically performed by combining a collection of subimages. Because of the combination, the expansion of the information contents is achieved by the increase in the bandwidth of the probing waveforms or the angular span of the data acquisition systems. This concept can also be applied to the optical domain, combining images from different viewing perspectives.





(b)

FIGURE 16.11 (a) System diagram of the terahertz imaging system. (b) Hardware setup of the system.

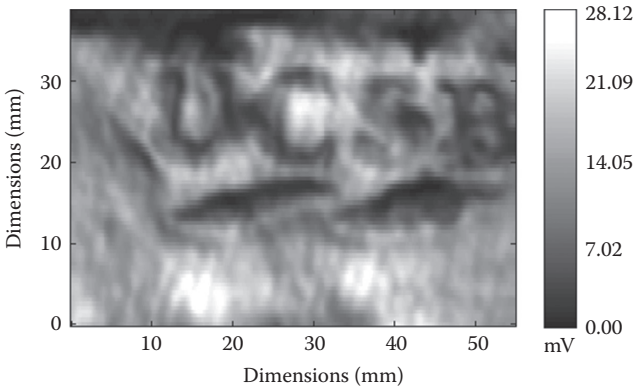


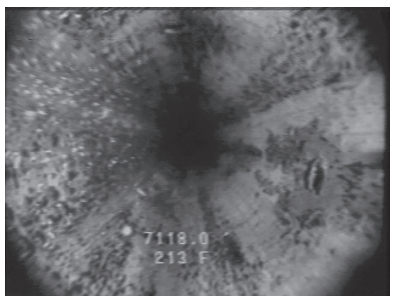
FIGURE 16.12 Terahertz subsurface image.

16.6.1 Oil Well Imaging

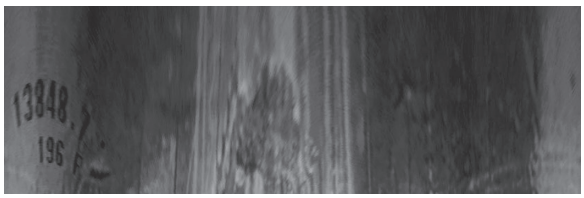
The first example is the vision system for the survey of the oil wells. The objective is to visualize the defects and deterioration of the wall of the wells. To conduct the survey, a wide-angle camera system is towed along the oil well at a steady rate. During the scan, a video sequence is recorded for the survey.

Figure 16.13a shows one image frame of the survey video. The camera is pointing toward the center axis of the oil well. Figure 16.13b is the unfolded version of the image frame, projecting the image frame onto the cylindrical wall surface. It can be seen that the resolution of the unfolded image is not uniform.

Platform motion is common due to the unsteady tow of the camera system. Thus, subsequently, motion estimation and compensation are conducted to calculate for the relative displacement between adjacent frames for accurate image registration. Figure 16.13c is the composite image using the high-resolution portion from 78 image frames, giving the high-resolution image of a section of the oil well wall.

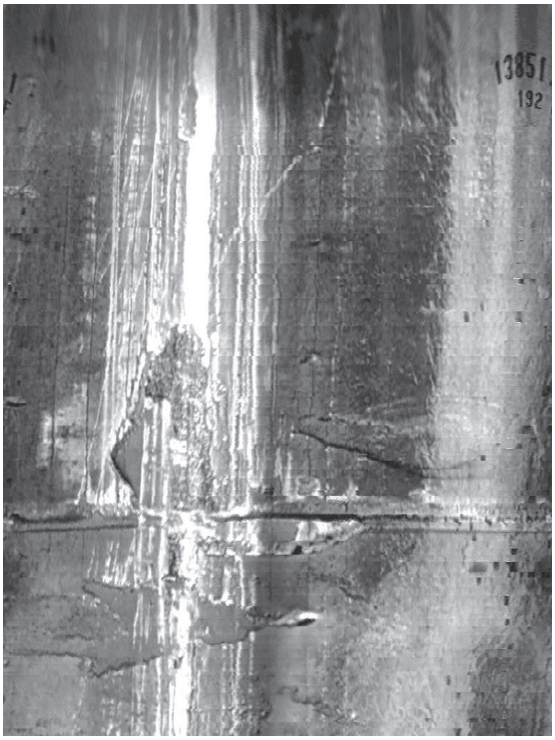


(a)



(b)

FIGURE 16.13 (a) Image frame of the survey video. (b) Unfolded version of the image frame. (*Continued*)

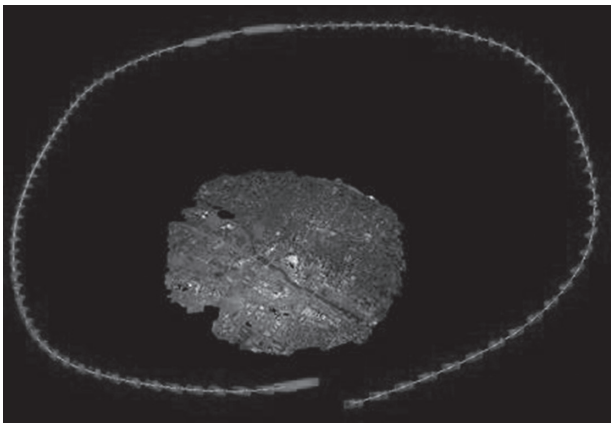


(c)

FIGURE 16.13 (***Continued***) (c) Composite image using the high-resolution portion from 78 image frames.

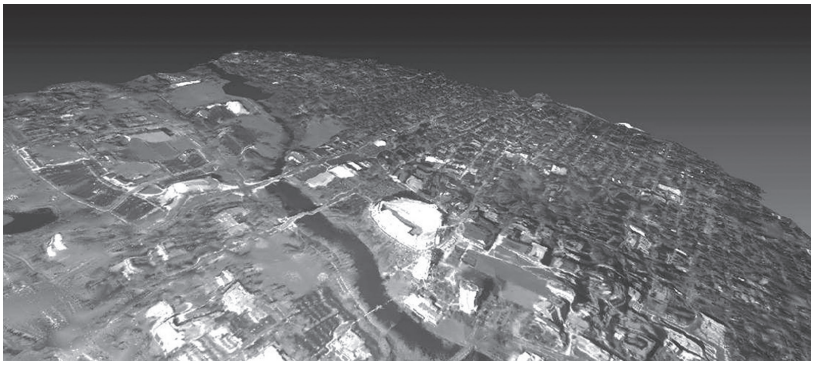
16.6.2 Construction of 3D Terrains from Image Sequences

The same concept can be applied to the survey of 3D terrains. Figure 16.14a shows the 3D composite image from the video sequence recorded during the survey flight. The relative positions between frames can be estimated directly from the image sequence, as part of the process. The data acquisition positions along the circular synthetic-aperture flight path are marked as a light shaded grayscale in the image.



(a)

FIGURE 16.14 (a) Three-dimensional composite image from a survey flight. (*Continued*)



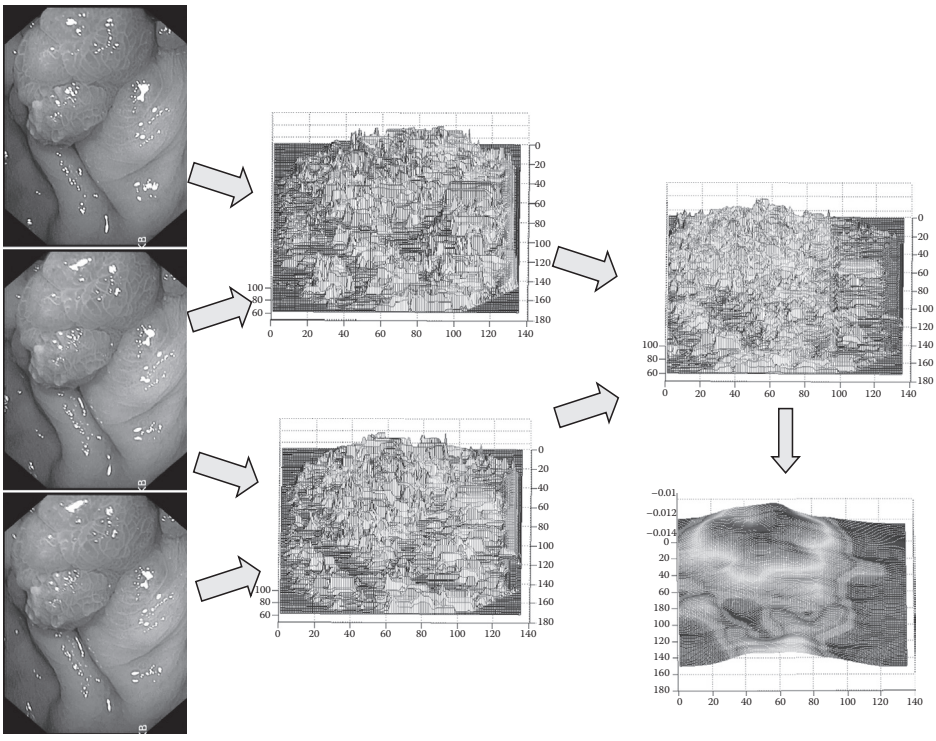
(b)

FIGURE 16.14 (*Continued*) (b) Enlarged version of the 3D reconstructed profile.

By combining the 2D image frames accurately from the video sequence, a 3D terrain profile can be constructed and stored. Then it can be viewed from any selected perspective. Figure 16.14b is the enlarged version of the 3D reconstructed profile, viewed from a selected perspective.

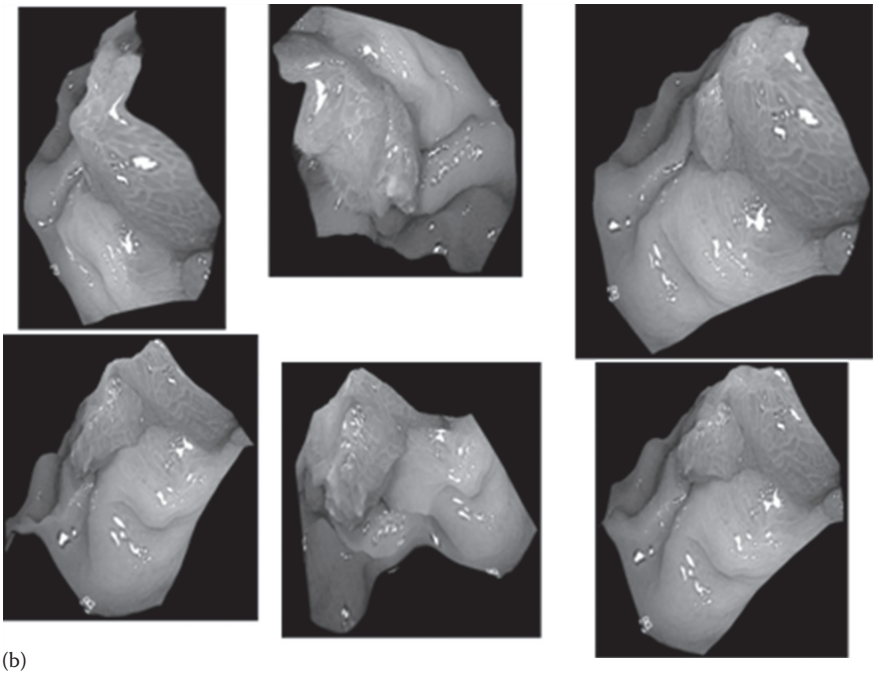
16.6.3 Three-Dimensional Endoscopy

The concept of constructing the 3D profile from a video sequence can also be applied medical endoscopy. Figure 16.15a shows the procedure of combining 2D image frames for the estimation of the 3D profile. After the formation of the 3D profile, one can view the 3D profile from any elected perspective.



(a)

FIGURE 16.15 (a) Combination of 2D image frames for the estimation of the 3D. (*Continued*)



(b)

FIGURE 16.15 (*Continued*) (b) Views from six different perspective angles.

Figure 16.15b shows the view of the 3D profile from six different perspective angles. This capability allows the surgeons to examine the region of interest from the preferred perspective to improve the effectiveness of hand–eye coordination.

This research program was supported by DHV International and Karl Storz Imaging. It demonstrates the feasibility of constructing 3D terrain profiles from a collection of 2D image frames. The key elements of this procedure include feature selection, estimation of feature correspondences, and superposition. This section presents three applications, including the inspection of the oil well walls, 3D terrain reconstruction from video sequences of the survey flight, and 3D endoscopic imaging. All three applications utilize camera systems operating in the synthetic-aperture mode within the optical domain. Thus, although the applications are seemingly different, the mathematical analysis and signal processing techniques are similar.

16.7 Summary

The main purpose of this chapter is to provide a comprehensive overview of the design, development, and optimization of high-performance tomographic imaging systems and the applications. Five imaging systems, including (1) STAM, (2) SAS imaging, (3) GPR imaging, (4) terahertz imaging, and (5) multiangle composite optical imaging, were selected to illustrate the commonalities and similarities. These five examples show the use of acoustic, microwave, or camera systems, with CW or pulse probing waveforms, physical arrays, or synthetic apertures, in transmission or reflection modes.

References

1. H. Lee and G. Wade, *Imaging Technology*, IEEE Press, New York, 1986.
2. Z.-C. Lin, H. Lee, and G. Wade, Back-and-forth propagation for diffraction tomography, *IEEE Transactions on Sonics and Ultrasonics*, SU-31(6), 626–634, 1984.

3. Z.-C. Lin, H. Lee, and G. Wade, Scanning tomographic acoustic microscope: A review, *IEEE Transactions on Sonics and Ultrasonics*, SU-32(2), 168–180, 1985.
4. H. Lee and C. Ricci, Modification of the scanning laser acoustic microscope for holographic and tomographic imaging, *Applied Physics Letters*, 49(20), 1336–1338, 1986.
5. Z.-C. Lin, H. Lee, G. Wade, M.G. Oravecz, and L.W. Kessler, Holographic image reconstruction in scanning laser acoustic microscopy, *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, UFFC-34(3), 293–300, 1987.
6. H. Lee and R.Y. Chiao, Holographic acoustic microscopy for quantitative velocity profile imaging, *Journal of Acoustical Society of America*, 85(3), 1375–1376, 1989.
7. R.Y. Chiao and H. Lee, Scanning tomographic acoustic microscopy, *IEEE Transactions on Image Processing*, 4(3), 358–369, 1995.
8. R.Y. Chiao, D. Kent, and H. Lee, Projection data error estimation and correction for high-resolution tomographic acoustic microscopy, *Journal of Acoustical Society of America*, 103(5), 2492–2501, 1998.
9. B.L. Douglas, J.M. Silkaitis, and H.Lee, Single-reference calibration technique for multiple-element array sonar imaging systems, *Journal of Acoustical Society of America*, 100(1), 400–407, 1996.
10. H. Lee, An overview of synthetic-aperture image reconstruction algorithms for GPR imaging with pulse-echo and step-frequency FMCW systems, *Journal of Environmental and Engineering Geophysics*, 8(2), 105–114, 2003.
11. B.L. Douglas and H. Lee, Synthetic-aperture active sonar imaging, *Proceedings of the 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. III, pp. 37–40, 1992.
12. B.L. Douglas and H. Lee, Synthetic-aperture sonar imaging with a multiple element receiver array, *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. V, pp. 445–448, 1993.
13. J.M. Silkaitis, B.L. Douglas, and H. Lee, Motion compensation in synthetic aperture sonar imaging, *Acoustical Imaging*, Vol. 21, J.P. Jones (ed.), Plenum Press, New York, pp. 745–754, 1995.
14. H. Lee, Synthetic-aperture technique for high-resolution composite imaging the inside walls of tubular specimens, *International Journal on Imaging Systems and Technology*, 14, 167–169, 2004.
15. H. Lee, Motion compensation by phase correction for synthetic-aperture side-scan sonar imaging, *International Journal on Imaging Systems and Technology*, 14, 159–261, 2005.
16. J.E. Mast, B. Edgar, T.W. Wall, J.P. Murtha, and H. Lee, Impulse radar imaging: Applications to historic buildings, *Proceedings of American Society for Nondestructive Testing Conference*, 1992.
17. M.S. D'Errico, B.L. Douglas, and H.Lee, Subsurface microwave imaging for nondestructive evaluation of civil structures, *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. V, pp. 453–456, 1993.
18. H. Lee and J.P. Murtha, Pulse-echo microwave tomographic imaging and object recognition for NDE of civil structures and materials, *Proceedings of the Second International Conference on Imaging Technologies: Techniques and Civil Engineering Applications*, 1997.
19. D. Koppel, Y.F. Wang, and H. Lee, Automated image rectification in video-endoscopy, *Proceedings of the Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2001.
20. D. Koppel, Y. Wang, and H. Lee, Image-based rendering and modeling in video-endoscopy, *Proceedings of the IEEE International Symposium on Biomedical Imaging*, 2004.
21. H. Lee, Synthetic-aperture technique for high-resolution composite imaging the inside walls of tubular specimens, *International Journal on Imaging Systems and Technology*, 14, 167–169, 2004.
22. D. Koppel, Y-F. Wang, and H. Lee, Image-based view rendering in endoscopy, *International Journal on Imaging Systems and Technology*, 14, 181–185, 2005.
23. Z.D. Taylor, R.S. Singh, M.O. Culjat, J.Y. Suen, W.S. Grundfest, H. Lee, and E.R. Brown, Reflective THz imaging of porcine skin burns, *Optics Letters*, 33(11), 1258–1260, 2008.
24. Z.D. Taylor, R.S. Singh, D.B. Bennett, P. Tewari, C.P. Kealey, N. Bajwa, M.O. Culjat et al., THz medical imaging: In vivo hydration sensing, *IEEE Transactions on Terahertz Science*, 1(1), 201–219, 2011.

17

Reconfigurable MIMO FMCW Imaging Technique and Applications in Acoustical and Microwave Imaging

Hua Lee
University of California

Michael Lee
University of California

17.1 Introduction **17**-1
17.2 Image Reconstruction Algorithm **17**-2
Step-Frequency FMCW System • Superposition of the Range Profiles
17.3 Resolution Analysis **17**-6
Case of Centered Aperture • Active Illumination • The Monostatic Case • Centered Aperture of the Monostatic Case • Step-Frequency FMCW Systems
17.4 FMCW Medical Ultrasound and Microwave Imaging **17**-13
17.5 Summary **17**-16
References **17**-17

17.1 Introduction

Most imaging systems are designed to have rigid and structured apertures, in linear, planar, or circular configuration. The structured aperture configurations enable the application of convolution operations for image reconstruction, which substantially simplifies the computation complexity with Fourier transform operations [1].

In many imaging applications, well-structured and organized data-acquisition arrays may not be physically feasible. In order to perform high-performance imaging under these conditions, one approach is to remove the requirement of structured aperture arrays to accommodate the physical nature of the data-acquisition regions. In return, the structured nature of the data-acquisition process is reallocated to the organization of the illumination waveforms.

This chapter introduces the concept and approach of the alternative operating modality, known as step-frequency FMCW imaging. In this format, the system transmits a sequence of FMCW pulses as the probing signals with a uniform frequency increment, instead of the traditional format of a single pulse. At each frequency step, the system functions exactly like a CW system. This allows the major portion of the computation to be shifted to the estimation of the range profiles, which indirectly provides the flexibility for reconfigurable data-acquisition array structures.

This chapter consists of three parts. The first section covers the image formation algorithm of the FMCW technique. It starts with a simple review of the concept of range estimation of traditional pulse-echo system. Then the probing pulse waveform is replaced with a sequence of FMCW signals to illustrate that the range profile can be obtained by a simple Fourier transform operation. And a subsequent linear superposition procedure converts all the range profiles into the final image [2–14].

The second section is the resolution analysis of the FMCW imaging modality. The resolution analysis is conducted in the spatial-frequency domain, based on the bandwidth coverage of the spectral distributions. It quantitatively defines the resolving capability in both range and cross-range directions. Part three is the laboratory experiments of the FMCW prototypes in the application areas of medical ultrasound and microwave subsurface imaging [15–19].

17.2 Image Reconstruction Algorithm

The pulse-echo model is the most fundamental imaging configuration. The concept is the foundation of many imaging techniques. To perform data acquisition of the imaging process in the pulse-echo mode, a probing signal $p(t)$ is transmitted:

$$E_T(t) = p(t) \quad \text{for } 0 \le t \le T \tag{17.1}$$

In practice, the probing signal has a finite time duration T and bandwidth B. Then the waveform from a single reflector detected at the receiver is in the form of

$$E_R(t) = \alpha E_T(t-\tau) = \alpha p(t-\tau) \tag{17.2}$$

where

α denotes the attenuation, which is due to the spread of the beam and propagation loss

τ is the time delay due to the propagation:

$$\tau = \frac{r}{v} \tag{17.3}$$

where

v is the propagation speed

r is the total travel distance from the transmitter to the target and then to the receiver, during the data-acquisition process

Subsequently, we perform cross-correlation of the returned signal with the transmitted waveform to estimate the time delay τ:

$$\begin{aligned} R_E(t) &= E_R(t) \times E_T \times (-t) \\ &= \alpha [p(t-\tau) \times p \times (-t)] \\ &\triangleq \alpha R_p(t-\tau) \end{aligned} \tag{17.4}$$

where $R_p(t)$ is the autocorrelation of the probing signal $p(t)$. $R_p(t)$ is Hermitian symmetrical with a peak centered at $t = 0$. Thus, the result of the cross-correlation process $R_E(t)$ is the time-delay profile, consisting a peak at the accurate time-delay position $t = \tau$, with amplitude α.

Now, if we generalize the case for continuous source distribution, the time-delay profile can be written in the form of $\alpha(t)$. The received signal is a combination of shifted versions of the probing waveform weighted by $\alpha(t)$, which can be written as the convolution of time-delay profile $\alpha(t)$ and probing waveform $E_T(t)$:

$$\begin{aligned} E_R(t) &= \int \alpha(\tau) E_T(t-\tau)\,d\tau \\ &= \int \alpha(\tau) p(t-\tau)\,d\tau \\ &= \alpha(t) * p(t) \end{aligned} \tag{17.5}$$

This implies that the received signal is the convolution of the unknown time-delay profile $\alpha(t)$ and the probing signal $p(t)$. If we describe the data-acquisition procedure as a filtering process, the input is $\alpha(t)$ and the impulse response of the filter is the probing signal $p(t)$. Now, the goal is to estimate the unknown time-delay profile $\alpha(t)$ from the received signal $E_R(t)$, with the knowledge of the probing signal $p(t)$.

The traditional time-delay estimation is the correlation of the received signal $E_R(t)$ with the probing waveform $p(t)$. Because the correlation with $p(t)$ can be formulated as a convolution with $p^*(-t)$, it becomes

$$\begin{aligned} R_E(t) &= E_R(t) * p^*(-t) \\ &= [\alpha(t) * p(t)] * p^*(-t) \\ &= \alpha(t) * [p(t) * p^*(-t)] \\ &= \alpha(t) * R_p(t) \end{aligned} \tag{17.6}$$

This also means that the estimation procedure for the time-delay profile is a matched filtering process, with the impulse response $p^*(-t)$. Then, the range–distance profile $R(r)$ can be formulated by rescaling the time-delay profile $R_E(t)$ based on the linear relationship $t = r/v$:

$$R(r) = R_E\left(\frac{r}{v}\right) \tag{17.7}$$

This formula shows that the result of the range estimation process is in a form of convolution of the true time-delay profile $\alpha(t)$ with the autocorrelation of the probing waveform $R_p(t)$. Thus, the autocorrelation of the probing waveform governs the quality of the range estimation process. Because the autocorrelation function is the Fourier transform of the power spectrum, the resolving capability of the range estimation is determined by the power spectrum of the probing signal and thus its bandwidth B. This suggests that wider bandwidth of the probing waveform results in better range resolution.

17.2.1 Step-Frequency FMCW System

Step-frequency FMCW is an alternative imaging modality. Instead of one single probing waveform, the step-frequency FMCW system transmits a complete sequence of N coherent signals, stepping through a defined frequency band B with frequency increment Δf:

$$f = f_0 + k\Delta f \quad \text{where } k = 0, 1, 2, \ldots, N-1 \tag{17.8}$$

During each frequency step, the transmitter functions exactly like a CW system. So, from this angle, the step-frequency FMCW modality can be regarded as an organized sequence of CW operations.

For simplicity, the transmitted signal can be written in the phasor form of

$$E_T(t) = E\exp(-j2\pi ft) \tag{17.9}$$

For each frequency step, this is equivalent to operating in the pulse-echo mode with a unity probing function and a carrier frequency *f*. In other words, it is to repeat the pulse-echo mode *N* times, with *N* different carrier frequencies. The pulse period is sufficiently long to be considered as a CW format. That is why FMCW imaging systems typically require longer data-acquisition period. In return, the quality of the final images is better due to lower noise level.

As indicated earlier, for a propagation distance *r*, the travel time is $\tau = r/v$, where *v* is the propagation speed. Then, responding to a single target, the reflected signal detected by the receiver is in the form of a weighted and delayed version of the transmitted waveform:

$$E_R(t) = \alpha E\exp(-j2\pi f(t-\tau)) = \alpha E\exp(-j2\pi f(t-r/v)) \tag{17.10}$$

where α denotes the weighting is due to the target reflectivity and propagation loss, and the delay is due to the round-trip travel time. At each frequency step, after demodulation, the received signal becomes

$$\begin{aligned} E_R(t)E_T^*(t) &= \alpha|E|^2\exp(j2\pi fr) \\ &= \alpha|E|^2\exp(j2\pi f(r/v)) \end{aligned} \tag{17.11}$$

Note that, after the demodulation, it is no longer a function of time. For each frequency, the output after the demodulation is a complex scalar. Through the *N* frequency steps, a complete illumination cycle produces an *N*-point sequence {*E*(*k*)} from the demodulated received waveforms:

$$\begin{aligned} E(k) &= E_R(t)E_T^*(t) \\ &= \alpha|E|^2\exp(j2\pi(f_0 + k\Delta f)(r/v)) \\ &= \alpha|E|^2\exp(j2\pi f_0(r/v))\exp(j2\pi k\Delta f(r/v)) \end{aligned} \tag{17.12}$$

The term $\exp(j2\pi k\Delta f(r/v))$ is the only one as a function of the frequency-stepping index *k*. To analyze this further, we match the core part of the sequence against the kernel of the *FFT* operator

$$\exp(j2\pi k\Delta f(r/v)) = \exp(j2\pi nk/N) \tag{17.13}$$

The matching results in a simple relationship:

$$\frac{n}{N} = \Delta f\frac{r}{v} \tag{17.14}$$

It is then simplified down to a linear relationship between the *FFT* index *n* and the propagation distance *r*:

$$n = N(\Delta fr/v) = (N\Delta f)r/v = \frac{Br}{v} \tag{17.15}$$

where B is the bandwidth of the waveform, defined as $B = N\Delta f$. This means, for a point reflector at distance r, the demodulated received signal can be represented as the *FFT* spectrum of a point sequence $\delta(n-n_o) = \delta(n-Br/v)$ and the complex amplitude of the point sequence is $\alpha|E|^2\exp(-j2\pi f_0 (r/v))$:

$$\begin{aligned}\mathrm{FFT}\left\{E_R(t)\,E_T^*(t)\right\} &= \alpha|E|^2\exp(j2\pi f_0(r/v))\,\mathrm{FFT}\left\{\exp(j2\pi k\Delta f(r/v))\right\}\\ &= \alpha|E|^2\exp(j2\pi f_0(r/v))\,\delta(n-Br/v)\\ &= \alpha|E|^2\exp(j2\pi f_0(r/v))\,\delta(n-n_o)\end{aligned} \tag{17.16}$$

where the location of the peak is at

$$n_o = \frac{B}{v}r \tag{17.17}$$

Thus, the estimation of the range profile from the stepped-frequency FMCW system can be achieved with a simple *FFT* operator. This suggests that the range profile and the sequence $E(k)$ are directly related as a Fourier transform pair that each index n of the spectral sequence is corresponding to the total travel distance r:

$$\mathrm{Range\,distance} = r = \frac{v}{B}n \tag{17.18}$$

Since the index n is an integer, the scaling factor v/B represents the increment of the range profile, which is often referred to as the range resolution:

$$\mathrm{Range\,resolution} = \Delta r = \frac{v}{B} \tag{17.19}$$

17.2.2 Superposition of the Range Profiles

At each receiver position (x_m, y_m, z_m), we compute the range profile $\hat{s}(x_m, y_m, z_m, r)$. For simplicity, we denote it as $\hat{s}_m(r)$. This represents the range profile estimated at the mth receiver position (x_m, y_m, z_m):

$$\hat{s}_m(r) = \hat{s}(x_m, y_m, z_m, r) \tag{17.20}$$

Conceptually, we can spread a range profile $\hat{s}_m(r)$ over the region of interest to form a subimage $\hat{s}_m(x,y,z)$, where (x, y, z) is the pixel position within the region of interest. This process can also include an additional weighting $w_m(r)$. With the weighting, the range profile can be modified in the form of $w_m(r)\,\hat{s}_m(r)$ prior to the formation of the subimage $\hat{s}_m(x,y,z)$. It is also common to expand the weighting function $w(r)$ to incorporate the following:

1. The range factor in Green's function
2. Normalization factor for limited aperture size
3. Beam patterns
4. Range compensation for the propagation loss due to distance

The formation of the mth subimage $\hat{s}_m(x,y,z)$ is a simple conversion from the mth range profile $\hat{s}_m(r)$. Suppose the mth profile is corresponding to the transmitter located at (x_t, y_t, z_t) and receiver

at (x_r, y_r, z_r). For each position (x, y, z) in the region of interest, we first compute the distances to the transmitter and to the receiver, respectively:

$$\begin{aligned} r &= r_1 + r_2 \\ &= \left[(x - x_t)^2 + (y - y_t)^2 + (z - z_t)^2 \right]^{1/2} + \\ &\quad + \left[(x - x_r)^2 + (y - y_r)^2 + (z - z_r)^2 \right]^{1/2} \end{aligned} \tag{17.21}$$

Then, we use the sum of the travel distances, $r = r_1 + r_2$, to seek and assign the complex amplitude from the mth range profile $\hat{s}_m(r)$ at $r = r_1 + r_2$ to the pixel of $\hat{s}_m(x, y, z)$ to form the mth subimage:

$$\hat{s}_m(x, y, z) = \hat{s}_m(r_1 + r_2) \tag{17.22}$$

Then, subsequent step of the image reconstruction procedure is the superposition of the M sets of range profiles to form the final image:

$$\hat{s}(x, y, z) = \frac{1}{M} \sum_{m=1}^{M} \hat{s}_m(x, y, z) \tag{17.23}$$

This is equivalent to summing up the values extracted from all the range profiles corresponding to the location (x, y, z).

This superposition procedure does not require extensive computation. For each pixel position (x, y, z), we compute r_1 and r_2 corresponding to each range profile, given its transmitter and receiver positions. From each range profile, the value of range profile at range bin of $r = r_1 + r_2$ is selected. The value of the final image at (x, y, z) is the average of the M selected values from M range profiles.

To improve the computation efficiency, the values of r_1 and r_2 are precomputed and stored for each pixel to form a lookup table. For each pixel position, the lookup table contains M range-bin addresses corresponding to the M range profiles. Thus, the main task of this superposition procedure becomes a simple lookup and retrieve process.

17.3 Resolution Analysis

For simplicity, we first examine the resolving capability of a passive coherent imaging system in two dimensions with a 1D aperture, based on the spatial-frequency spectral coverage of the wave-field data samples. Resulting from a point source at (x, z), the coherent wave-field pattern the receiver position at (x', z') over the aperture is in the form of

$$\left(\frac{1}{j\lambda r} \right) \exp\left(\frac{j2\pi r}{\lambda} \right) = \left(\frac{1}{j\lambda r} \right) \exp\left(\frac{j2\pi \left((z - z')^2 + (x - x')^2 \right)^{1/2}}{\lambda} \right) \tag{17.24}$$

where λ is the coherent wavelength and $r = ((z - z')^2 + (x - x')^2)^{1/2}$ is the propagation distance from the radiating point source at (x, z) to the receiver at (x', z'). By definition, the local spatial frequency, in the x-direction, of the wave field at the receiver position (x', z') is the derivative of the phase

$$\begin{aligned} f_x &= \frac{\partial}{\partial x'} \frac{\left((z - z')^2 + (x - x')^2\right)^{1/2}}{\lambda} \\ &= \frac{(1/\lambda)(x' - x)}{\left((z - z')^2 + (x - x')^2\right)^{1/2}} \\ &= \left(\frac{1}{\lambda}\right) \sin(\theta) \end{aligned} \tag{17.25}$$

where θ is the perspective angle, defined as $\sin(\theta) = (x' - x)/((z - z')^2 + (x - x')^2)^{1/2}$. The angle θ is bounded within the interval of $(\theta_{min}, \theta_{max})$, which is governed by the angular span of the aperture.

Similarly, the local spatial frequency, in the z-direction, of the wave-field pattern at the position (x', z') is

$$\begin{aligned} f_z &= \frac{\partial}{\partial z'} \frac{\left((z - z')^2 + (x - x')^2\right)^{1/2}}{\lambda} \\ &= \left(\frac{1}{\lambda}\right) \frac{(z' - z)}{\left((z - z')^2 + (x - x')^2\right)^{1/2}} \\ &= \left(\frac{1}{\lambda}\right) \cos(\theta) \end{aligned} \tag{17.26}$$

We define the 2D spatial-frequency vector $\boldsymbol{f}$ as

$$\begin{aligned} \mathbf{f} &= \left[f_x, f_z \right] \\ &= \left(\frac{1}{\lambda}\right) \left[\sin(\theta), \cos(\theta)\right] \end{aligned} \tag{17.27}$$

We should note here that the spatial-frequency vector for the 3D case will be in the form of $\boldsymbol{f} = [f_x, f_y, f_z]$.

It is important to point out that this spatial-frequency vector is a vector pointing from the source location (x, z) to the receiver position (x', z'), with a magnitude $1/\lambda$:

$$|\mathbf{f}| = \left(f_z^2 + f_x^2\right)^{1/2} = \frac{1}{\lambda} \tag{17.28}$$

This suggests the following:

1. The spatial-frequency vector is located on a circle in the spatial-frequency domain, and the radius of the circle is $1/\lambda$.
2. The location of the spatial-frequency vector on the circle is specified by the perspective angle θ, which is defined by the receiver position with respect to the location of the source.
3. Given M receivers, corresponding to each source location, there are M spectral samples along a circle of radius $1/\lambda$. And the locations of these M spectral samples are uniquely corresponding to the perspective angles, defined by the source location and the receivers.
4. The distribution of the spectral samples varies for different source location, which produces different resolving capability.

17.3.1 Case of Centered Aperture

Corresponding to the coverage of the aperture, the spatial-frequency content occupies an arc segment of the circle of radius $1/\lambda$, and the span of the segment governs the resolving capability of the imaging system. In the cross-range direction, the *projection* of spatial-frequency segment can be written as

$$\begin{aligned}\Delta f_x &= f_{x\max} - f_{x\min} \\ &= \left(\frac{1}{\lambda}\right)\sin(\theta_1) - \left(\frac{1}{\lambda}\right)\sin(\theta_2) \\ &= \left(\frac{1}{\lambda}\right)\left[\sin(\theta_1) - \sin(\theta_2)\right]\end{aligned} \tag{17.29}$$

where θ_1 and θ_2 denote the maximum and minimum of the angular span of the aperture in the x-direction, respectively. Similarly, in the range direction, we have

$$\begin{aligned}\Delta f_z &= f_{z\max} - f_{z\min} \\ &= \left(\frac{1}{\lambda}\right)\cos(\theta_3) - \left(\frac{1}{\lambda}\right)\cos(\theta_4) \\ &= \left(\frac{1}{\lambda}\right)\left[\cos(\theta_3) - \cos(\theta_4)\right]\end{aligned} \tag{17.30}$$

where θ_3 and θ_4 denote the maximum and minimum of the angular span defined by the aperture in the z-direction, respectively.

To illustrate the formula, we consider the example of a point source at $(x, z) = (0, 0)$ and a centered aperture with the span from $(-x_o, z_o)$ to (x_o, z_o). The spatial-frequency content is on the circle of radius $1/\lambda$, occupying the angular segment from $-\theta_o$ to θ_o. So, the projection of the spatial-frequency band in the x-direction is

$$\begin{aligned}\Delta f_x &= f_{x\max} - f_{x\min} \\ &= \left(\frac{1}{\lambda}\right)\sin(\theta_0) - \left(\frac{1}{\lambda}\right)\sin(-\theta_0) \\ &= \left(\frac{2}{\lambda}\right)\sin(\theta_0)\end{aligned} \tag{17.31}$$

where $\sin(\theta_0) = x_0/(z_0^2 + z_0^2)^{1/2}$. Thus, the resolution in the x-direction is

$$\Delta x = \frac{\lambda}{2}\sin(\theta_0) \tag{17.32}$$

which is in complete agreement with the *Rayleigh resolution limit*. And then in the z-direction, the spatial-frequency bandwidth is

$$\begin{aligned}\Delta f_z &= f_{z\max} - f_{z\min} \\ &= \left(\frac{1}{\lambda}\right) - \left(\frac{1}{\lambda}\right)\cos(\theta_0) \\ &= \left(\frac{1}{\lambda}\right)\left[1 - \cos(\theta_0)\right]\end{aligned} \tag{17.33}$$

The maximum $f_{z\max}$ is when $\theta = 0$. So, $f_{z\max}$ is typically $1/\lambda$ for the case of centered apertures. The corresponding resolution in the range direction becomes

$$\Delta z = \frac{\lambda}{\left(1-\cos(\theta_0)\right)} \tag{17.34}$$

This spatial-frequency coverage–based resolution analysis provides a clear and concise perspective of the resolving capability of an imaging system. It is important to point out that the resolution limit is fundamentally defined by the spectral distribution along a circle in the spatial-frequency domain. The projections of this arc segment onto the f_x and f_z axes formulate the *bandwidths* in the x- and z-directions, which subsequently translates into the range and cross-range resolutions.

It is important to point out that the arc segment in the spatial-frequency domain varies for different source location, since the angular perspective with respect to the aperture span changes. This formula also indicates that the resolution at different positions in the source region varies, although likely minor. In addition, one of the key elements of this approach is that it provides a quantitative and accurate formulation for resolution analysis and can be applied to various aperture configurations, continuous, detached, or discrete.

17.3.2 Active Illumination

For active multistatic systems, the region of interest is illuminated by the transmitted waveforms and the reflected wave field is then detected over the aperture. From a transmitter located at (x'', z''), the target region is modulated by the illumination waveform of

$$\begin{aligned} m(x,z) &= \left(\frac{1}{-j\lambda r}\right)\exp\left(\frac{-j2\pi r}{\lambda}\right)\dagger \\ &= \left(\frac{1}{-j\lambda r}\right)\exp\left(\frac{-j2\pi\left((z-z'')^2+(x-x'')^2\right)^{1/2}}{\lambda}\right) \end{aligned} \tag{17.35}$$

The illumination waveform can be regarded as a modulation pattern, and the local modulation frequency, in the neighborhood of the source (x, z), in the x-direction is

$$\begin{aligned} f_x &= \frac{\partial}{\partial x}\left[\frac{\left((z-z'')^2+(x-x'')^2\right)^{1/2}}{\lambda}\right]\dagger \\ &= \left(\frac{1}{\lambda}\right)\left[\frac{(x-x'')}{\left((z-z'')^2+(x-x'')^2\right)^{1/2}}\right] \\ &= \frac{\sin(-\theta')}{\lambda} = \frac{-\sin(\theta')}{\lambda} \end{aligned} \tag{17.36}$$

So, locally at (x, z), the component of the modulation waveform in the x-direction can be approximated as

$$m_x(x,z) \cong \left(\frac{1}{-j\lambda r}\right)\exp\left(-\frac{-j2\pi x\sin(\theta')}{\lambda}\right) \tag{17.37}$$

Also, in the z-direction, the modulation waveform can be approximated as

$$m_z(x,z) \cong \left(\frac{1}{j\lambda r}\right) \exp\left(\frac{-j2\pi z \cos(\theta')}{\lambda}\right) \tag{17.38}$$

This means, in combination, the illumination of the source region is mathematically equivalent to a modulation process with the spatial-frequency vector:

$$\begin{aligned} f_m &= \left[f_{mx}, f_{mz} \right] \\ &= \left[-(1/\lambda)\sin(\theta'), -(1/\lambda)\cos(\theta') \right] \\ &= (-1/\lambda)\left[\sin(\theta'), \cos(\theta') \right] \end{aligned} \tag{17.39}$$

In the image reconstruction process, subsequent to the formation of the subimage, a demodulation procedure is added for the removal of the effects of the illumination wave field, prior to the superposition process for the formation of the final image. The demodulation process is equivalent to shifting the spatial-frequency content of the subimage back by the frequency vector:

$$-f_m = \left(\frac{1}{\lambda}\right)\left[\sin(\theta'), \cos(\theta') \right] \tag{17.40}$$

As a result, the modulation–demodulation process associated with the multistatic illumination is equivalent to moving the original spatial-frequency distribution to the region of

$$f - f_m = \left(\frac{1}{\lambda}\right)\left[\sin(\theta) + \sin(\theta'), \cos(\theta) + \cos(\theta') \right] \tag{17.41}$$

17.3.3 The Monostatic Case

When $\theta' = \theta_1$, after the frequency shift, the maximum of the spatial-frequency content in the x-direction reaches

$$f_{x\max} = \left(\frac{1}{\lambda}\right)\left[\sin(\theta_1) + \sin(\theta_1) \right] = \left(\frac{2}{\lambda}\right) \sin(\theta_1) \tag{17.42}$$

Similarly, the minimum is achieved when $\theta' = \theta_2$:

$$\begin{aligned} f_{x\min} &= \left(\frac{1}{\lambda}\right)\left[\sin(\theta_2) + \sin(\theta_2) \right] \\ &= \left(\frac{2}{\lambda}\right) \sin(\theta_2) \end{aligned} \tag{17.43}$$

As a result, the bandwidth is changed to

$$\begin{aligned} \Delta f_x &= f_{x\max} - f_{x\min} \\ &= \left(\frac{2}{\lambda}\right)\left[\sin(\theta_1) - \sin(\theta_2) \right] \end{aligned} \tag{17.44}$$

And, similarly, in the range direction, we have

$$\begin{aligned}\Delta f_z &= f_{z\max} - f_{z\min} \\ &= (2/\lambda)\left[\cos(\theta_3) - \cos(\theta_4)\right]\end{aligned} \tag{17.45}$$

One simple and profound description of the modulation–demodulation effect is that the multistatic illumination moves the arc segment of the original spatial-frequency contents from the circle of radius 1/λ up to a bigger circle of radius 2/λ. Yet, the angular coverage defined by the aperture span remains exactly the same as the passive case. This means that the active multistatic illumination expands the spatial-frequency distribution by a factor of two. As a result, the resolution is also improved by the same factor.

17.3.4 Centered Aperture of the Monostatic Case

If we again consider the case of a target at $(x, z) = (0, 0)$ and a centered aperture, the spatial-frequency bandwidth in the x-direction is

$$\begin{aligned}\Delta f_x &= \left(\frac{2}{\lambda}\right)\left[\sin(\theta_0) - \sin(-\theta_0)\right] \\ &= \left(\frac{4}{\lambda}\right)\sin(\theta_0)\end{aligned} \tag{17.46}$$

And the resolution in the x-direction is

$$\Delta x = \frac{\lambda}{4\sin(\theta_0)} \tag{17.47}$$

Then, in the z-direction, the spatial-frequency bandwidth is

$$\Delta f_z = \left(\frac{2}{\lambda}\right)\left[1 - \cos(\theta_0)\right] \tag{17.48}$$

The corresponding resolution in the range direction becomes

$$\Delta z = \frac{\lambda}{2}\left(1 - \cos(\theta_0)\right) \tag{17.49}$$

17.3.5 Step-Frequency FMCW Systems

If we consider the case of FMCW systems with wideband waveforms with a temporal bandwidth B, instead of coherent signals,

$$\Delta f_t = B = f_2 - f_1 \tag{17.50}$$

where f_1 and f_2 are the lower and upper bounds of the temporal frequency band, respectively. Their corresponding wavelengths are

$$\lambda_{\max} = \frac{v}{f_1} \tag{17.51}$$

and

$$\lambda_{\min} = \frac{v}{f_2} \tag{17.52}$$

For consistency, we consider the same case of centered aperture as described previously. In the x-direction, the span of the spatial-frequency distribution ranges from $-2\sin(\theta_0)/\lambda_{min}$ to $2\sin(\theta_0)/\lambda_{min}$. The total spatial-frequency bandwidth in the x-direction is

$$\begin{aligned}\Delta f_x &= f_{x\max} - f_{x\min} \\ &= \left(\frac{2}{\lambda_{min}}\right)\left[\sin(\theta_0) - \sin(-\theta_0)\right] \\ &= \frac{4\sin(\theta_0)}{\lambda_{min}}\end{aligned} \tag{17.53}$$

Thus, the cross-range resolution is

$$\Delta x = \frac{\lambda_{min}}{4\sin(\theta_0)} = \frac{v}{4f_2\sin(\theta_0)} \tag{17.54}$$

For the wave-field component corresponding to the wavelength λ_{max}, the projection of the spatial-frequency contents along the f_z axis covers from $(2/\lambda_{max})\cos(\theta_0)$ up to $(2/\lambda_{max})$. And the swing is from $(2/\lambda_{min})\cos(\theta_0)$ up to $(2/\lambda_{min})$ for the wavelength λ_{min} and operating frequency f_2. In combination, the overall spatial-frequency span becomes

$$\begin{aligned}\Delta f_z &= f_{z\max} - f_{z\min} \\ &= (2/\lambda_{min}) - (2/\lambda_{max})\cos(\theta_0) \\ &= (2/\lambda_{min}) - (2/\lambda_{max}) + (2/\lambda_{max})\left[1 - \cos(\theta_0)\right] \\ &= 2(f_2 - f_1)/v + 2(f_1/v)\left[1 - \cos(\theta_0)\right] \\ &= (2/v)\left\{B + f_1(1 - \cos(\theta_0))\right\}\end{aligned} \tag{17.55}$$

This spatial-frequency bandwidth translates into the range resolution in the space domain:

$$\Delta z = \frac{v}{2}\left[B + f_1\left(1 - \cos(\theta_0)\right)\right] \tag{17.56}$$

This simple formula gives the range resolution in terms of both the waveform bandwidth and aperture coverage, which is of great interest and importance to the design of advanced imaging systems.

Traditional resolution analysis is conducted in the form of range and cross-range directions. However, the resolving capability of a system can also be estimated by the 2D spatial-frequency spectral coverage, for simplicity. The key advantage is that it can be coordinate independent. For example, for an FMCW imaging system, with temporal frequency band from f_1 to f_2, the operating wavelengths are from $\lambda_{min} = v/f_2$ to $\lambda_{max} = v/f_1$. The spatial-frequency spectral content is bounded between two circles, with radius $2/\lambda_{max}$ and $2/\lambda_{min}$, where the factor of two comes from the multistatic imaging modality.

If the angular span, defined by the aperture configuration, is $\Delta\theta = \theta_{max} - \theta_{min}$, the 2D spatial-frequency bandwidth coverage is

$$\begin{aligned} W &= \left(\frac{\Delta\theta}{2\pi}\right)\pi\left[\left(2/\lambda_{min}\right)^2 - \left(2/\lambda_{max}\right)^2\right] \\ &= 2\Delta\theta\left[\left(1/\lambda_{min}\right)^2 - \left(1/\lambda_{max}\right)^2\right] \\ &= 2\Delta\theta\left[\left(1/\lambda_{min}\right)^2 - \left(1/\lambda_{max}\right)^2\right] \\ &= \left(2\Delta\theta/v^2\right)\left[f_2^2 - f_1^2\right] = 4\Delta\theta B f_c/v^2 \end{aligned} \tag{17.57}$$

where f_c denotes the center frequency of the FMCW band as $f_c = (f_1 + f_2)/2$. This formula suggests that larger bandwidth, higher center frequency, and larger aperture span are governing parameters for wider bandwidth and consequently better resolution [2,3].

To simplify further, we can partition this formula into two parts:

$$W = \left(\frac{2B}{v}\right)\left(\frac{2\Delta\theta f_c}{v}\right) \tag{17.58}$$

The first term, $(2B/v)$, governs the range resolution, and the second term, $(2\Delta\theta\, f_c/v)$, defines the cross-range resolution. This simple formula is particularly accurate for systems with small apertures. Then, in the space domain, the resolution-cell size is

$$\Delta s = \left(\frac{v}{2B}\right)\left(\frac{v}{2\Delta\theta\, f_c}\right) \tag{17.59}$$

where the first term, $v/2B$, is the resolution-cell dimension in the range direction, and the second term, $v/2\Delta\theta\, f_c$, is the width in the cross-range direction.

17.4 FMCW Medical Ultrasound and Microwave Imaging

One of the direct applications of the FMCW imaging technique is in the area of medical ultrasound. Because this method conducts the estimation of the range profiles first, the geometrical configuration of the transceiver arrays becomes flexible and reconfiguration is feasible. Figure 17.1 is the micro flexible ultrasound transceiver array prototype for laboratory experiments. Figure 17.2 is the data-acquisition hardware of the imaging device.

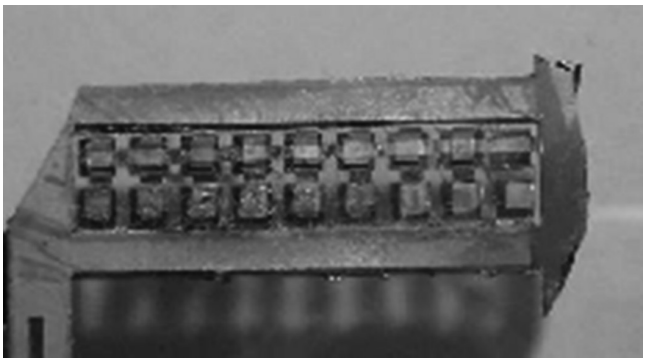


FIGURE 17.1 Micro flexible ultrasound transceiver array prototype.

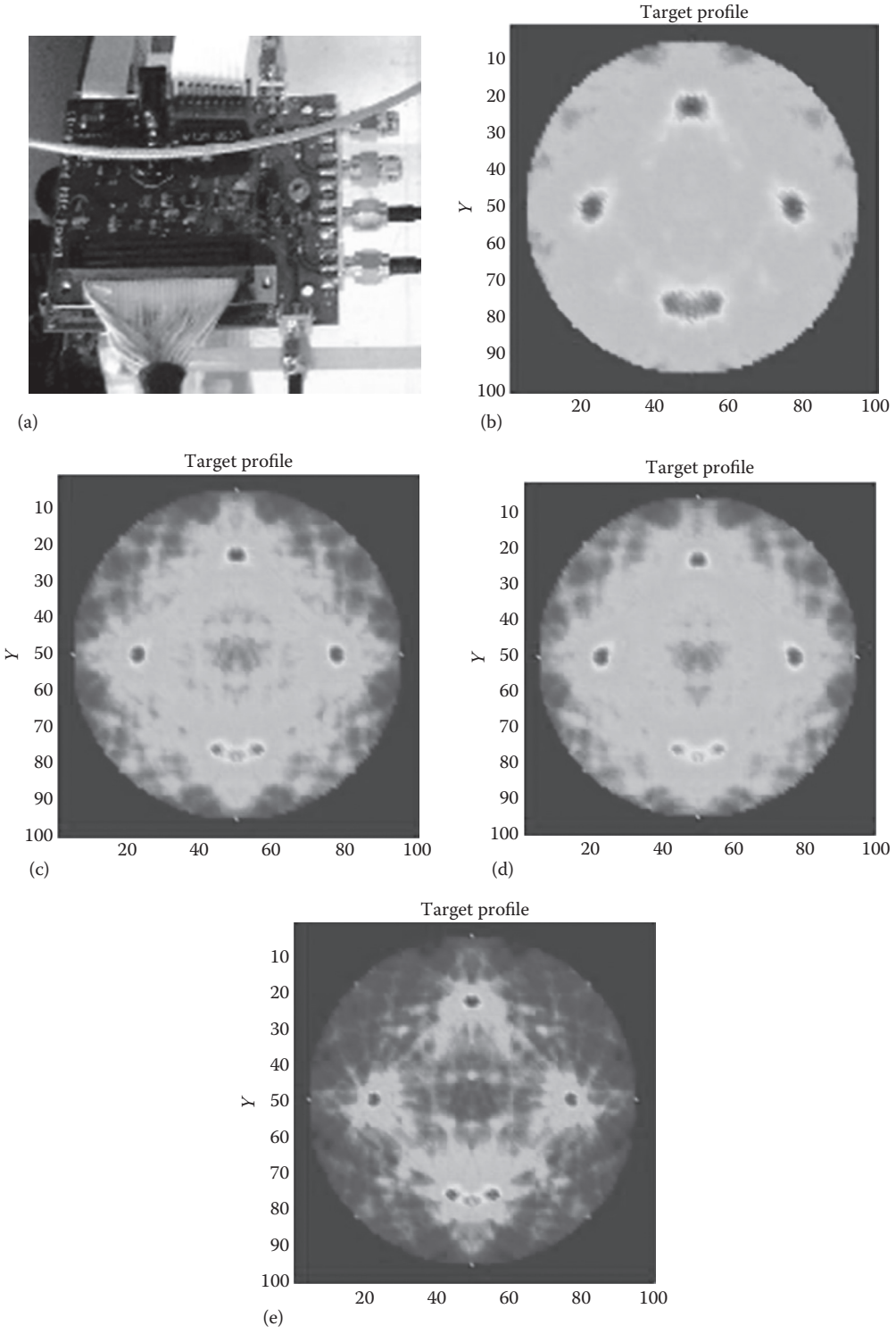


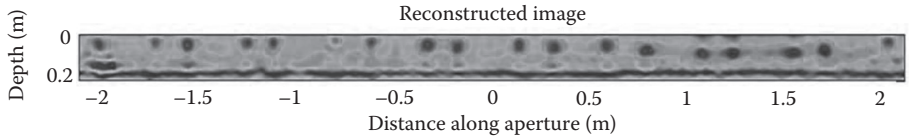
FIGURE 17.2 (a) Data-acquisition hardware of the ultrasound imaging device. (b–e) Reconstructed images of five point targets corresponding to various FMCW bandwidth.

The length of the FMCW sequence directly defines the bandwidth of the range profile, which governs the resolution in the range direction. Thus, at a given frequency increment, longer FMCW data sequences translate into improved resolution. Figure 17.2b–d shows the reconstructed images of five point targets from FMCW data of different lengths, illustrating the improvement of resolution as the bandwidth increases. The full circular aperture is employed to provide illustration with uniform resolution in all directions.

The same concept can be applied to microwave subsurface imaging. This experiment involved a simple monostatic FMCW system operating in the synthetic-aperture mode. The operating frequency is from 0.5 to 2 GHz. There are 200 data-acquisition positions along a 4 m linear scan path. At each position, the system transmits and receives at 128 frequency steps. Figure 17.3a shows the subsurface profile of a roadway during repair. Figure 17.3b is the reconstructed image after the repair process, and the roadway was covered with cement. The image shows the distribution of the rebar structure.

Figure 17.4a shows the second experiment setup with a different FMCW microwave imaging system. The system utilizes a four-antenna array, operating in the multistatic mode. The antennas used for the experiments are Vivaldi antennas. The operating bandwidth is the same, from 0.5 to 2 GHz. Figure 17.4b is the typical range profile from the experiment. Figure 17.4c is the reconstructed image, showing the two metal targets as well as the back wall.

(a)



(b)

FIGURE 17.3 (a) Subsurface profile of a roadway and (b) image of the subsurface profile.

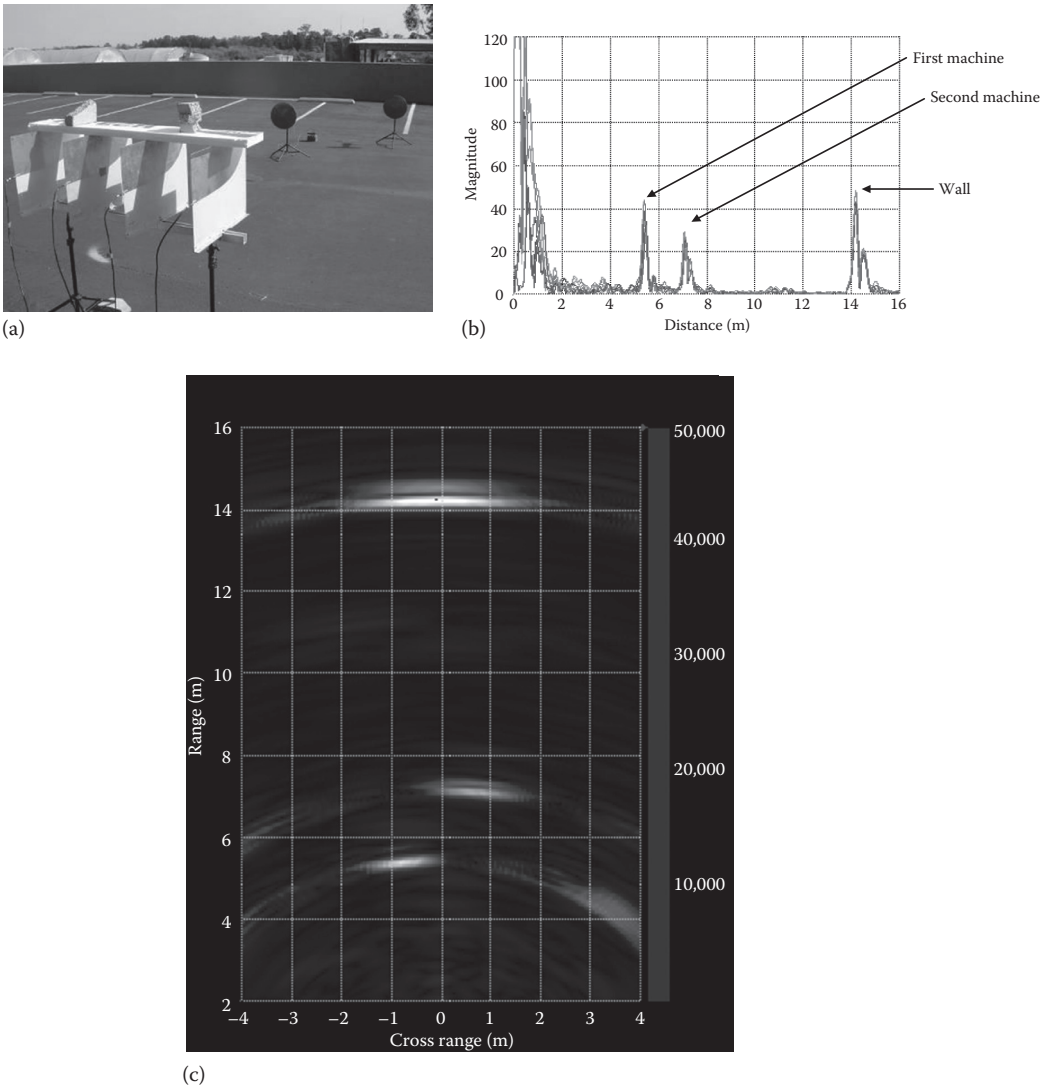


FIGURE 17.4 (a) Experiment setup of the FMCW microwave imaging system. (b) A typical range profile from the experiment. (c) Reconstructed image of the experiment.

17.5 Summary

The main objective of this chapter is to provide an overview of the foundation and concept of step-frequency FMCW imaging techniques. This approach allows us to redistribute the computation complexity in the image reconstruction process such that flexibility of the geometric configurations of the transceiver arrays becomes feasible.

After relaxing the configurations of the transceiver arrays, the reconfigurable capability of the probing waveforms can also be implemented with software-defined data-acquisition hardware. This allows the dynamic rearrangement of the distributions of the FMCW frequencies to operate the systems in different frequency bands and provide weighting and compensation to the data samples according to the propagation loss and beam-pattern variation directly.

This chapter covered the mathematical analysis, system modeling, and laboratory experiments. It starts with the overview of the procedure of the image reconstruction algorithm in the first section, which includes range estimation and linear superposition of the range profiles. Subsequently, a full-scale resolution analysis is given in the second section to provide the fundamental guidelines of the resolving capability of the FMCW imaging systems. It is then followed by results of laboratory experiments, with applications in biomedical ultrasound and microwave subsurface imaging.

References

1. H. Lee and G. Wade, *Imaging Technology*, IEEE Press, New York, 1986.
2. M. O. Culjat, D.B. Bennett, M. Lee, E.R. Brown, H. Lee, W.S. Grundfest, and R.S. Singh, Polyimide-based conformal ultrasound transducer array for needle guidance. *IEEE Sensors Journal*, 9(10), 1244–1245, 2009.
3. R.S. Singh, M.O. Culjat, S.P. Vampola, C.D. Utley, H. Lee, W.S. Grundfest, and E.R. Brown, Simulation, fabrication, and characterization of a novel flexible, conformable ultrasonic transducer array. *Proceedings of the 2007 IEEE Ultrasonics Symposium*, October 2007, pp. 1824–1827.
4. M.O. Culjat, R.S. Singh, C.D. Utley, S.P. Vampola, H. Lee, E.R. Brown, and W.S. Grundfest. A flexible, conformable ultrasound array for medical imaging. *MMVR 16*, J. D. Westwood, Ed., IOS Press, Amsterdam, Netherlands, 2008, pp. 95–97.
5. M. Lee, R.S. Singh, M.O. Culjat, S. Natarajan, B.P. Cox, E.R. Brown, W.S. Grundfest, and H. Lee. Waveform synthesis for the design and image reconstruction of FMCW ultrasound imaging systems with flexible conformal arrays. *SPIE Medical Imaging Conference*, 7265, 72650C, 2009.
6. D.B. Bennett, R.S. Singh, K. Williams, H. Lee, E.R. Brown, W.S. Grundfest, and M.O. Culjat. Micromachined flexible conformal ultrasound transducer arrays based on polyimide joints. *Proceedings of Health Monitoring of Structural and Biological Systems III Conference, 16th SPIE International Symposium on: Smart Structures and Materials & Nondestructive Evaluation and Health Monitoring*, 7295, 72951W, 2009.
7. M. Lee, R.S. Singh, M.O. Culjat, E.R. Brown, W.S. Grundfest, and H. Lee. Micro flexible ultrasound imaging array systems. *Journal of the Acoustical Society of America*, 125(4), 2514, 2009.
8. M.O. Culjat, A.E. Dann, M. Lee, D.B. Bennett, P.G. Schulam, H. Lee, W.S. Grundfest, and R.S. Singh. Transurethral ultrasound catheter-based transducer with flexible polyimide joints. *Proceedings of IEEE 2009 Ultrasonics Symposium*, 2009, pp. 2209–2212.
9. M. Lee, R.S. Singh, M.O. Culjat, S. Natarajan, B.P. Cox, E.R. Brown, W.S. Grundfest, and H. Lee. Space-time image reconstruction algorithm for diverse ultrasound transducer element distributions. *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 2009, pp. 177–180.
10. S. Natarjan, R.S. Singh, M. Lee, B.P. Cox, M.O. Culjat, H. Lee, and W.S. Grundfest. Step-FMCW signaling and target detection for ultrasound imaging systems with conformal transducer arrays. *SPIE BIOS Biomedical Optics*, 2010, 7555, 75550M, 2010.
11. M. Lee, R.S. Singh, M.O. Culjat, S. Stubbs, S. Natarajan, E.R. Brown, W.S. Grundfest, and H. Lee. Designing multistatic ultrasound imaging systems using software analysis. *Proceedings of SPIE Medical Imaging Conference*, 7629, 76290S, 2010.
12. S. Natarajan, R.S. Singh, M. Lee, B.P. Cox, M.O. Culjat, W.S. Grundfest, and H. Lee. Accurate step-FMCW ultrasound ranging and comparison with pulse-echo signaling methods. *Proceedings of SPIE Medical Imaging Conference*, 7629, 76290D, 2010.
13. R.S. Singh, M.O. Culjat, M. Lee, D.B. Bennett, S. Natarjan, B.P. Cox, E.R. Brown, W.S. Grundfest, and H. Lee. Conformal ultrasound imaging system. *Acoustical Imaging*, 30, 239–252, 2011.
14. M. Lee, W.S. Grundfest, and H. Lee. Image reconstruction algorithm for flexible FMCW MIMO ultrasound imaging systems. *Proceedings of International Telemetering Conference*, 2011.

15. M.-S. Wu and H. Lee. Remote detection and geolocation of breathing subjects by high-performance ultra-wideband FMCW MIMO microwave imaging system. *Proceedings of International Telemetering Conference*, 2011.
16. M. Lee, D. Doonan, M. Liebling, and H. Lee. Resolution analysis and system integration of a dynamically reconfigurable FMCW medical ultrasound imaging system. *Proceedings of International Telemetering Conference*, 2012.
17. M.-S. Wu and H. Lee. Dynamically reconfigurable imaging with flexible transceiver array and programmable probing waveforms. *Proceedings of International Telemetering Conference*, 2013.
18. M. Lee and H. Lee. Dynamic acoustical imaging systems with reconfigurable transceiver arrays and probing waveforms. *Program Summary of Annual Acoustical Society of America Meeting, JASA*, 134(5), 3997, 2013.
19. M. Lee, M. Liebling, and H. Lee. Modeling imaging performance of multistatic acoustic arrays nonuniform geometries. *Program Summary of Annual Acoustical Society of America Meeting, JASA*, 134(5), 4170, 2013.

18

Recent Advances in EO/IR Imaging Detector and Sensor Applications

18.1 Introduction **18**-1
18.2 UV, Visible, and IR Spectrum and Bands of Interest **18**-2
18.3 UV Nanostructured Detector Array Development: Applications of UV Imaging Technology **18**-3
Model Development for Passive and Active UV Systems Design • Systems Performance Metrics for UV Systems Design • ZnO/MgZnO Nanostructures for UV Applications • Characterization of ZnO NW Arrays Grown on the Various Substrates • Photoluminescence Measurements
18.4 Development of GaN UVAPD for UV Sensor Applications ... **18**-11
GaN/AlGaN UV APD Growth
18.5 Visible–NIR Detector Technologies **18**-15
$Si_{1-x}Ge_x$ (SiGe) Detector Arrays
18.6 SWIR Detector Technologies **18**-18
$In_xGa_{1-x}As$ Detector Array Development
18.7 Nanostructured Detector Technology for MWIR and LWIR Bands **18**-21
Design and Modeling of CNT-Based Bolometer Characteristics • Calculating the Thermal Response of the CNT Bolometer Film • Calculating the Electrical Response of the CNT Film • CNT Growth and Characterization
18.8 Summary **18**-28
Acknowledgment **18**-28
References **18**-28

Ashok K. Sood
Magnolia Optical Technologies Inc.

Yash R. Puri
Magnolia Optical Technologies Inc.

Nibir K. Dhar
Defense Advanced Research Project Agency

Dennis L. Polla
University of Minnesota

18.1 Introduction

This chapter covers recent advances in ultraviolet (UV), visible, and infrared (IR) detector technology, that is, materials and device for a variety of imaging, detection, and sensor applications. The detector array technology includes GaN/AlGaN, ZnO/MgZnO, Si, SiGe, Si micro electro mechanical system (Si MEMS), and carbon nano tube (CNT) for optical sensing applications.

IR technology is critical for military, commercial, and homeland security applications as well as used increasingly in many commercial products such as medical diagnostics, drivers enhanced vision, machine vision, and a multitude of other applications including consumer products. The key enabler of such IR products is the detector materials and designs used to fabricate focal plane arrays (FPAs). In the last decade, there have been many progresses toward the materials development and device

design innovations. In particular, significant advances have been made during the past decade in the bandgap engineering of various compound semiconductors leading to new and emerging detector architectures.

Advances in optoelectronics-related materials science, such as metamaterials and nanostructures, have opened doors for new approaches to apply detector design methodology that are expected to offer enhanced performance and low-cost products in a wide range of sensor applications.

This chapter reviews such advancements and presents different device architectures and analysis. The chapter briefly introduces the basics of IR detection physics and various IR wavelength band characteristics. The subject is divided into individual IR atmospheric transmission windows to address related materials, detector design, and performances. Advances in pixel scaling, junction formation, materials growth, and processing technologies are discussed. Single- and multispectral detector development and related applications are presented.

We cover the UV (200–350 nm), visible and near infrared (NIR) (400–1700 nm), shortwave infrared (SWIR) band (1–3 μm), and midwave infrared (MWIR) (3–5 μm) and long-wave infrared (LWIR) (8–14 μm) bands of interest for a variety of sensor applications.

We also discuss MWIR band that covers 3–5 μm and its applications. Some of the key work to be discussed will include InSb, HgCdTe, and III–V-based strained-layer superlattice (SLS) and barrier detector technologies. Each of these technologies has a place in the MWIR band where a variety of detector configurations are being used. We discuss the similarities and differences in each of these designs and their performance characteristic. We also discuss some of the exciting results that have been recently published through our technology development programs.

Long wavelength infrared (LWIR) band is of great deal of interest for a variety of applications that covers the wavelength range between 8 and 14 μm. The technologies that will be addressed are bolometer (microbolometer arrays), HgCdTe arrays, and a variety of very ingenious bandgap-engineered devices using III–V compound semiconductor materials. Theoretical and experimental consideration for the sensitivity and high signal-to-noise ratio is discussed. We also discuss various technologies that are being developed for microbolometer applications including nanostructures to increase temperature coefficient of resistance. The advances in bolometer technology leading to smartphone cameras are highlighted.

18.2 UV, Visible, and IR Spectrum and Bands of Interest

The UV spectrum has been of interest for a variety of sensors for defense and commercial applications. The UV band is from 250 to 400 nm as shown in Figure 18.1. This band can be further divided into UVA and UVB bands. Each of these bands has applications for sensors, detectors, and light-emitting diode (LED) applications.

The word "infrared" refers to a broad portion of the electromagnetic spectrum that spans a wavelength range from 1.0 μm to beyond 30 μm everything between visible light and microwave radiation. Much of the IR spectrum is not useful for ground- or sea-based imaging because the radiation is blocked by the atmosphere. The remaining portions of the spectrum are often called "atmospheric transmission windows" and define the IR bands that are usable on earth. The IR spectrum is loosely segmented into NIR (0.8–1.1 μm), SWIR (0.9–2.5 μm), MWIR (3–5 μm), LWIR) (8–14 μm), very long wave infrared (VLWIR, 12–25 μm), and far infrared (FIR, >25 μm). The MWIR–LWIR wave bands are important for the imaging of objects that emit thermal radiation, while the NIR–SWIR bands are good for imaging scenes that reflect light, similar to visible light.

Since NIR and SWIR are so near to the visible bands, their behavior is similar to the more familiar visible light. Energy in these bands must be reflected from the scene in order to produce good imagery, which means that there must be some external illumination source. Both NIR and SWIR imaging systems can take advantage of sunlight, moonlight, starlight, and an atmospheric phenomenon called "nightglow," but typically require some type of artificial illumination at night. In lieu of photon-starved scenes, arrays of IR LEDs can provide a very cost-effective solution for short-range illumination.

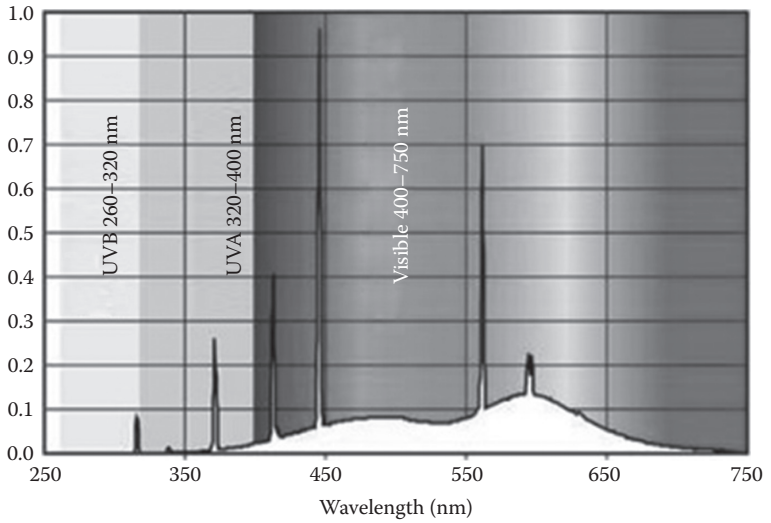


FIGURE 18.1 Overview of UV and visible spectral band. (From Spectral Charts, Raytheon Vision Systems, Goleta, CA.)

However, achieving good performance at distances of over hundreds of meters requires more directed illumination, such as a focused beam from a laser or specialized spotlight, although special consideration of eye-safety issues is required.

18.3 UV Nanostructured Detector Array Development: Applications of UV Imaging Technology

Imagery for identification of targets at various distances uses visible cameras, image intensifiers, SWIR cameras, and long-wave uncooled cameras. Each has distinct advantages and disadvantages and is useful under specific sets of conditions such as light level, thermal conditions, and level of atmospheric obscuration. The shortest wavelength is desired for spatial resolution that allows for small pixels and large formats [2–6].

Visible cameras, if adequate light level is present, can provide high resolution, but for long-range identification even under moonlit and starlit illuminations, long integration times and large optics are required, and dust, smoke, and fog easily defeat a single visible camera. Image intensifiers and SWIR cameras are useful in many conditions as the SWIR penetrates fog easily but requires fairly clear night skies for the upper atmospheric airglow light source, and image intensifiers require a certain level of celestial (starlight, moonlight) or light pollution irradiance. Both the SWIR and image intensifiers are limited by the diffraction resolution of the NIR to SWIR wavelengths [5,6].

For optimal resolution, the visible or UV spectrum is preferable; however, active (laser) illumination is required for long-range night imaging. Covert UV illumination is preferred over the visible, and the atmosphere transmits fairly well at the longer UV wavelengths. The covert active system for high-resolution identification modeled in this chapter consists of a UV laser source and a silicon CCD, AlGaN, or AlGaN avalanche photo diode (APD) FPA with pixels as small as 4 μm that are spectrally tuned for the solar-blind region of the UV spectrum. The solar-blind region is optimal as virtually all of the solar radiation is absorbed at the higher altitudes leaving a pitch dark terrain even in bright day, yet for sea-level path lengths of 1 km and shorter, the UV atmospheric transmittance is still acceptable.

This combination is ideal for exploitation by a UV illuminator and UV FPA sensor. Current UV lasers can provide either continuous or pulsed energy at levels detectable by solar-blind UV detectors under

relatively small optics and at 30 Hz frame rates, providing real-time high-resolution (on the order of 1 cm at 1 km) imagery. At these illumination levels and target ranges, both standard PN, PIN, and APD UV detectors and silicon CCDs can be used for target identification. The model has been developed and used to include the combined effects of detector and electronics, atmospheric transmittance and UV background radiance, target size, range and reflectance, and UV laser attributes to simulate and predict both CW and pulsed laser imaging performance and to assist in the design of this prototype system [6].

18.3.1 Model Development for Passive and Active UV Systems Design

The general equations for *Signal to Noise Ratio (SNR)* prediction for laser illumination and APD are derived:

$$SNR = \frac{G e_{Lret}}{[F^2 G^2 (e_{Lret} + e_{bk} + e_{dk}) + (\tilde{e}_{n,amp})^2]^{1/2}} \tag{18.1}$$

where G is the APD gain, F is the excess noise, and the noise electron terms are the laser return shot noise, the scene noise, the dark current noise, and the amp noise.

Two special and frequently occurring cases are (18.2) for the laser power noise-limited case and (18.3) for the *amp* noise-limited case:

$$SNR = \frac{[e_{Lret}]^{1/2}}{F} \tag{18.2}$$

$$SNR = \frac{G e_{Lret}}{(\tilde{e}_{n,amp})} \tag{18.3}$$

The laser return in electrons for CW assuming Lambertian reflection is

$$e_{Lret,ti} = [t_i \eta \tau_o] \left[P_{Low} \frac{\lambda}{hc} \right] \tau_a \frac{\Omega_{pix}}{\Omega_L} \rho_{tar} \tau_a^2 \frac{A_o}{\pi R^2} = [t_i \eta \tau_o] \left[P_{Low} \frac{\lambda}{hc} \right] \tau_a \frac{4A_{pix}}{\pi \theta_{fdiv}^2 f^2} \rho_{tar} \tau_a^2 \frac{A_o}{\pi R^2} \tag{18.4}$$

Or when separated into detector/optics, atmosphere, laser, and target attributes,

$$e_{Lret,ti} = \left[t_i \eta \tau_o \frac{A_{pix} A_o}{\pi f^2} \right] \tau_a^2 \frac{[\Phi_{Low}]}{\theta_{fdiv}^2} \frac{4\rho_{tar}}{\pi R^2} \tag{18.5}$$

If we allow for frame summing,

$$e_{Lret,ti+} = N_{fs} \left[t_i \eta \tau_o \frac{A_{pix} A_o}{\pi f^2} \right] \tau_a^2 \frac{[\Phi_{Low}]}{\theta_{fdiv}^2} \frac{4\rho_{tar}}{\pi R^2} \tag{18.6}$$

For pulse laser operation and using t_{bin} that equals t_{pulse} and the number of bins per frame $N_{bins/f}$,

$$e_{Limage} = \left[N_{bins/f} \eta \tau_o \frac{A_{pix} A_o}{\pi f^2} \right] \tau_a^2 \frac{P_{Lpulse} (\lambda/hc)}{\theta_{fdiv}^2} \frac{4\rho_{tar}}{\pi R^2} \tag{18.7}$$

18.3.2 Systems Performance Metrics for UV Systems Design

To model the sensor and system performance, we have assumed the pixel size for a high sensitivity, detector size of 5–20 μm for the UV detector array. The fill factor of 70% is assumed typical for these small pixels. Typical quantum efficiencies have been assumed to be in the 70% range for the PIN diode and APD [5,6]. The model uses, as default, an amp noise of 15 electrons per frame time, a dark current of 1e-15 amps for a 5 μm pixel, or 4 nA/cm^2 or 200 electrons or about 14 noise electrons, and scene noise is effectively zero in the solar-blind region.

The model from the MODTRAN runs; the daytime irradiance in the UV is insignificant in the solar-blind region. The drop-off from 0.30 to 0.26 μm that illustrates the requirement for a UV detector with spectral response is in the solar-blind region. Figure 18.1 shows the UV spectral radiance at midday and the comparative laser illumination of the target at 1 km for a 6 mrad beam divergence for powers of 1 and 10 mW. The left plot in the Figure 18.1 shows that the transmittance improves with longer UV wavelengths for all three levels of aerosols and is sufficient for 1 km lengths in our solar-blind region.

To achieve high-resolution day–night imaging and identification of targets, the following conditions and requirements must be met. While linear detection (no APD and no laser illumination) is fine for muzzle flashes and images of nearby combatants illuminated by live fire (a millisecond event), laser illumination is required for cold targets (facial recognition, profile recognition). A continuous laser and 33 ms integrations are adequate if enough laser power is available. If not, a pulsed laser with nanosecond integrations and APD detectors are required to reduce atmosphere scatter and improve detector sensitivity.

18.3.3 ZnO/MgZnO Nanostructures for UV Applications

Zinc oxide (ZnO) is a unique wide-bandgap biocompatible material system exhibiting both semiconducting and piezoelectric properties that has a diverse group of growth morphologies. Bulk ZnO has a bandgap of 3.37 eV that corresponds to emissions in the UV spectral band [7]. Highly ordered vertical arrays of ZnO nanowires (NWs) have been grown on substrates including silicon, SiO_2, GaN, and sapphire using a metal-organic chemical vapor deposition (MOCVD) growth process [7]. The structural and optical properties of the grown vertically aligned ZnO NW arrays have been characterized by scanning electron microscopy (SEM), x-ray diffraction (XRD), and photoluminescence (PL) measurements [7–10]. Compared to conventional UV sensors, detectors based on ZnO NWs offer high UV sensitivity and low visible sensitivity and are expected to exhibit low noise, high quantum efficiency (QE), and extended lifetimes and have low power requirements [11,12]. The photoresponse switching properties of NW array–based sensing devices have been measured with intermittent exposure to UV radiation, where the devices were found to switch between low and high conductivity states at time intervals on the order of a few seconds. Envisioned applications for such sensors/FPAs potentially include defense and commercial applications [13].

Zinc oxide is a versatile functional material that provides a biocompatible material system with a unique wide direct energy bandgap and exhibits both semiconducting and piezoelectric properties. ZnO is transparent to visible light and can be made highly conductive by doping. Bulk ZnO has a bandgap of 3.37 eV that includes emissions in the solar-blind UV spectral band (~240–280 nm), making it suitable for UV detector applications [7]. Over this wavelength range, solar radiation is completely absorbed by the ozone layer of the earth's atmosphere, so the background solar radiation at the earth's surface is essentially zero. This enhances the capability of UV sensors in missile warning systems to detect targets such as missile plumes and flames emitting in this region.

ZnO is the basis for the one of the richest families of nanostructures among all materials taking into accounts both structure and properties. ZnO growth morphologies have been demonstrated for NWs, nanobelts, nanocages, nanocombs, nanosprings, nanorings, and nanohelixes [7]. The development of

ZnO NW–based UV detectors offers high UV sensitivity and low visible sensitivity for missile warning–related applications. Demonstration of devices using single ZnO NW strands has been widely reported in literature [7–16]. However, the development of reliable 2D arrays of aligned ZnO NWs has proven more challenging. The demonstration of reliable 2D arrays requires (1) correlation of growth process and growth parameters with the material quality of ZnO NWs, (2) correlation of the electrical and optical performance with growth parameters and fabrication processes, and (3) addressing system design challenges [17,18].

With conventional NW growth methods including electrochemical deposition, hydrothermal synthesis, and molecular beam epitaxy (MBE), it is generally difficult to scale up and control NW growth. Electrochemical deposition is well suited for large-scale production but does not allow control over the NW orientation. Hydrothermal synthesis is a low-temperature and low-cost process that allows growth of NWs on flexible substrates without metal catalysts, but the direction and morphology of the NWs cannot be well controlled with this method [8–10]. The MBE method allows monitoring of the structural quality during NW growth; however, this type of synthesis often requires use of metal catalysts as a seed layer [10], which introduces undesired defects to the structure, decreasing the crystal quality [12–16]. Chemical vapor deposition (CVD) also requires catalysts at the NW tips, and using this method, the tips of the grown NWs were observed to be flat, with vertical alignment.

18.3.4 Characterization of ZnO NW Arrays Grown on the Various Substrates

The samples were characterized by SEM utilizing a Quanta FEG 250 system and XRD using Bruker D-8 Advance X-ray diffractometer with a wavelength of 1.5406 Å corresponding to the Cu Kα line. In addition, PL measurements were performed at room temperature using a Linconix HeCd UV laser emitting at a wavelength of 325 nm. An Si detector in conjunction with a lock-in amplifier and chopper was used to measure the PL from the beam reflected off the sample at the output over the desired wavelength range [18–20].

SEM was performed to explore the NW morphology. Figure 18.2 shows the synthesized ZnO NWs on the various substrates, which can be generally seen to have uniform distribution density. The ZnO NWs grown on sapphire (Figure 18.2a) had approximate diameters of 50–70 nm and lengths in the range of 1–2 μm. NWs grown on SiO_2 (Figure 18.2b) had diameters of 150–200 nm and lengths of 1–2 μm and were the least vertically oriented and associated with a relatively high lattice mismatch. NWs grown on the Si (111) substrate (Figure 18.2c) had a slightly random orientation, also having diameters in the range of 150–200 nm and lengths from 1 to 2 μm. Finally, the NWs grown on GaN (Figure 18.2d) showed strong vertical orientation, with diameters of 20–40 nm and lengths of 0.7–1.0 μm [20].

Figure 18.3 shows the XRD pattern for the ZnO NWs grown on p-Si, GaN, and SiO_2 substrates [10]. The inset shows dominant peaks related to ZnO (002). The peak at 34° (2θ) for ZnO grown on p-Si and SiO_2 substrates incorporated the overlapping of ZnO NWs (002) and ZnO thin film (002). An additional diffraction peak associated with GaN was present for the GaN/sapphire substrate. ZnO NWs oriented along the (002) direction had full widths at half maxima (FWHM) and c-lattice constants of 0.0498 (θ) and 5.1982 Å at 34.48° (2θ) for p-Si, 0.0497(θ) and 5.1838 Å at 34.58° (2θ) for GaN, 0.0865(θ) and 5.1624° at 34.38° (2θ) for SiO_2, and 0.0830° (θ) and 5.2011 Å at 34.46° (2θ) for sapphire.

The quality of the ZnO epilayers utilized as seed layers to grow ZnO NWs was also characterized. The ZnO thin films were oriented along (002) and had a maximum at 34.58° with FWHM of 0.0697 (θ) for p-Si, maximum of 34.58° with FWHM of 0.0684 (θ) for GaN, and maximum of 34.43° with FWHM of 0.0557 (θ) for SiO_2. Additional shallow diffraction peaks were observed for NWs grown on p-Si and SiO_2, which are attributed to ZnO (100, 101, 102, and 110) as can be seen from Figure 18.3. As shown in Figure 18.4, for ZnO NW growth on sapphire, major peaks were observed for ZnO (002) at 34.46° (2θ) and Al_2O at 37.91° (2θ), with a minor peak for ZnO (101) at 36.34° (2θ).

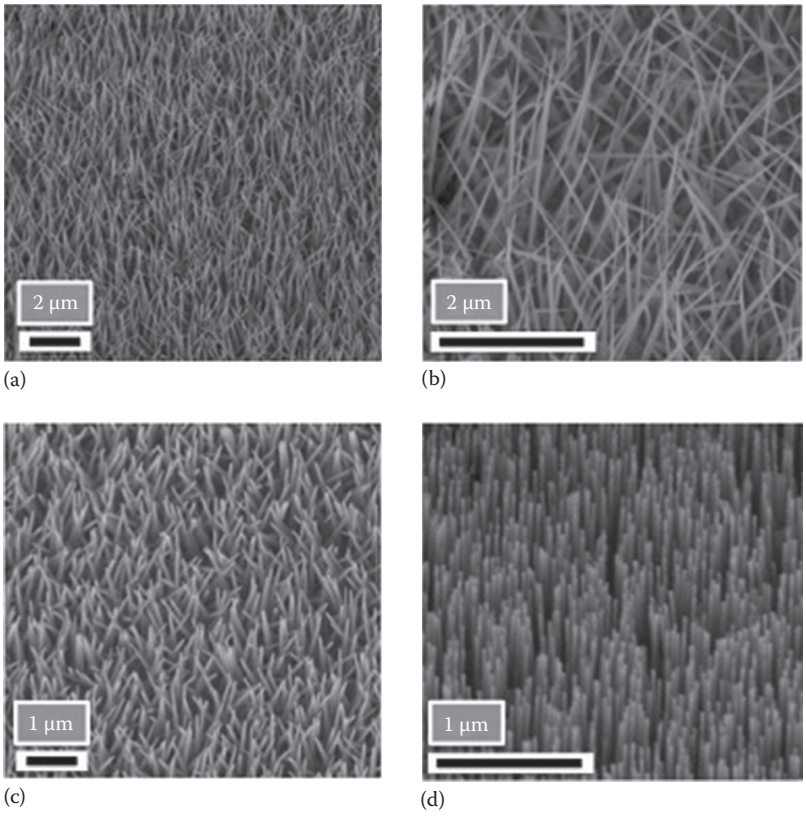


FIGURE 18.2 SEM images of NWs grown on the various substrates taken at room temperature, showing NWs grown on (a) ZnO/sapphire, (b) ZnO/SiO_2/p-Si, (c) ZnO/p-Si, and (d) ZnO/GaN/sapphire. (From Anwar, M. et al., *Proc. SPIE*, 8868, 8868B, 2013.)

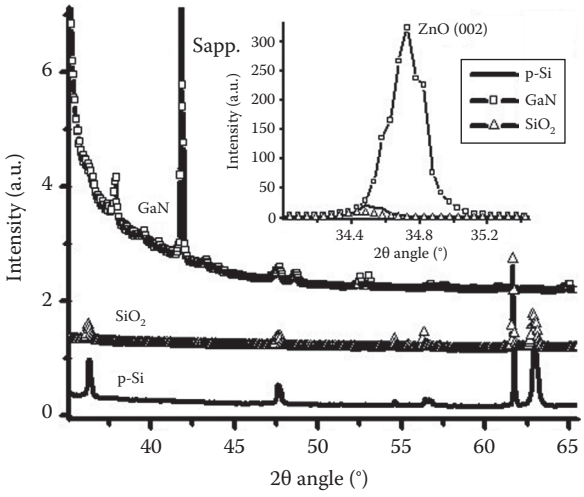


FIGURE 18.3 XRD of ZnO NWs grown using MOCVD on p-Si (solid), GaN/sapphire (square), and SiO_2 (triangle). The inset shows the ZnO peak associated with ZnO oriented along (002) and GaN. (From Anwar, M. et al., *Proc. SPIE*, 8868, 8868B, 2013.)

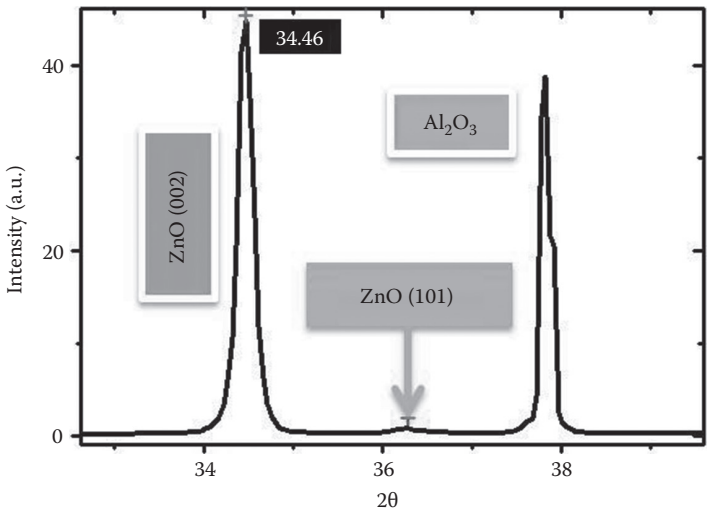


FIGURE 18.4 XRD of ZnO NWs grown using MOCVD on sapphire. (From Anwar, M. et al., *Proc. SPIE*, 8868, 8868B, 2013.)

18.3.5 Photoluminescence Measurements

Figure 18.5 shows the PL spectra for ZnO NWs grown on p-Si, GaN, and SiO_2 substrates [10]. The room temperature PL measurements were performed using a ~280 nm light source. Single peaks located at 380 nm having an FWHM of 14.69 nm and at 378 nm having an FWHM of 15 nm were observed for p-Si and SiO_2 substrates, respectively, corresponding to the recombination of excitons through an exciton–exciton collision process [18–20].

No defects related to Zn or O vacancies were observed, which can be attributed to the confinement of defects at the ZnO thin film/substrate interface. For the ZnO NWs grown on GaN, a predominant peak with an FWHM of 18.18 nm was observed at 378 nm. High stress was evident for ZnO NWs grown on GaN; this can contribute to the broadening of the peak in comparison to p-Si and SiO_2. Shallow peaks identified at 474 and 490 nm through Lorentzian decomposition are attributed to oxygen interstitial and oxygen vacancies, respectively [20].

A UV LED lamp acquired from Sensor Electronic Technology, Inc. was used to characterize the UV photoresponse of the ZnO NW arrays [20]. The lamp comprises eight separate AlGaN-based UV LEDs in a TO-3 package spanning the 240–370 nm wavelength range, with a customized power supply capable of independently monitoring and controlling the current of all or any of the LEDs. The photoresponse was determined by first applying voltage between indium contacts on the front and back sides of an Si NW sample and measuring the resulting current in the dark and then repeating this procedure while the sample was exposed to radiation from a UV LED at a specific wavelength.

Figure 18.6 shows the on–off switching characteristics of a ZnO vertical array NW device when exposed to radiation at 370 nm. This device was found to switch between low and high conductivity states in approximately 3 s, a faster response than most reported thus far for ZnO NW–based UV detectors.

Figure 18.7a shows a mounted and wire-bonded NW UV 3 × 9 pixel array detector device. Incorporation of Mg allows the detector response to be shifted to shorter wavelengths to provide detection in the solar-blind region. This device was tested by applying a bias between the top contacts on the pixels, which are apparent in Figure 18.7b, and the back contact.

ZnO NW–based arrays offer high sensitivity and have potential application in UV imaging systems. ZnO NW array-based UV detectors have no moving parts, high QE, extended lifetimes, low noise, and low power requirements and offer high sensitivity.

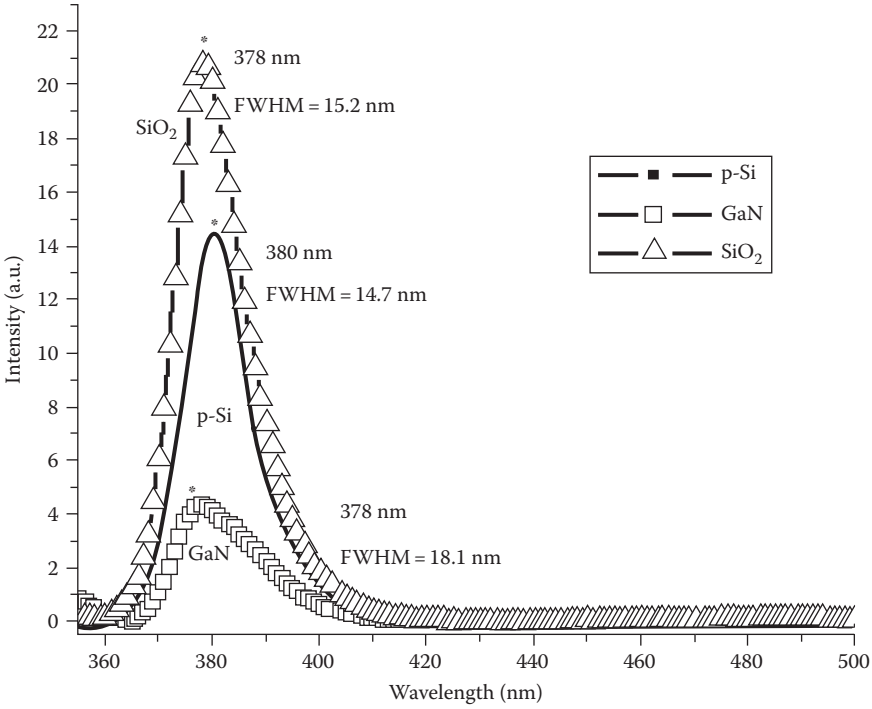


FIGURE 18.5 PL of ZnO NWs grown on p-Si (100) (solid) with a single peak at 380 nm, GaN (square) with a stronger peak at 378, and SiO_2 (triangle) with a single peak at 378 nm. (From Anwar, M. et al., *Proc. SPIE*, 8868, 8868B, 2013.)

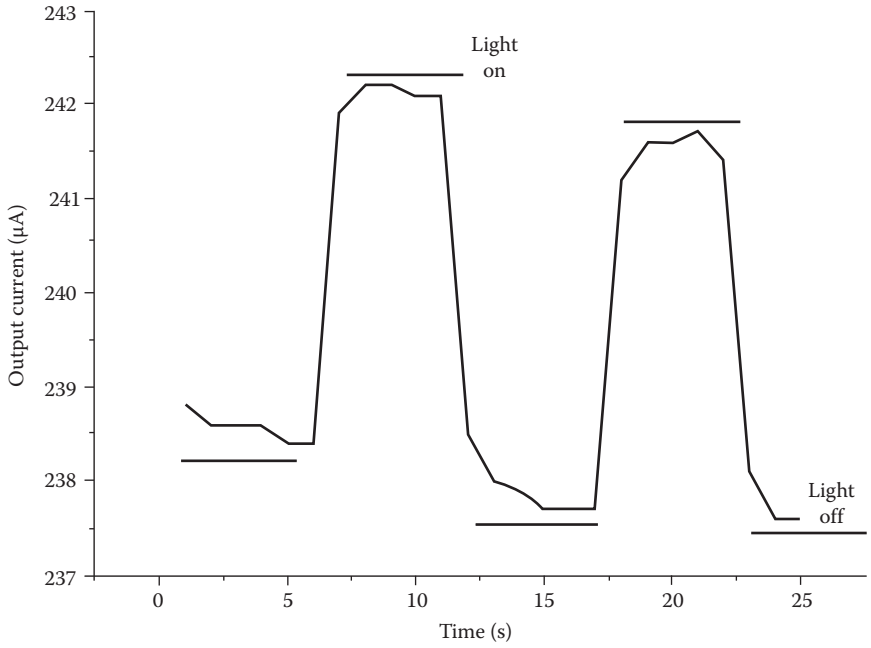
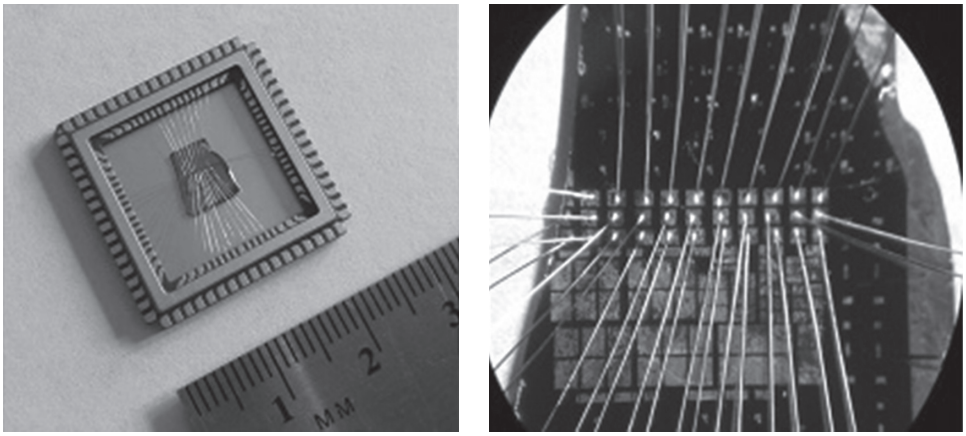


FIGURE 18.6 Switching photoresponse characteristics of ZnO NW device when UV LED source at ~370 nm turned on and off over approximately 10 s intervals. (From Anwar, M. et al., *Proc. SPIE*, 8868, 8868B, 2013.)



(a) (b)

FIGURE 18.7 (a) Mounted solar-blind NW UV 3 × 9 pixel array detector device and (b) close-up of device, showing wire-bonded pixels. (From Anwar, M. et al., *Proc. SPIE*, 8868, 8868B, 2013.)

ZnO NWs have also been evaluated for providing remote power for the stand-alone sensors. This type of application has been extensively studied by Professor Z.L. Wang and his team at Georgia Tech [21,22]. They have shown that ZnO NWs can be used as nanogenerators (NGs) for providing remote power using the piezoelectric effect. Photovoltaic cells or solar cells are a popular renewable energy technology, relying on approaches such as inorganic p–n junctions, organic thin films, and organic–inorganic heterojunction. However, a solar cell works only under sufficient light illumination, which depends on the location the devices will be deployed, as well as the time of the day and the weather.

Considering that mechanical energy is widely available in our living environment. They have demonstrated [21] the first hybrid cell (HC) for concurrently harvesting solar and mechanical energy through simply integrating a dye-sensitized solar cell (DSSC) and a piezoelectric NG on the two sides of a common substrate. After this, in order to solve the encapsulation problem from liquid electrolyte leakage in the first back-to-back integrated HC, early in 2011, Xu and Wang improved the prototype design of the HC and developed a compact solid-state solar cell. This innovative design convoluted the roles played by the NW array to simultaneously perform their functionality in an NG and a DSSC. The design and the performance are shown in Figure 18.8.

Based on these demonstrations of HCs for concurrently harvesting solar and mechanical energies, they have reported an optical fiber–based 3D HC, consisting of a DSSC for harvesting solar energy and an NG for harvesting mechanical energy; these are fabricated coaxially around a single fiber as a core–shell structure (Figure 18.8). The optical fiber, which is flexible and allows remote transmission of light, serves as the substrate for the 3D DSSC for enhancing the electron transport property and the surface area and making it suitable for solar power generation at remote/concealed locations. The inner layer of the HC is the DSSC portion, which is based on a radically grown ZnO NW array on an optical fiber with ITO as the bottom electrode. The dye-sensitized ZnO NW array was encapsulated by a stainless steel capillary tube with a Pt-coated inner wall as the photo-anode for the DSSC. The stainless steel tube also serves as the bottom electrode for the outer layer of the NG, with densely packed ZnO NWs grown on its outer wall.

Another exciting application of ZnO NWs is designing, fabricating, and integrating arrays of nanodevices into a functional system that are key to transferring nanoscale science into applicable nanotechnology as shown in Figure 18.9.

Recent work [22] on 3D circuitry integration of piezotronic transistors based on vertical zinc oxide NWs as an active taxel-addressable pressure/force sensor matrix for tactile imaging. Using the piezoelectric polarization charges created at a metal–semiconductor interface under strain to gate/modulate

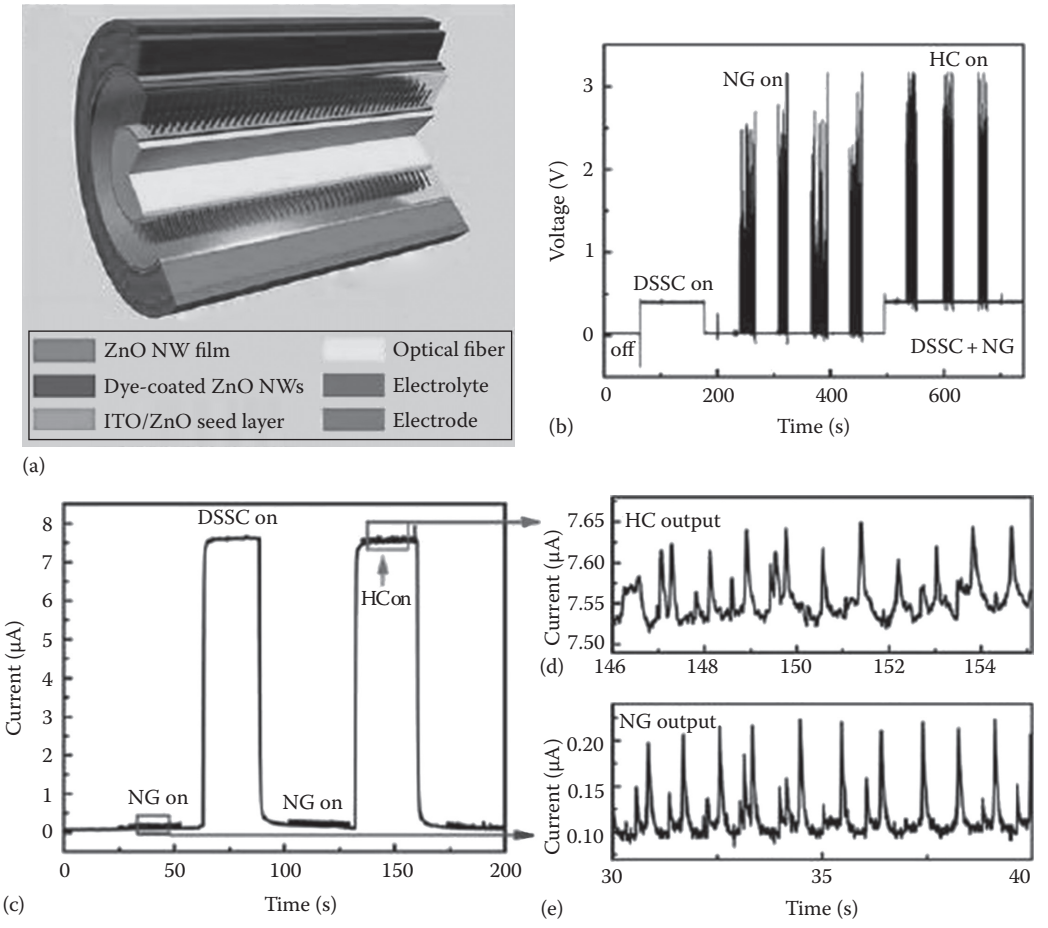


FIGURE 18.8 Design and performance of a 3D optical fiber–based HC consisting of a DSSC and an NG for harvesting solar and mechanical energy. (a) The 3D HC is composed of an optical fiber–based DSSC with capillary tube as counter electrode and an NG on top. (b) Open-circuit voltage (VOC) of the HC when the NG and the DSSC are connected in series, where VOC(HC) = VOC(DSSC)+ VOC(NG). (c) Short-circuit current (ISC) of the HC when the NG and the DSSC are connected in parallel. (d, e) Enlarged view of ISC(HC) and ISC(NG), clearly showing that ISC(NG) is 0.13 μA, the ISC(DSSC) is 7.52 μA, and the ISC(HC) is about 7.65 μA, nearly the sum of the output of the solar cell. (From Wang, Z.L. et al., *Mater. Today*, 15(12), 532, 2012.)

the transport process of local charge carriers, we designed independently addressable two-terminal transistor arrays, which convert mechanical stimuli applied to the devices into local electronic controlling signals.

The device matrix can achieve shape-adaptive high-resolution tactile imaging and self-powered, multidimensional active sensing. The 3D piezotronic transistor array may have applications in human–electronics interfacing, smart skin, and micro- and nano-electromechanical systems.

18.4 Development of GaN UVAPD for UV Sensor Applications

High-resolution imaging in UV bands has a lot of applications in defense and commercial applications. The shortest wavelength is desired for spatial resolution that allows for small pixels and large formats. Ultra Violet Avalanche Photodiodes (UVAPDs) have been demonstrated as discrete devices

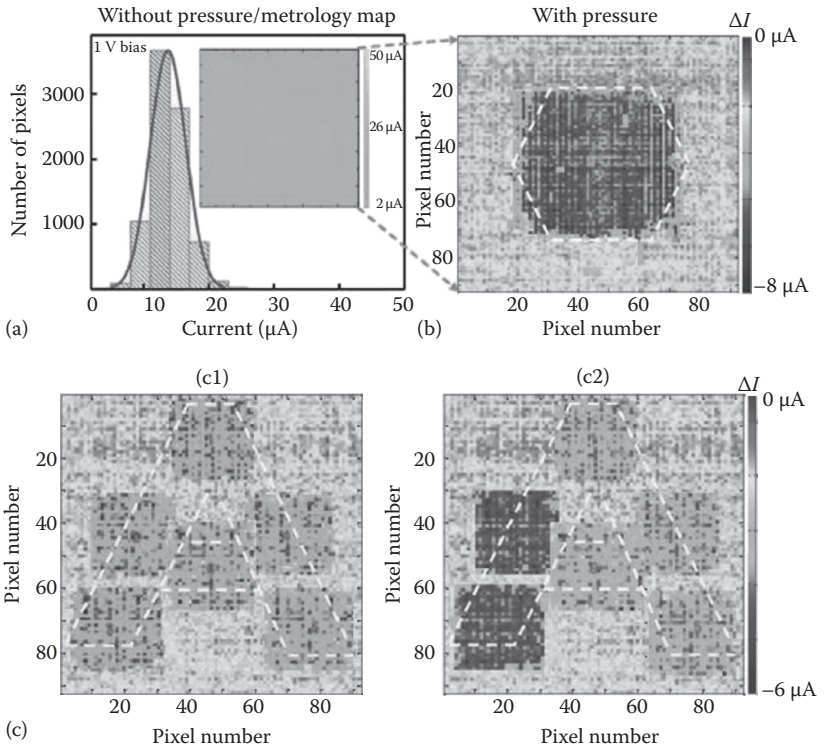


FIGURE 18.9 Tactile imaging and multidimensional sensing by the fully integrated 92 × 92 SGVPT array. (a) Metrology mapping (inset) and statistical investigation of the fully integrated Strain Gated Vertical Piezo-Electric Transistor (SGVPT) array without applying stress. (b) Current response contour plot illustrating the capability of SGVPT array for imaging the spatial profile of applied stress. The different shades (light to dark) of grayscale represent the current differences for each taxel before and after applying the normal stress. The physical shape of the applied stress is highlighted by the white dashed lines. (c) Multidimensional sensing by an SGVPT array exhibits the potential of realizing applications such as personal signature recognition with maximum security and unique identity. The shape of a "written" letter A is highlighted by the white dashed lines. (From Wu, W. et al., *Science*, 340(24), 2013.)

demonstrating gain. The next frontier is to develop UV APD arrays with high gain to demonstrate high-resolution imaging. We will discuss a model that can predict sensor performance in the UV band using APDs with various gain and other parameters for a desired UV band of interest. SNRs can be modeled from illuminated targets at various distances with high resolution under standard atmospheres in the UV band and the solar-blind region using detector arrays with unity gain and with high gain APDs [23–26].

Figure 18.10 presents the relationship between the alloy composition of gallium and aluminum in $Al_xGa_{1-x}N$ that determines the cutoff wavelength of the UV detector for p–i–n [23,24] and also for UV APDs. Deep ultraviolet (DUV) will require addition of larger composition of aluminum in $Al_xGa_{1-x}N$ [25].

18.4.1 GaN/AlGaN UV APD Growth

Figure 18.11 presents the high-temperature MOCVD system by Aixtron. This new reactor design and capability has the ability to grow high-quality GaN and AlGaN material with doping for GaN/AlGaN UVAPD applications [26].

Figure 18.12 presents the device structure of a backside-illuminated AlGaN UV APD. The substrate in this device structure is double-side-polished AlN substrate. The use of AlN substrate allows

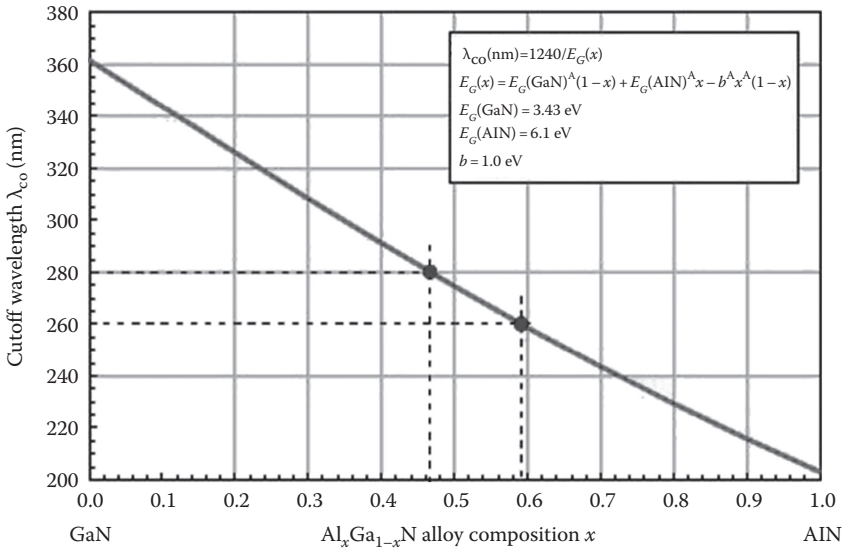


FIGURE 18.10 Relationship between alloy composition of AlGaN and the corresponding spectral cutoff for the UV detector arrays. (From Dupuis, R.D. et al., *Proc. SPIE*, 6739, 2006.)

(a) (b)

FIGURE 18.11 Photograph of new-generation AIXTRON CCS 3 × 2″ high-temperature III-nitride 3 × 2 MOCVD growth chamber open for loading wafers showing close-coupled showerhead. (From Sood, A.K. et al., *Proc. SPIE*, 8868, 88680T, 2013.)

the UV APD device structure to be backside illuminated and can be integrated with silicon CMOS electronics. Figure 18.13 presents the reciprocal space mapping of AlGaN on AlN substrate and sapphire substrate. The data for sapphire substrate show increased strain and mosaicity compared with AlN substrate.

Figure 18.14 presents the microscopic surface morphology using AFM on GaN p–i–n structure grown on GaN/sapphire template. No surface defects are observed. These results are encouraging to develop a low-cost backside-illuminated UV APD detector array.

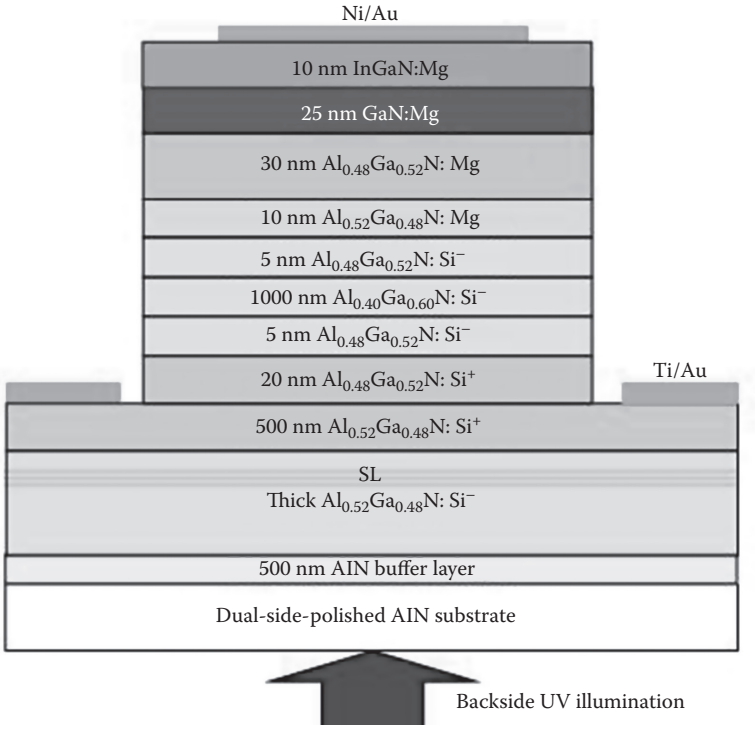


FIGURE 18.12 Device structure cross section of prototype backside-illuminated AlGaN UV APD. (From Sood, A.K. et al., *Proc. SPIE*, 7780, 77800E, 2010.)

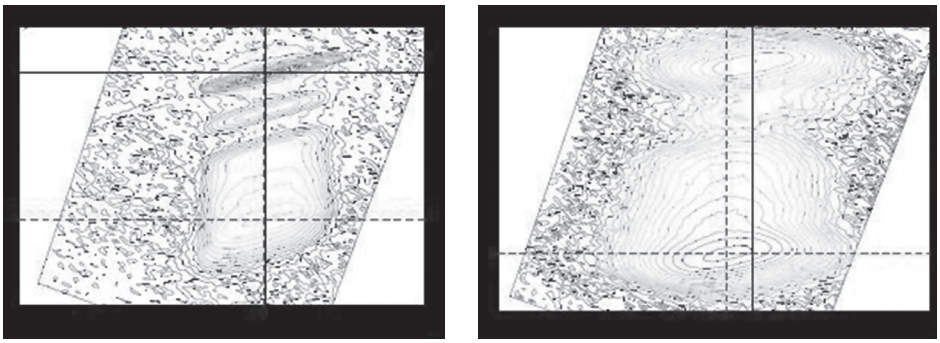


FIGURE 18.13 Reciprocal space mapping of AlGaN p–n junctions on (a) AlN and (b) sapphire substrates. (From Long, J.P., *Opto-Electron. Rev.*, 10(4), 251, 2002.)

In the SIMS analysis of GaN p–i–n structure on GaN/sapphire template, the data show controlled Si and Mg doping for n- and p-type layers. The data show low background doping concentration in GaN layer. The Mg doping is being increased for better p-type conductivity.

Figure 18.15 presents the current–voltage characteristics of AlGaN UV APD with spectral response of 300 nm. Further reduction in the dark current will be expected with surface passivation. The future effort is underway to improve the growth characteristics low defect density in substrates and high-quality epitaxial growth technologies that are the keys to the successful implementation of a robust high-performance APDs for UV FPAs [4,5,24,29,30].

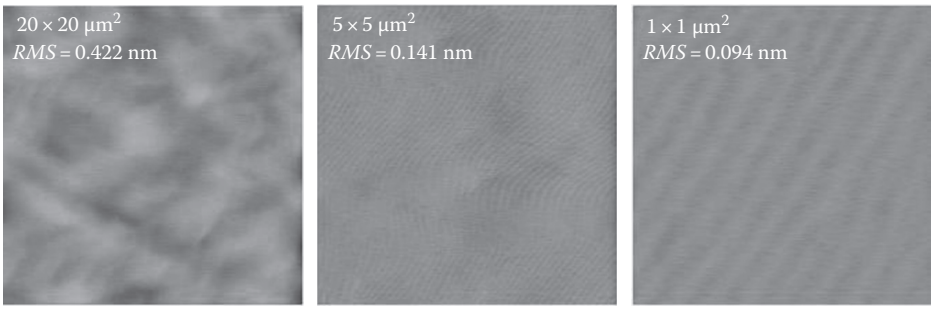


FIGURE 18.14 Microscopic surface morphology using AFM on GaN p–i–n structure grown on GaN/sapphire template. No specific surface defects are observed. (From Sood, A.K. et al., *Proc. SPIE*, 7780, 77800E, 2010.)

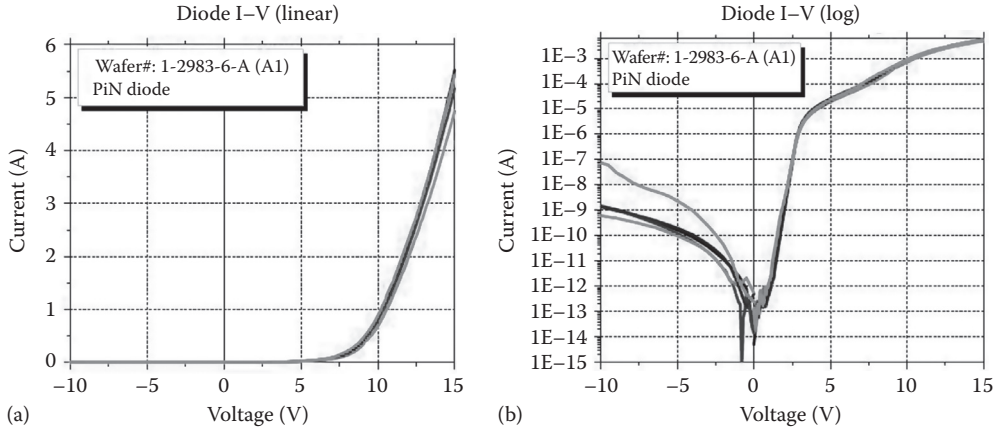


FIGURE 18.15 Current–voltage characteristics of AlGaN UV APD unpassivated test structure. Further reduction in the dark current will be expected with surface passivation. (From Crawford, M., *MRS Proc.*, 831, Fall 2004.)

18.5 Visible–NIR Detector Technologies

The visible–NIR band (0.4–1.7 μm) bridges the spectral gap between the visible and thermal bands in the electromagnetic spectrum. In this spectral band, the primary phenomenology of interest is the reflectance signature of the target, manifested as its variations either in brightness or spectral reflectance or both.

IR imaging in the NIR–SWIR band offers several advantages: it can detect reflected light, offering more intuitive, visible-like images; it is better suited for imaging in adverse environments and weather conditions, including fog, dust, and smoke; it can also see in low-light conditions and use eye-safe 1550 nm illumination; and it can generate digital video outputs thus offering a more dynamic range than traditional image intensifier night vision equipment. Under low-light conditions, the sensitivity of the FPA is ultimately determined by the R_0A product of the photodiode [31–33].

18.5.1 $Si_{1-x}Ge_x$ (SiGe) Detector Arrays

Like the other two alloy semiconductors mentioned earlier, SiGe is another example of material that can be used for the fabrication of IR detectors. The key attractive feature of SiGe IR detectors is that they can be fabricated on large diameter Si substrates with size as large as 12 in. diameter using

standard integrated circuit processing techniques. Furthermore, the SiGe detectors can be directly integrated onto low-noise Si readout integration circuits (ROICs) to yield low-cost and highly uniform IR FPAs.

Some of the earlier attempts in developing SiGe IR detectors focused on their LWIR applications [31–33]. Renewed efforts are now developing these detectors for application in the NIR–SWIR band [33]. For the SiGe material to respond to the SWIR band, its cutoff wavelength is tuned by adjusting the SiGe alloy composition. Si and Ge have the same crystallographic structure and both materials can be alloyed with various Ge concentration. The lattice constant of Ge is 4.18% larger than that of Si, and for an $Si_{1-x}Ge_x$ alloy, the lattice constant does not exactly follow Vegard's law. The relative change of the lattice constant is given by [33]

$$a_{Si_{1-x}Ge_x} = 0.5431 + 0.01992x + 0.0002733x^2 \text{ (nm)}$$

An $S_{i1-x}Ge_x$ layer with $x > 0$ on an Si substrate means that the layer is under compressive stress. A perfect epitaxial growth of such a strained heteroepitaxial layer can be achieved as long as its thickness does not exceed a critical thickness for stability. Beyond the critical thickness, the strain is relaxed through the formation of misfit dislocations that can cause an increase in the dark current.

Several approaches have been proposed to reduce the dark current in SiGe detector arrays by several orders of magnitude; these include superlattice, quantum dot, and buried junction designs [33–35]. Furthermore, some of these approaches have the potential of extending the wavelength of operation beyond 1.8–2.0 μm. The challenge is to take advantage of these innovative device designs and reduce the dark currents to 1–10 nA/cm^2. Figure 18.16 presents the SiGe/Ge detector array using buried junction approach to reduce the surface states and leakage current [33].

The SLS structure is being evaluated for longer detector array response to 2 μm. It consists of SiGe quantum wells and Si barrier layers, grown on p-type (001) Si substrates. Superlattices having differing Si barrier and Ge well thicknesses to control the strain are grown to optimize wavelength response and dark current.

The SiGe well thicknesses are kept below the critical layer thickness for dislocation formation. To complete the structure, the undoped superlattice is capped with a thin n+ Si cap layer to form the

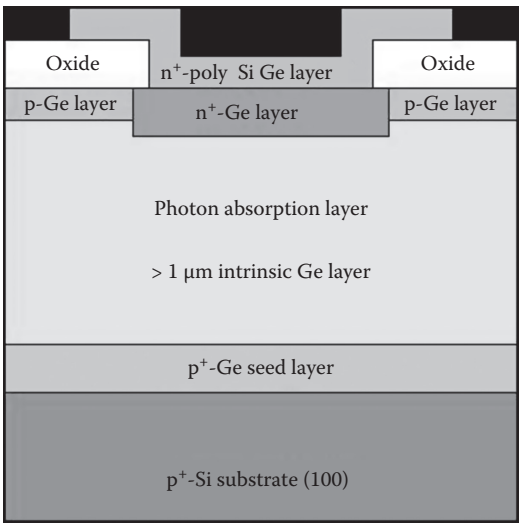


FIGURE 18.16 SiGe/Si-based buried junction approach to be evaluated for reduced surface states and leakage current. (From Sood, A.K. et al., *Proc. SPIE*, 7780, 77800F, 2010.)

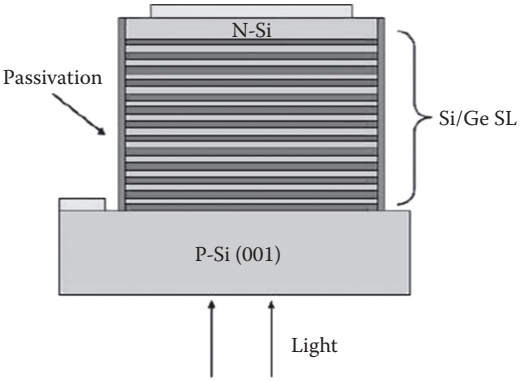


FIGURE 18.17 Schematic of detector array structure consisting of an SiGe/Si SLS grown on (001) silicon. (From Sood, A.K. et al., *Proc. SPIE*, 8012, 801240, 2011.)

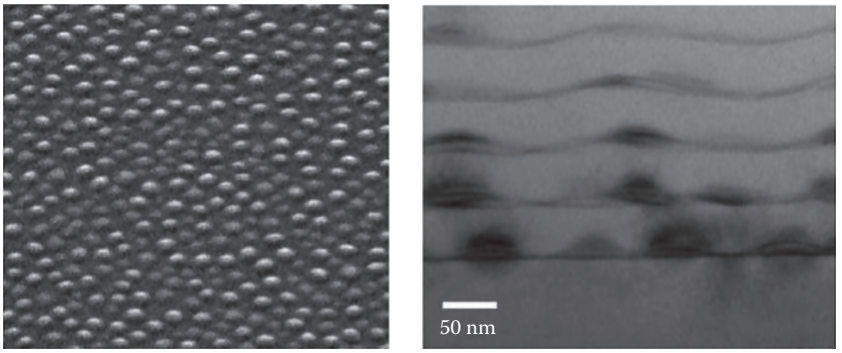


FIGURE 18.18 (a) SEM image (45° tilt) of a Ge quantum dot (QD) layer deposited on Si. The QDs are ~60 nm in diameter with a density of 1020 cm^2. (b) A cross-sectional TEM image of Ge/Si QDSL grown. Ge QDs appear with dark contrast compared to Si barriers. (From Sood, A.K. et al., *Proc. SPIE*, 8012, 801240, 2011.)

p–n junction. After growth, the devices are patterned with a top contact; mesas are etched to provide isolation, and the substrate contact is formed. The etched mesa can also be passivated to minimize surface recombination as indicated in Figure 18.17. The device shown in the Figure 18.17 uses substrate illumination, as is needed for use in FPAs, and short-wavelength response can be improved by thinning the Si substrate.

The SLS and quantum dot superlattice (QDSL) in the SiGe material system have the potential of developing visible–NIR detector arrays with longer cutoff wavelength and potentially lower dark current. The advantage of quantum dots is the potential to exploit the optical properties of Ge while avoiding dislocation formation. Ge QDs grown on Si in Stranski–Krastanov mode can be deposited well beyond the critical thickness without dislocation nucleation [36].

Figure 18.18 shows an SEM image of an array of Ge nanodots grown by MOCVD. These dots are typically 50–75 nm in diameter with area coverage of ~20%. To increase optical absorption and sensitivity, MOCVD-based growth techniques are being developed for the deposition of Ge/Si QDSLs, where Ge QDs are alternated with thin (10–30 nm) Si barrier layers. A cross-sectional transmission electron microscopy (TEM) image of QDSLs is shown in Figure 18.18b.

18.6 SWIR Detector Technologies

The SWIR band (0.9–2.5 μm) bridges the spectral gap between the visible and thermal bands in the electromagnetic spectrum. In this spectral band, the primary phenomenology of interest is the reflectance signature of the target, manifested as its variations either in brightness or spectral reflectance or both.

IR imaging in the SWIR band offers several advantages: it can detect reflected light, offering more intuitive, visible-like images; is better suited for imaging in adverse environments and weather conditions, including fog, dust, and smoke; can also see in low-light conditions and use eye-safe 1550 nm illumination that is totally undetectable by regular night vision equipment; and can generate digital video outputs thus offering a more dynamic range than traditional image intensifier night vision equipment. Under low-light conditions, the sensitivity of the FPA is ultimately determined by the R_0A product of the photodiode.

18.6.1 $In_xGa_{1-x}As$ Detector Array Development

For SWIR imaging, InGaAs is one of the widely used detector materials due to its low dark current. The detector material can be prepared using any of the following techniques: MBE, MOCVD, liquid-phase epitaxy (LPE), hydride-transport vapor-phase epitaxy (VPE), and atomic layer epitaxy (ALE). InGaAs layers are typically grown on lattice-matched InP substrates using an alloy composition of $x = 0.53$ [37–39].

The spectral response typically covers 0.9–1.7 μm at room temperature. By increasing the composition to $x = 0.82$, InGaAs is able to extend its cutoff to 2.6 μm. However, the crystal defects due to epitaxy and the decreased shunt resistance, due to a smaller bandgap, degrade performance at the longer cutoff wavelengths [40].

The bandgap (7–1) of the strained $In_xGa_{1-x}As$–InP structure can be tailored by varying the alloy composition during crystal growth according to the following equation:

$$E_g(eV) = \left(E_{g_{GaAs}} - \frac{\alpha_{GaAs}T^2}{T+\beta_{GaAs}} + \left(E_{g_{InAs}} - \frac{\alpha_{InAs}T^2}{T+\beta_{InAs}} - E_{g_{GaAs}} + \frac{\alpha_{GaAs}T^2}{T+\beta_{GaAs}}\right)x - 0.475x(1-x)\right) \quad (18.8)$$

where

E_g is the bandgap in (eV)
α and β are fitting parameters
x is the In–As ratio

The cutoff wavelength can be calculated from the expression $\lambda_{co} = hc/E_{gap}$.

The response can be extended to include the visible wavelength range by removing the InP substrate. There has been an intensive effort to develop InGaAs arrays for low-light level (LLL) SWIR imaging [37–43]. An example is in astrophysical space-based observatories that are very demanding on the detectors due to the very low IR flux levels. Such low flux levels represent the detection of few photons over long integration times and, therefore, require extremely low dark current photodiodes hybridized to a high-performance ROIC stage. For such LLL applications, there are challenges ahead to further lower noise, reduce pixel size, fabricate larger arrays, achieve higher operating temperatures, and reduce production cost.

The spectral response of InGaAs diodes at room temperature is in the 0.9–1.67 μm wavelength range that matches with the ambient nightglow spectrum. Imaging under such low-light conditions requires that the noise of the detector be extremely low. A significant portion of the noise is contributed by the dark current of the InGaAs detector and the readout noise. Dark current consists of unwanted thermally generated carriers that can cause the detector to produce a random varying output signal.

It is associated with interfacial, diffusional, G-R, and tunneling currents. The temperature dependence of the dark current is primarily due to the intrinsic carrier concentration that depends exponentially on the temperature. The dark current of the detector can be reduced through appropriate

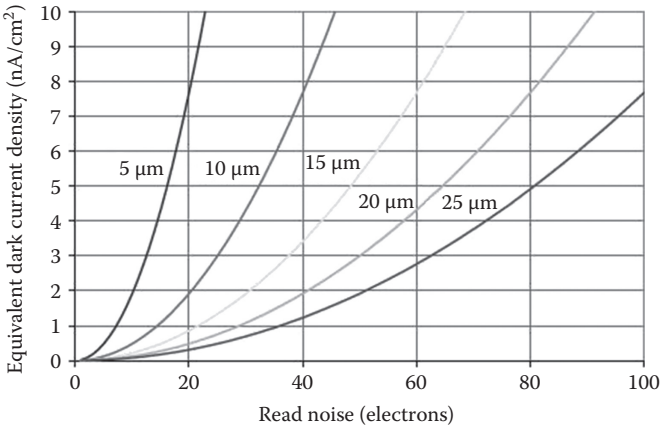


FIGURE 18.19 Dark current density versus read noise for different pixel pitches. (From MacDougal, M. et al., *Opt. Eng.*, 50(6), 061011, 2011.)

fabrication processes and device design. The impact of dark current noise as a function of read noise is shown in Figure 18.19, where the curves for different pixel pitch map the dark current noise into an equivalent read noise.

For a given read noise, the required dark current density increases as the pixel pitch is decreased. The challenge is to maintain a low dark current density as the pixel pitch is reduced. Simultaneously, the challenge for the readout circuit is to reduce the read noise. If the limitation is due to the detector and its noise level overwhelms the source signal, the solution may be to use an external illuminator or cool the detector. The choice of either solution will depend on a trade-off between size, weight, and power requirements (SWaP).

As mentioned earlier, the dark current of the detector can be reduced through appropriate fabrication processes and device designs. By focusing on the growth conditions for the InGaAs absorption layer, heterointerfaces, and the passivation layer, researchers have been able to demonstrate dark current density below 1.5 nA/cm² at 77°C for 15 μm pitch arrays as shown in Figure 18.20.

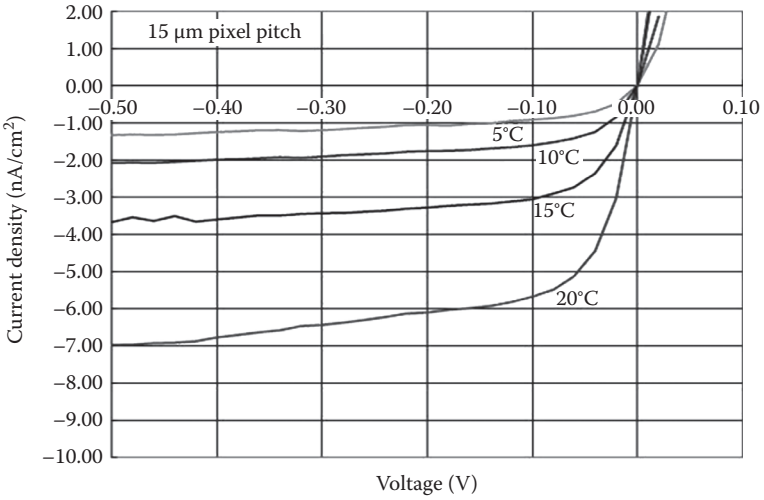


FIGURE 18.20 Dark current density at different temperatures using test structures on the wafer. Test arrays have 225 pixels (15 μm pitch), and the guard ring is not biased. (From Hood, A.D., *Proc. SPIE*, 8353, 83530A, 2012.)

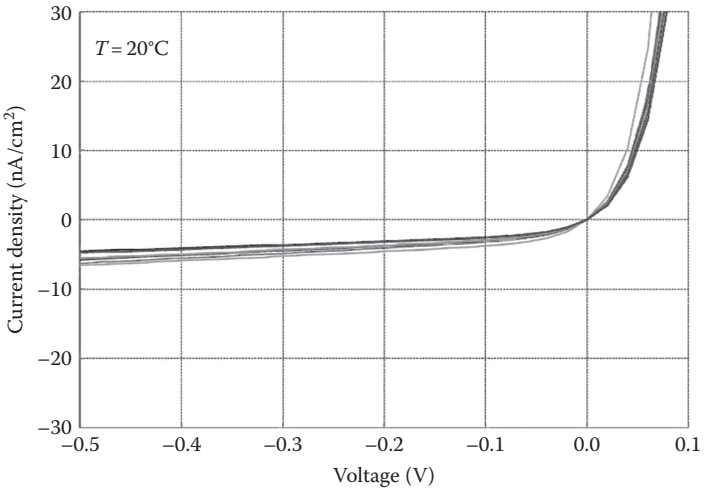


FIGURE 18.21 Experimental results for InGaAs test array demonstrating dark current density for eight separate 300 × 10, 15 μm pitch pixel test arrays measured across a wafer. The average dark current density for the test arrays at 100 mV reverse bias is 2.95 nA/cm^2 at 20°C. (From Hood, A.D., *Proc. SPIE*, 8353, 83530A, 2012.)

In scaling to small pixel pitch, further effort is continued to develop wafer processing parameters and methods that reduce surface-related perimeter effects and enable small pixel pitch InGaAs detectors with dark current densities comparable to large (25 μm) pixel pitch detectors [5,6]. Figure 18.21 presents a plot of dark current density, measured at 20°C, for eight different, 300 × 10 pixel test arrays distributed across a 3″ wafer. The average dark current density at −100 mV was 2.95 nA/cm^2.

Further effort is underway to demonstrate large format (>1 K × 1 K) and small pixel (<20 μm) InGaAs FPAs for a variety of LLL imaging applications such as night vision. These applications demand extremely low detector dark current and Si ROIC noise [44].

Recent work [44] has demonstrated significant progress in InGaAs detector array development on a 4″ wafer as shown in Figure 18.22; and also reducing dark current density for 10–20 μm pixel

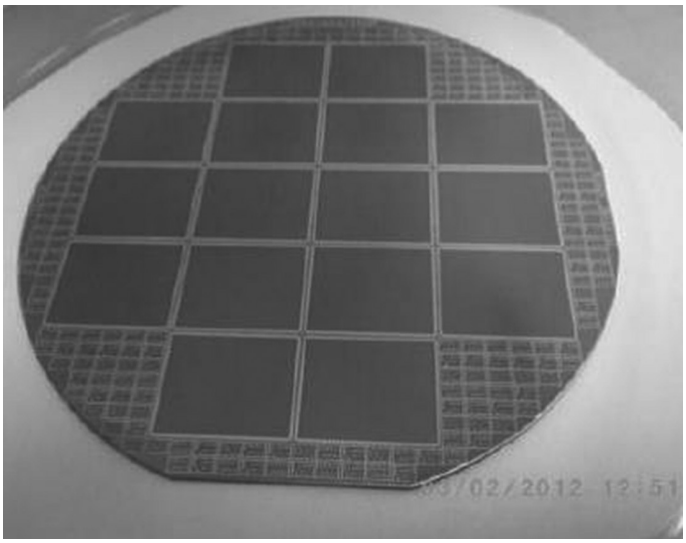


FIGURE 18.22 Experimental 1280 × 1024/15 μm arrays on 4″ wafer surrounded by various test mini-arrays with pitch sizes of 5–20 μm. (From Yuan, H. et al., *Proc. SPIE*, 6940, 69403C, 2008.)

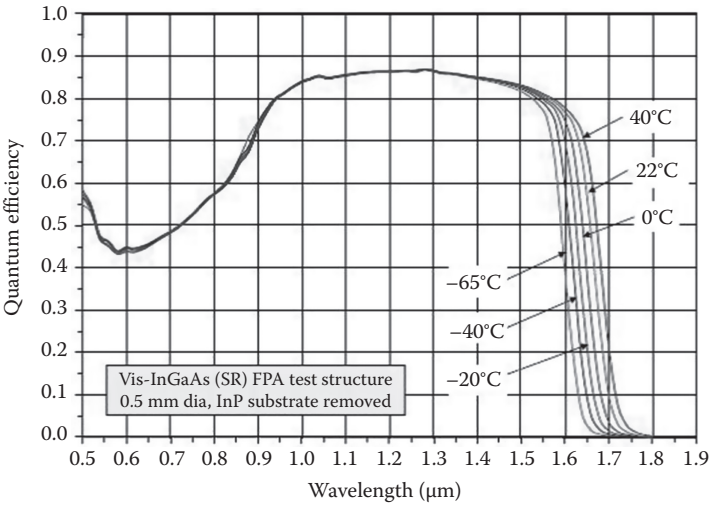


FIGURE 18.23 Spectral QE versus wavelength at different temperatures measured for backside-illuminated InGaAs photodiodes test array demonstrating visible–NIR response with InP substrate removed. (From Yuan, H. et al., *Proc. SPIE*, 6940, 69403C, 2008.)

arrays; developing sub-10 μm pixel array technology and demonstrating the feasibility of making 5 μm pixel arrays, and reducing the capacitance of small pixels [44]. Figure 18.23 demonstrates recent results for spectral QE as a function of wavelength measured on backside-illuminated InGaAs photodiodes test array at different temperatures demonstrating visible–NIR response with InP substrate removed [44].

18.7 Nanostructured Detector Technology for MWIR and LWIR Bands

Electro-optical (EO)/Infrared (IR) sensors and imagers using nanostructure-based materials are being developed for a variety of defense applications. In this section, we present recent work under way for development of next-generation carbon nanostructure–based IR detectors and arrays. We discuss detector concepts that will provide next-generation high-performance, high-frame-rate, and uncooled nanobolometer for MWIR and LWIR bands [45–48]. The critical technologies being developed include carbon nanostructure growth, characterization, and optical and electronic properties that show the feasibility for IR detection. Experimental results on CNT nanostructures are presented. Further discussion will be provided for the path forward to demonstrate enhanced IR sensitivity and larger arrays.

The microbolometer based on Si-MEMS device structure has been under development for over 20 years with support from DARPA and the US Army. Two most common Si-MEMS-based structures utilize VOx and amorphous silicon-based technologies. Several companies such as BAE systems and DRS Technologies are developing and producing 17 μm unit cell 640 × 480 and larger arrays using VOx [49–52]. Similarly, L-3 Communications and other groups are developing and producing 640 × 480 with 17 μm unit cell using amorphous silicon technology [51,52].

We discuss the use of carbon nanostructures as the high-performance bolometric element of the MWIR and LWIR bands. As part of this effort, we are exploring the development of smaller unit cell bolometer, that is, 5–10 μm unit cell, with higher temperature coefficient of resistance (TCR) and higher frequency response in the 1–10 kHz range. The feasibility of such an array can open up a larger number of defense and commercial applications. This section discusses the efforts under way to explore these possibilities.

18.7.1 Design and Modeling of CNT-Based Bolometer Characteristics

To optimize bolometer design, we need to consider several physical key phenomena. From a fundamental point of view, we take the absorbing material to have an extremely large response to IR radiation. The phonon modes of the material need to be able to easily couple to IR radiation. Furthermore, once this coupling has been achieved, the absorbed radiation should greatly increase the population of the phonon modes thereby significantly increasing the lattice temperature [45,46].

On a macroscopic scale, this large temperature increase is typically described in terms of a large thermal resistance. The higher the thermal resistance of the material, in general, the higher the resulting temperature will be after absorbing IR. Of course, higher thermal resistance may also give rise to larger thermal noise. This mitigating factor must be balanced with signal response in order to optimize IR sensitivity or the minimum detectable IR signal.

In summary, for DC operation, we work to maximize the thermal resistance R_{th} while achieving an acceptable noise, which maximizes the minimum signal that we can detect. In addition to DC operation, we try to maximize the thermal frequency response of the bolometer. This requires that we minimize the thermal capacitance C_{th} of the device and thereby minimize the thermal $R_{th}C_{th}$ time constant for the absorbing material.

18.7.1.1 Temperature-Dependent Electrical Characteristics

In addition to considering the DC and transient thermal characteristics of the absorbing material, we need to optimize the electrical response as well. To achieve this, we associate the electrical response with the thermal response by considering the temperature-dependent voltage–current characteristics of the electrical material.

A good response is obtained by utilizing a material that has an effectively large variation of electrical resistance with temperature. However, at the same time, we want the material to have a temperature coefficient of resistance that is relatively independent of the absolute value of the resistance itself. Therefore, we look at materials that have an exponential relationship between electrical resistance and temperature. For such materials, the temperature coefficient of resistance (TCR) is not strongly dependent on the absolute value of the electrical resistance of the material itself.

18.7.2 Calculating the Thermal Response of the CNT Bolometer Film

In this work, we are assessing the possibility of designing a bolometer using carbon nanotubes as both the IR-absorbing material and the electrical response material. Thus, our aim is to first determine the thermal response of the bolometer absorber that is composed of CNTs and then determine how the electrical characteristics of the CNT material depend on its changes in temperature after IR absorption.

While there are numerous geometries of CNT-based material we can consider, for this work, we focus on an absorbing material composed of a CNT film. The film is taken to consist of a random placement of CNTs that is two nanotubes thick.

18.7.2.1 Heat Flow Equation

To determine the temperature of the material in the presence of IR radiation, we start with the heat flow equation. This is a partial differential equation relating the time rate of change in temperature to the position and the rate of net heat that is absorbed by the material as a function of time and position:

$$C_v \frac{\partial T}{\partial t} = \kappa \nabla^2 T + H$$

where

C_v is the thermal capacitance (joules/degree-unit cm^3)
κ is the thermal diffusion coefficient in watts/degree cm
H is the net power absorbed by the material in watts per unit volume

To solve this equation for the CNT bolometer, we have to first determine C_v and κ for the CNT and the CNT film.

18.7.2.2 Thermal Capacitance of CNT Absorber

To determine the heat capacity of a carbon nanotube, we first determine the internal vibrational energy of the CNT and then take the derivative with respect to temperature. The internal energy is found by determining the energy of each vibrational mode, multiplying by the probability that the mode is populated using Bose–Einstein statistics, and then summing over all of the allowed modes.

The number of allowed modes will depend on the diameter and wrapping angle of the CNTs present, so we take a statistical sample. Multiplying the individual CNT heat capacity by the number of CNTs in the film provides a reasonable value for the heat capacity of the film. After following this procedure and inserting numerical values for physical constants, we arrive at the following average numerical value for thermal capacity of a CNT C_{vt}, where the length L is in microns and the diameter d is in nanometers:

$$C_{vt} \sim 1.4 \times 10^{-18} (Ld)$$

18.7.2.3 CNT Thermal Diffusion Coefficient

In addition to the thermal capacitance, we need to determine the thermal diffusion coefficient and eventually the thermal resistance of a single CNT. Experiments on isolated CNTs have fit the coefficient of thermal diffusion to data obtaining the following expression [45,46]:

$$\kappa(L,T) = \left\{ 3.7 \times 10^{-7} T + 9.7 \times 10^{-10} T^2 + \frac{9.3}{T^2}\left[1 + \frac{0.5}{L}\right] \right\}^{-1}$$

From κ, we can obtain the thermal resistance R_T of a single CNT using the following definition:

$$R_T = \frac{4L}{\kappa \pi d^2}$$

where L and d are the length and diameter of the CNT, respectively.

Using average values for CNTs gives the following numerical value for the thermal resistance of a CNT:

$$R_T \sim 5 \times 10^8 (L/d^2)$$

where

R_T is in units of Kelvin/watt
L is in units of microns
d is in units of nanometers

So a tube that is 1 μm long and 1 nm in diameter will have a thermal resistance R_T of approximately 5×10^8 K/W.

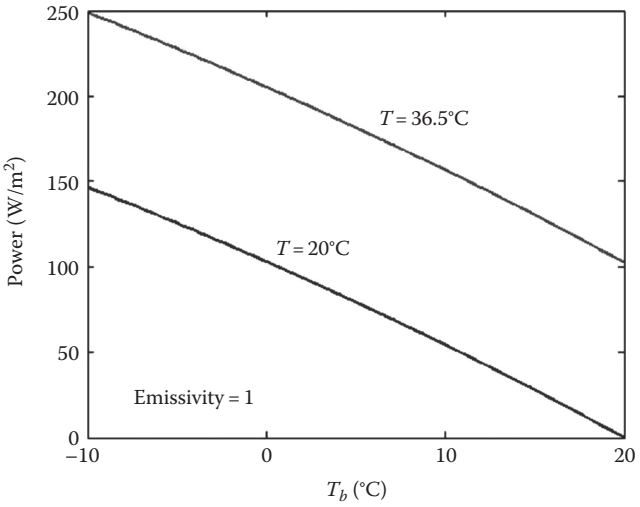


FIGURE 18.24 Net power received by bolometer as a function of bolometer temperature. (From Sood, A.K. et al., *Proc. SPIE*, 8353, 83533A, May 2012.)

18.7.2.4 Net IR Radiation Power Absorbed

Now that we have the thermal diffusion and capacitance, we are almost ready to begin solving the aforementioned heat flow equation to determine the temperature of the bolometer. However, before doing so, we need to determine H, the net IR power absorbed by the bolometer. We determine this power using the Stefan–Boltzmann law of blackbody radiation, which relates the net power absorbed to the temperatures of the subject and the bolometer using the following expression:

$$H_{net} = \sigma A \varepsilon \left(T_{obj}^4 - T_b^4\right)$$

where H_{net}, σ, A, ε, T_{obj}, and T_b are heat absorbed by the bolometer, Stefan–Boltzmann constant, cross-sectional area, emissivity, object of interest temperature, and bolometer absorber temperature, respectively. Figure 18.24 shows the net IR power absorbed by the absorber as a function of bolometer temperature for radiating objects at 20°C and 36.5°C. Cooling the bolometer by 30°C below room temperature allows for significantly more power to be absorbed, which can give rise to a much stronger signal.

18.7.2.5 Calculating the Bolometer Temperature Distribution

Using the aforementioned expressions for C_{vt}, K, and H, we expand on previous work and convert the heat flow equation earlier into a thermal network, illustrated in Figure 18.25 [6]. In actuality, there are thousands of nodes in the network for which we calculate the temperature for each.

In Figure 18.25, each resistor represents the thermal resistance of a CNT in series with the thermal resistance between adjacent CNTs. In addition, the capacitors represent the thermal capacity of a CNT, while the current sources represent the net IR radiation absorbed by each CNT. This thermal network contains thousands of nodes, and there is an equation relating the thermal resistance, capacitance, and net power for each node. This system of equations is then solved for the temperature as a function of position and time throughout the bolometer absorber [50].

Results of these calculations are shown in Figure 18.26 for different types of CNT networks. Here, we assumed that the net absorbed power is 1 nW and the pixel is tightly packed with the CNTs. The entire pixel's temperature map is obtained with a 100 × 100 temperature resolution. For the tubes, we used two different thermal resistance values: 5×10^8 and 1×10^9 K/W.

FIGURE 18.25 Illustration of thermal network superimposed on bolometer for calculating temperature map of CNT bolometer absorber. (From Fernandes, G. et al., *Proc. SPIE*, 8868, 88680S, September 2013.)

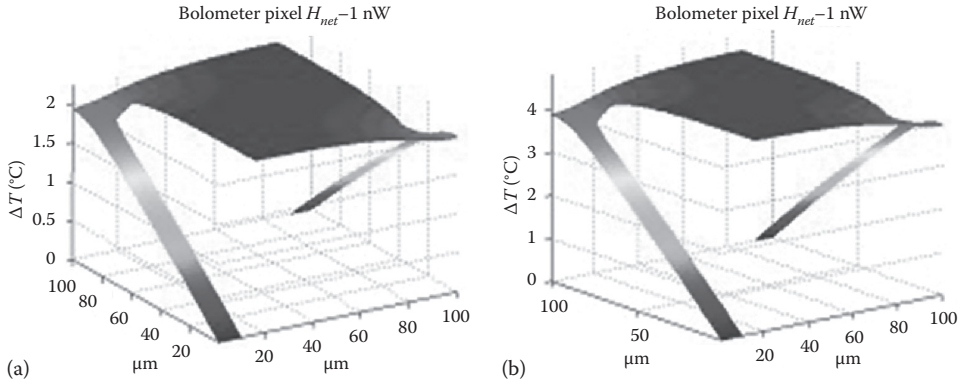


FIGURE 18.26 Temperature map of a bolometer pixel when the net absorbed power H is 1 nW and the CNT thermal resistance is 5×10^8 K/W (left) and 1×10^9 K/W (right). We assumed that the pixel is tightly packed with the CNTs. The temperature gradient of the contact legs connecting the film to the ROIC is clearly shown. (From Sood, A.K. et al., *Proc. SPIE*, 8353, 83533A, May 2012.)

As expected, the higher the thermal resistance, the higher the temperature difference from the ambient. We note that in general thermal resistance also rises with increasing temperature, resulting in further heating of hot spots compared to the case that this dependency is ignored. The temperature gradient of the contact legs connecting the film to the ROIC is clearly shown.

18.7.3 Calculating the Electrical Response of the CNT Film

To read the temperature that the bolometer pixel reaches after an exposure to IR radiation, one needs to measure the electrical resistance of the pixel. By comparing this resistance to a lookup table or using the a priori knowledge of temperature coefficient of resistance, the pixel temperature can be determined.

Therefore, in addition to having a large thermal resistance that translates into higher temperature rises, a large TCR is desirable to achieve a higher temperature resolution.

Here, TCR is defined as the change in electrical resistance per Kelvin divided by the absolute electrical resistance measured at the quiescent point, as follows:

$$\mathrm{TCR} = \frac{1}{R_e}\frac{dR_e}{dT}$$

Thus, the pixel electrical resistance after it reaches a temperature that is ΔT above its ambient becomes $R_e(T) = R_e(To)(1 + \mathrm{TCR})$. Using this relationship, the pixel temperature is calculated.

To obtain a high-temperature resolution, a large change in electrical resistance is needed upon heating. To achieve this, a substantial increase either in electron concentration or velocity (for a given electric field) is necessary. And to this end, materials with junctions where thermionic emission or tunneling is the electrical current bottlenecks offer a good solution. As the tunneling current exponentially rises with temperature, the effective change in their electrical resistance due to temperature becomes large compared to those observed in bulk materials where the change is proportional T^{γ} and γ is generally <2.

Here, a film of CNTs is proposed as the bolometer pixel material, since it is likely to have large thermal resistance and TCR values simultaneously. Both of these favorable properties are partially owed to the junctions between the tubes. As the electrical current flows along the mat, it needs to jump from one tube to the next where they intersect.

At this intersection, the carriers see a potential barrier that they need to tunnel through which gives rise to exponential increase in current upon heating. Assuming that the electron transport across this barrier is governed by a Fowler–Nordheim-type tunneling or thermionic emission, the expected TCR values can be calculated using the following expression:

$$I = q\int_0^{\infty} T_t(E)v(E)DOS(E)f(E,T)dE$$

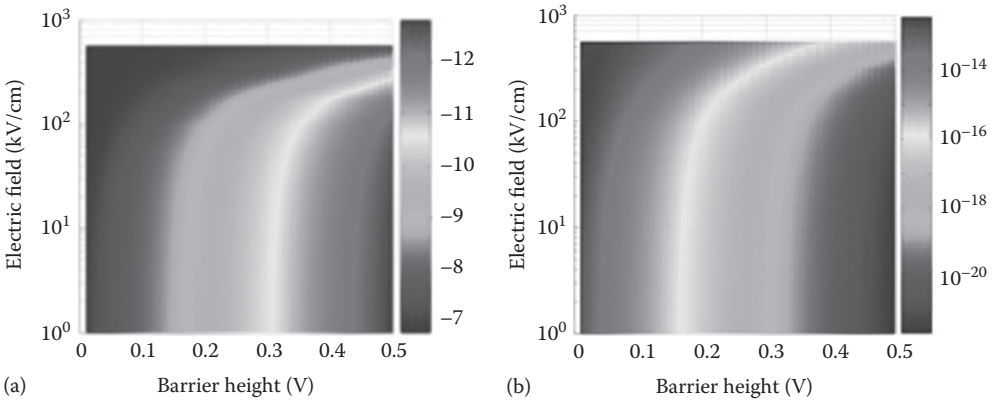


FIGURE 18.27 (a) Contour plot of TCR versus electric field and barrier height between CNTs of the film and (b) bolometer current also as a function of electric field and barrier height. The scales are the bars on the right of each contour plot in units of %TCR and amperes, respectively.

where T_t, v, DOS, and f are the transmission coefficient, thermal velocity, density of states, and distribution function, respectively, for electrons in the CNT. We performed these calculations as a function of barrier height and electric field. The results are shown in Figure 18.27. The figure on the left shows a contour plot of the theoretical values of the TCR for a CNT film.

The TCR is plotted as a function of electric field between the tubes and the barrier height. Theoretical calculations predict an extremely large TCR, which can be attributed to the relatively large barrier height between adjacent CNTs. If a lower barrier height is assumed, on the order of 0.06 eV, then a TCR of approximately 2.5% is obtained. It is also worth pointing out that such a large TCR comes at the price of extremely low output currents. The bolometer current densities, as a function of barrier height and electric field, are shown in Figure 18.27.

18.7.4 CNT Growth and Characterization

In this section, we discuss growth and characterization of single-wall carbon nanotubes (SWCNT) and multiwall carbon nanotubes (MWCNT) for use as the high-performance bolometric element for development of MWIR- and LWIR-sensitive detector elements.

Figure 18.28 presents growth of dense-oriented MWCNT "forestlike growth." The figure shows the CNT growth can be easily separated from the growth substrate. We have shown good length/diameter uniformity. Further work on the growth optimization is underway.

Figure 18.28 shows the SEM images of representative MWCNT films in the unsuspended left and suspended right form, respectively. Unlike their SWCNT counterparts, the MWCNT films contain substantial uncovered substrate areas. In addition, some minor deformation of recess is visible on suspended MWCNT films, which is similar to the SWCNT film case in the same thickness range. A TEM image of a representative individual MWCNT has a large hollow center of approximately 10–11 nm in diameter and contains approximately 40–50 CNT shells.

We have discussed recent efforts for modeling CNT–based bolometer and the experimental work for development of next-generation carbon nanostructure–based IR detectors and arrays. Our goal is to develop high-performance, high-frame-rate, and uncooled nanobolometer for MWIR and LWIR bands. We also discussed CNT growth system and its capability to grow samples of various orientations. We have also presented recent results on SWCNT and MWCNT samples that show promise for use of CNT for developing next-generation high-performance small-pixel bolometer arrays.

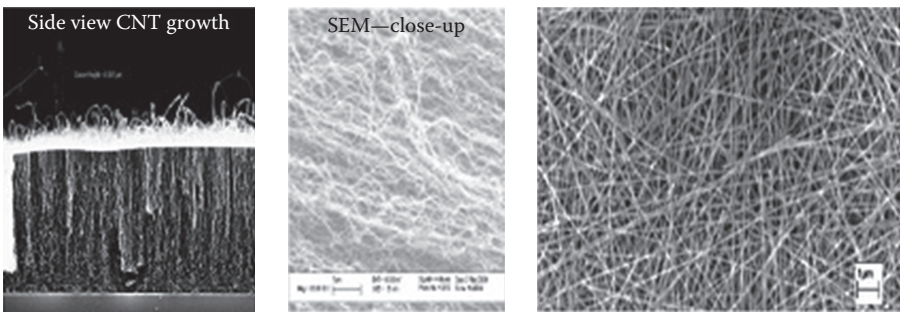


FIGURE 18.28 Growth of MWCNT forest with the ability to separate the CNT from the growth substrate with good length/diameter uniformity and the MWCNT released from the template.

18.8 Summary

In this chapter, we have discussed recent advances in detector technology, materials, and devices for optical sensing applications. The chapter has presented an overview of recent work underway on a variety of semiconductors and advanced materials such as GaN, ZnO, Si/SiGe, InGaAs, and CNT for optical sensing applications.

Optical sensing technology is critical for defense and commercial applications including optical communication. Advances in optoelectronics materials in the UV, visible, and IR, using nanostructures, and the use of novel materials such as CNT have opened doors for new approaches to apply device design methodology that are expected to offer enhanced performance and low-cost optical sensors in a wide range of applications.

We have covered the UV band (200–400 nm) and address some of the recent advances in nanostructure growth and characterization using GaN/AlGaN, ZnO/MgZnO-based technologies, and their applications. We have also discussed nanostructure-based Si/SiGe technologies (400–1700 nm) that cover various bands of interest in visible–NIR for detection and optical communication applications. The chapter has also discussed some of the theoretical and experimental results in these detector technologies.

Recent advancements in design and development of CNT-based detection technologies have shown promise for optical sensor applications. We have presented theoretical and experimental results on these device and their potential applications in various bands of interest. It is anticipated that the current research and development presented in this chapter will enable a host of new integrated technologies for a variety of defense and commercial applications.

Although numerous research activities are ongoing in the area of nanoscience and technology, we briefly made comments on such technologies to make readers aware of various research activities.

Acknowledgment

The authors gratefully acknowledge the contributions of the many distinguished scientists in the United States for development of EO/IR detector technology for optical sensor applications.

References

1. Spectral charts, Raytheon Vision Systems, Goleta, CA.
2. Dhar, N.K. IR material research at the army research laboratory. Invited keynote paper, *Proc. SPIE* 6542, 65420C (2007).
3. Long, J.P. UV detectors and focal plane array imagers based on AlGaN p-i-n photodiodes. *Opto-Electron. Rev.* 10(4), 251–260 (2002).
4. Crawford, M. Advances in AlGaN-based deep UV LEDs. *MRS Proc.* 831, E10.1 (Fall 2004).
5. Reine, M.B. Solar-blind AlGaN 256 × 256 p-i-n detectors and focal plane arrays. *Proc. SPIE* 6119, 611901 (2006).
6. Sood, A.K., Richwine, R.A., Puri, Y.R., Dhar, N.K., Polla, D.L., and Wijewarnasuriya, P.S. Multispectral EO/IR sensor model for evaluating UV, visible, SWIR, MWIR and LWIR system performance. *Proc. SPIE* 7300, 73000H (2009).
7. Rivera, A., Zeller, J., Manzur, T., Sood, A., and Anwar, M. MOCVD growth and characterization of ZnO nanowire arrays for UV detectors. *Proc. SPIE* 8540, 854008 (October 2012).
8. Liang, S., Sheng, S., Liu, Y., Huo, Z., Lu, Y., and Shen, H. ZnO Schottky ultraviolet photodetectors. *J. Cryst. Growth* 225, 110–113 (2001).
9. Zhang, J., Que, W., Jia, Q., Ye, X., and Ding, Y. Controllable hydrothermal synthesis of ZnO nanowires arrays on Al-doped ZnO seed layer and patterning of ZnO nanowires arrays via surface modification of substrate. *Appl. Surf. Sci.* 257(23), 10134–10140 (2011).

10. Lee, C.H., Yi, G.C., Zuev, Y.M., and Kim, P. Thermoelectric power measurements of wide band gap semiconducting nanowires. *Appl. Phys. Lett.* 94, 22106 (2009).
11. Falyouni, F., Benmamas, L., Thiandoume, C., Barjon, J., Lusson, A., Galter, P., and Sallet, V. Metal organic chemical vapor deposition growth and luminescence of ZnO micro- and nanowires. *J. Vac. Sci. Technol. B* 87, 1662 (2009).
12. Jeong, M.C., Oh, B.Y., Lee, W., and Myoung, J.M. Comparative study on the growth characteristics of ZnO nanowires and thin films by metal-organic chemical vapor deposition (MOCVD). *J. Cryst. Growth* 268, 149–154 (2004).
13. Kim, S.W., Fujita, S., and Fujita, S. ZnO nanowires with high aspect ratios grown by metal-organic chemical vapor deposition using gold nanoparticles. *Appl. Phys. Lett.* 86, 153119 (2005).
14. Lee, W., Jeong, M.C., and Myoung, J.M. Catalyst-free growth of ZnO nanowires by metal-organic chemical vapor deposition (MOCVD) and thermal evaporation. *Acta Mater*. 52, 3949–3957 (2004).
15. Liou, S.C., Hsiao, C.S., and Chen, S.Y. Growth behavior and microstructure evolution of ZnO nanorods grown on Si in aqueous solution. *J. Cryst. Growth* 274, 438 (2005).
16. Dong, J.W., Osinski, A., Hertog, B., Dabiran, A.M., Chow, P.P., Heo, Y.W., Norton, D.P., and Pearton, S.J. Development of MgZnO-ZnO-AlGaN heterostructures for ultraviolet light emitting applications. *J. Electron. Mater.* 34, 416–423 (2005).
17. Rivera, A., Zeller, J., Sood, A.K., and Anwar, A.F.M. A comparison of ZnO nanowires and nanorods grown using MOCVD and hydrothermal processes. *J. Electron. Mater.* 42, 894–900 (2013).
18. Ha, B., Ham, H., and Lee, C.J. Photoluminescence of ZnO nanowires dependent on O_2 and Ar annealing. *Phys. Chem. Solids* 69, 2453–2456 (2008).
19. Djurišić, A.B., Ng, A.M.C., and Chen, X.Y. ZnO nanostructures for optoelectronics: Material properties and device applications. *Prog. Quant. Electron.* 34, 191–259 (2010).
20. Anwar, M., Rivera, A., Mazady, A., Chou, H., Zeller, J., and Sood, A.K. ZnO solar blind detectors: From material to system. *Proc. SPIE* 8868, 8868B (2013).
21. Wang, Z.L., Zhu, G., Yang, Y., Wang, S., and Pan, C. Progress in nanogenerator for portable electronics. *Mater. Today* 15(12), 532–543 (December 2012).
22. Wu, W., Wen, X., and Wang, Z.L. Taxel-addressable matrix of vertical-nanowire and adaptive tactile imaging. *Science* 340(24), 952–957 (May 2013).
23. Dupuis, R.D., Ryou, H.J., and Yoder, D. High-performance GaN and AlGaN ultraviolet avalanche photodiodes grown by MOCVD on bulk III-N substrates. *Proc. SPIE* 6739, 67391B (2006).
24. Shen, S.C., Chang, Y., Limb, J.B., Ryou, J.H., Yoder, P.D., and Dupuis, R.D. Performance of deep UV GaN avalanche photodiodes grown by MOCVD. *IEEE Photonics Technol. Lett.* 19(21), 1744–1746 (November 2007).
25. Zhang, Y., Shen, S.C., Kim, H.J., Choi, S., Ryou, J.H., Dupuis, R.D., and Narayan, B. Low-noise GaN ultraviolet p-i-n photodiodes on GaN substrates. *Appl. Phys. Lett.* 94, 221109 (2009).
26. Sood, A.K., Richwine, R.A., Welser, R.E., Puri, Y.R., Dupuis, R.D., Ji, M.-H., Kim, J., Detchprohm, T., Dhar, N.K., and Peters, R.L. Development of III-N UVAPDs for ultraviolet sensor applications. *Proc. SPIE* 8868, 88680T (2013).
27. Sood, A.K., Richwine, R.A., Puri, Y.R., Dupuis, R.D., Dhar, N.K., and Balcerak, R.S. Development of GaN/AlGaN APD's for UV imaging applications. *Proc. SPIE* 7780, 77800E (2010).
28. Long, J.P. UV detectors and focal plane array imagers based on AlGaN p-i-n photodiodes. *Opto-Electron. Rev.* 10(4), 251–260 (2002).
29. Sood, A.K., Richwine, R.A., Puri, Y.R., Dhar, N.K., Polla, D.L., and Wijewarnasuriya, P.S. Multispectral EO/IR sensor model for evaluating UV, visible, SWIR, MWIR and LWIR system performance. *Proc. SPIE* 7300, 73000H (2009).
30. Zhang, Y., Shen, S.C., Kim, H.J., Choi, S., Ryou, J.H., Dupuis, R.D., and Narayan, B. Low-noise GaN ultraviolet p-i-n photodiodes on GaN substrates. *Appl. Phys. Lett.* 94, 221109 (2009).
31. Tsaur, B.-Y., Chen, C.K., and Marino, S.A. Long-wavelength Ge_xSi_{1-x}/Si heterojunction infrared detectors and focal plane array. *Proc. SPIE* 1540, 580–595 (1991).

32. Wada, H., Nagashima, M., Hayashi, K., Nakanishi, J., Kimata, M., Kumada, N., and Ito, S. 512 × 512 element GeSi/Si heterojunction infrared focal plane array. *Proc. SPIE* 3698, 584–595 (1999).
33. Sood, A.K., Richwine, R.A., Puri, Y.R., DiLello, N., Hoyt, J.L., Dhar, N., Balcerak, R.S., and Bramhall, T.G. Development of SiGe arrays for visible-near IR imaging applications. *Proc. SPIE* 7780, 77800F (2010).
34. Bublik, V.T., Gorelik, S.S., Zaitsev, A.A., and Polyakov, A.Y. Calculation on the binding energy of Ge-Si solid solution. *Phys. Status Solidi* 65, K79–K84 (1974).
35. Sood, A.K., Richwine, R.A., Puri, Y.R., DiLello, N., Hoyt, J.L., Akinwande, T., Dhar, N.K., Balcerak, R.S., and Bramhall, T.G. Characterization of SiGe detector arrays for visible-near IR imaging sensor applications. *Proc. SPIE* 8012, 801240 (2011).
36. Sood, A.K., Richwine, R.A., Pethuraja, G., Puri, Y.R., Lee, J.-U., Haldar, P., and Dhar, N.K. Design and development of wafer-level short wave infrared micro-camera. *Proc. SPIE* 8704, 870439 (2013).
37. van der A, R.J., Hoogeveen, R.W.M., Spruijt, H.J., and Goede, A.P.H. Low noise InGaAs infrared (1.O-2.4tm) focal plane arrays. *Proc. SPIE* 2957, 54–65 (1997).
38. Acton, D., Jack, M., and Sessler, T. Large format short-wave infrared (SWIR) focal plane array (FPA) with extremely low noise and high dynamic range. *Proc. SPIE* 7298, 72983E (2009).
39. Onat, B.M., Huang, W., Masaun, N., Lange, M., Ettenberg, M.H., and Dries, C. Ultra low dark current InGaAs technology for focal plane arrays for low-light level visible-shortwave infrared imaging. *Proc. SPIE* 6542, 65420L (2007).
40. Boisvert, J., Isshiki, T., Sudharsanan, R., Yuan, P., and McDonald, P. Performance of very low dark current SWIR PIN arrays. *Proc. SPIE* 6940, 69400L (2008).
41. MacDougal, M., Geske, J., Wang, J., and Follman, D. Short-wavelength infrared imaging using low dark current InGaAs detector arrays and vertical-cavity surface-emitting laser illuminators. *Opt. Eng.* 50(6), 061011 (2011).
42. Hood, A.D. Large InGaAs focal plane arrays for SWIR imaging. *Proc. SPIE* 8353, 83530A (2012).
43. Yuan, H., Meixell, M., Zhang, J., Bey, P., Kimchi, J., and Kilmer, L.C. Low dark current small pixel large format InGaAs 2-D photodetector array development. *Proc. SPIE* 8353, 835309 (2012).
44. Yuan, H., Apgar, G., Kim, J., Laquindanum, J., Nalavade, V., Beer, P., Kimchi, J., and Wong, T. FPA development: From InGaAs, InSb, to HgCdTe. *Proc. SPIE* 6940, 69403C (2008).
45. Pop, E., Mann, D., Wang, Q., Goodson, K., and Dai, H. Thermal conductance of an individual single-wall carbon nanotube above room temperature. *Nano Lett.* 6, 96 (2006).
46. Akturk, A., Goldsman, N., and Metze, G. Self-consistent modeling of heating and MOSFET performance in 3-D integrated circuits. *IEEE Trans. Electron. Dev.* 52(11), 2395–2403 (2005).
47. Sood, A.K., Egerton, E.J., Puri, Y.R. et al. Design and development of CNY based micro-bolometer for IR imaging applications. *Proc. SPIE* 8353, 83533A (May 2012).
48. Fernandes, G., Kim, J.H., Xu, J., Sood, A.K., Dhar, N.K., and Dubey, M. Unleashing giant TCR from phase–Changes in carbon nanotube composites. *Proc. SPIE* 8868, 88680S (September 2013).
49. Blackwell, R., Lacroix, D., Bach, T. et al. 17 micron microbolometer FPA Technology at BAE Systems. *Proc. SPIE* 7298, 72980P (2009).
50. Li, C., Skidmore, G., Howard, C., Clarke, E., and Han, J. Advancement in 17-micron pixel pitch uncooled focal plane arrays. *Proc. SPIE* 7298, 72980S (2009).
51. Schimert, T., Hanson, C., Brady, J. et al. Advanced in small-pixel, large-format alpha-silicon bolometer arrays. *Proc. SPIE* 7298, 72980T (2009).
52. Trouilleau, C., Fieque, B., Noblet, S. et al. High performance uncooled-amorphous silicon TEC less XGA IRFPA with 17 micron pixel pitch. *Proc. SPIE* 7298, 72980Q (2009).

19

Vehicular Sensor Networks

19.1 Introduction **19**-1
19.2 Vehicular Sensor Networks for Congestion Avoidance **19**-2
19.3 Driver Rerouting System for Congestion Avoidance **19**-3
Traffic Data Representation and Estimation • Congestion Prediction • Selection of Vehicles to Be Rerouted
19.4 Rerouting Strategies **19**-6
Single Shortest Path Strategies: Dynamic Shortest Path • Multiple Shortest Path Strategies • Rerouting Process • Dynamic Traffic Assignment
19.5 Evaluation of the Rerouting System **19**-15
Experimental Settings • Results and Analysis: Average Travel Time
19.6 Summary **19**-24
References **19**-25

Juan Pan
New Jersey Institute of Technology

Cristian Borcea
New Jersey Institute of Technology

19.1 Introduction

The global population spends a tremendous amount of time in traffic every day. For example, it is estimated that average American spends at least 50 min a day commuting [1]. Vehicular sensor networks, which leverage vehicular sensors and wireless communication, could improve our in-car experience by making travel safer and faster. In the past decades, with the advance of the new technologies, various electronics components started to be included in the automotive system such as acoustic, infrared, and still/motion video cameras. In recent years, most new vehicles come already equipped with GPS receivers and navigation systems. Car manufacturers such as Ford, GM, BMW, and Toyota have already announced efforts to include significant computing power within their automotive design [2–5]. After Google revealed its first self-driving car, a number of automakers began venturing into this research area. This trend is expected to continue, and, in the near future, the number of vehicles equipped with computing technologies and wireless network interfaces will increase dramatically. These vehicles will be able to run network protocols that will exchange messages for safer and more fluid traffic on the roads. Several protocols have been proposed for implementing vehicular communication. The most promising standard is IEEE 802.11p. The physical layer of 802.11p is dedicated short-range 5.9 GHz for communication among vehicles and with roadside infrastructure [6].

So far, a diversity of applications have been proposed and evaluated over vehicular sensor networks. Based on the functionality, those applications can be divided into three categories: road safety, local information sharing, and congestion monitoring/avoidance.

A typical safety-related application example is highway cooperative collision avoidance [7]. On highways, an initial collision is often followed by a series of collisions involving the following vehicles. This is due to the inability of drivers to react in time to emergency situations. By using wireless

communication, the initial car involved in the accident can warn the nearby vehicles such that their drivers can hit the brake faster. Another example is abnormal driver behavior (e.g., drunkenness and fatigue) detection in literature [8], where each vehicle's onboard unit calculates the score of abnormality for each driver through Dynamic Bayesian Networks based on the contextual information collected from various sources including traffic management center (TMC), road site units, and other vehicles' broadcasting messages. Once the abnormality score is above a certain threshold, an alert is sent to the driver to prevent accidents from happening.

A large body of applications focus on creating information sharing services to facilitate people's daily lives such as parking availability, taxi dispatching, and context-aware services. EZCab [9] describes a faster and more scalable system that allows people to book nearby cabs using their smartphones. When needed, a client sends a booking request to nearby vehicles, and the free cabs reply to the client by means of vehicular ad hoc communication. Similarly, the work in [10] proposes real-time parking navigation service in large parking lots through vehicular ad hoc communication. Each vehicle acts as a parking information sensor and communicates with the incoming vehicles to find the most convenient parking spot. The literature [11] presents a context-aware migratory service model for regional information sharing such as traffic jam, local advertisements, and parking lot availability. The information in a specific area is sensed by the passing vehicles and disseminated to other vehicles in proximity.

Recently, utilizing vehicular sensor networks for congestion monitoring/avoidance has drawn great attention. Vehicular sensor networks provide a cost-effective platform for traffic data collection and vehicle rerouting to avoid congestion and to improve the travel time. The rest of this chapter will focus on leveraging vehicular sensor networks for congestion avoidance.

19.2 Vehicular Sensor Networks for Congestion Avoidance

With the deployment of traffic surveillance infrastructure on more roads (e.g., loop detectors, video cameras), we have started to witness web-based services/applications that present the drivers with the current view of the traffic and let them decide which route to follow.

Projects such as Mobile Millennium [12,13], CarTel [14], JamBayes [15], Nericell [16], and surface street estimation [17] use vehicle probe data collected from onboard GPS devices to reconstruct the state of traffic and estimate the shortest travel time. Similarly, services such as INRIX [18] provide real-time traffic information at a certain temporal accuracy, which allows drivers to choose alternative routes if they are showing lower travel times. Systems such as Google Maps and Microsoft's Bing are able to forecast congestion and its duration by performing advanced statistical predictive analysis of traffic patterns. Differently, Waze [19] utilizes social networks to provide traffic information. In these applications, vehicles act as traffic sensors and relay traffic reports to TMC to obtain an accurate view of the current traffic. However, these solutions do not try to prevent congestions explicitly (i.e., they are reactive solutions) and provide the same guidance for all vehicles on the road at a certain moment as function of their destination (i.e., pull model in which drivers query for the shortest route to destination). Therefore, similar to route oscillations in computer networks, they could lead to unstable global traffic behavior: when it happens, congestion is switched from one route to another if a significant number of drivers use the guidance.

TrafficView [20] and StreetSmart [21] are examples of distributed systems that could help drivers avoid congestion: vehicles collect and disseminate traffic information to nearby vehicles, and thus, the drivers may change routes if they observe traffic congestions. PeerTIS [22,23] discusses an efficient and scalable architecture that incorporates a peer-to-peer overlay into a vehicular network based on cellular Internet communication to provide traffic information to drivers. Scientists are looking for inspiration in Biology and Internet protocols as well. Wedde et al. developed a road traffic routing protocol in work [24], Bee-JamA, based on honeybee behavior. Similarly, Tatomir et al. [25] proposed a route guidance system based on trial-laying ability of ants. Inspired by the well-known Internet routing protocols, Prothmann et al. [26] proposed decentralized organic traffic control. Some other distributed

applications include intelligent traffic light adjustment [27]. Nevertheless, these applications simply use the collected data for naive suboptimal shortest path computation that potentially switch congestion from one spot to another. Besides, for larger-scale networks, due to wireless contention, latency, and packet loss, vehicles are unable to acquire complete global traffic view and thus make suboptimal rerouting decisions.

Many of the aforementioned solutions expect the drivers to react to congestion by changing their routes. According to Wardrop's first traffic equilibrium principle [28], if accurate traffic time on each road segment can be obtained, then drivers could be able to reach user-optimum traffic equilibrium. It is known, however, that no true equilibrium can be found under congestion [12]. Even more important, the usefulness of such services is limited by their reactive nature: they cannot prevent congestions.

This situation could be avoided by new solutions based on user-optimal dynamic traffic assignment (DTA) that leads to either system-optimal or user-optimal route assignments. DTA research can be classified into two categories: analytical methods and simulation-based models. Analytical models, such as [29–31], formulate DTA as either nonlinear programming problems, optimal control problems, or variational inequalities. Although they provide theoretical insights, the computational intractability prevents their deployment in real systems [32]. Simulation-based approaches [33–36] have gained greater acceptance in recent years: simulations are used to model the theoretical insights that cannot be derived quickly from analytical approaches. Unfortunately, there is still a significant gap between the theoretical or simulation results and potentially deployable solutions: tractability for large-scale road networks given the computational burden associated with the simulator, capability of providing real-time guidance, effectiveness in the presence of congestion, and behavior of drivers who ignore the guidance.

In summary, despite good progress, existing solutions have not been able to provide an intelligent, cost-effective, easily deployable, real-time traffic guidance system. Such a system should possess the following properties: (1) fully utilize vehicles as mobile sensors to provide accurate traffic view, (2) be able to compute near-optimal routing paths for vehicles based on the collaborative knowledge collected from vehicles, (3) provide effective vehicle rerouting guidance to individual drivers in a timely manner, and (4) be robust and work well even when drivers do not follow the guidance. In the following section, we present a system that achieves these goals.

19.3 Driver Rerouting System for Congestion Avoidance

The objective is to implement and evaluate a real-time, cost-effective, and easily deployable vehicular traffic guidance system that reduces the effect of traffic congestions and lowers the trip times for all drivers. Implicitly, fuel consumption and pollution will be reduced as well. To achieve this goal, we developed a system consisting of smartphone-based vehicular networks and a back-end server infrastructure for traffic monitoring and coordination. Smartphones were chosen as the vehicular platform because they are already carried by drivers in many vehicles, are powerful (have several communication interfaces, GPS, accelerometer, powerful CPU, plenty of storage, etc.), and are easily programmable. Once they become widespread, vehicular computing systems could be considered instead of smartphones. Figure 19.1 presents a comparison between these existing solutions and the system. We propose to leverage the smartphone ubiquity to design and quickly deploy a cheap and effective traffic rerouting system. Such a system will benefit all of us through faster routes, less money spent on gas, and lower pollution.

Specifically, our traffic guidance system is composed of (1) a centralized traffic monitoring and rerouting service (which can physically be distributed across several servers) and (2) a vehicle software stack for periodic traffic data reporting (position, speed, direction) and showing alternative routes to drivers. Vehicles run this software either on a smartphone or an embedded vehicular system. Vehicles are equipped with GPS receivers and can communicate with the service over the Internet when needed. When starting a trip, each vehicle informs the service of its current position and destination; the service sends back a route computed according to its strategy. It is assumed that the service knows the road network as well as the capacity and legal speed limits on all roads.

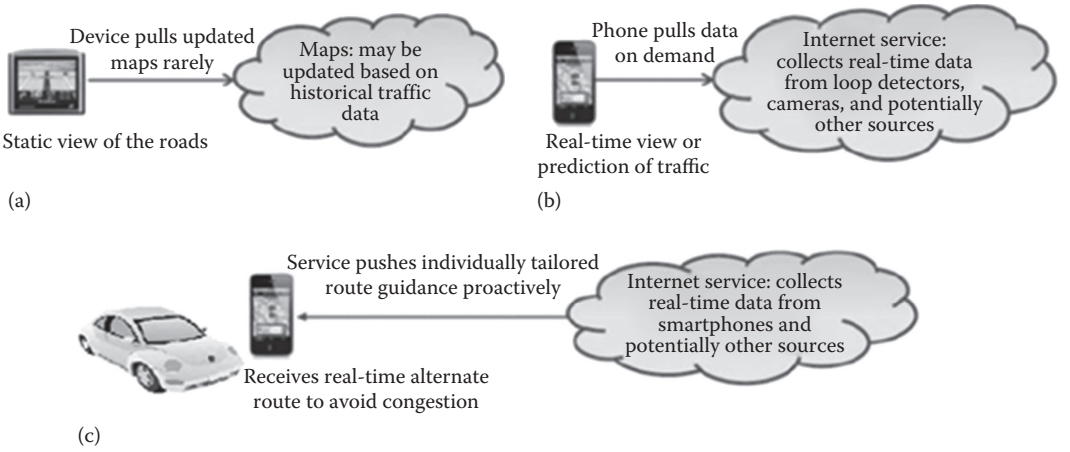


FIGURE 19.1 The system overview: (a) in-car navigation system, (b) web-based services, and (c) proposed system. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.,* 62(8), 3551, 2014.)

Logically, the traffic guidance system operates in four phases executed periodically: (1) data collection and representation, (2) traffic congestion prediction, (3) vehicle selection for rerouting, and (4) alternative route assignment for each vehicle and pushing the guidance to the vehicles. Since data collection has been studied extensively in the literature, this issue is not addressed, and it is assumed that the centralized service receives traffic data from vehicles and roadside sensors where available.

19.3.1 Traffic Data Representation and Estimation

The road network is represented as a directed, weighted graph, where nodes correspond to intersections, edges to road segments, and weights to estimated travel times. The weights are updated periodically as new traffic data become available. Several methods can be employed to estimate the travel time over a road segment. For instance, using vehicle probe data collected from onboard GPS devices to reconstruct the state of traffic is a well-studied topic [13,37]. We use Greenshield's model [38] to estimate the travel time since it is used extensively in DTA models by transportation researchers. The model considers that there is a linear relationship between the estimated road speed V_i and the traffic density K_i (vehicles/m) on road segment i, as in the following equation:

$$V_i = V_f\left(1 - \frac{K_i}{K_{jam}}\right), \quad T_i = \frac{L_i}{V_i}, \tag{19.1}$$

where

K_{jam} and V_f are the traffic jam density and the free flow speed for road segment i, respectively
T_i and L_i are the estimated travel time and length for the same segment, respectively

The free flow speed V_f is defined as the average speed at which a motorist would travel if there were no congestion or other adverse conditions. To simplify our implementation, we consider that the free-flow speed is the road speed limit. Basically, K_i/K_{jam} is the ratio between the current number of vehicles N_i and the max number of vehicles N_{max}. The current number of vehicles N_i is obtained from the traffic data collected by the service, whereas the N_{max} = length of road/(average vehicle length + min gap).

19.3.2 Congestion Prediction

Periodically, the service checks the road network to detect signs of congestion. A road segment is considered to exhibit congestion signs when $K_i/K_{jam} > \delta$, where $\delta \in [19]$ is a predefined threshold value. Choosing the right value for delta is particularly important for the service performance. If it is too low, the service could trigger unnecessary rerouting; this may lead to an increase in the drivers' travel times. If it is too high, the rerouting process could be triggered too late, and congestion will not be avoided. The evaluation in Section 19.5.2 confirms these hypotheses.

19.3.3 Selection of Vehicles to Be Rerouted

When a certain road segment presents signs of congestion, the service looks for nearby vehicles to reroute. Specifically, we select vehicles from incoming segments (i.e., segments that bring traffic into the congested one). To decide how far from congestion to look for candidates for rerouting, the service uses a parameter L (level). This parameter denotes the furthest distance (in number of segments) a candidate vehicle can be away from the congested segment. In practice, L could be computed as function of the severity of congestion; for example, we can use the "level of service" (LOS) defined in the *Highway Capacity Manual* [39]. L's value has to be large enough to mitigate congestion. If L is too high, however, more vehicles than necessary will be selected for rerouting, which can have undesired consequences (e.g., creating congestion in another spot). Since our focus is on the rerouting algorithms and the analysis of their performance, we decided to consider L a tuning parameter that is varied during our experiments.

The service performs a breadth-first search (BFS) on the inverted network graph (i.e., the road network graph is directed), starting from the congested segments with maximum depth L, and considers all these cars as candidates for rerouting. The process is illustrated in Figure 19.2. Assuming L = 2 and signs of congestion are detected on the segment Sc, the system recursively selects the vehicles situated on the incoming segments of the congested segment in two steps. First, the vehicles located on segments S1 are included in the candidate set, followed by the vehicles situated on segments S2. We select the union of all the vehicles affected by all the congested segments in the whole network within the same level, then perform vehicle ranking, path computations, and path assignments detailed in the following sections.

19.3.3.1 Ranking of the Selected Vehicles

The selected vehicles need to be ranked and assigned to alternative paths according to their rank. In this way, the performance of the strategies improves. The impact a congested road segment has on a vehicle's

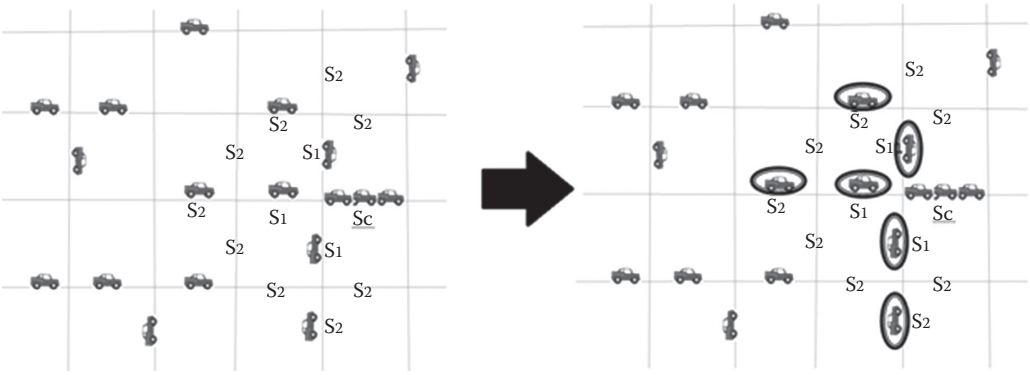


FIGURE 19.2 The vehicle selection process. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

travel time is different depending on the remaining distance to the vehicle's destination. Intuitively, the drivers that are close to their arrival point may have a different perception of the congestion than the drivers that are far away from their destination. Our system uses an urgency function to rank the vehicles that are selected for rerouting. Hence, the vehicles with higher urgency are rerouted first and get relatively better routes.

Definition 19.1

Given a set of vehicles $V = (v_1, v_2, v_3, \ldots, v_m)$ to be rerouted, we define two urgency functions to compute the rerouting priority of a vehicle in V:

- *Absolute congestion impact (ACI)*: ACI = RemTT–RFFTT
- *Relative congestion impact (RCI)*: RCI = (RemTT–RFFTT)/RFFTT where RemTT is the remaining travel time and RFFTT is the remaining free flow travel time for the vehicle

ACI measures the impact of all the (congested) segments of the remaining journey of a vehicle, since (*RemTT–RFFTT*) is the absolute increase in the travel time with respect to the free-flow travel time. With *ACI*, the longer the remaining distance to the destination for a vehicle is, the higher is the probability for that vehicle to get a higher rank (as the difference *RemTT–RFFTT* normally increases with the *RFFTT*). On the other hand, *RCI* weighs the congestion impact on a vehicle relative to its remaining travel time. Hence, *RCI* gives a higher priority to vehicles that are close to their destination. Since the further the vehicle is from its destination, the higher are the number of alternative paths and the potential benefit of rerouting, we expect *ACI* to perform better than RCI in our system.

19.3.3.2 Alternative Route Computation and Assignment

There are two main requirements for the rerouting algorithm: (1) compute an alternative route for each driver that improves both single driver's trip time and global network efficacy and (2) push the guidance to drivers fast to allow them enough time to switch on the new route. Essentially, a best effort algorithm is required, which finds good enough alternatives with real-time constraint.

Accordingly, two types of solutions were developed. The first solution is "single shortest path strategies," and the second one is "multiple shortest path strategies." The former computes one single path for each vehicle collaboratively according to the other vehicles' paths. The latter computes k loopless shortest paths [40–42] according to current travel time and picks the optimal one for each vehicle. This is because one of the main goals of the system is to prevent moving congestion from one path to another. If all vehicles are rerouted on their current shortest paths, the algorithm might end up doing exactly that. For this reason, vehicles are provided with alternative paths that lower their currently predicted travel time, but these paths do not have to be the shortest. The specification of the rerouting algorithms is presented in the following section.

19.4 Rerouting Strategies

Recent research has proved that real-time traffic flow data and road travel time can be determined based on data reported by vehicles or roadside sensors [13,15,16]. The question is how to utilize this knowledge in an intelligent fashion to avoid congestion and reduce the drivers' travel times. This section presents five rerouting strategies that we classify into two categories. The first, presented in Section 19.4.1, includes two rerouting strategies that compute a single, alternative new path for each rerouted vehicle. The strategies are based on the well-known Dijkstra algorithm and on the A* algorithm with a modified heuristic, respectively. Section 19.4.2 presents the second category consisting of three rerouting

strategies that compute multiple, alternative new paths for each of the rerouted vehicles. Then, different heuristics are used to choose the best alternative path to be assigned to a vehicle. Based on the five proposed strategies, we describe the main rerouting process executed by the traffic guidance system in Section 19.4.3. Finally, Section 19.4.4 presents a DTA strategy [36] that we use as a baseline to measure the effectiveness and efficiency of the proposed strategies.

19.4.1 Single Shortest Path Strategies: Dynamic Shortest Path

Dynamic shortest path (DSP) is a classical rerouting strategy that assigns the selected vehicles to the path with the lowest travel time. However, different from the existing systems, the system described in this system takes a proactive approach. Specifically, each time a road segment presents signs of congestion, the service obtains the set of cars whose paths intersect this road segment and computes for each car a new shortest path based on the current travel time in the road network. Therefore, the path of each car can be periodically updated on an event-driven basis. The advantage of this strategy lays in its simplicity and consequently reasonable computational cost, that is, O(E + Vlog(V)) [43], where E is the number of road segments and V is the number of intersections of the road network. This strategy is expected to provide good results when the number of rerouted vehicles is low, since in this case, the risk of switching congestion from one spot to another is low. Hence, locally redirecting the traffic when congestion happens should be sufficient in this case. On the other hand, when the traffic density is higher, there is an increased risk of switching the congestion from one road to another. Moreover, the rerouting frequency for a driver is likely to increase in this case, which can be annoying to drivers.

19.4.1.1 A* Shortest Path with Repulsion

The DSP strategy only takes into account the current view of the traffic when performing rerouting, without considering the impact the rerouting will have on the future traffic. To address this limitation, A* shortest path with repulsion (AR*) is proposed, which modifies the A* search algorithm to include the prior rerouting decisions into the computation of the current shortest path. A* [44] uses best-first search and heuristic function to determine the order in which the network nodes (road intersections in this case). Given a node x, a heuristic function F(x) is computed as the sum G(x) + H(x). G(x) is the path cost from the start node to x, which corresponds to the travel time in this case, while H(x) is a heuristic estimation of the remaining travel time from x to the destination node. In addition, H(x) has to be "admissible" (i.e., it must not overestimate the remaining travel time to the destination) to produce the shortest path between the source and the destination. Therefore, H(x) is computed as the Euclidean distance divided by the maximum speed in the road network.

Future congestion occurs if many drivers take the same road segment within the same future time window. As the drivers share their route information with the central service, it is possible to estimate the future footprint of each driver in the road network.

Definition 19.2

A weighted footprint counter, fc_i, of a road segment i is defined as $fc_i = n_i \times \omega_i$, where n_i is the total number of vehicles that are assigned to paths that include segment i and w_i is a weight associated with i; $\omega_i = len_{avg}/(len_i \times lane_i) \times Vf_{avg}/Vf_i$, where len_{avg} is the average road segment length in the network, Vf_{avg} is the average free-flow speed of the network, len_i is the length of i, Vf_i is the free-flow speed of i, and $lane_i$ is the number of lanes of i.

In the formula, n_i represents the discretized future traffic flow on road i. The weights were incorporated into the formula in order to count for different road characteristics. For example, suppose there are two road segments r_i and r_j. Although $n_i = n_j$, the segments should not be treated equally since r_i has

higher capacity (more lanes or longer length); thus, the possibility of causing congestion is lower. In other words, the impact of the traffic flow n_i on road r_i is lower than n_j on r_j even though $n_i = n_j$.

In AR*, the heuristic function F(x) is modified to include the other vehicles sharing the same path as a repulsive force. Specifically, the repulsive score R(x) of a node x is defined as the sum of the weighted footprint counters (cf. Definition 19.2) from the starting node to the node x. Thus, the path-cost function becomes $F(x) = (1 - \beta) \times (G(x) + H(x)) + \beta \times R(x)$, where G(x) and H(x) are computed same as in the original algorithm. G(x) + H(x) measures the travel time factor, while R(x) reflects the impact of other vehicle traces on the examined path. Since the travel time and the repulsive force use different metrics, the values are normalized, and F(x) is a linear combination of the two factors. Parameter β presents the weight of the repulsion. If β is too high, the resulting path will be diverted too far away from the optimal. Similarly, if β is too low, it will be the same as the naive shortest path (DSP) strategy. Therefore, the beta value was varied from 0.01 to 0.5, and it was observed that the interval [0.05, 0.2] results in better performance in Section 19.5.2.

The complete algorithm is presented as pseudocode in Algorithm 19.1. Starting from the initial node, the algorithm maintains a queue of nodes to be traversed, denoted as the open set (lines 3–5). At each iteration, the node with the lowest F score value is removed from the queue (lines 16–19), the values of its neighbors are updated accordingly (lines 27–28), and these neighbors are added to the queue (line 30). The algorithm continues until the end node has been reached or until the queue is empty. The normalization of the travel time and the repulsive force factors is done at line 14. A path is returned at line 18 if found, otherwise an empty path is returned at line 48.

Algorithm 19.1: A star shortest path with repulsion rerouting.

```
1:  procedure AstarRepulsion(start,end)
2:  P[start]=empty {the reverse pointer of the path, which is used to re-construct the path}
3:  closedset = set() {The set of nodes already evaluated}
4:  openset = set()
5:  openset.add(start) {The set of tentative nodes to be evaluated, initially containing the start node}
6:  Gscore[start] = 0.0 {Travel time cost from start along best known path}
7:  ILscore[start] = Euclidean(start, end)/maxspeed
8:  Rscore[start] = 0.0
9:  Fscore[start] = 1.0
10: while openset is not empty do
11:     sumF=S umFscore (openset)
12:     sumR=S umRscore (openset)
13:     for all node in openset do
14:         Fscore [node]=(1-/β) *Fscore [node] / SumF + β* Rscore [node] / Sum R
15:     end for
16:     current=get, least Fscore (openset)
17:     if current==end then
18:         return (Fscore[current] ,P)
19:     end if
20:     openset .remove (current)
21:     closedset.add(current) {add current to closedset}
22:     for all edge in current.outEdges do
23:         node=edge.endnode
24:         if node in closedset then
25:                 continue
26:         end if
```

```
27:         tentative_g_score=Gscore [current] + edge.actualtime
28:         tentative_r_score=Rscore[current] + edge.weight .foot prints
29:         if node not in openset then
30:                 openset.add (node)
31:                 Hscore[node]=Euclidean(node, end)/maxspeed
32:                 tentative is_better=True
33:         else
34:             if tentative_g_score<Gscore[node] then
35:                 tentative is_better=True
36:             else
37:                 tentative is_better=False
38:             end if
39:         end if
40:         if tentative_is_better == True then
41:             P[node]=edge
42:             Gscore[node]=tentative_g_score
43:             Rscore[node]=tentative_r_score
44:             Fscore[node] =Gscore [node]+Hscore [node]
45:         end if
46:     end for
47: end while
48: return (0.0,{})
49: end procedure
```

Figure 19.3 illustrates a simple example of how AR* is used in rerouting. It is assumed that vehicles v_1, v_2, and v_3 have the same origin and destination, i.e., from ab to ij, and need to be re-routed and that urgency (v_1) > urgency (v_2) > urgency (v_3). At the beginning, since no vehicle has been assigned any path, AR* performs normal A* search and assigns the shortest path ab, bc, cd, di, and ij to vehicle v_1. When computing the shortest path for vehicle v_2, AR* will find ab, bg, gh, hi, and ij. Although v_2 has the same destination as v_1, the path found by AR* is different since it considers the footprints produced by v_1 as a repulsion. Hence, AR* avoids the already assigned paths as much as possible, while still keeping the new path as short as possible. Finally, the procedure is repeated for vehicle v_3, and the path ab, bc, cd, di, and ij is obtained for the same reasons.

Note that AR* has to be employed by the rerouting system in an iterative manner. Namely, after the selected vehicles to be rerouted have been ranked based on their urgency, the system calculates

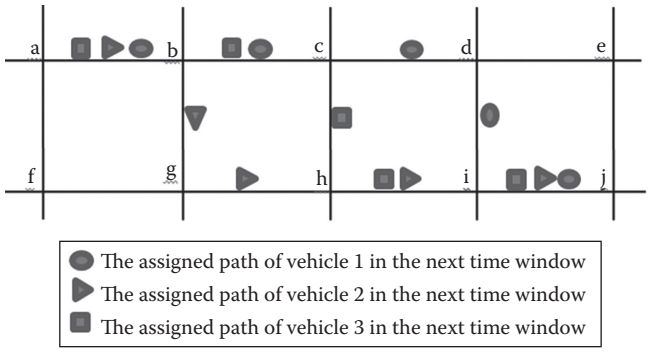


FIGURE 19.3 AR* rerouting example. All road segments have the same weight and β = 0.5. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.,* 62(8), 3551, 2014.)

sequentially each vehicle's route starting from the most urgent one. Therefore, in the case of AR*, the computation time increases linearly with the number of rerouted vehicles. On the other hand, as explained in the next sections, the rest of the proposed rerouting methods optimize this phase by grouping the vehicles to be rerouted based on their origin–destination (OD), which leads to lower computational complexity.

19.4.2 Multiple Shortest Path Strategies

The two strategies proposed previously compute a single path for each rerouted vehicle. However, the two methods have opposite behaviors. Firstly, DSP sacrifices effectiveness (since it does not consider the impact of rerouting on the future traffic) to optimize the computational cost (by grouping the rerouted vehicles on their OD). Secondly, AR* trades efficiency (since it computes an alternative path for each vehicle) for effectiveness (by taking into account the future traffic configuration). In this section, a new class of rerouting strategies designed to obtain the best trade-off between efficiency and effectiveness are introduced. For these strategies, the rerouting process is divided into two steps. First, k-shortest paths are computed for each selected vehicle based on the travel time in the road network, where k is a predefined parameter. Compared to DSP, this approach involves a higher computation time, but it still permits to group vehicles on their OD. Therefore, the computation time is expected to be lower than AR*. Second, the vehicles are assigned to one of their k-shortest paths in the order of their ranking. Among the k-shortest paths, the algorithm selects the path the least employed by other vehicle traces. Hence, this strategy is expected to have effectiveness similar to AR*. Three heuristics for the selection of the best path among the k-shortest paths are proposed.

19.4.2.1 Random k-Shortest Paths

Random k-shortest paths (RkSP) assign each selected vehicle to one of the k paths randomly. The goal is to avoid switching congestion from one spot to another by balancing the rerouted traffic among several paths. Compared to DSP, the price to pay is a higher computational complexity, $O(kV(E + V\log(V)))$ [40], which increases linearly with k. Although a larger k will allow better traffic balancing, it also increases the difference in the travel time among the k paths. Therefore, to prevent an excessive increase in the travel time for some drivers, RkSP limits the maximum allowed relative difference between the fastest and the slowest path to 20%.

19.4.2.2 Entropy Balanced k-Shortest Paths

While RkSP addresses the main potential shortcoming of DSP (i.e., moving congestion to another spot), it has its own deficiencies. First, it increases the computational time, which matters because the alternative paths must be computed and pushed to vehicles before they pass the rerouting intersection. Second, it assigns paths randomly to vehicles, which is far from optimal both from a driver point of view and from the global traffic point of view. To address this second shortcoming of RkSP, the entropy-balanced k-shortest path (EBkSP) strategy is proposed. The idea is to perform a more intelligent path selection by considering the impact that each selection has on the future density of the affected road segments. The more intelligent path selection comes at the cost of a slightly increased complexity.

However, this optimization is expected to improve the traffic from a global point of view. In addition, as in AR*, EBkSP ranks the cars to be rerouted based on an urgency function that quantifies the degree to which the congested road affects the driver travel time. Thus, the more affected vehicles will have priority and be rerouted first.

The entropy idea comes from the Shannon information theory [45]. Several works [46,47] have successfully applied it to compute the popularity of a visited area among all the users. To avoid creating new congestions through rerouting, a "popularity" measure is associated with road segments in EBkSP. Entropy is used to define the popularity of a path as follows.

Definition 19.3

Let $(p_1, \ldots, p_k)$ be the set of paths computed for the vehicle that will be assigned next. Let $(r_1, \ldots, r_n)$ be the union of all segments of $(p_1, \ldots, p_k)$, and let $(fc_1, \ldots, fc_n)$ be the set of weighted footprint counters associated with these segments. The popularity of p_j is defined as $Pop(p_j) = e^{E(p_j)}$. $E(p_j)$ is the weighted entropy of p_j and is computed as $E(p_j) = -\sum_{i=1}^{n} (fc_i/N) \ln(fc_i/N)$, $N = \sum_{i=1}^{n} n_i$. The value of $E(p_j)$ measures the probability that a number of vehicles will be on the path p_j in a time window. According to this definition, the value of $Pop(p_j)$ is $0 < Pop(p_j) < m$, where m is the number of vehicles. $Pop(p_j)$ has the maximum value m when every previously assigned vehicle traverses entirely p_j (i.e., they take the same path). $Pop(p_j)$ has the minimum value when no one takes the path p_j. Intuitively, the higher the popularity of a path, the higher the probability that more drivers will take this path.

After vehicle selection and ranking, the central server assigns each vehicle to the least popular path among its k-shortest paths in order to avoid potential future congestions. Specifically, the first vehicle is assigned the current best path without considering others. Then, the road network footprints are updated based on the new path. When assigning the second vehicle, the popularity score of its k-shortest paths is calculated, and the least popular path will be chosen. The process is then repeated for the rest of the rerouted vehicles.

Figure 19.4 illustrates an example of EBkSP rerouting. It is assumed that vehicles (v_1, v_2, v_3) have been assigned to their paths before v_4, and each road has the same weight (i.e., $\omega_i = 1$). The footprints of (v_1, v_2, v_3) in the next time window are (fg, gh, hi, ij), (ab, bg, gh, hi, ij), and (ch, hk), respectively. For v_4, which travels from ab to ij, there are three alternative paths with similar travel times: p_1(ab, bg, gh, hi, ij), p_2(ab, bc, ch, hi, ij), and p_3(ab, bc, cd, di, ij). The union of their segments is the set (ab, bg, gh, hi, ij, bc, ch, cd, di), and their weighted footprint counters are (1, 1, 2, 2, 2, 0, 1, 0, 0). Consequently, N = 11, $E(p_1) = 2.29$, $E(p_2) = 1.67$, and $E(p_3) = 0.53$. Hence, v_4 will be assigned to p_3 because it is the least popular.

19.4.2.3 Flow-Balanced k-Shortest Paths

RkSP and EBkSP distribute the traffic load of the rerouted vehicles by randomly choosing between alternative paths or by balancing the system entropy among multiple paths. Since the key idea is load balancing, an alternative approach is designed to directly balance the traffic load, that is, the weighted footprint counters, through local search optimization [48]. The goal of the "local search" is to find the path assignment in which the sum of the weighted footprint counters is minimal, that is, to minimize

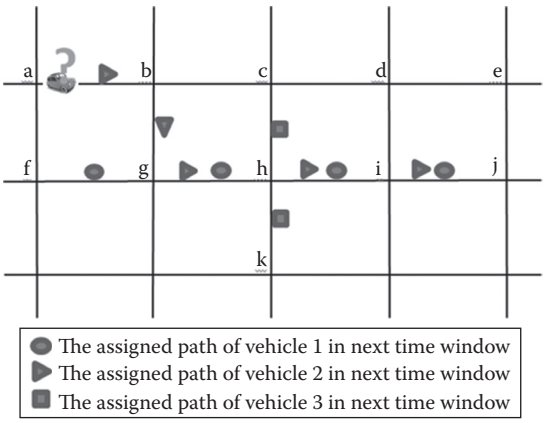


FIGURE 19.4 An EBkSP rerouting example. All segments have the same weight. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

in a network region, where S is the set of all region segments. As described in Definition 19.2, weighted footprint counter fc_i indicates the impact of the traffic flow on road segment r_i (i.e., the possibility of generating future congestion on r_i). Therefore, the summation of the weighted footprint counters of all the road segments measures the risk of congestion of the whole network. In other words, as a weighted footprint counter indicates the future flow magnitude, minimizing the sum of the weighted footprint counters means having balanced flows on all paths, and thus, reducing the risk of producing congestion.

Figure 19.5 illustrates how the path assignment affects the total number of weighted footprint counters. Assume that initially the vehicles (v_1, v_2, v_3) are assigned to the paths (ab, bc, cd, di, ij), (fg, gh, hi, ij), and (ab, bc, ch), respectively, and that the road segments have different weights (cf. Figure 19.5). Then, the sum of the weighted footprint counters in this network region is 18 (cf. Figure 19.5a). However, if v_1 switches to the path (ab, bg, gh, hi, ij), the sum of the weighted footprint counters is reduced to 16 as shown in Figure 19.5b. Therefore, the system will select the latter assignment.

To implement the optimization of the total number of footprints in a road network region, a random search strategy (cf. Algorithm 19.2) is implemented. The system generates first a good path assignment solution for all selected vehicles by assigning to each vehicle the path with the current least number of footprints (lines 2–8 in Algorithm 19.2). This initial assignment does not necessarily guarantee the minimum sum of footprint counters of the considered network region, that is, the union of all segments of the k-shortest paths of the rerouted vehicle. Therefore, the system randomly modifies the initial assignment in order to improve it (lines 14–16). If the new assignment reduces the total number of weighted footprint counters in the network region, the new assignment is accepted (lines 18–19). Otherwise, the assignment is rejected. This process runs iteratively until the limit number of iterations is attained (line 26).

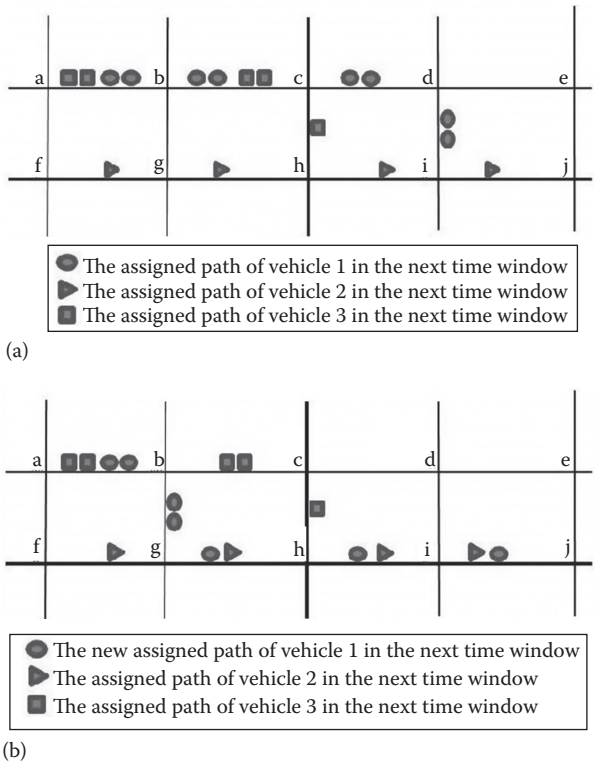


FIGURE 19.5 An FBkSP example. $\omega_{fg} = \omega_{gh} = \omega_{hi} = \omega_{ij} = \omega_{ch} = 1$, $\omega_{ab} = \omega_{bc} = \omega_{cd} = \omega_{de} = \omega_{af} = \omega_{bg} = \omega_{di} = \omega_{ej} = 2$: (a) the old assignment and (b) the new assignment. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.,* 62(8), 3551, 2014.)

Algorithm 19.2: Flow-balanced k-shortest path (FBkSP) rerouting.

```
1:  procedure LpcaLQptAssign (allkPaths, sorted Vehicles)
    {generate initial solution}
2:  for all vehicle in sorted Vehicles do
3:      {origin. dest}=getVehicleOD(vehicle)
4:      nowpath = pickPath_leastfootprints(allkPaths, origin, dest)
5:      reduction =getReduction ()
6:      vehicle .selectedpath=newpath
7:      updateFootprintf vehicle)
8:  end for
    {locally optimize the initial solution}
9:  iter=0
10: repeat
11:     for all vehicle in sortedVehicles do
12:         {origin, dest}=getVehicleOD(vehicle)
13:         newpath= pick path_random(allkpath,orgiiL, dest)
14:         newreduction=getReduction(newpath,vehicle.selectedpath)
15:         if iiewReduction<reduction then
16:             vehicle.select,edpatb=newpath
17:             iipdateFootprintf vehicle)
18:             reduction=newReduct ion
19:         end if
20:     end for
21:     iter=iter+1
22: until iter > MaxIteration {MaxIteration is a constant, set as 10 here,}
23: end procedure
```

19.4.3 Rerouting Process

In this section, the global rerouting process is presented, which was the basis for the traffic guidance system described in this thesis. The process is presented in Algorithm 19.3. The system periodically looks for signs of congestion in the road network (line 4). If signs of congestion are detected, then the system selects the vehicles situated near to the congested road segments and ranks them based on the urgency function. Finally, alternative routes are computed for the selected vehicles by using one of the five proposed rerouting strategies. It is worth noticing that except AR*, all the other rerouting strategies optimize the alternative path search by grouping the vehicles on their OD (line 10). This can lead to a significant reduction of the computational cost as showed in Section 19.5.1.

Algorithm 19.3: The main process.

```
1:  procedure main
2:  while true do
3:      updateEdge Weights()
4:      congested Roads=detect Congestion (edge Weights)
5:      if # congested Roads >0 then
6:          for all road in congestion Roads do
7:              selected Vehicles =selected Vehicles U selectVehicles(road)
```

```
8:          end for
9:          sorted Vehicles=sortByUrgency(selected Vehicles)
10:         al1paths= Empty
11:         if not AR* then
12:             odPairs=update OD Pairs (selectedVehicles)
13:             if DSP then
14:                 allPaths=Dijkstra(odPairs)
15:             else
16:                 al1 Paths=compuate_al1_kShortestPaths (o d Pairs)
17:     end if
18:             do Reroute (all Paths, sortedVehides)
19:             else
20:             for all vehicle in sorted Vehicles do
21:                 {origin, dest}=getVehieleOD(vehicle)
22:                 newPath=Astar Repulsion (origin,dest)
23:                 if newPath is not empty then
24:                     set Route (vehic1e, newPath)
25:                 end if
26:             end for
27:         end if
28:     end if
29:     wait (period) {The process executes periodically.}
30: end while
31: end procedure
32: procedure do Reroute ( all Paths, sorted Vehicles)
33: if FBkSP then
34:     LocalOpt Assign (all Paths, sorted Vehicles}
35: else
36:     for all vehicle in sorted Vehicles do
37:         {origin, dest}=getVehicleOD (vehicle)
38:         if DSP then
39:             new Path = allPaths[origin][dest][0]
40:         end if
41:         if RkSP then
42:             new Path = pickPath_random( allPaths [origin] [dest])
43:         end if
44:         if EBkSP then
45:             newPath = pickPath_least Popular (allPaths[origin] [dest])
46:             updateFootprint(vehicle, newPath)
47:         end if
48:         setRoute (vehic1e, newPath)
49:         end for
50: end if
51: end procedure
```

19.4.4 Dynamic Traffic Assignment

The work on DTA algorithms is essential for the problem considered here, that is, improving the individual travel time through traffic rerouting and guidance. Nevertheless, as explained in Section 19.2, DTA is not yet the most viable solution for real-time traffic guidance, mainly because of the DTA's very high

computational complexity coupled with the high dynamics of the traffic and the imperfections in traffic knowledge. In spite of this, DTA can offer valuable information as, for example, the level of improvement in the travel time that can be achieved in an ideal situation (i.e., where computational cost is not an issue and the traffic information is perfect). Therefore, DTA is employed to obtain a lower bound on the optimization of the travel time for comparison with the results produced by the proposed strategies.

The DTA model used here tries to achieve stochastic user equilibrium (SUE) through an iterative simulation process and mathematical modeling (see Section 19.2). Given the traffic demand, it chooses some initial routes assuming zero traffic. Then, it calculates the network load and the travel times by simulation and updates the route choices of the drivers. This process is repeated until the travel times are stationary or a maximum number of iterations are reached. The simulation-based DTA tool employed here was proposed in [36,49]. At least three parameters have to be given as input: a road network, a set of trips, and the maximum number of iterations. The higher the number of iterations is, the higher is the probability to achieve an SUE traffic state. In these experiments, the maximum number of iterations is defined to 50, since that was the value specified in [50]. The DTA algorithm, as defined in [36], is summarized next:

Step 1: Initialize the route of each driver by the optimal route in the empty network.
Step 2: Calculate the time-dependent costs of the road segments by simulation.
Step 3: Recalculate the optimal routes of a certain portion p of the drivers using the time-dependent costs from step 2.
Step 4: If routes have changed in step 3, go to step 2.

Note that the DTA algorithm involves not only the shortest path graph computations but also simulations. The purpose of the simulation is to help DTA acquire a relative accurate estimation of the travel times given the assignment of the previous iteration. Then, the estimated travel times are used to adjust the assignment in the next iteration.

However, this inevitably leads to increased computational burden. In comparison, the approach used here prompts alternative routes to drivers during their entire journey based on the dynamic conditions in the road network, and most of the computation is spent on the shortest path graph algorithms. Therefore, this approach is expected to be more efficient than DTA.

19.5 Evaluation of the Rerouting System

The main objective of this simulation-based evaluation is to study the performance of the five rerouting strategies under various scenarios, specifically, to address the following questions:

- Which strategy leads to the most benefits for drivers in terms of travel time and the number of reroutings?
- What is the trade-off between strategy effectiveness and their efficiency in terms of computation time? How do the proposed strategies compare to a DTA-based approach in terms of effectiveness and efficiency?
- Which strategies scale better with the number of cars?
- How robust is the system under various compliance rates (i.e., percentage of drivers who follow the guidance) and penetration rates (i.e., percentage of vehicles that have this software)?

The experimental settings are firstly introduced in Section 19.5.1. Afterwards, the results are presented and analyzed in Section 19.5.2.

19.5.1 Experimental Settings

Both simulation of urban mobility (SUMO) [51] and TraCI [52] were employed for these simulations. SUMO is an open source, highly portable, microscopic road traffic simulation package designed to

handle large road networks. TraCI is a library providing extensive commands to control the behavior of the simulation including vehicle state, road configuration, and traffic lights. The rerouting strategies algorithms were implemented using TraCI. Essentially, when SUMO is called with the option to use TraCI, SUMO starts up, loads the scenario, and then waits for a command. Thus, variables in the simulation can be changed (e.g., new paths assigned to certain vehicles). Then, a new command can be sent with how many seconds to run the simulation before stopping and waiting for another command.

Two urban road maps were downloaded from OpenStreetMap [53] in OSM format. One is a section of Brooklyn, NY, and the other is in Newark, NJ. The Netconvert tool in SUMO was used to convert the maps into a SUMO usable format and the Trafficmodeler tool [54] to generate vehicle trips. Netconvert removes the pedestrian, railroad, and bus routes and sets up a static traffic light at each intersection to make the simulations more realistic (as the maps do not have STOP signs). All roads have the same speed limit (13.9 m/s); some roads have one lane in each direction, while others have just one lane based on the specification in the OpenStreetMap OSM file. The statistics of the two networks are shown in Table 19.1. By default, the shortest travel time paths are automatically calculated and assigned to each vehicle at the beginning of simulation based on the speed limit. Figure 19.6 illustrates the simulation process. Figure 19.7a and b shows the traffic flow in both networks. Trafficmodeler was used to generate a total of 1000 cars in the Brooklyn network from the left area to the right area in an interval of 1000 s. The origins and the destinations are randomly picked from the left area and the right area, respectively. In the Newark network, 908 cars were generated having the origins picked randomly from the peripheral road segments and the destinations on the road segments inside the hot spot circle.

In the simulations, the default settings in SUMO 15.0 were used for vehicle length = 5 m, the minimal gap = 2.5 m, the car following model [55], and the driver's imperfection = 0.5. For each scenario, the results are averaged over 20 runs. Initially, an ideal scenario is assumed, in which all drivers have the system and accept the route guidance. These assumptions are relaxed in the last part of the evaluation. Table 19.2 defines the parameters used in this evaluation.

A DTA-based rerouting strategy (cf. Section 19.4.4) was also implemented by using a DTA tool provided with the SUMO generator. Contrary to the proposed approach, the DTA strategy computes the routes leading to user equilibrium for all the vehicles in one shot, before any vehicle starts its journey. Thus, the DTA tool produces a file containing the routes of all the simulated vehicles, which is supplied to the SUMO simulator. Based on this route file, SUMO generates a single, continuous simulation,

TABLE 19.1 Statistics of the Two Road Networks

	Brooklyn	Newark
Network area (km^2)	75.85	24.82
Total number of road segments	551	578
Total length of road segments (km)	155.55	111.41
Total number of intersections	192	195

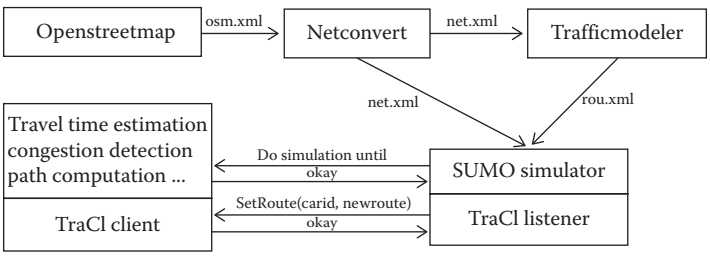


FIGURE 19.6 The simulation process. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

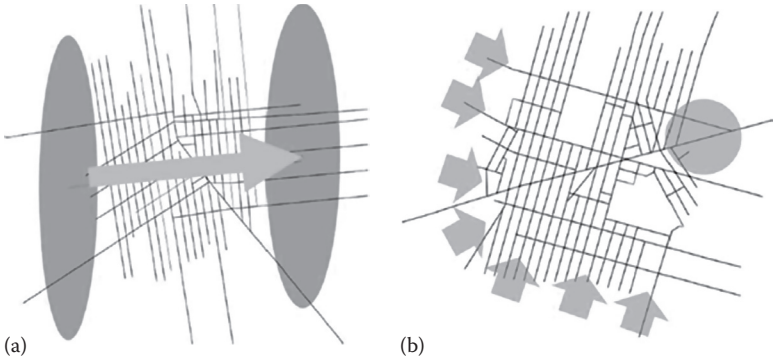


FIGURE 19.7 Traffic flow in the road networks: (a) Brooklyn and (b) Newark. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

TABLE 19.2 Parameters in Centralized Rerouting Algorithms

Period	The frequency of triggering the rerouting, by default period = 450 s.
Threshold δ	Congestion threshold; if $K_i/K_{jam} > \delta$, the road segment is considered congested; by default $\delta = 0.7$.
Urgency	Urgency policy: *RCI* or *ACI*.
Level L	Network depth to select vehicles for rerouting starting from the congested segment and using BFS on the inverted network graph.
# Paths k	The max number of alternative paths for each vehicle, by default k = 4.
Repulsion weight β	The weight of repulsion in AR*, by default $\beta = 0.05$.

that is, without any other route changes as in the case of the proposed strategies. Hence, the CPU time measured in the next section indicates, in the case of DTA, the time required to produce the route file, whereas in the case of the proposed strategies, the cumulated time (i.e., over the whole simulation) required to compute alternative paths for the rerouted vehicles.

19.5.2 Results and Analysis: Average Travel Time

Figure 19.8 presents the average travel time obtained with the five strategies and with DTA on both networks. The "no-reroute" bars indicate the travel time in the absence of any rerouting. The results show that all the proposed strategies improve the travel time significantly. In most cases, the proposed strategies obtain travel times at least two times lower than no rerouting. For instance, with a selection level of 3, compared to "no reroute," EBkSP reduces the travel time by 2.2 times and 4.5 times on Brooklyn and Newark, respectively. As expected, DTA has the best average travel time since it can achieve user equilibrium. Based solely on the obtained average travel time, the five strategies are ranked as follows: DTA > AR* > (EBkSP, FBkSP) > RkSP > DSP > no rerouting. The results confirm the hypotheses laid out in Section 19.4 with statistical significance of 95% confidence interval. DSP can improve the travel time, since it reroutes dynamically the vehicles by considering the traffic conditions. However, in some cases, for example, if many vehicles have similar current positions and destinations, respectively, new congestions can be created by the rerouting process. RkSP avoids this shortcoming since it balances the traffic flow over several paths. Nevertheless, a randomly picked path is not necessarily the best one. EBkSP and FBkSP offer even better performance by carefully selecting the path for each rerouted vehicle. Finally, AR* has the best performance among the proposed strategies as it considers all the other vehicles in the road network in the computation of a new route.

The experiments also demonstrated that setting the depth level to 3 or 4 is best for selecting a relatively optimal number of vehicles for rerouting (the two values lead to similar performance for Brooklyn,

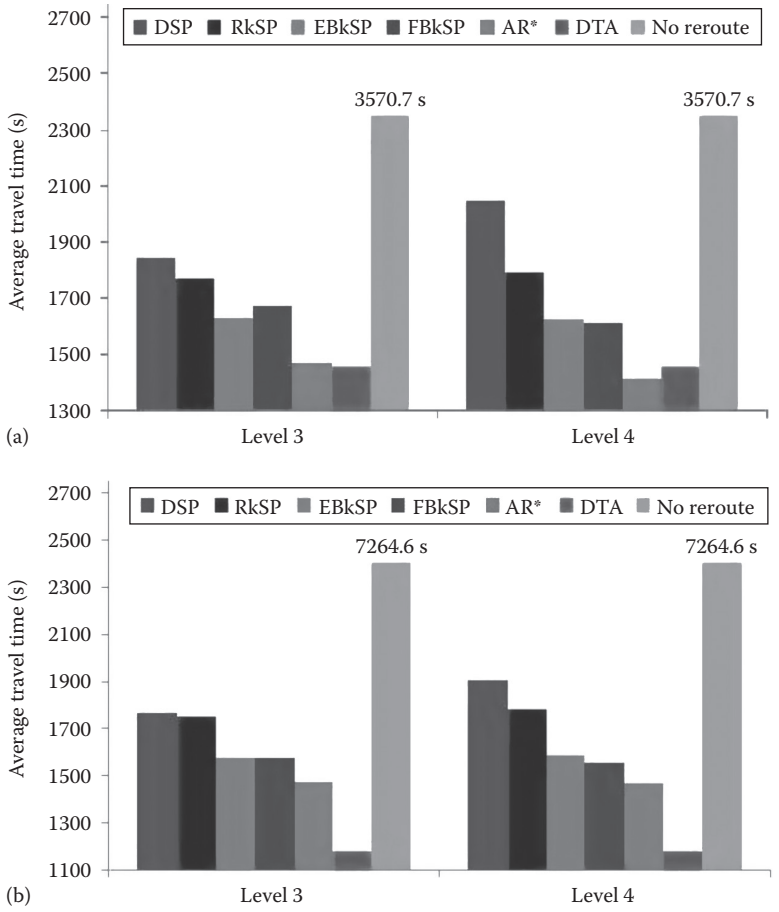


FIGURE 19.8 Average travel time (L = (3, 4), k = 4, urgency = ACI, period = 450 s, δ = 0.7, and β = 0.05): (a) Brooklyn and (b) Newark. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

while level 3 is better for Newark). Lower-level values do not select enough cars, whereas higher values increase the number of reroutings (see Figure 19.9). Therefore, the level parameter is set to 3 in the remaining experiments.

19.5.2.1 Average Number of Reroutings

It is important that the rerouting frequency for a given vehicle during a trip stays low. From the driver point of view, changing the path to the destination too often can be distracting and annoying. From the system point of view, having a low number of reroutings means decreasing the computational burden because the rerouting process is costly. Figure 19.9 compares the number of reroutings across the five proposed strategies. The statistical analysis shows that AR* < (EBkSP, FBkSP) < RkSP < DSP in terms of average rerouting number with 95% confidence interval. For example, compared to DSP, AR* reduces the average number of reroutings by up to 2.0 and 1.5 times, while compared to RkSP, AR* is better by 1.6 and 1.3 times on Brooklyn and Newark, respectively. The reason is that by considering future path information in the rerouting decision, EBkSP, FBkSP, and AR* cannot only mitigate the current congestion but also avoid creating new congestions, hence, the lower necessity for recurrent rerouting.

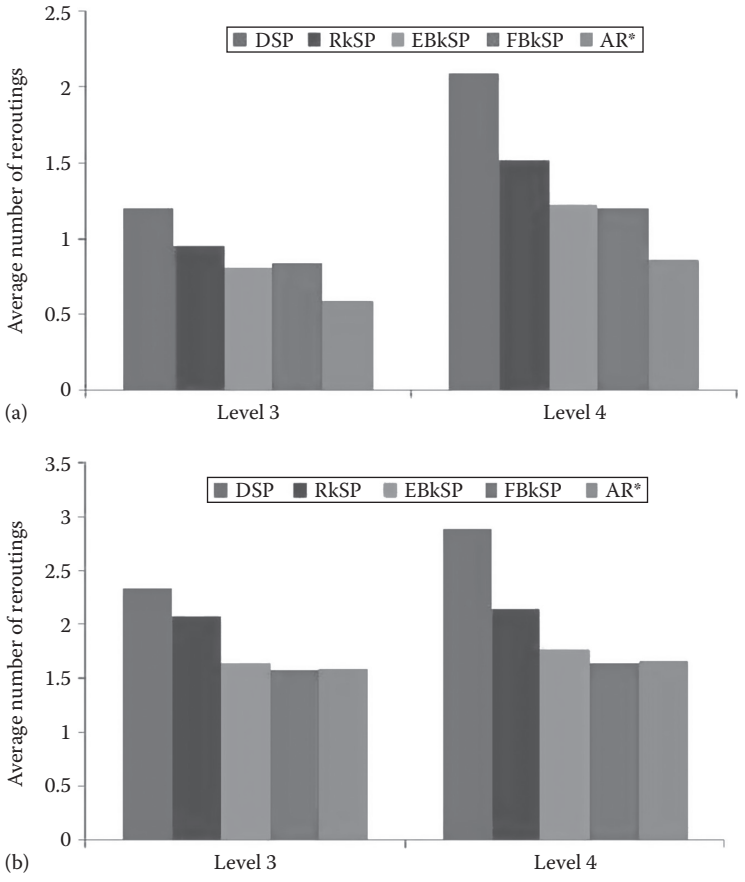


FIGURE 19.9 Average number of reroutings (L = (3, 4), k = 4, urgency = ACI, period = 450 s, δ = 0.7, and β = 0.05): (a) Brooklyn and (b) Newark. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.,* 62(8), 3551, 2014.)

19.5.2.2 CPU Time

At this point, the results indicate that AR* produces the best average travel times (near to the DTA times), followed closely by EBkSP, FBkSP, and, in some cases, RkSP. An important question is, what is the computational performance among all the proposed five strategies? If the computational complexity of the algorithms that the strategies are based on is considered, the complexities of the Dijkstra shortest path (used by DSP), k loopless shortest paths (used by RkSP, EBkSP, and FBkSP), and A* (used by AR*) algorithms must be evaluated. The Dijkstra shortest path and k loopless shortest paths require O(E + Vlog(V)) and O(kV(E + Vlog(V))), respectively, while A* was proven to be faster than Dijkstra [56]. However, this complexity analysis is pertinent only when the selection of an alternative path for one single vehicle is considered. From the system point of view, the global computational complexity also depends on the number of reroutings processed in a time window; this number is a function of the number of congested road segments and the congestion severity (i.e., how many vehicles are selected for rerouting). Moreover, DSP, RkSP, FBkSP, and EBkSP compute the shortest paths after grouping the vehicles on their OD, whereas AR* calculates a new path for each vehicle. Therefore, AR* could require a larger computation time than the other methods.

Figure 19.10a shows the global CPU time consumed for rerouting by the five methods and by DTA. Note that the experiments were conducted on a 64-bit Ubuntu machine with Intel Core i5-2467M CPU (1.6 GHz) and 4 GB of memory. The following four observations were noted regarding the CPU time results:

- DSP requires the least CPU time for rerouting, mainly due to the low complexity of the shortest path algorithm (compared to the k-shortest paths algorithm) and to grouping the rerouted vehicles.
- AR* consumes significantly more CPU time. Specifically, it requires 2.0 and 2.3 times more CPU time than RkSP and EBkSP on Brooklyn. The main reason is that AR* cannot group the rerouted vehicles like the other methods as stated in Section 19.4.1.
- EBkSP, FBkSP, and RkSP are situated in between the aforementioned methods from the CPU time point of view. Interestingly, EBkSP and FBkSP require less computation time than RkSP even though they execute more complex path selection algorithms in addition to the k-shortest path computation. The explanation is that EBkSP and FBkSP decrease the total number of reroutings processed in a period. This decrease becomes apparent when the number of OD pairs involved in the computation was examined, as indicated in Figure 19.10b. The total number of OD pairs is

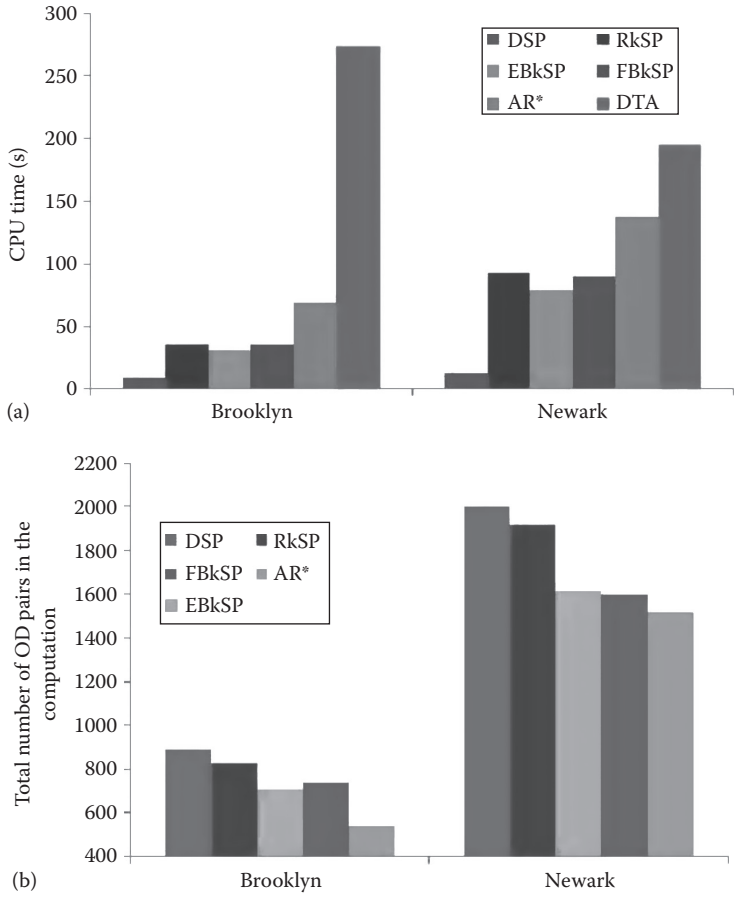


FIGURE 19.10 CPU time for both the networks (L = 3, k = 4, urgency = ACI, period = 450 s, δ = 0.7, β = 0.05): (a) CPU time and (b) the number of OD pairs. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

lower for EBkSP and FBkSP than for RkSP. Figure 19.10b also shows that although DSP leads to the largest number of OD pairs, it still has the lowest CPU time because of the much lower computational complexity of the Dijkstra algorithm compared to the k-shortest path algorithm.

- DTA has the largest CPU time and scales poorly with an increasing number of vehicles (in terms of CPU time) when compared to AR* or the other proposed methods (as shown in Figure 19.11b). Also, it is worth noticing that DTA assumes all vehicles in the system known at the beginning (i.e., when it computes its routes). However, in real life, vehicles may appear at any time, and DTA would be required to perform its expensive computation over and over again. Therefore, due to its very high computational cost in real life, DTA may be impractical (i.e., it may not be able to compute alternative routes fast enough in order to mitigate congestions).

In conclusion, if both the average travel time and the CPU time are considered, EBkSP and FBkSP appear to be the best strategies since they offer the best trade-off between rerouting effectiveness and computational efficiency. If computational cost is not an issue, one can use the AR* strategy, while in the opposite case, DSP is the most appropriate choice.

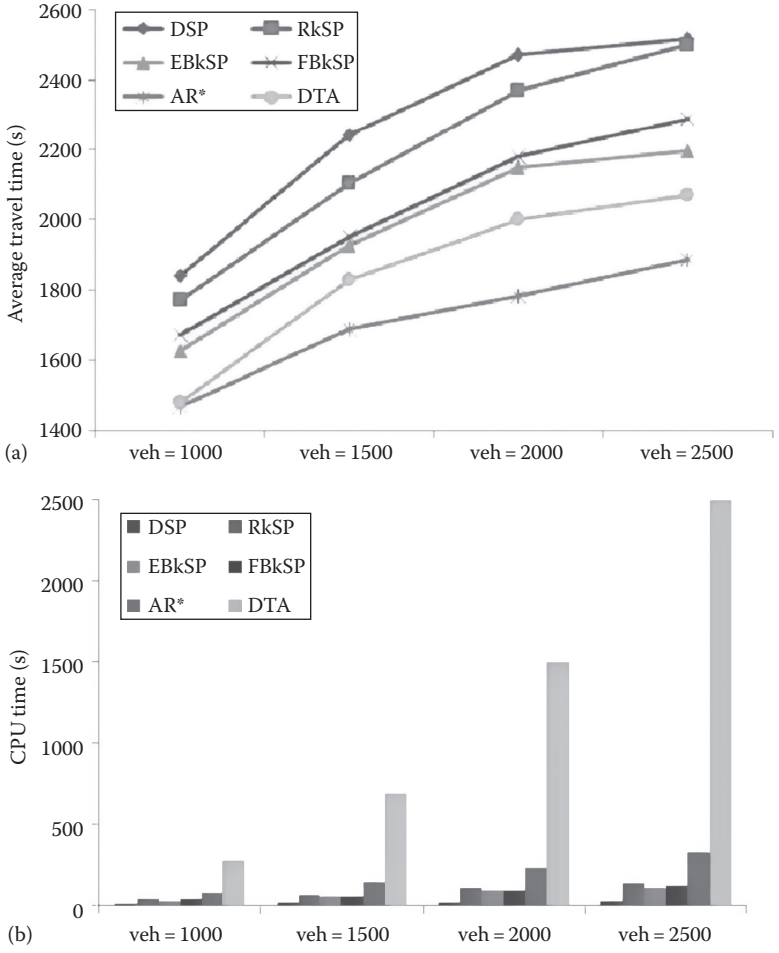


FIGURE 19.11 (a) Average travel time for different traffic density. (b) CPU time for different traffic density. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

19.5.2.3 Traffic Density

The results presented up to here already offer a good idea about the capabilities of the proposed rerouting strategies to alleviate traffic congestions. Yet there is an important aspect that still needs to be explored, that is, how the proposed methods scale with the increase in the traffic volume. To respond to this question, another set of experiments were conducted on the Brooklyn network, where the number of vehicles was increased from 1000 to 2500. Figure 19.11 shows the obtained results both for the average travel time and the CPU time for different traffic densities. AR* and DTA present the best scalability from the average travel time point of view. However, these methods are also the least scalable from the CPU time point of view. It was apparent that DTA exhibits particularly poor scalability compared to the proposed strategies, confirming the hypothesis that DTA is not yet a suitable approach for real-time traffic management. Also, somewhat interestingly, AR* obtained better average travel times than DTA (see Figure 19.11a) when the number of vehicles was above 1500. This is certainly due to the fact that the 50 iterations limit, set in the DTA tool, is not sufficient to achieve user equilibrium for higher traffic densities. Therefore, a higher number of iterations are needed in this case, which will lead evidently to even higher CPU times.

19.5.2.4 Compliance Rate

It is unrealistic to assume that every driver follows the rerouting guidance. The drivers' compliance rate (i.e., the possibility for the driver to accept the guidance) is an important factor for the rerouting strategy design. Therefore, the average travel time was measured, while varying the compliance rate for the five proposed strategies and for DTA. Specifically, for the strategies, given a compliance rate of x%, at each rerouting period, each of the selected vehicles changes their routes with x% possibility. As for DTA, x% of the vehicles are randomly selected to follow the DTA-assigned route, while the rest of the vehicles follow the shortest time route.

Figure 19.12a indicates that the average travel time can be significantly improved by all five strategies even under low compliance rates. This is due to the fact that even under low compliance rates, the drivers who comply with the guidance can still receive more rapid routes, which in turn can improve the travel time for the rest of the drivers. Figure 19.12b shows the average travel time for a wide range of compliance rates.

In particular, when the compliance rate is low, RkSP, EBkSP, FBkSP, and AR* show significantly better travel time than DTA. The reason is that when compliance is low, the drivers who comply benefit much more from this guidance than from the DTA guidance. In the DTA approach, the route computation is done once before any vehicle enters the network. If the compliance rate is low, the DTA-computed routes are far from user equilibrium, inclusively for the compliant drivers. Differently, these strategies can adjust the vehicles' routes periodically based on the current traffic information. Therefore, although the noncompliant drivers create congestion in the network, the compliant ones can still receive fairly good paths, which implicitly reduces the congestion level in the network.

19.5.2.5 Penetration Rate

To understand how easy is to deploy this solution in real life, the effect of the penetration rate on the average travel time was studied. Specifically, before the system starts, each vehicle was predefined with x% possibility of owning the system (e.g., providing position information and receiving guidance). Once the system starts, only those the x% vehicle will have the chance to be rerouted. The penetration rate is a parameter of major importance for two reasons. First, since only the vehicles that have the system provide position and route information to the server, the accuracy of congestion detection and road travel time estimation depend directly on this parameter (i.e., the lower the penetration rate is, the lower the accuracy is and vice versa). Second, similar to the compliance rate, the effectiveness of the load balancing mechanism implemented by the rerouting strategies increases with the percentage of the vehicles that use the system.

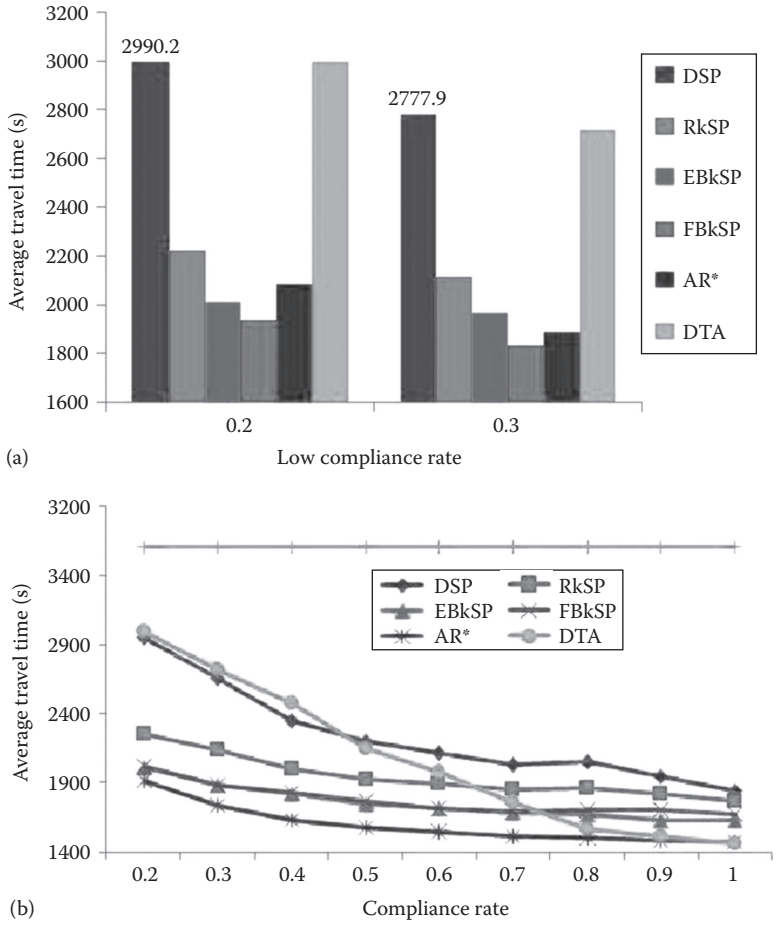


FIGURE 19.12 (a) Average travel time for low compliance rate. (b) Average travel time for all compliance rate. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.*, 62(8), 3551, 2014.)

Figure 19.13a shows the average travel time for various penetration rates, when traffic data are collected only from vehicles (i.e., no support from roadside sensors) [57]. When the penetration rate is low, the performance of the proposed methods is the same as "no reroute." In this case, the service does not have enough data to accurately detect signs of congestion. Once the penetration rate is greater than 0.4, the system is able to improve the travel time. For penetration rates above 0.6, EBkSP, FBkSP, and AR* start to perform better than DSP and RkSP, since a larger number of vehicles are rerouted, which requires a more advanced load-balancing mechanism. Compared to these methods, DTA performs better under low penetration rates since the DTA rerouting is not triggered by the congestion detection as in this approach.

To boost the adoption of the system, it is possible that data from roadside sensors (in conjunction with data from vehicles) can be leveraged to detect congestion more accurately in the case of low penetration rates. When road sensors are present, the road traffic density can be measured, the congestion can be detected, and the travel time can be estimated more accurately. Figure 19.13b demonstrates that these methods can significantly improve the travel time even under low penetration rates if roadside sensor information is available. Moreover, EBkSP, FBkSP, and AR* perform better than DTA in this case. When penetration rate is low (x%), DTA distributes evenly the x% vehicles without considering the rest, which can still create congestion. Therefore, the alternative routes proposed by DTA are not as effective

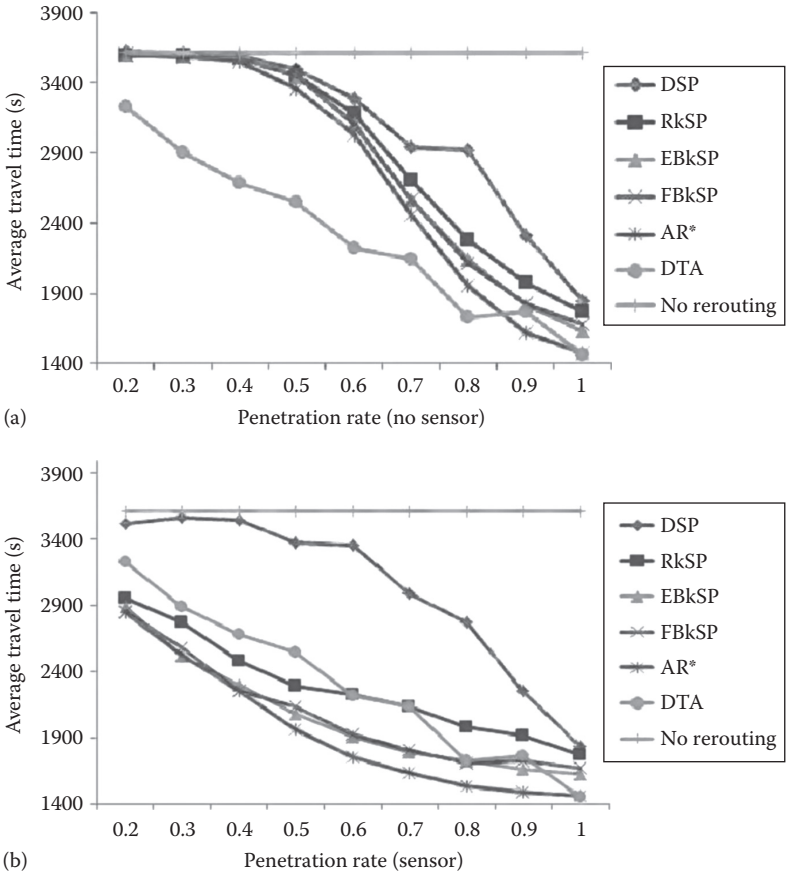


FIGURE 19.13 (a) Average travel time for all penetration rates with no sensor. (b) Average travel time for all penetration rate with sensors. (From Pan, J. and Borcea, C., *IEEE Trans. Veh. Tech.,* 62(8), 3551, 2014.)

in alleviating congestion. By comparison, these strategies can take advantage of the sensor information to divert the x% of the drivers and to reduce congestion.

19.6 Summary

The ubiquity of mobile devices such as smartphones or onboard vehicle units is leading to real-life vehicular sensor networks. This chapter presented a novel approach to tackle the ever more stringent problem of traffic congestion. This approach is based on a traffic guidance system that monitors traffic and proactively pushes individually tailored rerouting guidance to vehicles when there are signs of congestion. The system is responsible for several functions such as traffic data representation, congestion prediction, and selection of the vehicles to be rerouted. We chose to focus in this chapter on a key element of our rerouting system, that is, the rerouting strategies. We proposed five rerouting strategies to compute alternative routes for vehicles. Then, we conducted an extensive set of simulation-based experiments to validate our approach. The results showed that the proposed rerouting algorithms are very effective in mitigating congestion and adapt well to the dynamic nature of the traffic, being also more efficient and scalable than existing approaches. In addition, our traffic guidance system remains useful even with low compliance rate and moderate penetration rate.

As future work, we intend to investigate a hybrid architecture that off-loads parts of the computation and decision process in the network and uses vehicle-to-vehicle communication to better balance the need for privacy, scalability, and low overhead with the main goal of low average travel time.

References

1. Pan, J., Popa, I. S., Zeitouni, K., & Borcea, C. (2013). Proactive vehicular traffic rerouting for lower travel time. Vehicular Technology, IEEE Transactions on, 62(8), 3551-3568. Accessed December 14, 2013.
2. Car-to-car communication. 2012. Available from: http://www.bmw.com/com/en/insights/technology/technology_guide/articles/cartocar_communication.html. Accessed December 14, 2013.
3. Santo, M., Toyota, Audi promise driverless car demos at CES 2013. 2013. Available from: http://www.examiner.com/article/toyota-audi-promise-driverless-car-demos-at-ces-2013. Accessed December 14, 2013.
4. Vijayenthiran, V., GM participating in U.S. car to car communications trial. 2012. Available from: http://www.motorauthority.com/news. Accessed December 14, 2013.
5. Vijayenthiran, V., Ford powers ahead with development of car-to-car communication technology. 2012. Available from: http://www.motorauthority.com/news. Accessed December 14, 2013.
6. Jiang, D. and L. Delgrossi, IEEE 802.11p: Towards an international standard for wireless access in vehicular environments, in *IEEE Vehicular Technology Conference*, Singapore, 2008, pp. 2036–2040.
7. Biswas, S., R. Tatchikou, and F. Dion, Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety. *IEEE Communications Magazine*, 2006. 44(1): 74–82.
8. Al-Sultan, S., Al-Bayatti, A. H., and Zedan, H. Context-aware driver behavior detection system in intelligent transportation systems. *Vehicular Technology, IEEE Transactions on*, 2013, 62(9), 4264–4275.
9. Zhou, P. et al., EZCab: A cab booking application using short-range wireless communication, in *IEEE Third International Conference on Pervasive Computing and Communications, 2005* (*PerCom'05*), Kauai Island, HI, 2005. IEEE, pp. 27–38.
10. Lu, R. et al., SPARK: A new VANET-based smart parking scheme for large parking lots, in *IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, 2009. IEEE, pp. 1413–1421.
11. Riva, O. et al., Context-aware migratory services in ad hoc networks. *IEEE Transactions on Mobile Computing*, 2007. 6(12): 1313–1328.
12. Hunter, T. et al., Path and travel time inference from GPS probe vehicle data, in *NIPS Workshop on Analyzing Networks and Learning with Graphs*, Whistler, British Columbia, Canada, 2009.
13. Work, D.B. et al., An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices, in *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Mexico, 2008, pp. 5062–5068.
14. Eriksson, J. et al., The pothole patrol: Using a mobile sensor network for road surface monitoring, in *Proceedings of the Sixth International Conference on Mobile Systems, Applications, and Services* (*MobiSys'08*), Breckenridge, CO, 2008, pp. 29–39. ACM.
15. Horvitz, E. et al., Prediction, expectation, and surprise: Methods, designs, and study of a deployed traffic forecasting service, in *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence*, Edinburgh, Scotland, 2005, pp. 244–257.
16. Mohan, P., V.N. Padmanabhan, and R. Ramjee, Nericell: Rich monitoring of road and traffic conditions using mobile smartphones, in *Proceedings of the Sixth ACM Conference on Embedded Network Sensor Systems*, Raleigh, North Carolina, USA, 2008, pp. 323–336.
17. Yoon, J., B. Noble, and M. Liu, Surface street traffic estimation, in *Proceedings of the Fifth International Conference on Mobile Systems, Applications and Services*, San Juan, Puerto Rico, USA, 2007, pp. 220–232.
18. INRIX. Available from: http://www.inrix.com. [Online; accessed on December 14, 2013].
19. Waze. Available from: http://www.waze.com. [Online; accessed on December 14, 2013].
20. Nadeem, T. et al., TrafficView: Traffic data dissemination using car-to-car communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 2004. 8(3): 6–19.
21. Dornbush, S. and A. Joshi, StreetSmart traffic: Discovering and disseminating automobile congestion using VANETs, in *IEEE 65th Vehicular Technology Conference*, Montreal, Quebec, Canada, 2007, pp. 11–15.

22. Rybicki, J. et al., Challenge: Peers on wheels-a road to new traffic information systems, in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, Dublin, Ireland, 2007, pp. 215–221.
23. Rybicki, J. et al., PeerTIS: A peer-to-peer traffic information system, in *Proceedings of the Sixth ACM International Workshop on VehiculAr InterNETworking*, Beijing, China, 2009, pp. 23–32.
24. Senge, S. and H. Wedde, Bee inspired online vehicle routing in large traffic systems, in *Proceedings of the Second International Conference on Adaptive and Self-Adaptive Systems and Applications* (*ADAPTIVE'10*), Lisbon, Portugal, 2010, pp. 78–83.
25. Tatomir, B. et al., Dynamic routing in traffic networks and MANETs using ant based algorithms, in *Proceedings of the Seventh International Conference on Artificial Evolution*, Lille, France, 2005.
26. Prothmann, H. et al., Decentralized route guidance in organic traffic control, in *Proceedings of the Fifth IEEE International Conference on Self-Adaptive and Self-Organizing Systems* (*SASO'11*), Ann Arbor, Michigan, USA, 2011, pp. 219–220.
27. Gradinescu, V. et al., Adaptive traffic lights using car-to-car communication, in *IEEE 65th Vehicular Technology Conference*, Dublin, Ireland, 2007, pp. 21–25.
28. Wardrop, J.G., Some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers, Part II*, 1952. 1(36): 252–378.
29. Friesz, T.L. et al., A variational inequality formulation of the dynamic network user equilibrium problem. *Operations Research*, 1993. 41: 179–191.
30. Merchant, D.K. and G.L. Nemhauser, Optimality conditions for a dynamic traffic assignment model. *Transportation Science*, 1978. 12(3): 200–207.
31. Merchant, D.K. and G.L. Nemhauser, A model and an algorithm for the dynamic traffic assignment problems. *Transportation Science*, 1978. 12(3): 183–199.
32. Peeta, S. and A.K. Ziliaskopoulos, Foundations of dynamic traffic assignment: The past, the present and the future. *Networks and Spatial Economics*, 2001. 1(3): 233–265.
33. Chiu, Y.C. et al., Dynamic traffic assignment: A primer. *Transportation Research E-Circular*, 2011. E-C153.
34. Taylor, N.B. *CONTRAM 5: An Enhanced Traffic Assignment Model*. Transport and Road Research Laboratory, Crowthorne, U.K., 1990.
35. Mahmassani, H.S., T.-Y. Hu, and R. Jayakrishnan, Dynamic traffic assignment and simulation for advanced network informatics (DYNASMART), in *Proceedings of the Second International CAPRI Seminar on Urban Traffic Networks*, Capri, Italy, 1992.
36. Gawron, C., *Simulation-Based Traffic Assignment-Computing User Equilibria in Large Street Networks*. University of Cologne, Köln, Germany, 1999.
37. Maerivoet, S. Modeling traffic on motorways: State-of-the-art, numerical data analysis, and dynamic traffic assignment, PhD thesis, Katholieke Universiteit Leuven, Department of Electrical Engineering ESAT-SCD (SISTA), 2006.
38. Banks, J.H., *Introduction to Transportation Engineering*. McGraw-Hill, New York, 2002.
39. Manual, H.C. *Highway Capacity Manual*. Transportation Research Board, Washington, DC, 2000.
40. Lawler, E.L., A procedure for computing the k best solutions to discrete optimization problems and its application to the shortest path problem. *Management Science*, 1972: 401–405.
41. Martins, E.Q.V. and M.M.B. Pascoal, A new implementation of Yen's ranking loopless paths algorithm. *A Quarterly Journal of Operations Research*, 2003. 1(2): 121–133.
42. Perko, A., Implementation of algorithms for K shortest loopless paths. *Networks*, 1986. 16(2): 149–160.
43. Fredman, M.L. and R.E. Tarjan, Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM*, 1987. 34(3): 596–615.
44. Hart, P.E., N.J. Nilsson, and B. Raphael, A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 1968. 4(2): 100–107.

45. Shannon, C.E., A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 2001. 5(1): 3–55.
46. Cranshaw, J. et al., Bridging the gap between physical location and online social networks, in *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, Beijing, China, 2010, pp. 119–128.
47. Xu, T. and Y. Cai, Feeling-based location privacy protection for location-based services, in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, Chicago, IL, USA, 2009, pp. 348–357.
48. Hoos, H.H. and T. Stützle, *Stochastic Local Search: Foundations & Applications*. Elsevier, Amsterdam, the Netherlands, 2004.
49. DuaIterate. Available from: http://sumo.sourceforge.net/doc/current/docs/userdoc/Tools/Assign.html [Online; accessed on December 14, 2013].
50. Behrisch, M., D. Krajzewicz, and Y.P. Wang, Comparing performance and quality of traffic assignment techniques for microscopic road traffic simulations, in *Proceedings of the International Symposium on Dynamic Traffic Assignment* (*DTA'08*), Leuven, Belgium, 2008.
51. Behrisch, M. et al., SUMO—Simulation of urban mobility: An overview, in *Proceedings of the Third International Conference on Advances in System Simulation* (*SIMUL'11*), Barcelona, Spain, 2011.
52. Wegener, A. et al., TraCI: An interface for coupling road traffic and network simulators, in *Proceedings of the 11th Communications and Networking Simulation Symposium*, Ottawa, ON, Canada, 2008, pp. 155–163.
53. Haklay, M. and P. Weber, OpenStreetMap: User-generated street maps. *IEEE Pervasive Computing*, 2008. 7(4): 12–18.
54. Papaleondiou, L.G. and M.D. Dikaiakos, Trafficmodeler: A graphical tool for programming microscopic traffic simulators through high-level abstractions, in *IEEE 69th Vehicular Technology Conference*, Barcelona, Spain, 2009, pp. 1–5.
55. Krauss, S., P. Wagner, and C. Gawron, Metastable states in a microscopic model of traffic flow. *Physical Review E*, 1997. 55(5): 5597.
56. Schultes, D., *Route Planning in Road Networks.* Universität Karlsruhe (TH), Fakultät für Informatik, Institut für Theoretische Informatik, Algorithmik II, Karlsruhe, Germany, 2008.
57. Pan J. and C. Borcea, *Proactive vehicular traffic rerouting for lower travel time. IEEE Transactions on Vehicular Technology*, 2014. 62(8): 3551–3568.

20

Acoustic Sensing System for Underwater Geolocation and Navigation

20.1 Introduction **20**-1
20.2 Acoustic Sensor Unit **20**-2
20.3 Double-Integration Method **20**-4
20.4 Normalization **20**-8
20.5 Reversed System **20**-10
20.6 Extension to Active Systems **20**-11
20.7 Summary **20**-12
References **20**-12

Hua Lee
University of California

20.1 Introduction

This chapter presents a simple high-performance approach to underwater geolocation and navigation. The simplicity of the algorithm provides excellent stability, accuracy, and computation efficiency. In addition, the chapter also describes the reversed version of the system for the enhancement of system performance and the conversion to active system for unmanned underwater vehicle (UUV) collision avoidance.

The original goal of this research program was to improve the accuracy and stability of the conventional polarity estimation algorithm for UUV homing and docking exercises underwater. The assumption is that the relative left–right position of the UUV to the base stations can be estimated from the polarity of the beacon waveforms. The traditional technique conducts the polarity estimation based on the phase term of the first peak of the received beacon signal. In the underwater environment, this approach has been problematic with low stability and accuracy because of the difficulty especially in locating the first peak from the interference signal pattern under substantial multipath and background noise. After a series of experiments for improving existing algorithms, the effort was then redirected toward the design and development of new techniques with simple structures and computation complexity. As a result, the *double-integration method* was developed, and the results showed superior stability and accuracy.

During the investigation, it was also noted that the reversed version of the existing system could be even more effective. This involves the reversal of the configuration of the transmitter and receiver. The alternative arrangement further simplifies the hardware and software. Besides, the new configuration enables us to continue the improvement of system performance with added receiver elements without increasing the computation complexity.

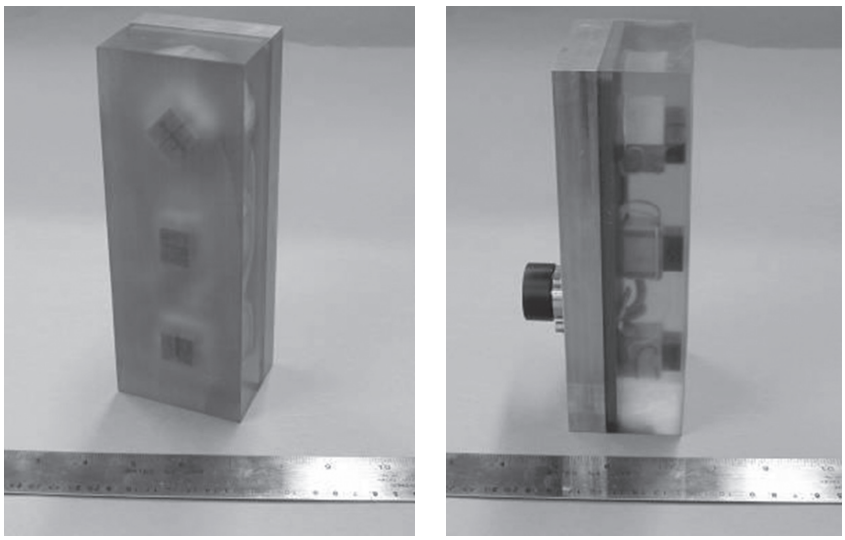
One important extension is the conversion of the existing system to the active modality by placing the transmitter at the center of the circular receiver array. Because the transmitted signal is available as the reference waveform, this enables us to integrate the estimation of the range distance into the algorithm. For the superior computation efficiency in dynamic sensing, this system will be an excellent candidate for added applications to collision avoidance.

20.2 Acoustic Sensor Unit

In mobile autonomous sensing, the geolocation capability for navigation and guidance can be achieved by estimating the position of the sensor unit with respect to the underwater base stations. The locations of the underwater base systems are constantly estimated and updated with respect to the interface stations over the ocean surface. And the interface stations are supported by the global positioning system (GPS) with the direct microwave link to surface communication infrastructure. Thus, one of the most critical elements is the mobile sensor's capability of dynamic estimation and updating of its relative position with respect to the underwater base stations. In 3D underwater geolocation tasks, the objective parameters include mainly the *range distance* and a multidimensional *bearing angle vector*.

Figure 20.1 shows the laboratory prototype of the three-component transmitter of the base stations. Each of the three components consists of four small square transmitting elements. And Figure 20.2 shows the single-element receiver at the mobile platform.

With the three transmission components, the transmitter unit sends out a sequence of three signals, $\{T_1(t), T_2(t), T_3(t)\}$. The first signal $T_1(t)$, from the top transducer element, is an in-phase reference signal, of which all four elements send out the same signal. The second signal $T_2(t)$, from the middle transducer, is a beacon signal, transmitting a pair of signals with a left–right polarity of 180° phase offset, for the estimation of the bearing angle in the horizontal direction. Similarly, the third signal $T_3(t)$, from the bottom transducer, is with a top–bottom polarity of a 180° phase offset for the estimation of the bearing angle in the vertical direction.



(a) (b)

FIGURE 20.1 (a) Front and (b) side view of the laboratory prototype of the three-component transmitter of the base stations.

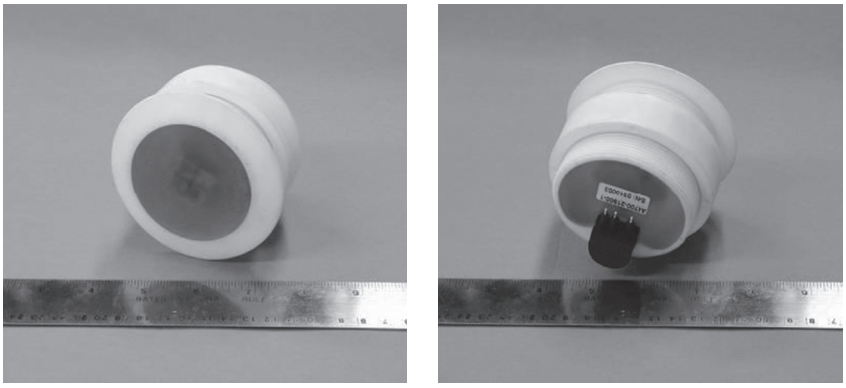


FIGURE 20.2 (a) Front and (b) back view of the single-element receiver.

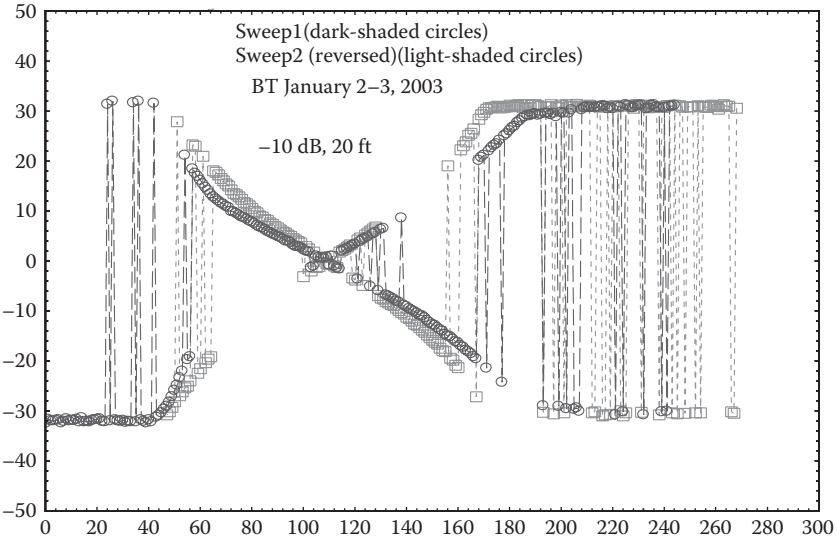


FIGURE 20.3 Bearing angle estimation experiment with the conventional method.

Because of the complexity and sensitivity of the underwater acoustic propagation and serious multipath interference, the accuracy of geolocation has been an extremely difficult problem. For the estimation of polarity, the conventional approach is to search for the first peak of the matched filtered received signal and identify its phase term. Figure 20.3 shows a typical result of the horizontal bearing angle estimation experiment with the conventional method. The experiment was conducted in a laboratory pool. The receiver unit scans along a circular path with a 60° angular span, from –30° to +30°. The horizontal axis is the bearing angle, and the vertical axis is the estimate. The ideal result is a monotonically increasing pattern. This chart describes the results of two experiments, marked in light and dark shades, respectively. The rapid hopping between the top and bottom curves represents the failure of correctly estimating the polarity of the bearing angles. This difficult task has long been regarded as the bottleneck of high-precision underwater geolocation and navigation.

20.3 Double-Integration Method

To start the analysis, the basic structure of the received beacon signal is examined. From a particular receiver position, the twin beacon signal, with the polarity of 180° phase offset, received from the transmitter array can be written in the form of

$$T_2(t) = \left[h\left(t+\frac{\Delta}{2}\right) - h\left(t-\frac{\Delta}{2}\right)\right]$$
$$= \left[\delta\left(t+\frac{\Delta}{2}\right) - \delta\left(t-\frac{\Delta}{2}\right)\right] * h(t) \tag{20.1}$$

where $h(t)$ is the designated transmission waveform. For the laboratory prototype, the signal $h(t)$ has a carrier frequency of 73.9 kHz with a 15.8 kHz bandwidth. The positive-polarity term $h(t + \Delta/2)$ is transmitted from the element at the right, and the negative-polarity term from the left transmitter element is $-h(t - \Delta/2)$.

The time-delay term Δ is the relative propagation lag due to the separation distance D between the transmitter elements:

$$\Delta = \frac{D}{v}\sin(\theta) \tag{20.2}$$

where

θ is the unknown bearing angle
v is the propagation speed

The width of the separation D is 1 cm. It is important to point out that the time-shift term Δ can be either positive or negative, depending on the value of the bearing angle. If the receiver is at the right-hand side of the transmitter, the angle θ is positive, and Δ becomes a positive value accordingly. Then the positive-polarity term $\delta(t + \Delta/2)$ arrives at the receiver first, followed by the negative-polarity term $-\delta(t - \Delta/2)$, Δ seconds later. On the other hand, when the receiver is at the left-hand side of the transmitter, the angle θ becomes negative, and so is Δ. Then the negative-polarity term $-\delta(t - \Delta/2)$ arrives at the receiver first, followed by the positive-polarity term $\delta(t + \Delta/2)$.

The objective of the algorithm is the accurate estimation of the term Δ, which gives the estimate of the polarity as well as the bearing angle. The detected signal at the receiver is in the form of

$$s(t) = T_2(t-d)$$
$$= c\left[\delta\left(t-d+\frac{\Delta}{2}\right) - \delta\left(t-d-\frac{\Delta}{2}\right)\right] * h(t) \tag{20.3}$$

where

d is the time delay due to the propagation from the center of the transmitter-array unit to the receiver
c is the attenuation factor due to propagation loss [1–3]

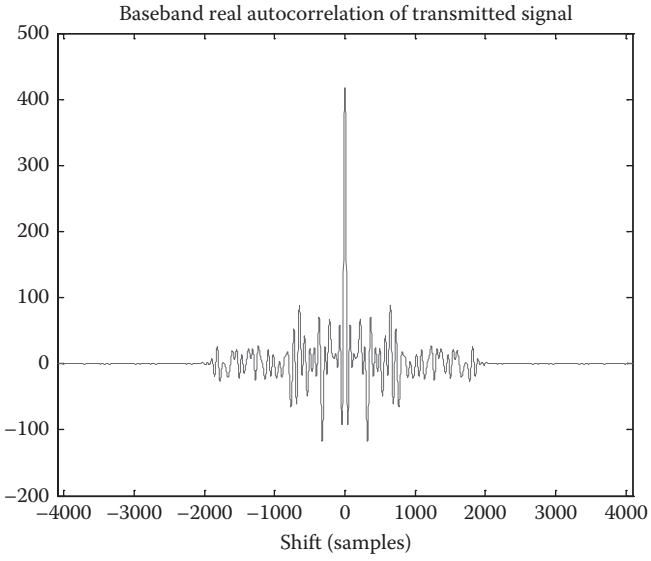


FIGURE 20.4 Autocorrelation function $R_0(t)$ used in the field experiment.

At the receiving end, matched filtering is first applied to the received signal. Mathematically, the matched filtering process can be formulated in the form of a convolution with $h^*(-t)$:

$$\begin{aligned} r_0(t) &= s(t) \times h^*(-t) \\ &= c\left[\delta\left(t-d+\frac{\Delta}{2}\right)-\delta\left(t-d-\frac{\Delta}{2}\right)\right] * \left[h(t)\times h^*(-t)\right] \\ &= c\left[\delta\left(t-d+\frac{\Delta}{2}\right)-\delta\left(t-d-\frac{\Delta}{2}\right)\right] * R_0(t) \\ &= c\left[R_0\left(t-d+\frac{\Delta}{2}\right)-R_0\left(t-d-\frac{\Delta}{2}\right)\right] \end{aligned} \tag{20.4}$$

where $R_0(t)$ is the autocorrelation of $h(t)$. Figure 20.4 shows the autocorrelation function $R_0(t)$ used in the field experiment.

Because $R_0(t)$ is zero at $t = \pm\infty$, the initial and final values of the matched filtered signal $r_0(t)$ are also zero:

$$r_0(-\infty) = r_0(\infty) = 0 \tag{20.5}$$

Figure 20.5 shows a typical received signal from the laboratory experiment after matched filtering. It can be seen that the fluctuation of the signal explains the difficulty in accurate estimating the bearing angle with the conventional approach.

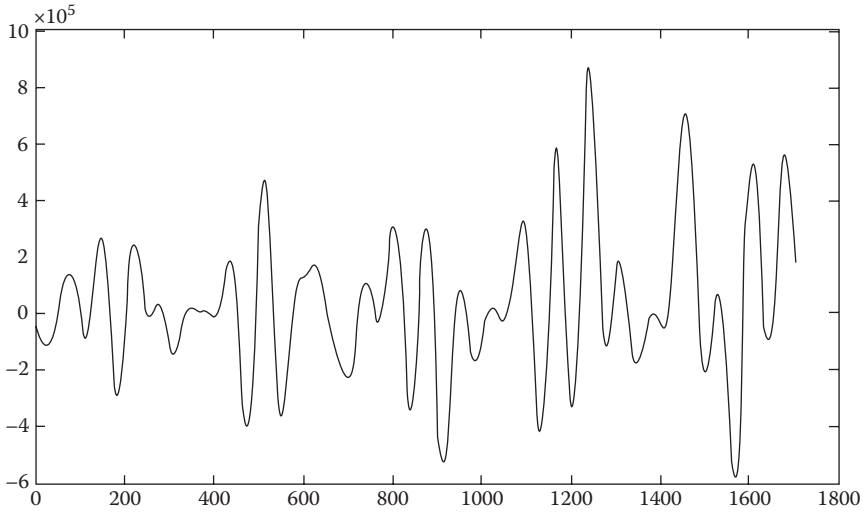


FIGURE 20.5 A received signal after matched filtering.

Now, we introduce a new function $r_1(t)$ by integrating $r_0(t)$:

$$\begin{aligned} r_1(t) &= \int_{-\infty}^{t} r_0(\tau)\, d\tau \\ &= c\left[u\left(t-d+\frac{\Delta}{2}\right)-u\left(t-d-\frac{\Delta}{2}\right)\right] * R_0(t) \\ &= \pm c\, p_\Delta(t-d) * R_0(t) \end{aligned} \tag{20.6}$$

where $p_\Delta(t)$ is a pulse of unit amplitude with pulse duration $|\Delta|$. Note that the duration of the pulse is independent of the time-delay term d. The ±polarity of the term depends solely on the value of Δ. And the polarity of the term follows exactly the sign of the time delay Δ.

Equation 20.6 also shows that $r_1(t)$ is the result of a convolution of the autocorrelation function $R_0(t)$ with a finite-length pulse with a ±polarity and length $|\Delta|$. If we perform the second integration, it results in

$$\begin{aligned} r_2(t) &= \int_{-\infty}^{t} r_1(\tau)\, d\tau \\ &= \pm c\, p_\Delta(t-d) * \int_{-\infty}^{t} R_0(\tau)\, d\tau \\ &= \pm c\, p_\Delta(t-d) * R_1(t) \end{aligned} \tag{20.7}$$

where $R_1(t)$ is the result of integration of $R_0(t)$ as

$$R_1(t) = \int_{-\infty}^{t} R_0(\tau)\, d\tau \tag{20.8}$$

Thus, the final value of $R_1(t)$ is the direct current (DC) term of $R_0(t)$. Since $R_0(t)$ is the autocorrelation of $h(t)$, the DC term of $R_0(t)$ is $|H(0)|^2$, where $H(j\omega)$ is the Fourier spectrum of $h(t)$ [4]. Hence,

$$R_1(\infty) = |H(0)|^2 \tag{20.9}$$

Subsequently, we find the final value of $r_2(t)$ as

$$\begin{aligned} \text{Final value} = r_2(\infty) &= c\Delta \cdot R_1(\infty) \\ &= c\Delta \cdot |H(0)|^2 \\ &= c\frac{D\sin(\theta)}{v}|H(0)|^2 \\ &= \frac{cD}{v}|H(0)|^2 \sin(\theta) \end{aligned} \tag{20.10}$$

Note the separation of the transmitter elements D, propagation velocity v, and propagation attenuation c. Figure 20.4 shows the autocorrelation function $R_0(t)$ used in the field experiment. It should be noted that the term $|H(0)|^2$ is the DC term of the power spectrum of the transmitted signal $h(t)$.

Thus, from Equation 20.10, the final value of $r_2(t)$ is linearly related to sin(θ). This means that the final value of $r_2(t)$ can uniquely characterize the polarity of the signal as well as the bearing angle of the receiver with respect to the transmitter. Figure 20.6 is the result of the double integration, showing steady convergence to a collection of values in proportion to sin(θ) of the bearing angles, indicated by Equation 20.10.

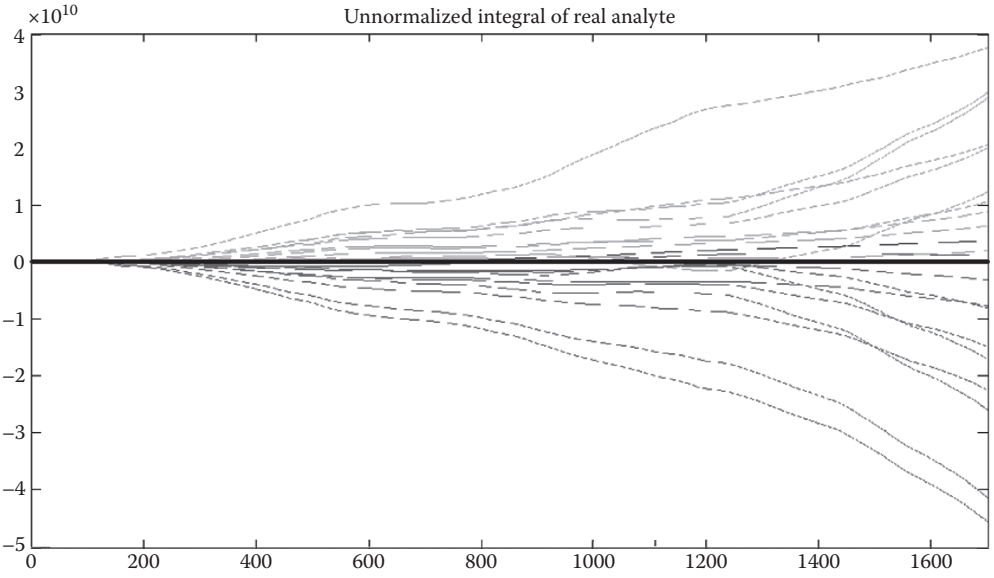


FIGURE 20.6 Results of the double-integration algorithm.

20.4 Normalization

The parameters in the constant term, D, v, and $|H(0)|^2$, are fully defined from the system design. The propagation loss term c is the only uncertain element. The propagation loss is known to be a function of the range distance as well as beam patterns. When the mobile sensor unit changes its position, the range distance and beam pattern change, and the value of the c term varies also as a result, which introduces ambiguity to the estimation of the bearing angle. Thus, to achieve high-precision estimation of the bearing angle, the c factor needs to be isolated from the estimation process.

The reference signal, the first pulse from the transmitter array, is in the form of

$$\begin{aligned} T_1(t) &= \left[h\left(t+\frac{\Delta}{2}\right)+h\left(t-\frac{\Delta}{2}\right)\right] \\ &= \left[\delta\left(t+\frac{\Delta}{2}\right)+\delta\left(t-\frac{\Delta}{2}\right)\right] * h(t) \end{aligned} \tag{20.11}$$

The corresponding detected signal at the receiver is in the form of

$$\begin{aligned} s(t) &= T_1(t-d) \\ &= c\left[\delta\left(t-d+\frac{\Delta}{2}\right)+\delta\left(t-d-\frac{\Delta}{2}\right)\right] * h(t) \end{aligned} \tag{20.12}$$

After matched filter, the signal is

$$\begin{aligned} r_0(t) &= s(t) \times h^*(-t) \\ &= c\left[\delta\left(t-d+\frac{\Delta}{2}\right)+\delta\left(t-d-\frac{\Delta}{2}\right)\right] * R_0(t) \\ &= c\left[R_0\left(t-d+\frac{\Delta}{2}\right)+R_0\left(t-d-\frac{\Delta}{2}\right)\right] \end{aligned} \tag{20.13}$$

After one integration, it becomes

$$\begin{aligned} r_1(t) &= \int_{-\infty}^{t} r_0(\tau)\,d\tau \\ &= c\int_{-\infty}^{t} R_0\left(\tau-d+\frac{\Delta}{2}\right)+R_0\left(\tau-d-\frac{\Delta}{2}\right)d\tau \\ &\approx c\left[R_1\left(t-d+\frac{\Delta}{2}\right)+R_1\left(t-d-\frac{\Delta}{2}\right)\right] \end{aligned} \tag{20.14}$$

Then the final value of $r_1(t)$ is

$$\begin{aligned} \text{Final value} &= r_1(\infty) = 2cR_1(\infty) \\ &= 2c|H(0)|^2 \end{aligned} \tag{20.15}$$

Now, it can be seen that the ratio of the final values, after the double integration of the beacon signal and the first integration of the reference waveform, is

$$\rho = \frac{D}{2v}\sin(\theta) \tag{20.16}$$

The normalization removes the ambiguity factor due to propagation loss and as a result isolates the bearing angle estimation from sources of errors. Since the separation distance D and propagation speed v are known, then the bearing angle can be estimated with high accuracy. This algorithm was applied to the same dataset, and the result is documented in Figure 20.7, which shows superior performance and accuracy. Figure 20.7a is the result of the double-integration algorithm without the normalization procedure, and Figure 20.7b is the result after normalization.

Ideally, based on the design of the experiments, the estimates of the bearing angles would be a linear function, from −30° to +30°. The deviation of the result from the line represents the estimation errors of the algorithm. From the figure, the distributions of the estimated bearing angles seem similar and in proportion. Note that the scale of the top figure is 10^4 of that of the bottom figure. Thus, from the scales of the two plots, the rescaling by the normalization process significantly reduces the estimation errors. Figure 20.8 shows the estimation of the bearing angles after the nonlinear conversion:

$$\theta = \sin^{-1}\left(\frac{2v\rho}{D}\right) \tag{20.17}$$

The accuracy at the edges of the curve, corresponding to the angles at ±30°, is mainly due to the beam patterns of the transceivers, and the noise is amplified by the nonlinear *arcsine* operation in the conversion.

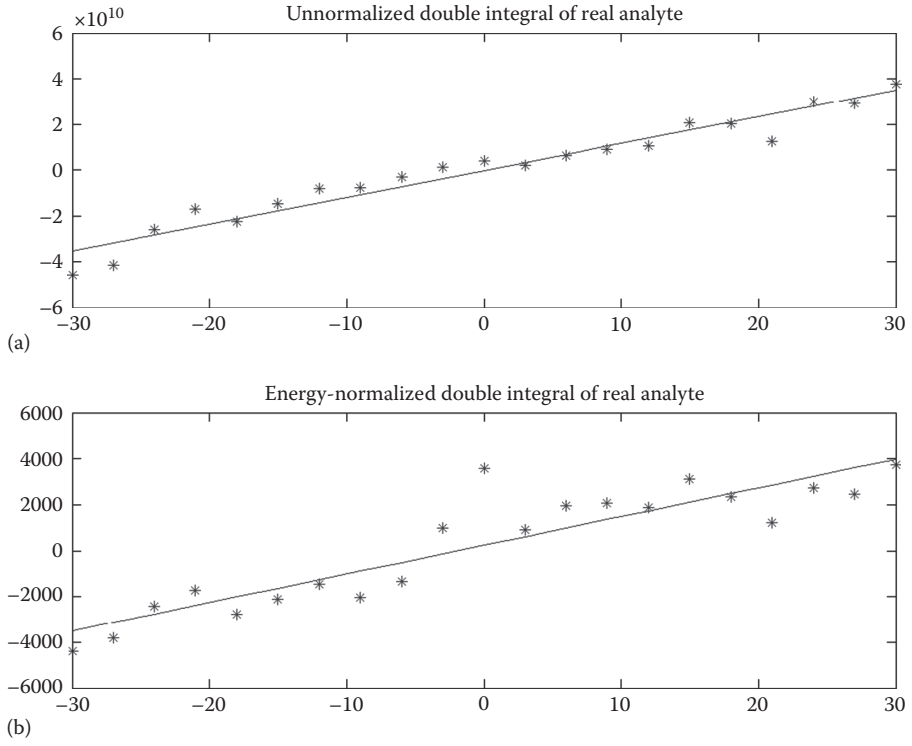


FIGURE 20.7 Estimation of the bearing angles (a) before the normalization and (b) after the normalization.

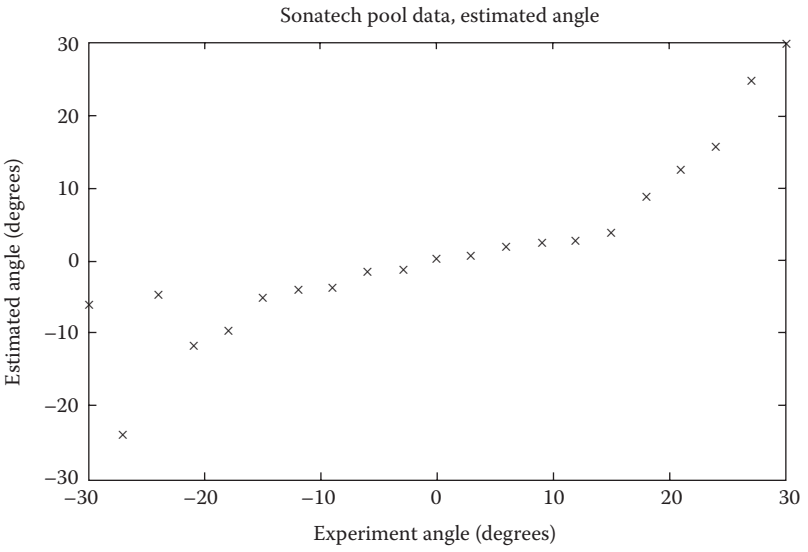


FIGURE 20.8 Estimation of the bearing angles after the nonlinear conversion.

20.5 Reversed System

It can be seen that the acoustic geolocation tasks can function even more effectively with the reversed version of the system. This is to use a single-element transmitter and a multiple-element receiver array. The reversed version simplifies the transmission operation so that it is required to transmit only one signal instead of a sequence of three. It also makes the transmitter electronics more energy efficient.

Suppose the receiver is a four-element circular array. The diameter of the array is kept *D*. The received and matched filtered signals from the four receiver elements are $\{r_{-x0}(t), r_{x0}(t), r_{-y0}(t), r_{y0}(t)\}$, respectively. We can first obtain the normalization factor as the final value of the integral of the average signal $r_s(t)$, where

$$r_s(t) = \frac{1}{4}\left(r_{x0}(t) + r_{-x0}(t) + r_{y0}(t) + r_{-y0}(t)\right) \tag{20.18}$$

Then we partition the four elements into two pairs and apply the double-integration method to the difference $[r_{x0}(t) - r_{-x0}(t)]$ and $[r_{y0}(t) - r_{-y0}(t)]$. This will give the result of the bearing angle vector $[\theta_x, \theta_y]$ directly. To be more computationally effective, we can apply the double-integration procedure to the weighted version of the differential

$$r_d(t) = \left(r_{x0}(t) - r_{-x0}(t)\right) + j\left(r_{y0}(t) - r_{-y0}(t)\right) \tag{20.19}$$

The result of the double-integration procedure produces a complex scalar. After normalization, the real and imaginary parts of the complex scalar represent the bearing angle vector. The computation involved is so simple that it can be implemented in the form of first-order op-amp circuits.

To improve the accuracy of the estimation, one approach is to increase the number of receiver elements. For conventional methods, the increase in the number of receiver elements couples with the increase in computation complexity. However, the effect to this technique due to array size is very minor.

To compute for the normalization factor, we recognize the combined signal $r_s(t)$ is the average of the received signals. The differential signal $r_d(t)$ can be regarded as the inner product of the receiver position vector and the received signal vector:

$$\begin{aligned} r_d(t) &= \left(r_{x0}(t) - r_{-x0}(t)\right) + j\left(r_{y0}(t) - r_{-y0}(t)\right) \\ &= \left[1, -j, -1, +j\right]\left[r_{x0}(t), r_{y0}(t), r_{-x0}(t), r_{-y0}(t)\right]^T \end{aligned} \tag{20.20}$$

If we extend this concept to a circular array with N uniformly spaced receiver elements, the normalization factor can be obtained from the final value of the integral of the average signal $r_s(t)$:

$$r_s(t) = \frac{1}{N}\left(r_{01}(t) + r_{02}(t) + r_{03}(t) + \cdots + r_{0N}(t)\right) \tag{20.21}$$

and the differential signal is in the form of

$$r_d(t) = \sum_{k=1}^{N} \exp\left(\frac{-j2\pi k}{N}\right) r_{0k}(t) \tag{20.22}$$

The complex weighting coefficients $\{\exp(j2\pi k/N)\}$ are fully defined by the relative positions of the receivers on the 2D plane. This means that both $r_s(t)$ and $r_d(t)$ are linear combinations of the received signals. After these two simple superposition steps, the computation procedure is identical. This implies that the computation complexity of this algorithm remains largely the same for varying number of receiver elements, which enables significant enhancement of accuracy of bearing angle estimation without the increase in computation complexity in hardware as well as software.

Traditional algorithms for the estimation of bearing angle are based on the relative time delays among the receivers. This is largely accomplished by cross-correlation procedures, which require substantial computation. In comparison, the double-integration method is much simpler in terms of computation complexity and hardware structure, with excellent stability and accuracy. By nature, the integration operations of the algorithm provide good tolerance against noise. The simplicity also translates into savings in computation time and power consumption.

20.6 Extension to Active Systems

The existing system is designed for geolocation, navigation, homing, and docking operations, with separate transmitter and receiver system. The receiver component functions as a passive data acquisition device [5–10]. Yet one extremely interesting extension is to modify it into an active system. This can be achieved by placing the transmitter at the center of the circular receiver array.

With the same integration method, this system is capable of accurately estimating the bearing angle of the reflected waveform. Because the transmitted signal is readily available, the differential between the average received waveform $r_s(t)$ and transmitted signal $T_1(t)$ can be obtained, and the relative time delay can be estimated by using the double-integration method in a similar manner. Subsequently, the range distance of the target can be calculated from the time delay. With the capability of dynamic estimation of both the range distance and bearing angle in real time with low-power-consumption level, this system can be deployed for collision avoidance for UUV or unmanned aerial vehicles (UAV).

20.7 Summary

In this chapter, a simple technique was presented for bearing angle estimation for UUV underwater geolocation and navigation. The algorithm was originally designed as the replacement for the conventional peak-phase approach for the systems with twin transmission waveforms and the polarity of 180° phase offset. The accuracy, stability, and especially the simplicity of this algorithm made the system, in both hardware and software, significantly more effective.

It turns out that this algorithm can also function effectively with the reversed version of the system. The conventional techniques for passive multireceiver acoustic arrays were largely structured, based on the computationally extensive cross-correlation method. This simple algorithm showed remarkable system performance in laboratory tests, and the simplicity in computation remains for a large number of receiver elements.

The most interesting extension of this simple signal processing technique is the application to the active version of the system, with combined transmitter and receivers in the same unit. The accuracy, stability, and computational efficiency make it an excellent system for UUV collision avoidance. The analysis in the paper has been organized in the pulse-echo mode. Yet, with slight modifications, this algorithm can function equally well with frequency-modulated continuous-wave (FMCW) signaling formats.

References

1. H. Lee and G. Wade, *Imaging Technology*, IEEE Press, New York, 1986.
2. H. Lee and G. Wade, *Modern Acoustical Imaging*, IEEE Press, New York, 1986.
3. J.W. Goodman, *Introduction to Fourier optics*, McGraw Hill, New York, 1968.
4. A.D. Whalen, *Detection of Signals in Noise*, Academic Press, New York, 1971, pp. 39–41.
5. H. Lee, Localization and tracking techniques for navigation and guidance for underwater autonomous sensing systems, *Journal of the Acoustical Society of America*, 116(4), 2633, 2004.
6. D. Doonan, C. Utley, and H. Lee, Signal processing algorithms for high-precision navigation and guidance for underwater autonomous sensing systems, *Proceedings of 2004 International Telemetry Conference*, 2004.
7. T. Fu, D. Doonan, C. Utley, R. Iltis, R. Kastner, and H. Lee, Design and development of a software-defined underwater acoustic modem for sensor networks for environmental and ecological research, *OCEANS 2006*, Iss, 1–6, 2006.
8. T. Fu, D. Doonan, C. Utley, and H. Lee, Field testing of a spread spectrum acoustic modem with sparse channel estimation, *Proceedings of IEEE International Conference on Acoustic, Speech, and Signal Processing*, pp. 5292–5295, 2008.
9. H. Lee, Signal processing algorithms for UUV navigation and collision avoidance, *Journal of the Acoustical Society of America*, 125(4), 2537, 2009.
10. H. Lee, Underwater acoustical imaging and sensing systems for homing and docking, navigation, and collision avoidance, *Acoustical Imaging*, 30, 295–304, 2011.

21

Load-Balanced Virtual Backbones in Wireless Sensor Networks

21.1 Introduction **21**-1
Virtual Backbone • Load-Balanced Virtual Backbone • Deterministic Wireless Sensor Networks and Probabilistic Wireless Sensor Networks • Chapter Organization
21.2 Related Work **21**-4
Centralized Algorithms for CDS • Distributed Algorithms for CDS • Other Algorithms for CDSs • Other Load-Balancing-Related Work
21.3 LBVB under DNM **21**-6
Problem Statement • Load-Balanced CDS • Load-Balanced Allocation of Dominatees • Performance Evaluation
21.4 LBVB under PNM **21**-21
Network Model and Problem Definition • LBVBP-MOGA • Performance Evaluation
21.5 Summary **21**-36
References **21**-36

Jing (Selena) He
Kennesaw State University

21.1 Introduction

21.1.1 Virtual Backbone

Wireless sensor networks (WSNs) are deployed for monitoring and controlling of systems where human intervention is not desirable or feasible. One typical characteristic of WSNs is the possibility of deploying many nodes in an area to ensure sufficient coverage of an area and/or to have redundancy against node failures. However, in a relatively crowded network, many problems are aggravated: (1) many nodes interfere with each other, (2) there are a lot of possible routes, (3) nodes might needlessly use large transmission power to talk to distant nodes directly, and (4) routing protocols might have to recompute routes even if only a small number of nodes changed their locations. These problems can be overcome by selecting some nodes as a *virtual backbone* (VB) for a network, in which only the links within this backbone and direct links from other nodes to the backbone nodes are mainly used in the WSN. Usually, we use a *dominating set* (DS) to serve as a backbone for a WSN, which is a subset of nodes in the network where every node is either in the subset or a neighbor of at least one node in the subset. For a backbone to be useful, it should be connected, namely, by a *connected DS* (CDS). The nodes in a CDS are called *dominators*, otherwise *dominatees*. In a WSN with a CDS as its VB, dominatees only forward their data to their connected dominators. Moreover, the CDS with the smallest size (the number of nodes in the CDS) is called a *minimum-sized CDS* (MCDS). In this chapter, CDS and VB are used interchangeably.

Since only dominators need to maintain the routing information, CDS-based routing becomes much easier and more qualified to accommodate topology change. To be specific, the search space for the available routes is reduced within a CDS. Moreover, if there is no topology change in the subgraph induced by the CDS, there is no need to update the routing information at all. In addition to routing [1,2], a CDS has many other applications in WSNs, such as data collection [3,4], broadcasting [5,6], topology control [7], coverage [8,9], data aggregation [10], and query scheduling [11]. Clearly, the benefits of a CDS can be magnified by making its size smaller. Therefore, it is desirable to build an MCDS to reduce the number of nodes and links involved in communication. As a matter of fact, constructing a CDS, especially an MCDS for WSNs, is one way to extend network lifetime.

Ever since the idea of employing a CDS for WSNs was introduced in [12], a huge amount of effort has been made to find CDSs with variety of features for different applications, especially the MCDS. In the seminal work [13], Guha and Khuller first modeled the problem of computing the smallest CDS as the MCDS problem in a general graph, which is a well-known NP-hard problem [14]. After that, to make a CDS more resilient in mobile WSNs, the fault tolerance of a VB is considered. In [15–17], *k*-connected and *m*-dominated sets are introduced as a generalized abstraction of a fault-tolerance VB. In [18], the authors proposed a minimum routing cost CDS (MOC-CDS), which aims to find a minimum CDS while assuring that any routing path through this CDS is the shortest in WSNs. Additionally, the authors investigate the problem of constructing a quality CDS in terms of size, diameter, and average backbone path length (ABPL) in [19,20].

21.1.2 Load-Balanced Virtual Backbone

Unfortunately, all the aforementioned works did not consider the *load-balance* factor when they construct a CDS. If the workload on each dominator in a CDS is not balanced, some heavy-duty dominators deplete their energy quickly. Then, the whole network might be disconnected. Hence, intuitively, we not only have to consider constructing an MCDS but also need to consider constructing a load-balanced CDS (LBCDS). An illustration of an LBCDS is depicted in Figure 21.1, in which dominators are marked as black nodes, while white nodes represent dominatees. Moreover, the number beside each node represents the node's degree. In Figure 21.1b and c, solid lines represent that the dominatees are allocated to the connected dominators, while the dashed lines represent the communication links in the original graph shown in Figure 21.1a. According to the traditional MCDS construction algorithms, a CDS $\{s_4, s_7\}$ with size 2 is obtained for the network shown in Figure 21.1a. However, there are two severe drawbacks of the CDS shown in Figure 21.1a. For convenience, the set of neighboring dominatees of a dominator s_i is denoted by $ND(s_i)$. First, $ND(s_4) = \{s_1, s_2, s_3, s_5, s_6\}$, which represents that dominator s_4 connects to five different dominatees, and $ND(s_7) = \{s_6, s_8\}$. If every dominatee has the same amount of data to be transferred through the connected dominator at a fixed data rate, dominator s_4 must deplete its energy much faster than dominator s_7, since dominator s_4 has to forward the data collected from five connected dominatees. Second, dominatee s_6 connects to both dominators. If s_6 chooses dominator s_4 as its data forwarder, obviously, only one dominatee s_8 can forward its data to dominator s_7. In this situation, the workload imbalance in the CDS is further amplified. Consequently, the entire network lifetime is shortened. We show a counterexample in Figure 21.1b, where the constructed CDS is $\{s_3, s_6, s_7\}$. According to the topology shown in Figure 21.1b, we can get the dominatee sets of each dominator: $ND(s_3) = \{s_1, s_2, s_4\}$, $ND(s_6) = \{s_4, s_5\}$, and $ND(s_7) = \{s_4, s_8\}$. Compared with the MCDS constructed in Figure 21.1a, the numbers of dominatees of all the dominators in Figure 21.1b are very similar. For convenience, we use $A(s_i) = \{s_j | s_j$ is a dominatee and s_j forward its data to $s\}$ to represent the dominatees allocated to a dominator s_i. Thus, we can have two different dominatee allocation schemes shown in Figure 21.1b and c, respectively. One is $A(s_3) = \{s_1, s_2, s_4\}$, $A(s_6) = \{s_5\}$, and $A(s_7) = \{s_8\}$. The other one is $A(s_3) = \{s_1, s_2\}$, $A(s_6) = \{s_4, s_5\}$, and $A(s_7) = \{s_8\}$. Apparently, the workload on each dominator is almost evenly distributed in the CDS constructed in Figure 21.1c. Intuitively, the construction algorithm and dominatee allocation scheme shown in Figure 21.1c can extend network lifetime notably, since the traffic loads on each dominator shown in Figure 21.1c are almost evenly distributed.

Hence, to benefit from the CDS-based VB in WSNs and also take the load-balance factor into consideration, it is better to construct a load-balanced VB (LBVB).

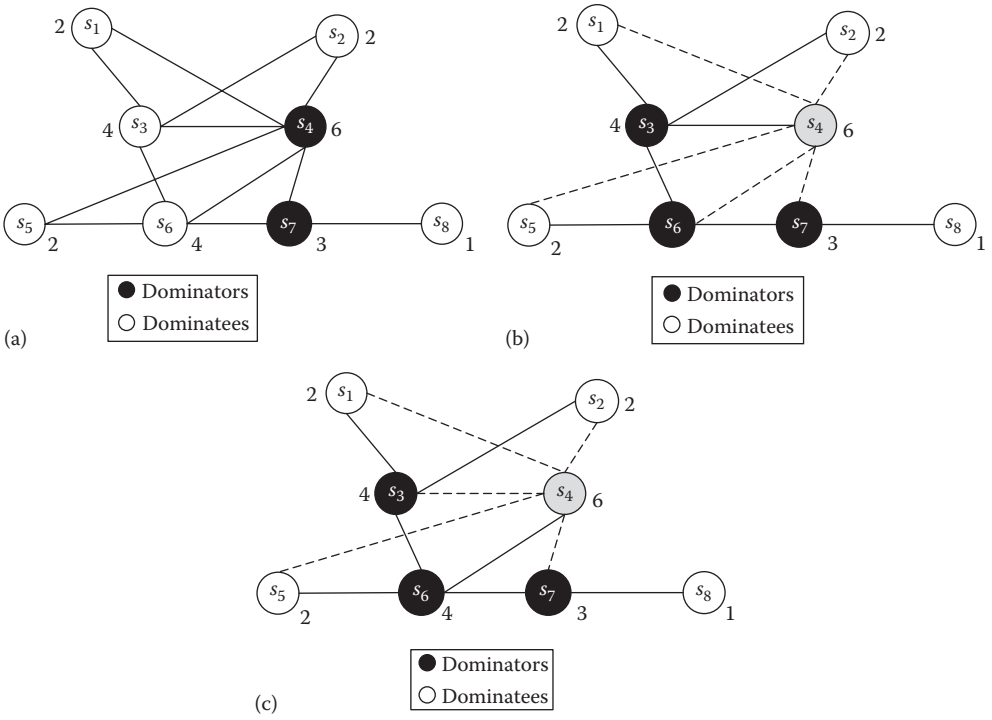


FIGURE 21.1 (a) A regular CDS; (b) a load-balanced CDS with an imbalanced dominate allocation scheme; (c) a load-balanced CDS with a load-balanced dominate allocation scheme.

21.1.3 Deterministic Wireless Sensor Networks and Probabilistic Wireless Sensor Networks

WSNs are usually modeled using the deterministic network model (DNM) in recent literature. Under this model, there is a transmission radius of each node. According to this radius, specific pairs of nodes are always connected to be neighbors if their physical distance is less than this radius, while the rest of the pairs are always disconnected. The unit disk graph (UDG) model is a special case of the DNM model if all nodes have the same transmission radius. When all nodes are connected to each other, via a single-hop or multihop path, the WSN is said to have full connectivity. In most real applications, however, the DNM model cannot fully characterize the behavior of wireless links. This is mainly due to the transitional region phenomenon that has been revealed by many empirical studies [21–24]. Beyond the *always connected* region, there is a transitional region where a pair of nodes is probabilistically connected. Such pairs of nodes are not fully connected but reachable via the so-called lossy links [24]. As reported in [24], there are often many more lossy links than fully connected links in a WSN. Additionally, in a specific setup [25], more than 90% of the network links are lossy links. Therefore, their impact can hardly be neglected.

The employment of lossy links in WSNs is not straightforward, since when the lossy links are employed, the WSN may have no guarantee of full network connectivity. When data transmissions are conducted over such topologies, they may degrade the node-to-node delivery ratio. Usually, a WSN has large node density and high data redundancy; thus, this certain degraded performance may be acceptable for many WSN applications. Therefore, as long as an expected percentage of the nodes can be reached, that is, the node-to-node delivery ratio satisfies some preset requirement, lossy links are tolerable in a WSN. In other words, full network connectivity is not always a necessity. Some applications can trade full network connectivity for a higher energy efficiency and larger network capacity [25].

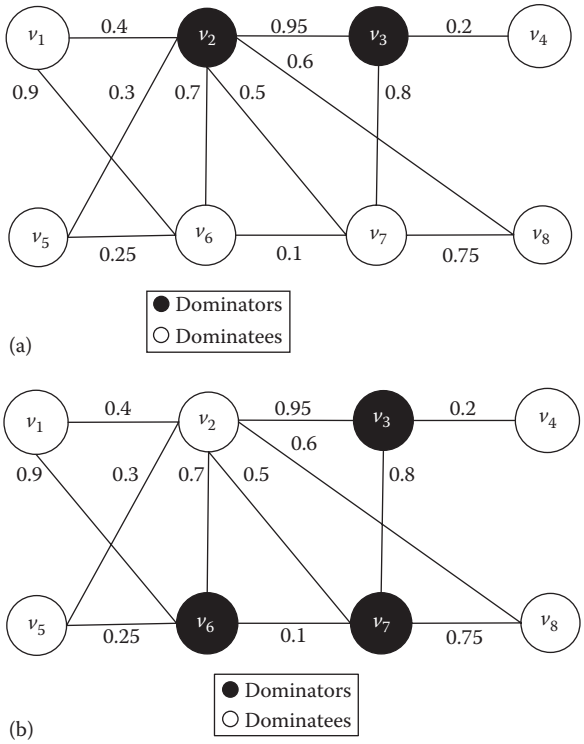


FIGURE 21.2 (a) A regular VB under PNM; (b) a Load-Balanced VB (LBVB) under PNM.

Therefore, a more practical network model for WSNs is the probabilistic network model (PNM). Under this model, there is a *delivery ratio* (γ_{ij}) associated with each link connecting a pair of nodes v_i and v_j, which is used to indicate the probability that v_i can successfully deliver a package to v_j (an example is shown in Figure 21.2). For convenience, the WSNs considered under the DNM/PNM are called deterministic/probabilistic WSNs.

For probabilistic WSNs, how to measure the traffic load on each node is different compared to deterministic WSNs. For example, in Figure 21.2, $\gamma_{27} = 0.5$, which means the probability that v_2 can successfully deliver a packet to v_7 is 50%. Then the expected number of transmissions to guarantee v_2 delivered one packet to v_7 is 1/0.5 = 2. In other words, the less the γ_{ij} value, the more potential traffic load on the link from v_j to v_i. Hence, in this chapter, LBVB problems are investigated under both deterministic and probabilistic WSNs.

21.1.4 Chapter Organization

The rest of this chapter is organized as follows: in Section 21.2, we review some related works on CDSs. In Section 21.3, we investigate the LBVB problem under DNM. In Section 21.4, the LBVB problem is studied under PNM. Finally, the chapter is concluded in Section 21.5.

21.2 Related Work

The following section reviews related work on CDSs.

21.2.1 Centralized Algorithms for CDS

Constructing an MCDS for a WSN is a well-known NP-hard problem even under the UDG model [26]. Many research works have been devoted to achieve a better performance ratio. For centralized CDS algorithms, they generally can be categorized into two types: one-stage algorithms and two-stage algorithms.

One-stage algorithms [27] aim to construct a CDS directly. In contrast, two-stage algorithms construct a CDS in two stages. The first stage is to select a minimum DS, and the second stage is to construct a CDS using the technique of Steiner tree [28]. In [27], two centralized greedy algorithms were proposed. The first algorithm is a one-stage strategy with an approximation ratio of $2H(\Delta) + 2$ where Δ is the maximum node degree in a network and $H(\cdot)$ is a harmonic function. The second strategy is a two-stage strategy with an approximation ratio of $H(\Delta) + 2$. Later, based on the main idea of [27], Ruan et al. [29] proposed a one-stage algorithm with an approximation ratio of $3 + \ln(\Delta)$.

21.2.2 Distributed Algorithms for CDS

Due to the instability of network topology in WSNs, it is necessary to update topology information periodically. However, to deal with the network scalability and dynamical changes in a centralized way, the cost is extremely high. Therefore, many distributed algorithms are proposed.

These distributed algorithms can be classified into two categories: subtraction based [30,31] and addition based [10,32,33]. The subtraction-based algorithms begin with the set of all the nodes in a network and then remove some nodes according to predefined rules to obtain a CDS. The best-known algorithms in this category include Wu et al. [30] and Dai's [31] algorithms.

The addition-based CDS algorithms start from a subset of nodes (usually disconnected) and then include additional nodes to form a CDS. Depending on the type of the initial subset, the addition-based CDS algorithms can be further divided into maximal independent set (MIS) based [10] and tree based [32,33]. The MIS-based CDS algorithm [10] obtains a CDS by selecting an MIS first (which is also a DS actually) and then finding connectors between the MIS nodes. For convenience, we use *opt* to denote the size of any optimal MCDS. For UDGs, Wan et al.'s [34] approach guarantees that the approximation factor on the size of a CDS is at most $4\,opt + 1$. Later, many attempts tried to improve this approximation factor on the size of a CDS based on this idea. Wu et al. reported an approximation factor of $3.8\,opt + 1.2$ [35]. Yao et al. improved it to $3.67\,opt + 1.33$ [36]. The factor is further improved to $3.478\,opt + 4.874$ [37] and $3.4306\,opt + 4.8185$ [38], subsequently. The current best is that the size of an MIS is at most $3.399\,opt + 4.874$ [39].

The tree-based CDS algorithms, for instance, single-initiator (SI) version in [32] and multi-initiator (MI) version in [33], start from a subset of nodes called initiators and grow a dominator tree from each of the initiators. Particularly, there are three phases in the tree-based algorithms. In phase one, a number of initiators are elected. In phase two, each initiator utilizes a timer to grow a tree so that the nodes with more neighbors can be added to the tree. In phase three, additional bridge nodes are added to connect neighboring trees. It has been shown in [33] that the addition-based algorithms generally produce smaller CDSs than the subtraction-based algorithms. Moreover, the tree-based algorithms incur less communication overhead.

21.2.3 Other Algorithms for CDSs

Because a CDS can benefit a lot to WSNs, a variety of other factors are considered when constructing a CDS. More than one CDS can be found for each WSN. To conserve energy, all CDSs are constructed and each CDS serves as the VB duty cycled in [40]. For the sake of fault tolerance, k-connect m-DSs [41] are constructed, where k-connectivity means between any pair of backbone nodes there exists at least k independent paths, and m-dominating represents that every dominatee has at least m adjacent dominator neighbors. To minimize delivery delay, a special CDS problem—MOC-CDS [18]—is proposed, where each pair of nodes in MOC-CDS has the shortest path. The work [19] considers more than one factor—size, diameter, and ABPL [19]—in order to construct a CDS with better quality.

21.2.4 Other Load-Balancing-Related Work

In this subsection, we summarize the recent work to improve the load balancing of other applications in WSNs.

The authors in [42] proposed an even energy dissipation protocol (EEDP) for efficient cluster-based data gathering in WSNs. In EEDP, sensor data are forwarded to the base station (BS) via multiple chains

of cluster heads. Each chain uses a rotation scheme to balance energy consumption among cluster heads and avoid the formation of a hot spot. Achieving efficient bandwidth utilization in multichannel sensor networks is a challenging research problem. In [43], the authors presented a cognitive load-balance algorithm for single-hop multichannel sensor networks. Based on the load distribution of all BSs, the proposed algorithm dynamically alternated the communication channels. As a result, the extra load from overloaded channels is directed to underloaded channels with a computed switch probability. Moreover, the authors also proved that a high throughput can be achieved if the load is balanced. In order to balance power usage in heterogeneous sensor networks, the load-balancing group clustering (LBGC) strategy is proposed in [44] based on the clustering model. The LBGC protocol periodically selects cluster heads and implements dynamic route calculation according to the condition of energy distributing of network, which could make full use of the heterogeneous energy to realize load balance. The authors in [45] proposed a novel approach for load balance using compressive sensing. The proposed approach offered accurate recovery of sampled data from a small number compressed data. Simulation results show that the proposed method achieves a good performance in terms of energy balancing among all nodes in a network, which outperforms the tree-based collection protocols. In [46], a three-layer framework was proposed for mobile data collection in WSNs, which includes the sensor layer, cluster head layer, and mobile collector. The framework employs distributed load-balanced clustering and MIMO uploading techniques. The objective is to achieve good scalability, long network lifetime, and low data collection latency.

21.3 LBVB under DNM

21.3.1 Problem Statement

In this section, we introduce the network model and define the LBCDS problem and the load-balanced allocation of dominatee (LBAD) problem formally.

21.3.1.1 Network Model

We assume that a static connected WSN is deployed in a square with area size $A = cn$, where c is a constant and the WSN is consisting of n sensors, denoted by $s_1, s_2, \ldots, s_n$, respectively. All sensors are independent identically distributed (i.i.d.) over the whole network. We also assume that all nodes have the same transmission range. We modeled the WSN as a connected undirected general graph $G = (V, E)$, in which V represents node set and E represents the link set. $\forall u, v \in V$, there exists an edge (u, v) in G if and only if u and v are in each other's transmission range. In this chapter, we assume that edges are undirected (bidirectional), which means that two linked nodes are able to transmit and receive information from each other.

21.3.1.2 Preliminary

The *load-balance* factor is our major concern in this work. Thus, finding an appropriate measurement to evaluate load balance is the key to solve the LBCDS and LBAD problems. We use *p-norm* to measure load balance in this chapter. The definition of *p-norm* is given as follows.

Definition 21.1: *p-Norm*

The p-norm of an $n \times 1$ vector $= (x_1, x_2, x_n)$ is

$$\left|\mathbf{X}\right|_p = \left(\sum_{i=1}^{n} \left|\mathbf{x}_i\right|^p\right)^{1/p} \tag{21.1}$$

The authors in [47] stated that p-norm shows interesting properties for different values of p. If p is close to 1, the information routes resemble the geometric shortest paths from the sources to the sinks. For $p = 2$, the information flow shows an analogy of an electrostatic field, which can be used to measure the load balance among x_i. More importantly, the smaller the *p-norm* value, the more load balanced the interested feature vector . For simplicity, we use $p = 2$ in this chapter.

In this chapter, we use node degree (Definition 21.2) and the number of dominatees connected to a dominator (Definition 21.3) of the interested node set as the information vector , since the degree of each node and the number of the dominatees connected to a dominator are only a potential indicator of traffic load.

We still use the WSN shown in Figure 21.1 to illustrate how to use *p-norm* to measure the load balance of CDSs. Two different CDSs for the same network are identified in Figure 21.1. The degree of the node s_i is denoted by d_i. Thus, $|d_i - \bar{d}|$ are used as the information vector , where $\bar{d}$ is the mean degree of the graph in Figure 21.1. Therefore, the *p-norm* value of the CDS shown in Figure 21.1a is $\sqrt{9}$. Similarly, in Figure 21.1b, the *p-norm* value is $\sqrt{2}$. Clearly, $\sqrt{2} < \sqrt{9}$, which implies that the CDS in Figure 21.1b is more load balanced than the CDS in Figure 21.1a.

After we construct an LBCDS, the next step is to allocate dominatees to each dominator in the LBCDS. The *p-norm* can again be used to measure the load balance of different allocation schemes, in which the number of dominatees connected to a dominator of the interested node set is used as the information vector . An illustration example is shown in Section 21.3.3.

21.3.1.3 Problem Definition

Now, we give the formal definition of the problems we investigate in this chapter.

Definition 21.2: *LBCDS*

For a WSN represented by graph $G = (V, E)$, the LBCDS problem is to find a node set $D \subseteq V$, $D = \{s_1, s_2, \ldots, s_M\}$ such that

1. $G[D] = (D, E')$, where $E' = \{e | e = (u, v), u \in D, v \in D, (u, v) \in E)\}$ is connected
2. $\forall u \in V$ and $u \notin D$, $\exists v \in D$, such that $(u, v) \in E$
3. $\min |D|_p = \left(\sum_{i=1}^{M} |d_i - \bar{d}|^p \right)^{1/2}$

Definition 21.3: *LBAD*

For a WSN represented by graph $G = (V, E)$ and a CDS $D = \{s_1, s_2, \ldots, s_M\}$, the number of the dominatees connecting to each dominator s_i $(1 \le i \le M)$ is denoted by $|A(s_i)|$, and the expected allocated dominatees of each dominator is denoted by $|\bar{A}|$, the LBAD problem is to find M disjoint sets on V, that is, $A(s_1)$, $A(s_2)$, …, $A(s_M)$, such that

1. Each set $A(s_i)$ $(1 \le i \le M)$ contains exactly one dominator s_i
2. $\bigcup_{i=1}^{M} A(s_i) = V, A(s_i) \cap A(s_j) = \varnothing \ (1 \le i \ne j \le M)$
3. $\forall u \in A(s_i)$ $(1 \le i \le M)$ and $u \ne s_i$ such that $(u, s_i) \in E$
4. $\min |D|_p = \left(\sum_{i=1}^{M} \| |A(s_i)| - |\bar{A}| \|^p \right)^{1/2}$

21.3.2 Load-Balanced CDS

21.3.2.1 Algorithm Description

In essence, we design a greedy algorithm to solve the LBCDS problem. The algorithm starts from an empty dominator set (DS). Each time, it adds the node into the DS set that has the smallest $|d_i - \bar{d}|$ value (where $1 \le i \le n$). If there exists a tie on the $|d_i - \bar{d}|$ value, we use greater d_i value to break the tie, since the nodes with higher degree can make the algorithm converge faster. The algorithm terminates when the nodes in the DS set form a CDS.

The pseudocode of the greedy algorithm is shown in Algorithm 21.1.

LBCDS-approximate algorithm as shown in Algorithm 21.1 is a centralized one-phase greedy algorithm. Initially, all the nodes are white. All black nodes form an LBCDS finally. We use the following terms in the algorithm: d_i, the degree of node s_i, and $\bar{d}$, the mean degree of G.

Initially, all nodes in the WSN are marked as dominatee nodes. Then, sorting the n sensors based on their $|d_i - \bar{d}|$ values, the sorted node IDs are stored in an array called A (shown in Line 2). Starting from the lowest subscript i of the sorted array A, if the nodes represented by $A[i]$ and all its 1-hop neighbors are not dominated, then mark the node represented by $A[i]$ as the dominator node (shown in Line 5). Keep continuing the process till all the black nodes form a CDS, then an LBCDS is constructed (from Lines 3 to 10).

Algorithm 21.1: LBCDS approximate

Require: A WSN represented by graph $G = (V, E)$, node degree d_i, and mean degree of the graph $\bar{d}$.
1: Make all nodes white {dominatee nodes}.
2: Sort the n sensors based on their $|d_i - \bar{d}|$ values. If there exists a tie, use greater d_i value to break the tie, where $1 \le i \le n$. The node IDs are stored in the sorted array denoted by A.
3: **for** $i = 1$ **to** n **do**
4: **if** the node represented by A_i and all its 1-hop neighbors are not dominated **then**
5: Mark the node represented by A_i black {dominator node}.
6: **if** All black nodes form a connected dominating set ((CDS) **then**
7: **return** all black nodes.
8: **end if**
9: **end if**
10: **end for**

The time complexity of the sorting process shown in Line 2 is $O(n\log(n))$, whereas the time complexity of the process to form an LBCDS (shown from Lines 3 to 10) is $O(n)$. Hence, the time complexity of Algorithm 21.1 is $O(n\log(n))$. Since Algorithm 21.1 is a centralized algorithm, no extra storage memory is required for each sensor in the WSN.

21.3.2.2 Example Illustration

We use the WSN shown in Figure 21.1 to illustrate how to build an LBCDS. Based on each node's degree, we can calculate $\bar{d} = 3$. According to the aforementioned LBCDS construction algorithm, in the first round, all the nodes with degree 3 are added into the DS set (shown in Figure 21.3a). Thus, node s_7 is added into the DS set, since $d_7 = \bar{d} = 3$. In the next round, nodes s_3 and s_6 with degree 4 are added into the DS set (shown in Figure 21.3b), since s_3 and its 1-hop neighbors, s_6 and its 1-hop neighbors are not dominated. Nodes s_1 and s_2 are not added into the DS set, because they have smaller degree values than

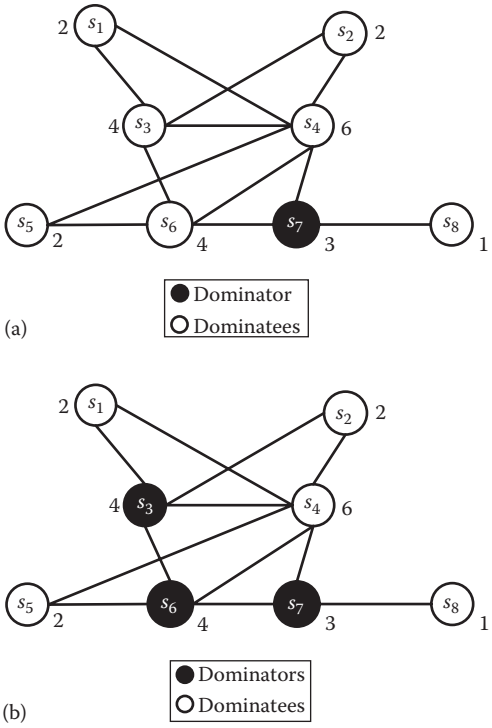


FIGURE 21.3 (a) The first round of running Algorithm LBCDS approximate; (b) The output of running Algorithm LBCDS approximate.

nodes s_3 and s_6. So far, there are three nodes in the DS set, which forms a CDS; therefore, the algorithm terminates. Finally, we get an LBCDS, which is $\{s_3, s_6, s_7\}$.

21.3.2.3 Remarks

According to Definition 21.2, *p-norm* is the measurement of the load balance for CDSs. The smaller the *p-norm* value, the more load balanced the CDS is. Moreover, $|d_i - \bar{d}$ is the information vector in Equation 21.1. The LBCDS construction algorithm greedily searches the dominators with the smallest $|d_i - \bar{d}$ values. Based on this greedy criterion, the algorithm can output a CDS with a small *p-norm* value.

21.3.3 Load-Balanced Allocation of Dominatees

Constructing an LBCDS is the foundation to solve the LBAD problem. In this section, we introduce how to use an existing LBCDS for LBADs.

21.3.3.1 Terminologies

In a traditional/naive way, such as the work in [10], each dominatee sets its data forwarder to be the connected dominator with the smallest ID. Thus, the load-balance factor is not taken into account. In some environment, the dominator with the smallest ID, which is chosen by majority dominatees, probably has a heavier workload than the other dominators with a smaller number of dominatees. Therefore, the node degree cannot imply the potential workload precisely. In a WSN with a CDS as the VB, only the dominator and dominatee links contribute to the workload. Based on this observation, we define the following.

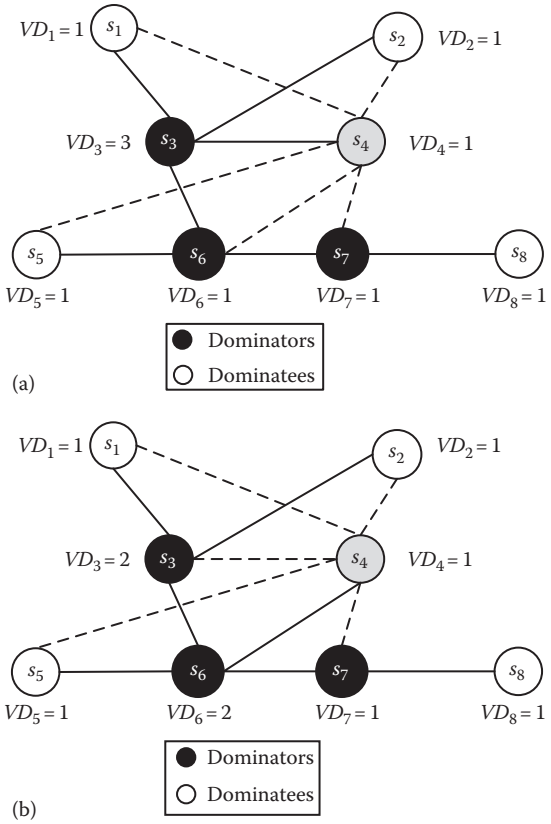


FIGURE 21.4 (a) A imbalanced dominate allocation scheme; (b) A load-balanced dominate allocation scheme.

Definition 21.4: *Valid degree* (*VD*)

For each dominatee s_i, VD_i is the number of its connected dominators. For each dominator s_j, VD_j is the number of its allocated dominatees.

Figure 21.4a and b illustrates an imbalanced and a balanced allocations of dominatees. The number beside each node *s* is its VD_i value. Using $|VD_i - \bar{d}|$ as the information vector , we still can use *p-norm* to measure the load-balance factor of the dominatee allocation scheme. Therefore, the *p-norm* value of the allocation scheme shown in Figure 21.4a is $\sqrt{8}$. Similarly, in Figure 21.4b, the *p-norm* value is $\sqrt{6}$ Clearly, $\sqrt{6} < \sqrt{8}$, which implies the allocation scheme shown in Figure 21.4b, is more load balanced than the scheme shown in Figure 21.4a.

Due to the instability of network topology, it is not practical to always allocate one dominatee to one dominator. In order to adapt to network topology change, a terminology *expected allocation probability* (*EAP*) is proposed as follows.

Definition 21.5: *EAP*

For each dominatee and dominator pair, there is an *EAP*, which represents the expected probability that the dominatee is allocated to the dominator.

The *EAP* value associated on each dominatee and dominator pair directly determines the load-balance factor of each allocation scheme. We conclude the properties of the *EAP* values as follows:

1. For each dominatee s (as shown in Figure 21.5),

$$\sum_{j=1}^{|NE(s_i)|} EAP_{ij} = 1 \tag{21.2}$$

   where
   $NE(s_i)$ is the set of neighboring dominators of dominatee s_i
   $|NE(s_i)|$ is the cardinality of set $NE(s_i)$

2. The most load-balanced allocation scheme, which is obtained when the expected numbers of allocated dominatees of all the dominators are the same, can be formulated as follows:

$$EAP_{i1} \times VD_1 = \cdots = EAP_{i|NE(s_i)|} \times VD_{|NE(s_i)|} \tag{21.3}$$

An example about how to calculate *EAP* values is shown in Figure 21.6. The gray nodes, that is, s_6 in Figure 21.6a and s_4 in Figure 21.6b, are dominatees connected to more than one dominator. The numbers shown on the links are the *EAP* values of each dominatee and dominator pair. If a dominatee only connects to one dominator, the *EAP* value associated with the pair is equal to X. Otherwise, Equation 21.3 can be used to calculate the *EAP* values of all connected dominator links.

As mentioned in Definition 21.4, the degree of each dominator is not a good indicator of workload. Hence, after allocating dominatees through the *EAP* scheme, the information vector in Equation 21.1 is the summation of the *EAP* values of dominator s_j minus the expected number of allocated dominatees of each dominator, which is formulated by $\sum_{i=1}^{ND(s_j)} |EAP_{ij} - \bar{p}|$, where $\bar{p} = (n - M)/M$ representing the expected number of allocated dominatees of each dominator. In Figure 21.4, $\bar{p} = 5/3$. Therefore, the *p-norm* value of the allocation scheme shown in Figure 21.4a is $\sqrt{1.16}$. Similarly, the *p-norm* value of the allocation scheme shown in Figure 21.4b is $\sqrt{0.57}$. Apparently, $\sqrt{0.57}$ is smaller than $\sqrt{1.16}$, which means that the allocation scheme in Figure 21.4b is more load balanced than the scheme in Figure 21.4a. There are two reasons to have a very small *p-norm* value in Figure 21.4b. First, an LBCDS is used. Second and more important, we adopt the probability-based dominatee allocation scheme.

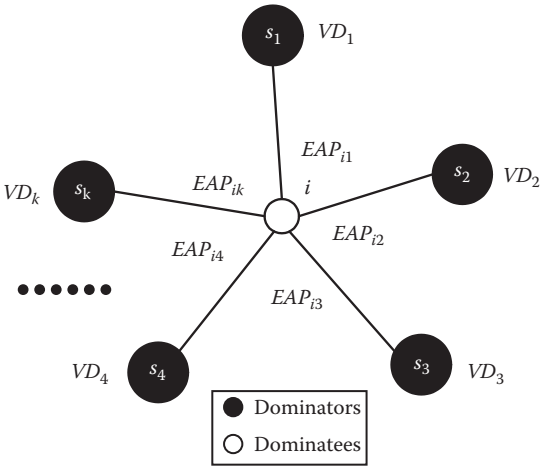


FIGURE 21.5 Property of *EAP*.

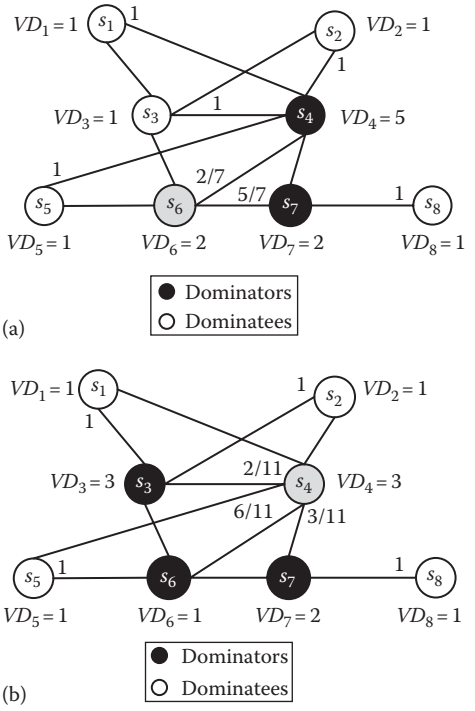


FIGURE 21.6 (a) Calculating EAP with the dominate connecting to two dominators; (b) Calculating EAP with the dominate connecting to three dominators.

The allocation criterion is based on making the expected number of allocated dominatees of each dominator the same. The criterion implies that, on average, the expected number of allocated dominatees of all the dominators is the same. If every dominatee has the same amount of data to be transferred through the allocated dominator at a fixed data rate, then the probability-based allocation scheme can achieve the maximized load balance on the expected workload among dominators.

21.3.3.2 Algorithm Description

The allocation system starts from finding an LBCDS using the aforementioned LBCDS construction algorithm. Then the *EAP* value is calculated for each dominatee and dominator pair. *EAP* only indicates the probability the dominatee will be assigned to the dominator for each dominator and dominatee pair. Thus, the final step is to allocate the dominatees to the dominators. We use the stochastic allocation, which is that a dominatee is randomly assigned to an adjacent dominator based on the *EAP* value.

Figure 21.6b shows an example about how to perform the stochastic dominatee allocation. In Figure 21.6b, only dominatee s_4 connects to more than one dominator and its associated *EAP* values are $EAP_{43} = 2/11 = 0.18$, $EAP_{46} = 6/11 = 0.55$, and $EAP_{47} = 3/11 = 0.27$. Dominatee s_4 generates a random number $\delta = 0.358$. If $\delta \in [0, 0.18]$, s_4 chooses dominator s_3; else, s_4 chooses dominator s_6 if $\delta \in (0.18, 0.73]$; otherwise, s_4 chooses dominator s_7 if $\delta \in (0.73, 1]$. Since $\delta = 0.358$, dominatee s_4 is assigned to dominator s_6.

Each time a dominatee that is connected to more than one dominator wants to send data, it must redo the last step to pick a proper dominator based on the *EAP* probability and then forward its data.

In Section 21.3.2, the detailed description of how to construct an LBCDS is introduced. Additionally, the third step, allocating the dominatees to the dominators, is a trivial process as we just explained. In the rest of this section, we design two algorithms to implement the second step, namely, how to calculate the *EAP* value for each dominator and dominatee pair. We introduce the centralized algorithm first.

21.3.3.3 Centralized Algorithm

We propose a constrained nonlinear programming scheme to solve the LBAD problem. The essence of allocating dominatees is to achieve maximum load balance among dominators. We use the *p-norm* value to measure the load-balance factor. Consequently, the objective of the optimization problem is to minimize the *p-norm* value of the dominatee allocation scheme. In addition, the constraint is to guarantee Property 1 of *EAP* values (Equation 21.2). To conclude, the optimization problem is formulated as follows:

$$\begin{aligned}
&\text{Minimize: } |EAP|_p = \sum_{j=1}^{M} \left(\sum_{i=1}^{|ND(s_j)|} |EAP_{ij} - \bar{p}| \right)^2 \\
&\text{Subject to: for dominatee}_i, \sum_{j=1}^{|ND(s_j)|} EAP_{ij} = 1 \\
&\text{Where: } \quad 0 \le EAP_{ij} \le 1
\end{aligned} \tag{21.4}$$

The centralized algorithm is shown in Algorithm 21.2.

The objective function in Equation 21.4 is the *p-norm* value on *EAP* values. The constraint states that the sum of the *EAP* values on each dominatee is equal to 1, which is the first property of *EAP* (Equation 21.2). The centralized algorithm can guarantee to find the optimal solution. However, solving the nonlinear programming is too time and energy consuming. If precision is the major concern, we can solve the nonlinear programming formulas at the BS. Nevertheless, if the energy and time are the primary concern, a distributed algorithm to find a near-optimal solution is preferred. We, therefore, propose the distributed algorithm as follows.

Algorithm 21.2: LBAD centralized

Require: A WSN represented by a graph $G = (V, E)$; an LBCDS: $G[D] = (D, E')$.

1: Solve the constrained nonlinear programming formulated in Equation 21.4. Let EAP_{ij} be the optimal solution of the nonlinear programming.
2: **for** each dominatee s_i **do**
3: Generate a number δ between 0 and 1
4: **if** $\delta \in \left[\sum_{j=0}^{k-1} EAP_{ij}, \sum_{j=0}^{k} EAP_{ij} \right]$, where $0 < k \le NE(s_i)$ **then**
5: mark the link between dominatee s_i and dominator s_k black
6: **end if**
7: **end for**
8: **return** all black links.

21.3.3.4 Distributed Algorithm

The objective of the LBAD problem is to find a load-balanced dominatee allocation scheme. The most load-balanced allocation scheme is that the expected number of allocated dominatees of all the dominators is the same, which is formulated in Equation 21.3. Additionally, we guarantee Property *X* of *EAP* values (Equation 21.2). By listing all the equations, we can solve them to get EAP_{ij} of each connected dominatee s_i and dominator s_j, which is formulated as follows:

$$\begin{aligned}
&EAP_{i1} : EAP_{i2} : \ldots : EAP_{i|ND(s_i)|} \\
&= VD_2 \times VD_3 \times \cdots \times VD_{|ND(s_i)|} : \ldots : \\
&\prod_{j=1, j \ne j}^{|ND(s_i)|} VD_j : \ldots : VD_1 \times VD_2 \times \cdots \times VD_{|ND(s_i)|-1}
\end{aligned} \tag{21.5}$$

Therefore, the distributed LBAD problem can be transformed to calculate the *EAP* value of each dominatee locally. The dominatee stochastic allocation step is the same as the centralized algorithm.

The distributed algorithm is a localized two-phase algorithm where each node only needs to know the connectivity information within its 1-hop-away neighborhood. All the nodes get the *VD* values by broadcasting messages to all its neighbor nodes and then store the values locally. Each dominatee calculates the *EAP* values using Equation 21.5.

The pseudocode is given in Algorithm 21.3. We call it *LBAD-distributed* algorithm.

We use the following terms in Algorithm 21.3:

VD_k: The *VD* value of each node s_k
$ND(s_k)$: The set of neighboring dominatees of dominator s_k
$|ND(s_k)|$: The number of the nodes in set $ND(s_k)$
$NE(s_k)$: The set of the neighboring dominators of dominatee s_k
$|NE(s_k)|$: The number of the nodes in set $NE(s_k)$
EAP_{ij}: The *EAP* value of each connected dominatee s_i and dominator s_j pair

Each node s_i maintains the following data structures:

1. s_i's *ID*, initialized to 0.
2. The dominator/dominatee flag *f*: 1 means dominator, and 0 means dominatee. It is initialized to 0.
3. $|ND(s_i)|$, if s_i is a dominator; $|NE(s_i)|$, if s_i is a dominatee, initialized to 0.
4. Neighboring dominator/dominatee lists. A list contains a dominator/dominatee's *ID*, its *VD* value, and EAP_{ij}, initialized to À.

Initially, each node initializes its data structures and broadcasts a *hello* message containing its *ID*, *VD*, and *f* to its 1-hop neighbors to exchange neighbors' information. All the nodes run the following:

- For any dominator s_i, upon receiving a *hello* message from node s_j: if s_j is a dominator, ignore the message. If s_j is a dominatee, update $|ND(s_i)|$ and dominatee s_j's *ID* and *VD* value in the neighboring dominatee list of the dominator s_i.
- For any dominatee s_i, upon receiving a *hello* message from node s_j: if s_j is a dominatee, ignore the message. If s_j is a dominator, update $|NE(s_i)|$ and dominator s_j's *ID* and *VD* value in the neighboring dominator list of the dominatee s_i. Calculate and store EAP_{ij} based on the *VD* values stored in the neighboring dominator list using Equation 21.5.

Algorithm 21.3: LBAD distributed

1: ***Initialization Phase***:
2: For each dominatee s_i, get the number of neighboring dominators (denoted by $|NE(s_i)|$) and store locally.
3: For each dominator s_j, get the number of neighboring dominatees (denoted by $|ND(s_j)|$) and store locally.
4: ***Allocation Phase***:
5: For each dominatee s_i, calculate its neighboring dominators' EAP_{ij} by the following formula:
6: $$EAP_{i1} : EAP_{i2} : \ldots : EAP_{i|ND(s_i)|} = VD_2 \times VD_3 \times \cdots \times VD_{|ND(s_i)|} : \ldots :$$
$$VD_1 \times VD_2 \times \cdots \times VD_{|ND(s_i)|-1} = \prod_{j=1, j \neq j}^{|ND(s_i)|} VD_j$$
7: A dominatee is assigned to an adjacent dominator based on the above calculated EAP_{ij} value.

The distributed algorithm is a two-phase algorithm. The first phase is the initialization phase, where all the nodes get their neighborhood information and update their own data structure locally. The message complexity of this phase is $O(n)$, since each node only needs to communicate with its 1-hop neighbors. In practice, it is hard to decide when the initialization phase completes. Hence, we set a timer. If the timer expires, the second phase, allocation phase, starts to work. In the allocation phase, every dominatee calculates the *EAP* values of its connected dominators using Equation 21.5. We only use 1-hop-away neighborhood information to calculate the *EAP* values locally. Therefore, it is an easy and efficient algorithm. In other words, the time complexity of the allocation phase is $O(1)$. Nevertheless, only using the 1-hop-away neighborhood information to calculate the *EAP* values may lead us to find a local optimal solution instead of a global optimal solution. Note that we sacrifice the storage space to save the time complexity in the distributed algorithm. Every node stores neighbor's dominatee/dominator's list locally.

21.3.3.5 Analysis

Based on the assumptions mentioned in Section 21.3.1.1, n sensors are i.i.d. in a square with area size $A = cn$. The communication range of each sensor is 1. Thus, we denote the unit circle associated with each sensor s_i by c_i. According to the network model, the following lemma can be proved.

Lemma 21.1

For any unit circle c_i, let the random variable Z_i denote the number of the sensors within it. Then, the probability that c_i contains more than ln n sensors is no greater

$$\frac{\exp((\exp(\gamma)-1)\times(\pi/c))}{\exp(\gamma\times\ln n)},\quad \text{i.e., } \Pr[Z_i > \ln n] \le \frac{\exp((\exp(\gamma)-1)\times(\pi/c))}{\exp(\gamma\times\ln n)},\quad \text{for any } \gamma > 0$$

Proof: Since all the sensors are i.i.d., the number of the sensors in c_i satisfies the binomial distributions with parameters $(n, \pi/A)$ [48]. Applying the Chernoff bound and for any $\gamma > 0$, we have

$$\begin{aligned}
\Pr[Z_i > \ln n] &\le \frac{E[\exp(\gamma Z_i)]}{\exp(\gamma\ln n)} = \frac{[1+(\exp(\gamma)-1)(\pi/A)]^n}{\exp(\gamma\ln n)} \\
&\le \frac{\exp((\exp(\gamma)-1)(\pi/A)\times n)}{\exp(\gamma\ln n)} \quad (\text{by } 1+x\le e^x) \\
&= \frac{\exp((\exp(\gamma)-1)\times(\pi/c))}{\exp(\gamma\times\ln n)} \quad (\text{by } A = cn)
\end{aligned}$$

■

From Lemma 21.1, the probability that a unit circle contains more than ln n sensors is zero when $n \to \infty$. Hence, we can use ln n as the upper bound of the number of the sensors in a unit circle in our analysis. Then, we can get the following theorem that states the upper and lower bounds of the *p-norm* of the distributed algorithm.

Theorem 21.1

The upper bound of the p-norm value in the distributed algorithm is $M(\ln n - 1 - (2\pi/(3\times c)) - (\sqrt{3}/(2\times c)) - ((n-M)/M))^2$. The lower bound of the p-norm value in the distributed algorithm is $M((\pi/c)-((n-M)/M))^2$.

Proof: According to Definition 21.3, the *p-norm* value of the allocated dominatees with *EAP* is formulated as

$$\left|\mathrm{EAP}\right|_p = \sum_{j=1}^{M}\left(\left|\sum_{i=1}^{|ND(s_j)|}\mathrm{EAP}_{ij} - \overline{p}\right|\right)^2$$

The *p-norm* value depends on how many dominatees are adjacent to each dominator, namely, $ND(s_j)$ in the formula. So the upper bound and lower bound of the number of neighboring dominatees are the key challenges to analyze the performance ratio. The upper bound of the number of the sensors in a unit circle occurs when there is only one dominator in a unit circle and all the other dominatees connect to the dominator, and then we can get the upper bound of the *p-norm* value. However, we are considering a CDS. In order to maintain the connectivity, at least two dominators must be within each other in the transmission range, namely, in one unit circle. Figure 21.7 illustrates the situation. There are some overlapped areas shown by the gray in the figure. The gray area is a sector with 120°, and its size is $(2\pi/3) - (\sqrt{3}/2)$. Because all the sensors are i.i.d., the expected number of the sensors is $\frac{\frac{2\pi}{3} - \frac{\sqrt{3}}{2}}{c \times n} \times n = \frac{2\pi}{3 \times c} - \frac{\sqrt{3}}{2 \times c}$. Therefore, the upper bound of the *p-norm* value is

$$\left|\mathrm{EAP}\right|_p \geq \sum_{j=1}^{M}\left(\ln n - 1 - \frac{2\pi}{3 \times c} - \frac{\sqrt{3}}{2 \times c} - \frac{n-M}{M}\right)^2$$

$$= \left(\ln n - 1 - \frac{2\pi}{3 \times c} - \frac{\sqrt{3}}{2 \times c} - \frac{n-M}{M}\right)^2$$

The lower bound of the number of the sensors in each unit circle can be estimated by $(\pi/(c \times n)) \times n = \pi/c$. Hence, the lower bound of the *p-norm* value is

$$\left|\mathrm{EAP}\right|_p \geq \sum_{j=1}^{M}\left(\frac{\pi}{c} - \frac{n-M}{M}\right)^2 = M\left(\frac{\pi}{c} - \frac{n-M}{M}\right)^2$$ ■

21.3.4 Performance Evaluation

In this section, we validate our proposed algorithms through implementing the CDS-based data aggregation protocol and CDS-based data collection protocol.

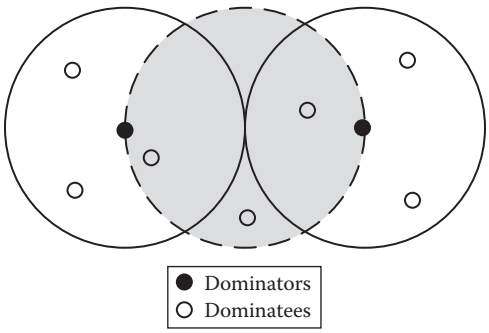


FIGURE 21.7 Connectivity guarantee.

21.3.4.1 Scenario 1: Data Aggregation Communication Mode

In this subsection, we evaluate our proposed algorithms by comparing our work with the work in [10], in which each dominatee chooses the neighboring dominator of the smallest ID as its parent. Four different schemes are implemented:

- LBCDSs with LBAD, noted by LB-A
- LBCDSs with the smallest ID dominator selection scheme, noted by LB-ID
- MIS-based CDSs with LBAD, noted by MIS-A
- MIS-based CDSs with the smallest ID dominator selection scheme, which is the work in [10], noted by MIS-ID

We compare them in terms of the *p-norm* value; network lifetime, which is defined as the time duration till the first dominator's energy is depleted; and the standard derivation of the residual energy of all the nodes.

21.3.4.2 Simulation Environment

We build our own simulator where all the nodes have the same transmission range (10 m). n nodes are randomly deployed in a fixed area of 100 m × 100 m. n is incremented from 200 to 450 by 50. For a certain n, 100 instances are generated. The results are averaged among 100 instances. Moreover, we use the CDS-based data aggregation as the communication mode.

21.3.4.3 Simulation Results

Figure 21.8 shows the *p-norm* values of the four schemes. The *X*-axis represents the number of the nodes. The *Y*-axis represents the *p-norm* values of the four schemes. With the increase in the number of the sensor nodes, the *p-norm* values increase correspondingly. This is because when the number of the nodes increases, we need more nodes to build an LBCDS. According to Definition 21.1, more nodes imply more sum subitems, so the *p-norm* values increase. As mentioned in Section 21.3.1.1, the smaller the *p-norm* value is, the more load balanced the scheme is. From Figure 21.9a, we know that the MIS-ID scheme has the largest *p-norm* values while the LB-A scheme has the smallest *p-norm* values. This is because the MIS-ID scheme did not consider the load-balance factor when building a CDS and allocating dominatees to dominators. To clearly see the *p-norm* values of the LB-A scheme, we redraw the curve using a smaller scale in Figure 21.8b for LB-A. Additionally, Figure 21.8 demonstrates that the LBAD algorithm fits for any type of CDSs. The MIS-A scheme still has smaller *p-norm* values than the other two schemes using the smallest ID allocation scheme, namely, LB-ID and MIS-ID.

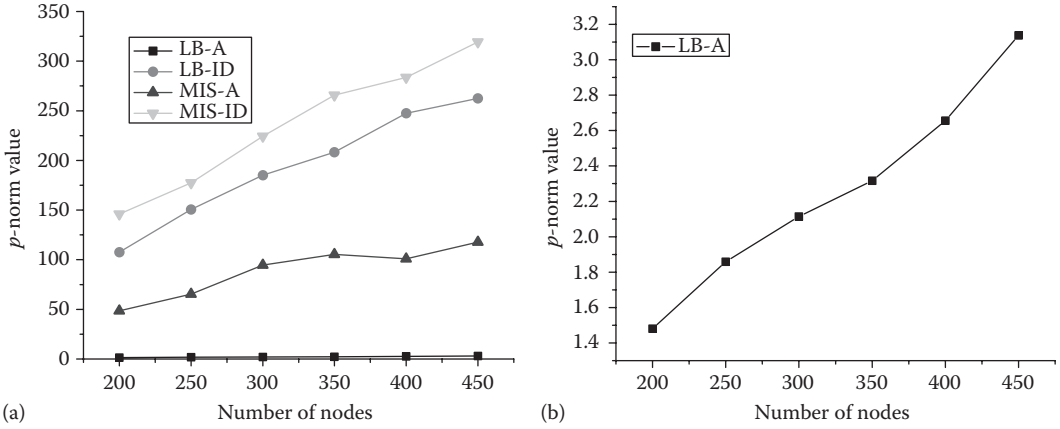


FIGURE 21.8 (a) *p*-norm values of the four schemes when number of nodes increase; (b) *p*-norm values of LB-A when number of nodes increase.

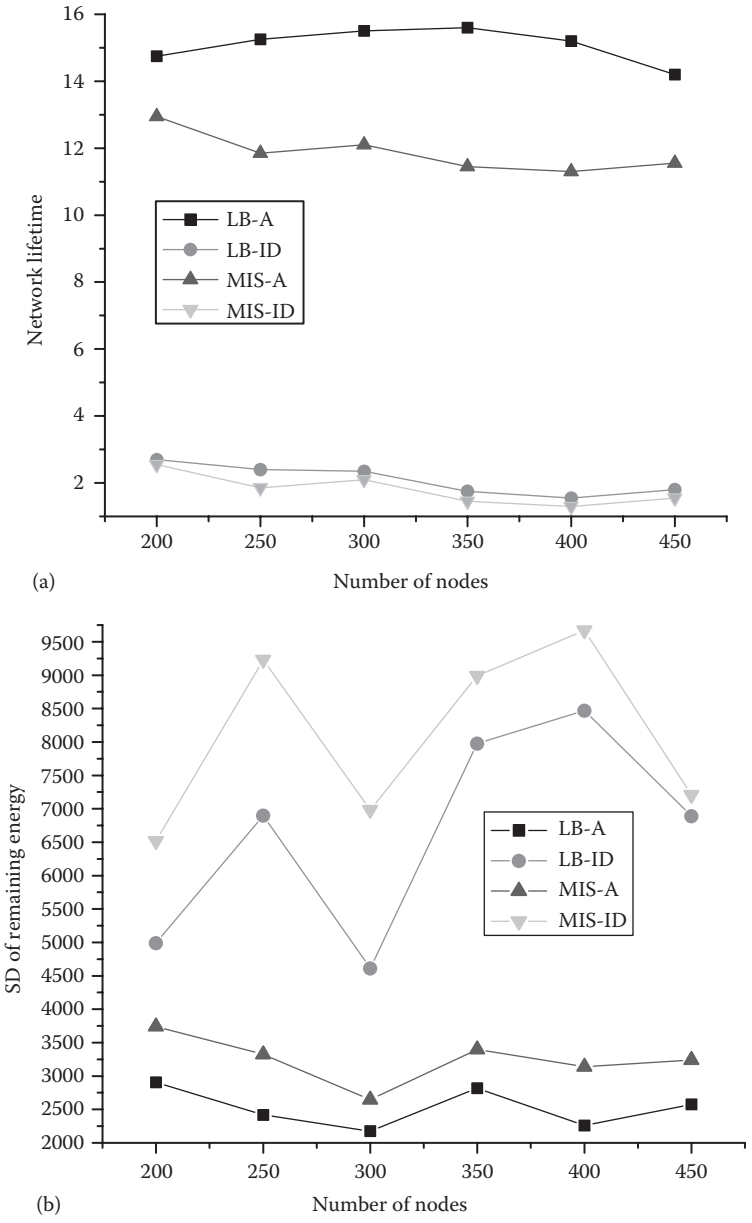


FIGURE 21.9 Simulation results for a square area of 100 m × 100 m: the node transmission range is 10 m, and the number of nodes changes from 200 to 450—(a) network lifetime and (b) SD of residual energy.

Figure 21.9a shows the network lifetime of the four schemes. The *X*-axis represents the number of the nodes, and the *Y*-axis represents the network lifetime of the four schemes. The simulated energy consumption model is that every node has the same initial 100 units of energy. Receiving a packet consumes 1 unit of energy, while transmitting a packet consumes 2 units of energy. From Figure 21.9a, we can see the load-balanced schemes (LB-A and MIS-A) prolong network lifetime by 80% compared to nonbalanced schemes (LB-ID and MIS-ID). When the number of the node increases, there is no obvious increase or decrease trend of network lifetime, since the locality of the network topology mainly decides

the network lifetime. A network topology is generated randomly, so we cannot control the locality of the network. From Figure 21.9a, we also find the network lifetimes of imbalanced schemes (LB-ID and MIS-ID) are close to 1, 2, or 3. This is because some critical smaller ID dominators are connected to many dominatees. They deplete energy very quickly, and then the whole network is disconnected.

Figure 21.9b shows the standard derivation of the residual energy of the four schemes. The *X*-axis represents the number of the nodes. The *Y*-axis represents the standard derivation of the average residual energy of all the nodes. We use the standard derivation here to observe whether the residual energy is balanced or not. From Figure 21.9b, we know the balanced schemes (LB-A and MIS-A) have more balanced residual energy than imbalanced schemes (LB-ID and MIS-ID). This is because we consider the load-balance factor when building a CDS and allocating dominatees to dominators.

The simulation results can be summarized as follows:

- The LB-A scheme always has the best performance according to the *p-norm* value, network lifetime, and the standard derivation of residual energy. The results demonstrate building an LBCDS and then load-balancedly allocating dominatees can increase network lifetime significantly.
- The load-balanced dominatee allocation algorithm can be applied to not only LBCDSs but also imbalanced CDSs to achieve good performances. The LB-A and MIS-A schemes have better performances over the LB-ID and MIS-ID schemes among all measures, namely, the *p-norm* value, network lifetime, and the standard derivation of residual energy.
- The balanced schemes (LB-A and MIS-A) have better scalability than the imbalanced schemes (LB-ID and MIS-ID).

21.3.4.4 Scenario 2: Data Collection Communication Mode

In this subsection, we evaluate our proposed algorithms by comparing our work denoted by LBCDS with the tree-based CDS construction work [10], denoted by tree based, and the subtraction-based CDS construction work [31] denoted by subtraction. We compare them in terms of the number of dominators, network lifetime that is defined as the time duration till the first dominator's energy is depleted, and the average residual energy among all the dominators.

21.3.4.5 Simulation Environment

We build our own simulator where all the nodes have the same transmission range (10 m). n nodes are randomly deployed in a fixed area of 100 m × 100 m. n is incremented from 200 to 700 by 100. For a certain n, 100 instances are generated. The results are averaged among 100 instances. Moreover, we use the CDS-based data collection as the communication mode. The simulated energy consumption model is that every node has the same initial 2000 units of energy. Receiving and transmitting a packet both consume 1 unit of energy.

21.3.4.6 Simulation Results

From Figure 21.10a, we can see that, with the increase in the number of sensor nodes, the number of dominators almost keeps stable for all the three algorithms (tree-based, LBCDS, and subtraction). This is because the area of the network deployed region keeps fixed. The result implies that, if the network deployed area keeps unchanged, the density of the WSN has little effect on the size of the constructed CDS.

Despite few changes in the number of dominators, different dominatee allocation schemes do affect network lifetime as shown in Figure 21.10b. From Figure 21.10b, we know that the network lifetime decreases for all algorithms with the number of nodes increasing. This is because there are more neighbors of each dominator in a more and more crowded network. As shown in Figure 21.10a, the number of dominators almost remains unchanged. It implies that with the number of neighbors increasing, the collected data on each dominator become heavier. Hence, the network lifetime decreases for all

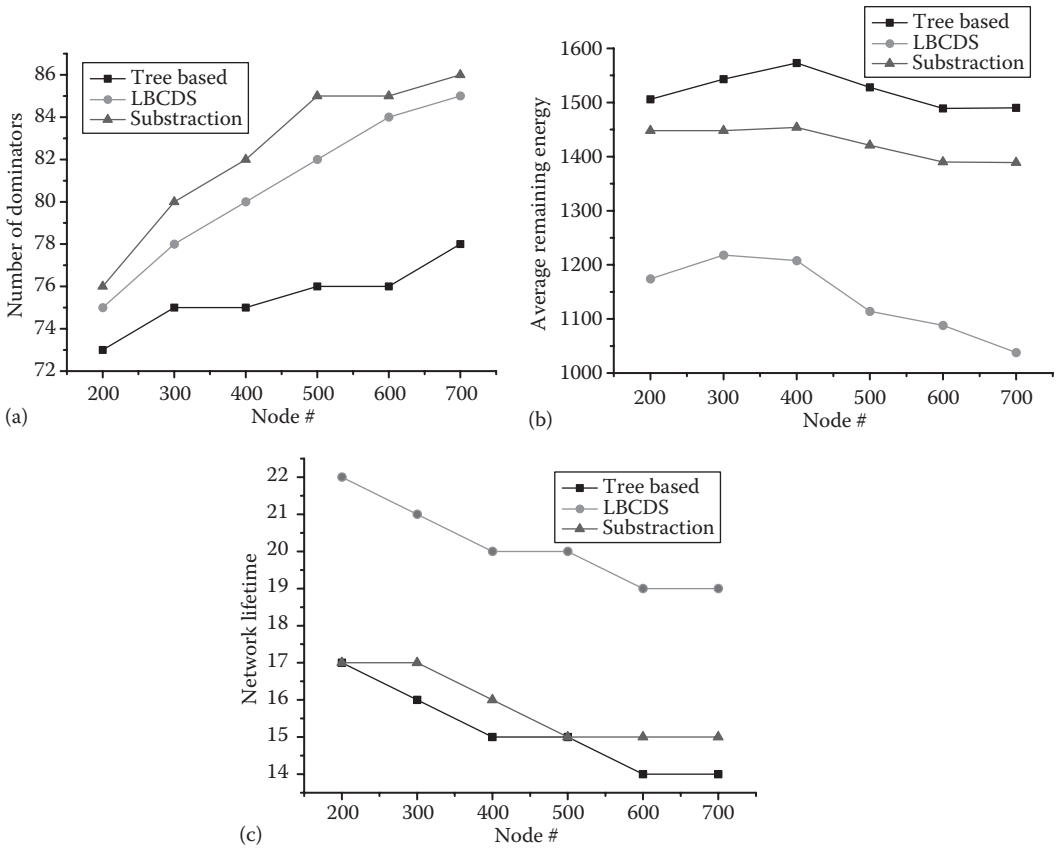


FIGURE 21.10 Simulation results for a square area of 100 m × 100 m: the node transmission range is 10 m, and the number of nodes changes from 200 to 700—(a) the number of dominators, (b) average residual energy of dominators, and (c) network lifetime.

the three algorithms. Additionally, we can see LBCDS outperforms than tree based and subtraction. Furthermore, LBCDS prolongs network lifetime by 31% on average compared with tree based and by 26% on average compared with subtraction. The results demonstrate that load-balancedly allocating dominatees to dominators can improve network lifetime notably. On the other hand, subtraction outperforms than tree based, since the size of the constructed CDS by subtraction is larger than the size of the constructed CDS by tree based.

Figure 21.10c shows the average residual energy among all the dominators of the three algorithms. With the increase in the number of nodes, the average residual energy decreases for all algorithms. As the WSN becomes denser and denser, the dominators collect more and more data in the WSN. From Figure 21.10c, we know that LBCDS has less average residual energy than tree based and subtraction. This is because LBCDS considers the load-balance factor when building a CDS and allocating dominatees to dominators. Thus, the lifetime of the whole network is extended, which means that the average residual energy of the dominators is less than tree based and subtraction. Additionally, subtraction has less average residual energy than tree based. Since subtraction has longer network lifetime than tree based as shown in Figure 21.10b, subtraction has less residual energy than tree based. In summary, Figure 21.10 indicates that constructing an LBCDS can balance the energy consumption on each backbone node and make the lifetime of the whole network prolonged considerably.

21.4 LBVB under PNM

21.4.1 Network Model and Problem Definition

In this section, we first present the assumptions and then introduce the PNM. Subsequently, we formally define some terminologies. Finally, we give the problem definition and make some remarks about the proposed problem.

21.4.1.1 Assumptions

We assume a static connected WSN with the set of n nodes $= \{v_0, v_1, \ldots, v_{n-1}\}$. All the nodes in the WSN have the same transmission range. The *delivery ratio* γ_{ij} associated with each link connecting a pair of nodes v_i and v_j is available, which can be obtained by periodic hello messages or be predicted using link quality index (LQI) [49]. We also assume that the γ_{ij} values are fixed. This assumption is reasonable as many empirical studies have shown that LQI is pretty stable in a static environment [50]. Furthermore, no node failure is considered since it is equivalent to a link failure case. No duty cycle is considered either. We do not consider packet collisions or transmission congestion, which are left to the MAC layer.

21.4.1.2 Network Model

Under the PNM, we model a WSN as an undirected graph $(, , \Upsilon( ))$, where $$ is the set of n nodes, denoted by v_i, where $0 \le i < n$. i is called the node ID of v_i in the chapter. $$ is the set of lossy links. $\forall v_i$, $v_j \in$, there exists a link (v_i, v_j) in $$ if and only if (1) v_i and v_j are in each other's transmission range and (2) $\gamma_{ij} > 0$. For each link $(v_i, v_j) \in$, γ_{ij} indicates the probability that node v_i can successfully directly deliver a packet to node v_j, and $\Upsilon( ) = \{\gamma_{ij} | (v_i, v_j) \in , 0 < \gamma_{ij} \le 1\}$. We assume that the links are undirected (bidirectional), which means two linked nodes are able to transmit and receive information from each other with the same γ_{ij} value.

Because of the introduction of γ_{ij}, the traditional definition of the node neighborhood has changed. Hence, we first give the definition of the 1-hop neighborhood and then extend it to the h-hop neighborhood.

Definition 21.6: *One-hop neighborhood* ($_1(v_i)$)

$\forall v_i \in$, the 1-hop neighborhood of node v_i is defined as

$$N_1(v_i) = \{v_j \mid v_j \in V, \gamma_{ij} > 0\}$$

The physical meaning of 1-hop neighborhood is the set of the nodes that can be reached from node v_i via 1-hop neighbors with positive probability. In this chapter, we use $| _1(v_i)|$ to represent the cardinality of the 1-hop neighborhood set of node v_i.

Definition 21.7: *h-Hop neighborhood* ($_h(v_i)$)

$\forall v_i \in$, the h-hop neighborhood of node v_i is defined as

$$N_h(v_i) = N_{h-1}(v_i) \cup \left\{ v_k \mid \exists v_j \in N_{h-1}(v_i), v_k \in N_1(v_j), v_k \notin \bigcup_{i=1}^{h-1} N_i(v_i) \right\}$$

The physical meaning of the h-hop neighborhood is that the set of nodes that can be reached from node v_i by passing maximum h number of *lossy links* with positive probability.

21.4.1.3 Preliminary

Since load balance is the major concern of this work, the measurement of traffic load balance under PNM is critical to solve the load-balanced VB problem (LBVBP) construction problem. Hence, in this subsection, we first define a novel metric called *potential traffic load* to measure the potential traffic load on each node.

Without knowing the communication protocol, the number of neighboring nodes of a node (i.e., $|N_1(v_i)|$) is a potential indicator of the traffic load on each node. However, it is not the only factor to indicate the potential traffic load on each node in probabilistic WSNs. As we mentioned in Section 21.1.1, the less the γ_{ij} value, the more potential traffic load on v_j from v_i. Therefore, a more reasonable and formal definition of the potential traffic load is given as follows.

Definition 21.8: *Potential traffic load* (ι_i)

$\forall v_i \in$, the potential traffic load of v_i is defined as $\iota_i = \sum_{v_j \in N_1(v_i)} (1/\gamma_{ij})$

After knowing the *potential traffic load* of each node, how to measure load balance of a constructed VB is another challenge. We use *p-norm* to measure load balance in this chapter. The definition of *p-norm* is given as follows.

Definition 21.9: *p-Norm*

The p-norm of an $n \times 1$ vector $= (x_1, x_2, \ldots, x_n)$ is

$$|X|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

The authors in [47] stated that p-norm shows interesting properties for different values of p. If p is close to 1, the information routes resemble the geometric shortest paths from the sources to the sinks. For $p = 2$, the information flow shows an analogy to an electrostatics field, which can be used to measure the load balance among x_i. More importantly, the smaller the *p-norm* value, the more load balanced the interested feature vector . For simplicity, we use $p = 2$ in this chapter.

In this chapter, we use *potential traffic load* (Definition 21.8) as the feature vector . According to Definition 21.9, we define the *VB p-norm* as follows.

Definition 21.10: *VB p-norm* ($|\ |_p$)

For WSN (, , Υ( )), and a VB $= \{v_1, v_2, \ldots, v_m\}$. The *VB p-norm* of an $m \times 1$ vector $= (\iota_1, \iota_2, \ldots, \iota_m)$ is

$$|B|_p = \left(\sum_{i=1, v_i \in B}^{m} |\iota_i - \bar{\iota}|^p \right)^{1/p} \tag{21.6}$$

where

m is the cardinality of set

ι_i represents the potential traffic load of each node in set

$\bar{\iota} = \left(\sum_{j=1, v_j \in B}^{m} \iota_j \right) / m$ is the average potential traffic load on set

As we mentioned earlier, the smaller the *VB p-norm* value, the more load balanced the constructed VB. We still use the WSN shown in Figure 21.2 to illustrate how to use *VB p-norm* to measure the load balance of constructed VBs. Two different VBs (represented by the black nodes) for the same network are identified in Figure 21.2a and b, in which the VB shown in Figure 21.2a is a typical MCDS-based VB. According to Definition 21.8, in Figure 21.2a, $\iota_2 = 20.28$ and $\iota_3 = 8.55$. Hence, $\bar{\iota} = (\iota_2 + \iota_3)/2 = 14.42$. Therefore, based on Equation 21.6, the *VB p-norm* value of the VB shown in Figure 21.2a is $\sqrt{68.80}$. Similarly, we can get the *VB p-norm* value of the VB shown in Figure 21.2b, which is $\sqrt{34.68}$. Clearly, $\sqrt{34.68} < \sqrt{68.80}$, which implies that the VB shown in Figure 21.2b is more load balanced than the VB shown in Figure 21.2a.

Actually, if one dominatee is adjacent to more than one dominator, one of the adjacent dominators is chosen by the dominatee to perform data transmission. Hence, it is important to load-balancedly allocate dominatees to each dominator to further balance the traffic loads among each dominator. In a traditional/naive way [10], each dominatee is allocated to the neighboring dominator with the smallest ID. Obviously, the load-balance factor is not taken into account. In some environments, the dominator with the smallest ID, which is chosen by majority dominatees, tends to have heavier workload than the other dominators. Therefore, neither node ID nor *potential traffic load* can reflect the actual workload precisely. In a WSN with a CDS as the VB, only the dominator and dominatee links contribute to the actual traffic load. Based on this observation, we define the following concepts.

Definition 21.11: *Dominator partition* (*3*)

Dominator partition ($\mathcal{P}$). For a WSN represented by graph (, , Υ( )) and *aVB* = $\{v_1, v_2, \ldots, v_m\}$, m disjoint sets are identified of , that is, (v_1), (v_2), …, (v_m), such that

1. Each set (v_i) ($1 \le i \le m$) contains exactly one dominator v_i
2. $\bigcup_{i=1}^{m} P(v_i) = v$ and $P(v_i) \cap P(v_j) = \emptyset (1 \le i \ne j \le m)$
3. $\forall v_u \in$ (v_i) ($1 \le i \le m$) and $v_u \ne v_i$, (v_u, v_i) $\in$

A dominator partition is $\mathcal{P} = \{$ (v_i) $\mid v_i \in$, $1 \le i \le m\}$.

We also use the WSN shown in Figure 21.2 to explain the concept of *dominator partition*. Three different dominator partitions are shown in Figure 21.11, in which only dominator and dominatee links are presented in the figure. According to Definition 21.11, we have (v_3) = $\{v_2, v_4\}$, (v_6) = $\{v_1, v_5\}$, and (v_7) = $\{v_8\}$ for the partition 3 = { (v_3), (v_6), (v_7)} shown in Figure 21.11a. Without considering the *delivery ratio* on each dominator and dominatee link, it is obvious that the dominator partition shown in Figure 21.11b is the most unbalanced of the workloads on each dominator. Moreover, without further information, it is hard to reveal which partition is more balanced than the other shown in Figure 21.11a and c. According to the previous observations, we define the following concepts and metric to measure the load balance of a dominator partition.

Definition 21.12: *Authorized link set* ($_i$)

$\forall v_i \in$, the *authorized link set* of dominator v_i is the set of the dominator and dominatee links formed by nodes in (v_i), that is,

$$L_i = \left\{ (v_i, v_j) \mid v_j \in P(v_i), 1 \le i \le m \right\}$$

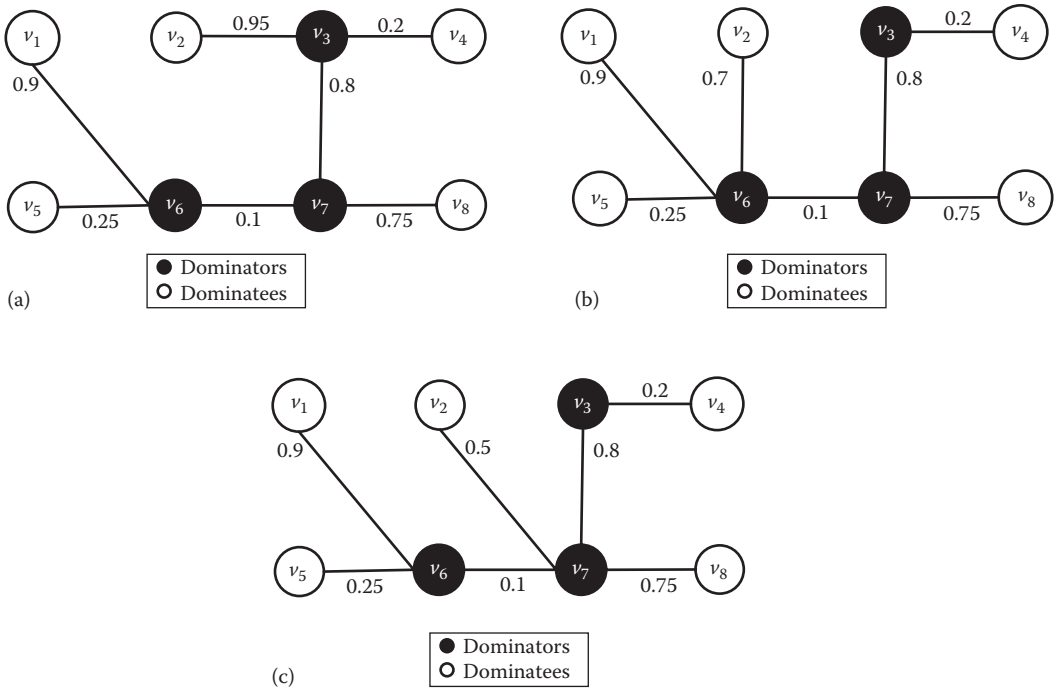


FIGURE 21.11 (a) A imbalanced dominator partition; (b) Another imbalanced dominator partition; (c) A load-balanced dominator partition.

As we have already known, l_i is only the indicator of the potential traffic load on each dominator v_i. The actual traffic load can be determined when a dominator partition is decided. In other words, the *authorized link set* $_i$ along with the corresponding *delivery ratio* of each link is an indicator of the actual traffic load on each dominator v_i. According to this observation, we give the following definition.

Definition 21.13: *Actual traffic load* (l_i)

$\forall v_i \in$, the actual traffic load of v_i is defined as $1_i = \sum_{(v_i, v_j) \in L(v_i)} (1/\gamma_{ij})$.

In this chapter, we use *partition p-norm* to measure the load balance of different dominator partitions, in which the *actual traffic load* l_i of each dominator v_i is used as the feature vector shown in Definition 21.9. The definition of the *partition p-norm* is given as follows.

Definition 21.14: *Partition p-norm* ($|3|_p$)

For a WSN represented by graph (, , Υ( )), a VB = {v_1, v_2, …, v_m}, and a dominator partition 3, the *partition p-norm* is

$$|3|_p = \left(\sum_{i=1, v_i \in B}^{m} \left| 1_i - \bar{1} \right|^p \right)^{1/p} \tag{21.7}$$

where $\bar{1} = \left(\sum_{j=1, v_j \in B}^{m} 1_i \right) / m$ is the average actual traffic load on set .

It is worth mentioning that the smaller the *partition p-norm* value, the more load balanced the dominator partition. Figure 21.11 illustrates unbalanced and balanced dominator partitions. We can use *partition p-norm* to reveal which partition is the most balanced one. In Figure 21.11a, the *actual traffic loads* of dominators are

$$l_3 = \frac{1}{\gamma_{32}} + \frac{1}{\gamma_{34}} = 6.05, \quad l_6 = \frac{1}{\gamma_{61}} + \frac{1}{\gamma_{65}} = 5.11$$
$$l_7 = \frac{1}{\gamma_{78}} = 1.33, \quad \bar{l} = \frac{l_3 + l_6 + l_7}{3} = 4.16$$

According to Equation 21.7, we obtain the *partition p-norm* of the dominator partition shown in Figure 21.11a, which is $\sqrt{12.48}$. In the same way, the *partition p-norm* values of the dominator partitions shown in Figure 21.11b and c are $\sqrt{14.33}$ and $\sqrt{1.99}$, respectively. Clearly, $\sqrt{1.99} < \sqrt{12.48} < \sqrt{14.33}$, which implies the dominator partition shown in Figure 21.11c is the most balanced partition among all three dominator partitions shown in Figure 21.11. The partition shown in Figure 21.11b is the most unbalanced, which further confirms the observation we mentioned earlier.

21.4.1.4 Problem Definition

Finally, we are dedicated to construct an LBVB. The physical meaning of an LBVB is that the constructed VB has the minimum *VB p-norm* and minimum *partition p-norm* values under PNM. The formal definition of the LBVBP is as follows.

Definition 21.15: ***LBVBP in probabilistic WSNs***

For a WSN represented by graph (, , ϒ( )), the *LBVBP* is to find a minimum-sized node set ⊆ and a dominator partition *3*, such that

1. [ ] = (, ′), where ′ = {*e*|*e* = (*u*, *v*), *u* ∈ , *v* ∈ , (*u*, *v*) ∈ }, is connected
2. ∀*u* ∈ and *u* ∉ , ∃*v* ∈ , such that (*u*, *v*) ∈
3. Minimize{| |$_p$, |3|$_p$}

The LBVBP construction problem is NP hard, since it still belongs to the MCDS problem. Based on Definition 21.15, the key issue of the LBVBP construction problem is to seek a VB that satisfies multiple constraints, that is, the minimum size, the minimum *VB p-norm*, and the minimum *partition p-norm*. In reality, the multiple objectives are potentially in conflict. Conflicting objectives result in a set of compromised solutions, known as the Pareto-optimal set. Since none of the solutions in this set can be considered as better than the others with respect to all the objectives, the goal of the multiobjective optimization problem is to find as many as Pareto-optimal solutions as possible. The multiobjective genetic algorithm (MOGA) is a powerful tool to fulfill the previously mentioned requirements due to its inherent parallelism and its ability to exploit the similarities among solutions by recombination [51]. Hence, in the following, a novel MOGA, named LBVBP-MOGA, is proposed to solve the LBVBP construction problem.

21.4.2 LBVBP-MOGA

This section provides the detailed design of the LBVBP-MOGA. The first part provides an overview of the MOGAs and some notions to be used in the chapter. The second part introduces the LBVBP-MOGA step by step. The last part analyzes the convergence of the proposed LBVBP-MOGA.

21.4.2.1 Overview of MOGAs

21.4.2.1.1 Multiobjective Problem Definitions and Overview

Similar definitions for multiobjective problems (MOPs) are given in related literature [51]. We merely introduce three important concepts in this subsection. Without loss of generality, we only consider minimization problems in this chapter, and it is easy to convert a maximization problem into a minimization problem.

Definition 21.16: ***MOP***

In general, an MOP minimizes $\boldsymbol{F}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x}))$ subject to $g_i(\boldsymbol{x})$, $i = 1, \ldots, m$, $\boldsymbol{x} \in \Omega$ (Ω is the *decision variable space*). An MOP solution minimizes the components of a *objective vector* $\boldsymbol{F}(\boldsymbol{x})$, where $\boldsymbol{x} = (x_1, \ldots, x_n)$ is an n-dimensional *decision variable vector* from some universe Ω.

Because of multiple objectives and constraints, a key concept in determining a set of MOP solutions is *Pareto optimality*. To ensure understanding and consistency, we mathematically define *Pareto dominance* and *Pareto optimality* as follows.

Definition 21.17: ***Pareto dominance*** **($\preccurlyeq$)**

A vector $\boldsymbol{u} = (u_1, \ldots, u_k)$ is said to dominate $\boldsymbol{v} = (v_1, \ldots, v_k)$ (denoted by $\boldsymbol{u} \preccurlyeq \boldsymbol{v}$) if and only if $\boldsymbol{u}$ is partially less than $\boldsymbol{v}$, that is, $\forall i \in \{1, \ldots, k\}, u_i \leq v_i \wedge \exists i \in \{1, \ldots, k\}, u_i \leq v_i$.

Definition 21.18: ***Pareto optimality***

A solution $x \in \Omega$ is said to be *Pareto optimal* with respect to $\boldsymbol{\Omega}$ if and only if there is no $x' \in \boldsymbol{\Omega}$ for which $\boldsymbol{F}(\boldsymbol{x}') \preccurlyeq \boldsymbol{F}(\boldsymbol{x})$.

The set of all Pareto-optimal decision vectors is called the *Pareto-optimal set*. The corresponding set of the objective vectors is called the *Pareto-optimal front*.

21.4.2.1.2 GA Overview

GAs work with a population of *chromosomes*, each representing a possible solution to a given problem. Each chromosome is assigned a *fitness score* according to how good a solution to the problem it is. The fittest chromosomes are given opportunities to *reproduce*, by *crossover* with other chromosomes in the population. This produces new chromosomes as *offsprings*, which share some features taken from each *parent*. The least chromosomes of the population are less likely to be selected for reproduction, and so they *die out*. A whole new population of possible solutions is thus produced by selecting the best chromosomes from the current *generation* and mating them to produce a new set of chromosomes. In this way, over many generations, good characteristics are spread throughout the population. If the GA has been designed well, the population will converge to an optimal solution to the problem. The overview of GAs is pictured in Figure 21.12.

21.4.2.1.3 MOGA Overview

MOGA is a recently developed algorithmic tool to solve MOPs. MOGAs are very attractive because they have the ability to search partially ordered spaces for several alternative trade-offs. Additionally, an MOGA can track several solutions simultaneously via its population.

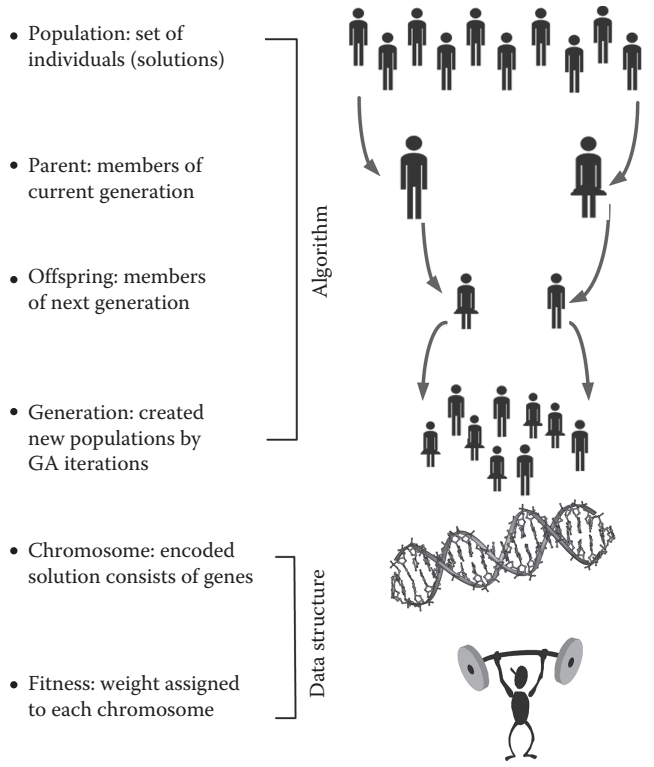


FIGURE 21.12 Genetic algorithm overview.

According to the definition of Pareto optimality, we can discover distinct differences between MOPs and single-objective problems (SOPs). The solutions in the objective space of SOPs are scalar numbers, and their relations have only two possibilities: less than and greater than or equal. However, the solutions of MOPs are vectors and their relations have three possibilities: $\boldsymbol{u} \preccurlyeq \boldsymbol{v}$, $\boldsymbol{v} \preccurlyeq \boldsymbol{u}$, and nondominated. This difference requires that MOGAs have more complicated fitness assignment rules and replacement rules.

21.4.2.1.4 Design of LBVBP-MOGA

In this section, we design and explain the MOGA to solve the LBVBP called LBVBP-MOGA step by step.

21.4.2.1.5 Representation of Chromosomes

A chromosome is a possible solution of the LBVBP. Hence, when designing the encoding scheme of chromosomes, we need to identify dominators and dominatees in a chromosome and a *dominator partition* in a chromosome as well. For convenience, the set of neighboring dominators of each dominatee $v_s \in$ is denoted by $(v_s) = \{v_r | v_r \in$, $(v_r, v_s) \in \}$. In the proposed LBVBP-MOGA, each node is mapped to a gene in the chromosome. A gene value g_i indicates whether the sensor represented by this gene is a dominator or not. If the sensor is a dominator (i.e., $\forall v_i \in$), the corresponding gene value is $g_i = 1$. Otherwise, the corresponding gene value is the two tuples, which represent the allocated dominator of the dominatee v_i and the neighboring dominators set of the dominatee v_i, respectively. Hence, a generation of chromosomes with gene values is denoted as $^{G} = \{C_j | 1 \le j \le G, C_j = (g_1, g_2, \ldots, g_i, \ldots, g_n)\}$, where G is the number of chromosomes in each generation of population, and for $1 \le i \le n$,

$$g_i = \begin{cases} 1, & \forall v_i \in B \\ < \forall v_t \in D(v_i) \,|\, D(v_i) >, & \forall v_i \in W \end{cases}$$

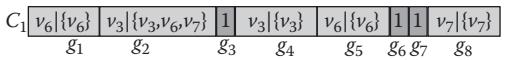


FIGURE 21.13 A chromosome with meta-genes and genes.

Through the previous description, we know that, as long as choosing a specific dominator from the neighboring DS (v_i), $\forall v_i \in$, we can decide a specific *dominator partition*. Additionally, all the nodes with $g_i = 1$ form a VB $= \{v_i | g_i = 1, 1 \le i \le n\}$. We still use the probabilistic WSN shown in Figure 21.2b to illustrate the encoding scheme. There are eight nodes, and the VB is $= \{v_3, v_6, v_7\}$. Moreover, according to the topology shown in Figure 21.2b, we can get (v_i), $\forall v_i \in$ easily. Thus, the *dominator partition* shown in Figure 21.11b can be encoded using eight genes in a chromosome, that is, $C_1 = (v_6|\{v_6\}, v_7|\{v_3, v_6, v_7\}, 1, v_3|\{v_3\}, v_6|\{v_6\}, 1, 1, v_7|\{v_7\})$ (shown in Figure 21.13). In conclusion, C_j records one possible VB and one possible dominator partition associated with the VB, while ^{G} represents the G different solutions to the LBVBP.

21.4.2.1.6 Population Initialization

GAs differ from most optimization techniques because of their global searching effectuated by one population of solutions rather than from one single solution. Hence, a GA search starts with the creation of the first generation, that is, a population with G chromosomes denoted by P_1. This step is called population initialization. A general method to initialize the population is to explore the genetic diversity. That is, for each chromosome, all dominators are randomly generated. However, the dominators must form a VB. Therefore, we start to create the first chromosome C_1 by running an existing MCDS method, for example, the latest MCDS construction algorithm [10], and then generate the population with G chromosomes by modifying C_1. We call the procedure, generating the whole population by modifying one specific chromosome, inheritance population initialization (IPI) [52].

21.4.2.1.7 Fitness Function

Given a solution, its quality should be accurately evaluated by the fitness score, which is determined by the fitness function. In MOGAs, multiple conflict objectives need to be achieved. Hence, in our algorithm, three different fitness functions are defined as follows:

$$\begin{cases} f_1(C_j) = \min\{|B|\} \\ f_2(C_j) = \min\{|B|_b\} \\ f_3(C_j) = \min\{|3|_b\} \end{cases}$$

As we have mentioned, different from the relations of the solutions in SOPs, the relations of the solutions of MOPs have three possibilities. We use the following operator to summarize the relations.

Definition 21.19: *Fitter operator*

$$\text{Fitter}(C_i, C_j) = \begin{cases} 1 & F(C_i) \preceq F(C_j) \\ -1 & F(C_j) \preceq F(C_i) \\ 0 & \text{nondominated} \end{cases} \tag{21.8}$$

where the objective vector is $\mathbf{F}(C_k) = (f_1(C_k), f_2(C_k), f_3(C_k))$, $1 \le k \le G$.

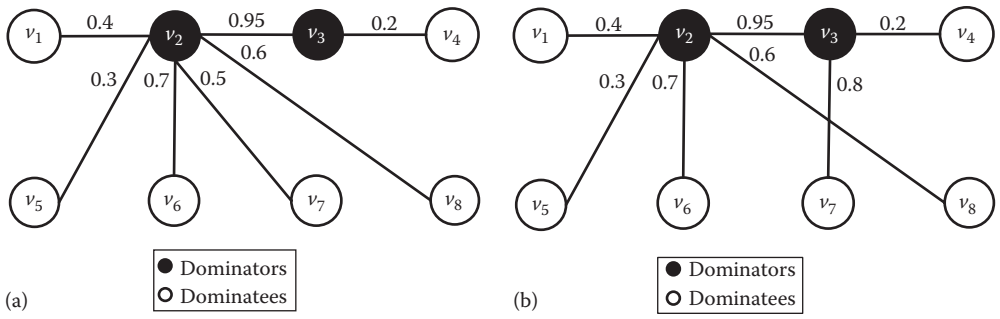


FIGURE 21.14 (a) A imbalanced dominator partition; (b) a load-balanced dominator partition.

Chromosomes	f_1	f_2	f_3
C_1 (Figure 21.14(a))	2	8.29	4.19
C_2 (Figure 21.14(b))	2	8.29	1.89
C_3 (Figure 21.11(a))	3	5.89	3.53
C_4 (Figure 21.11(b))	3	5.89	3.79
C_5 (Figure 21.11(c))	3	5.89	1.41

(a)

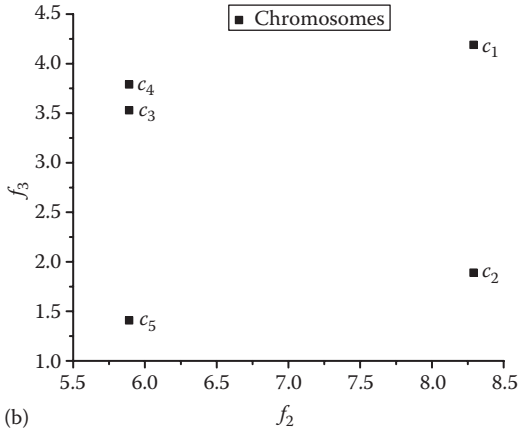


FIGURE 21.15 (a) Multiple objective fitness values of five chromosomes and (b) five chromosomes in the 2D objective space.

When the objective vector of C_i *Pareto dominates* that of C_j, the *fitter operator Fitter*(C_i, C_j) returns 1; when the objective vector of C_i is *Pareto dominated* by that of C_j, the *Fitter operator Fitter*(C_i, C_j) returns –1; and when they are nondominated, the *Fitter operator Fitter*(C_i, C_j) returns 0.

We will use the feasible solutions (encoded as chromosomes) shown in Figures 21.14 and 21.11 to illustrate the *Fitter operator*. The three fitness values of the five different chromosomes are listed in Figure 21.15a. To be more specific, we draw the five chromosomes in the 2D objective space (f_2 vs. f_3) in Figure 21.15b. From Figure 21.15b, we know C_2 Pareto dominates C_1, C_5 Pareto dominates C_3, and C_3 Pareto dominates C_4.

21.4.2.1.8 Selection Scheme and Replacement Policy

During the evolutionary process, election plays an important role in improving the average quality of the population by passing the high-quality chromosomes to the next generation. Therefore, in MOPS, the selection operator needs to be more carefully formulated to ensure that better chromosomes (the chromosomes closer to the Pareto-optimal set) of the population have a greater probability of being selected for mating. We adopt *dominating tree* (*DT*) [53] to select parent chromosomes. A DT is a binary tree, in which each node has three fields: *id*, *left-link*, and *right-link*. The left-link field links to its left subtree whose root node is *dominated* by the node, and the right-link field links to its right subtree

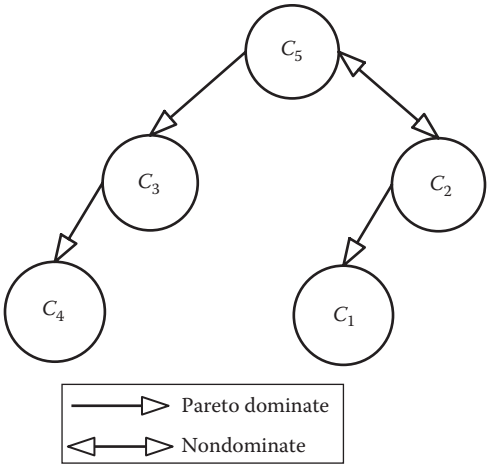


FIGURE 21.16 Illustration of a DT.

whose root node is *nondominated* by the node. A *sibling chain* of a DT is defined as a chain constituted by its root and the root's right-link nodes. A DT has some useful features [53]:

- The sibling chain of a DT consists of and only consists of all Pareto-optimal nodes in the DT.
- The root of a DT Pareto dominates all nodes in its left subtree.
- The leftmost node in the DT can be regarded as the *worst* node of the DT.

Figure 21.16 demonstrates a DT consisting of the five encoded chromosomes shown in Figures 21.14 and 21.11. According to these features, we have the following:

- C_5 and C_2 (the sibling chain) are Pareto-optimal nodes in the DT.
- C_5 Pareto dominates C_3 and C_4 in the DT.
- C_4 can be considered as the *worst* node in the DT.

Based on the earlier description, in each generation, we random select a pair of parent chromosomes in the set of Pareto-optimal nodes (i.e., *from sibling chain*). The two new-generated offsprings will be inserted into the DT one by one and the *worst* (left-most) individual will be deleted from the DT each time. This process will be repeated until the stopping criterion (e.g., a desired number of total generation is reached) is satisfied.

21.4.2.1.9 Genetic Operations

The performance of a GA relies heavily on two basic genetic operators, *crossover* and *mutation*. Crossover exchanges parts of the parent chromosomes in order to find better ones. Mutation flips the values of genes, which helps a GA keep away from local optimum. In the LBVBP, we can adopt classical operations; however, the new obtained solutions may not be valid (the DS represented by the chromosome is not a CDS) after implementing the crossover and mutation operations. Therefore, a correction mechanism [52] needs to be performed to guarantee the validity of all the new-generated offspring solutions.

The purpose of crossover operations is to produce more valid VBs represented by the offspring chromosomes. At this stage, we do not need to care about dominator partitions. Therefore, when performing crossover operations, we can logically assume that all gene values of dominatees are 0. In the LBVBP-MOGA, we adopt three crossover operators called single-point crossover, two-point crossover, and uniform crossover, respectively (shown in Figure 21.17). With a crossover probability p_c, each time we select two chromosomes from the set of Pareto-optimal nodes as parents to perform one of the three crossover operators randomly. As mentioned earlier, after crossover operation, the new-generated offsprings may not be a valid solution (the constructed VB is not a CDS). Thus, we need to perform the correction mechanism.

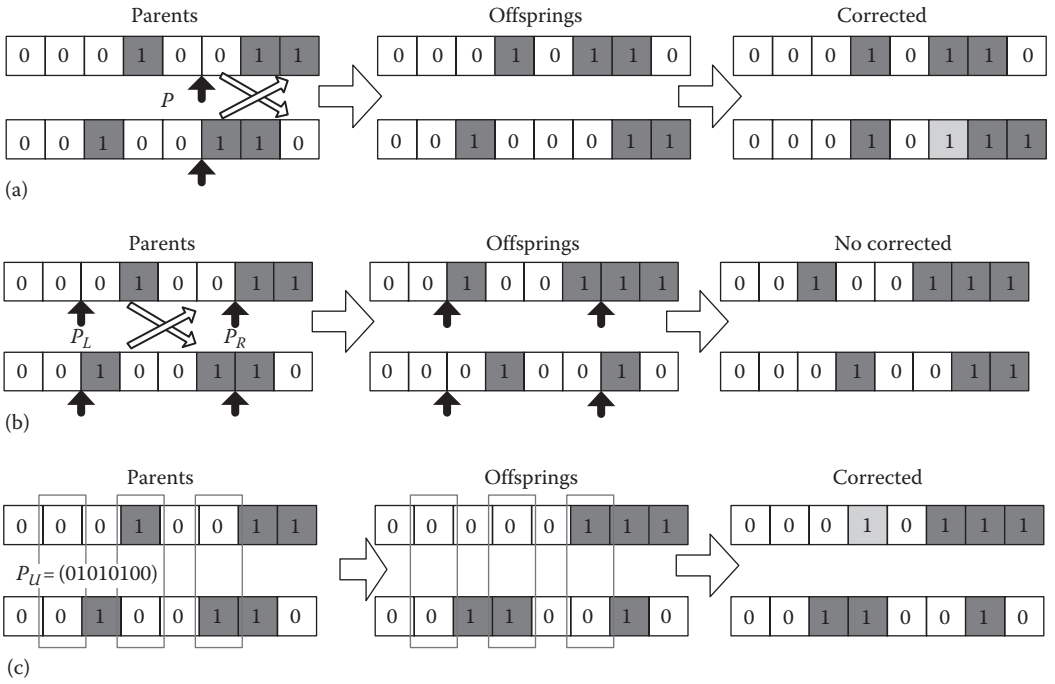


FIGURE 21.17 Illustration of crossover operations: (a) single-point crossover, (b) two-point crossover, and (c) uniform crossover.

The mechanism starts with scanning each gene g_i on the offspring chromosome from the position of the crossover point, till the end of the chromosome. If g_i value is different from the corresponding value of its parent, then the mechanism corrects the value. The processes of the correction mechanism are shown in Figure 21.17a and c.

Suppose that two parent chromosomes (00010011) and (00100110) are selected by the RS scheme from the population. By the single-point crossover (shown in Figure 21.17a), the genes from the randomly generated crossover point $P = 6$ to the end of the two chromosomes exchange with each other to get (00010110) and (00010111). After crossing, the first offspring (00010110) is a valid solution. However, the other one (00100011) is not valid; thus, we need to perform the *correction mechanism*. The mechanism starts with scanning each gene on the offspring chromosome, denoted by C_o^g till the end of the chromosome. If the value of the current scanned gene is 0, that is, $g_i = 0$, and the gene value is different from the original chromosome, denoted by C_s^g, without doing crossover and mutation operations, we change the gene value to 1. Whenever the DS represented by the corrected chromosome is a CDS, stop the mechanism. Otherwise, keep repeating the process till the end of C_o^g is reached. The idea behind the correction mechanism is that the DS represented by C_s^g is a CDS. If C_o^g is not valid, then add the dominators represented by C_s^g into the DS represented by C_o^g one by one. Finally, the corrected chromosome must be valid. For example, for the specific invalid offspring chromosome (00100011), when scanning the gene at position P, that is, $g_6 = 0$, we find the value of g_6 is different after crossing. Therefore, we correct it by setting $g_6 = 1$. Then the corrected chromosome (00010111) is now a valid solution. Consequently, the correction mechanism stops, and we get two valid offspring chromosomes (00010110) and (00010111). The correction mechanism is the same for crossover and mutation operations.

By the two-point crossover (shown in Figure 21.17b), the two crossover points are randomly generated, which are $P_L = 3$ and $P_R = 6$; and then the genes between P_L and P_R of the two parent chromosomes are exchanged with each other. The two offsprings are (00100111) and (00010010), respectively. Since both of the offspring chromosomes are valid, we do not need to do any correction.

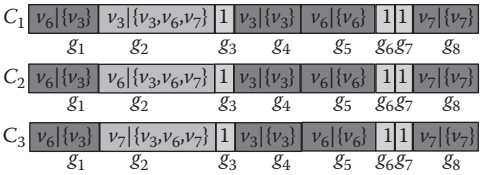


FIGURE 21.18 Illustration of dominatee mutation.

For the uniform crossover (shown in Figure 21.17c), the vector of uniform crossover P_U is randomly generated, which is P_U = (01010100), indicating that g_2, g_4, and g_6 of the two parent chromosomes exchange with each other. Hence, the two offsprings are (00000111) and (00110010). Since the first offspring is not a valid solution, we need to perform the correction mechanism mentioned before, and the corrected chromosome becomes (00110010), which is a valid solution.

The population undergoes the gene mutation operation after the crossover operation is performed. With a mutation probability p_m, we scan each gene g_i on the offspring chromosomes. If the mutation operation needs to be implemented, the value of the gene flips, that is, 0 becomes 1 and 1 becomes 0. The same correction mechanism needs to be performed if the mutated chromosomes are not valid.

In order to increase the diversity of possible dominator partitions, we propose an additional step called *dominatee mutation* in LBVBP-MOGA to generate more feasible dominator partitions.

As known, as long as we choose a specific node from the neighboring DS (v_i), $\forall v_i \in$, we can easily explore a dominator partition. According to the observation, we design the following *dominatee mutation*. The original population without doing crossover and gene mutation operations will undergo the dominatee mutation operation. If the number of neighboring dominators of a dominatee v_i is greater than 1, that is, $|(v_i)| \geq 2$, then randomly pick a node from the set (v_i). We use the VB shown in Figure 21.2b to illustrate the dominatee mutation. According to the topology, we get $|(v_2)| = |\{v_3, v_6, v_7\}| = 3 > 1$, which satisfies the condition to perform the dominatee mutation. Therefore, we randomly pick one dominator from the set (v_2). If v_3 is selected from (v_2), it means that dominatee v_2 is allocated to dominator v_3. The corresponding dominator partition is shown in Figure 21.11a. Similarly, if dominatee v_2 is allocated to dominator v_6, or v_2 is allocated to dominator v_7, the dominator partitions are shown in Figure 21.11b and c, respectively. In summary, the process of dominatee mutation for v_2 is shown in Figure 21.18.

The pseudocode of LBVBP-MOGA is shown in Algorithm 21.4.

Algorithm 21.4: LBVBP-MOGA

Require: p_c, p_m, K {K represents the number of total generations}.
1: $t = 0$ {t represents the current generation number};
2: Initialize population $P(0)$ using the IPI algorithm;
3: **while** $t < K$ **do**
4: Calculate the fitness vector of each chromosome in population $P(t)$;
5: Construct *dominating tree* (*DT*) T for population $P(t)$;
6: Select two parent chromosomes from $P(t)$ in the set of Pareto optimal nodes {from the *sibling chain of T*};
7: Perform dominatee mutation;
8: Crossover with p_c;
9: Mutation with p_m;
10: Insert new offsprings to the dominating tree T one by one;
11: Delete the "worst" (left-most) one from T;

12: Calculate the fitness vector of each chromosome in interim population $P'(t)$;
13: $P(t + 1) = P'(t)$;
14: t++;
15: **end while**
16: **return** the fittest individual in population $P(K)$ {the root of the constructed DT}.

21.4.2.1.10 Convergence Analysis

If MOGA can find the global optimum (i.e., convergence to the *Pareto-optimal set*) of an MOP with probability one, we say that the MOGA converges. In this subsection, we show that our proposed LBVBP-MOGA converges.

Given that **x** is a decision variable, $\mathbb{S}$ is the space of all feasible decision variable, $\Phi: \mathbb{S} \to \mathbb{R}^k (k > 2)$ is a multiobjective fitness function, t is the generation number, and $P(t)$ is the population of the tth generation. Back et al. prove that a GA converges with probability one if it fulfills the following conditions in [54]:

1. $\forall \mathbf{x}, \mathbf{x}'$ is reachable from **x** by means of crossover and mutation operations.
2. The population sequence $P(0), P(1), \ldots$ is monotone, that is,

$$\forall t : \min\left\{\Phi\left(x(t+1)\right) \mid x(t+1) \in P(t+1)\right\} \leq \min\left\{\Phi(x(t)) \mid x(t) \in P(t)\right\} \tag{21.9}$$

This definition is appropriate in the context of single-objective GAs. However, in the context of MOGAs, a more general definition of monotonicity is needed. At any given generation t, let $P_{\text{found}}(t)$ the set of Pareto-optimal solutions that can be found by the MOGA through generation t. Moreover, let $P_{\text{true}}(t)$ denote the true set of Pareto-optimal solution that can be found through generation t. The definition of monotonicity is given by the condition as follows:

$$P_{\text{found}}(t) = \left\{x \in P_{\text{true}}(t) \mid \forall x' \in P_{\text{true}}(t) \text{ s.t. } F(x) \preceq F(x')\right\}$$

with $P_{\text{found}}(0) = \text{À}$. It can be shown by induction on t that under this condition, $P_{\text{found}}(t)$ consists of the set of solutions evaluated through generation t that are Pareto optimal with respect to the set of all such solutions. Thus, $P_{\text{found}}(t + 1)$ either retains or improves upon solutions in $P_{\text{found}}(t)$. In this manner, this condition ensures that $P_{\text{found}}(t)$ monotonically moves toward to the Pareto-optimal set P^* of an MOP.

Theorem 21.2

An MOGA satisfying the following two conditions converges to P^* of an MOP with probability one, that is,

$$\text{Prob}\left\{\lim_{t \to \infty} \left\{P^* = P_{\text{found}}(t)\right\}\right\} = 1 \tag{21.10}$$

$\forall \boldsymbol{x}, \boldsymbol{x}'$ is reachable from $\boldsymbol{x}$ by means of crossover and mutation operations

$$P_{\text{found}}(t) = \left\{x \in P_{\text{true}}(t) \mid \forall x' \in P_{\text{true}}(t) \text{ s.t. } F(x) \preceq F(x')\right\} \tag{21.11}$$

Proof 21.1: An MOGA can be viewed abstractly as a Markov chain consisting of two states. In the first state, $P^* = P_{\text{found}}(t)$, and in the second state $P^* \neq P_{\text{found}}(t)$. By Equation 21.11, there is zero probability of

transiting from the first state to the second state. Thus, the first state is absorbing. By Equation 21.10, there is a nonzero probability of transiting from the second state to the first state. Thus, the second state is transient. Based on the earlier analysis, we can conclude that an MOGA can be absorbed to the first state (i.e., converges to P^*) within finite states if the previous two conditions (Equations 21.10 and 21.11) are satisfied.

Theorem 21.3

The proposed LBVBP-MOGA is convergent.

Proof 21.2: The offspring chromosomes (i.e., $\mathbf{x}'$) are created only by crossover and mutation operations, which satisfy Equation 21.10. Moreover, according to the proposed selection scheme and replacement policy, in each generation, we randomly select parent chromosomes in the set of Pareto-optimal nodes (i.e., $\mathbf{F}(\mathbf{x}) \preccurlyeq \mathbf{F}(\mathbf{x}')$). Newly generated offsprings will be inserted into the DT one by one and the *worst* individual will be deleted from the DT each time. This process satisfies the monotonicity condition, which is shown in Equation 21.11. According to Theorem 21.2, our proposed LBVBP-MOGA is convergent.

21.4.3 Performance Evaluation

Since there are no existing works studying the LBVB construction problem for probabilistic WSNs currently, in the simulations, the results of LBVBP-MOGA (denoted by MOGA) are compared with the recently published MCDS construction algorithm [10] denoted by MCDS and the LBCDS-GA algorithm proposed in [52] denoted by GA. We compare the three algorithms in terms of network lifetime, which is defined as the time duration until the first dominator runs out of energy.

21.4.3.1 Simulation Environment

We build our own simulator where all the nodes have the same transmission range and all the nodes are deployed uniformly and randomly in a square area. For each specific setting, 100 instances are generated. The results are averaged over these 100 instances (all results are rounded to integers). Moreover, a random value between 0.5 and 0.98 is assigned to the delivery ratio (γ_{ij}) value associated to a pair of nodes (v_i and v_j) inside the transmission range; otherwise, a random value between (0, 0.5) is assigned to γ_{ij} associated to a pair of nodes beyond the transmission range. Moreover, we use the VB-based data aggregation as the communication mode. The simulated energy consumption model is that every node has the same initial 1000 units of energy. Receiving and transmitting a packet both consume 1 unit of energy. In the simulation, we consider the following tunable parameters: the node transmission range, the total number of nodes deployed in the square area, and the side length of the square area. Subsequently, we show the simulation results in two different scenarios.

21.4.3.2 Simulation Results

Figure 21.19 shows the network lifetime of three methods (MOGA, MCDS, and GA) under two different scenarios. From Figure 21.19a, we know that the network lifetime increases for all the three algorithms with the side length of the deployed area increasing. It is obvious that the density of the network becomes thinner with the side length of the deployed area increasing. As to a data aggregation, the thinner the network is, the fewer neighbors of each dominator. In other words, the aggregated data are less on each dominator when the network becomes thinner. Hence, network lifetime is increasing for all the three algorithms. Additionally, we can see that both MOGA and GA outperform MCDS. Furthermore, MOGA prolongs network lifetime by 42% on average compared with MCDS and by 20% on average compared with GA. The results demonstrate that load-balancedly allocating dominatees to dominators can improve network lifetime significantly. On the other hand, MOGA outperforms GA, since MOGA

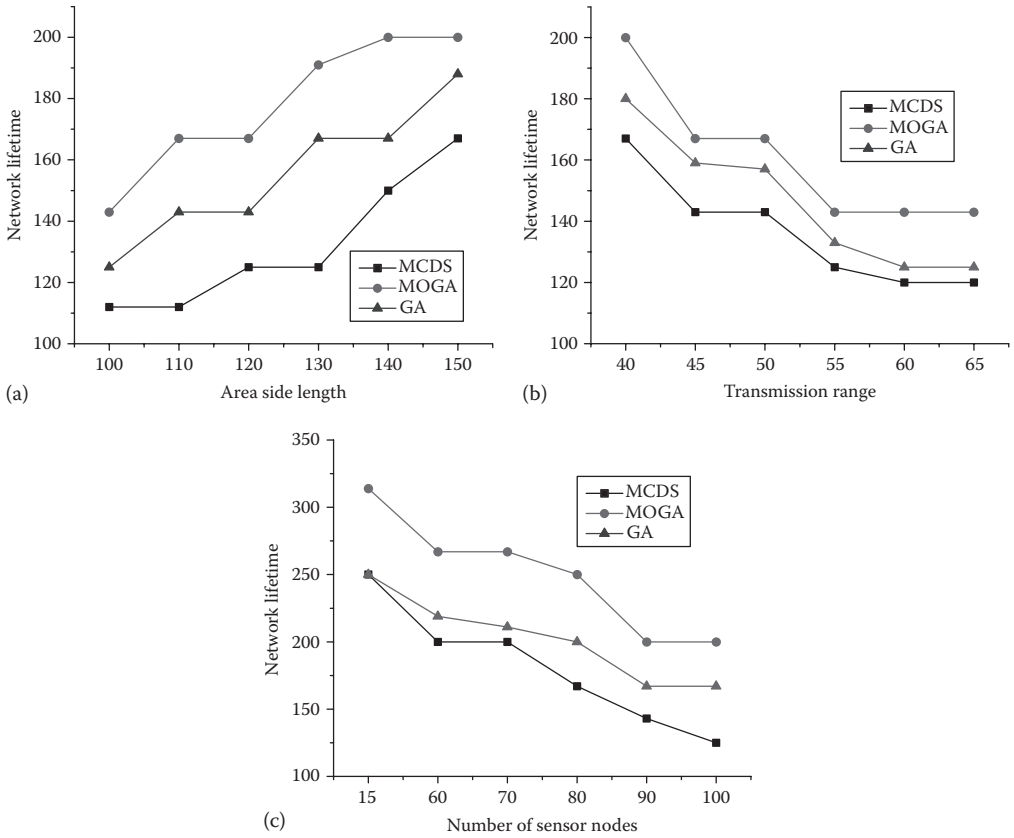


FIGURE 21.19 Network lifetime: (a) the node transmission range is 20 m, the number of nodes is 100, and the side length of the deployed area changes from 100 m × 150 m; (b) the square area is 300 m × 300 m, the number of nodes is 100, and the node transmission range changes from 40 to 65 m; and (c) the square area is 300 m × 300 m, the node transmission range is 50 m, and the number of nodes changes from 50 to 100.

takes multiple objectives into consideration simultaneously, making the MOGA easier to converge to a global optimum. Additionally, the local optimal solution found by GA might not be the same as the global optimal solution. Hence, the results shown in Figure 21.19a indicate that our proposed MOGA can find a solution that is closer to the optimal solution than GA.

From Figure 21.19b, we know that the network lifetime decreases for all algorithms with the node transmission range increasing. The fact is that the network becomes denser with the node transmission range increasing. The denser the network is, the more number of neighbors on each dominator. Since we use data aggregation as the communication mode in the simulations, the aggregated data are increasing on each dominator when the network becomes denser. Hence, network lifetime is decreasing for all the three algorithms. Similar results indicate that both MOGA and GA outperform MCDS. To be specific, MOGA prolongs network lifetime by 25% on average compared with MCDS and by 6% on average compared with GA. The reasons are the same as analyzed before.

From Figure 21.19c, we know that the network lifetime decreases for all the three algorithms with the number of nodes increasing. This is because we perform data aggregation in a more crowded network. Intuitively, the denser the network is, the more number of neighbors of each dominator. With the number of neighbors increasing, the aggregated data on each dominator become heavier. Hence, the network lifetime decreases for all the three algorithms. Additionally, we can see that both MOGA and GA outperform MCDS. Furthermore, MOGA prolongs network lifetime by 69% on average compared with MCDS and by 47% on average compared with GA. The reasons are the same as analyzed before.

21.5 Summary

The LBVB problem in WSNs is introduced in this chapter. How to solve the problem under deterministic WSNs and probabilistic WSNs is presented subsequently. The extensive simulation results demonstrate that using an LBCDS as a VB and then load-balancedly allocating dominatees to dominators can prolong network lifetime significantly.

References

1. W. Di, Q. Yan, and T. Ning. Connected dominating set based hybrid routing algorithm in ad hoc networks with obstacles. In *IEEE International Conference on Communications*, Istanbul, Turkey, 2006, Vol. 9, pp. 4008–4013.
2. W. El-Hajj, A. Al-Fuqaha, M. Guizani, and H.H. Chen. On efficient network planning and routing in large-scale MANETs. *IEEE Transactions on Vehicular Technology*, 58(7):3796–3801, 2009.
3. S. Ji, Y. Li, and X. Jia. Capacity of dual-radio multi-channel wireless sensor networks for continuous data collection. In *Proceedings of IEEE INFOCOM*, Shanghai, China, 2011, pp. 1062–1070.
4. S. Ji, R. Beyah, and Y. Li. Continuous data collection capacity of wireless sensor networks under physical interference model. In *IEEE Eighth International Conference on Mobile Ad Hoc and Sensor Systems*, Valencia, Spain, 2011, pp. 222–231.
5. M. Li, L. Ding, Y. Shao, Z. Zhang, and B. Li. On reducing broadcast transmission cost and redundancy in ad hoc wireless networks using directional antennas. *IEEE Transactions on Vehicular Technology*, 59(3):1433–1442, 2010.
6. B.K. Polat, P. Sachdeva, M.H. Ammar, and E.W. Zegura. Message ferries as generalized dominating sets in intermittently connected mobile networks. *Pervasive and Mobile Computing*, 7(2):189–205, 2011.
7. B. Deb, S. Bhatnagar, and B. Nath. Multi-resolution state retrieval in sensor networks. In *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications*, Anchorage, Alaska, 2003, pp. 19–29.
8. K.P. Shih, D.J. Deng, R.S. Chang, and H.C. Chen. On connected target coverage for wireless heterogeneous sensor networks with multiple sensing units. *Sensors*, 9(7):5173–5200, 2009.
9. H.M. Ammari and J. Giudici. On the connected k-coverage problem in heterogeneous sensor nets: The curse of randomness and heterogeneity. In *29th IEEE International Conference on Distributed Computing Systems*, Montreal, Québec, Canada, 2009, pp. 265–272.
10. P.J. Wan, S.C.H. Huang, L. Wang, Z. Wan, and X. Jia. Minimum-latency aggregation scheduling in multihop wireless networks. In *Proceedings of the 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, New Orleans, Louisiana, 2009, pp. 185–194.
11. M. Yan, J. He, S. Ji, and Y. Li. Minimum latency scheduling for multi-regional query in wireless sensor networks. In *2011 IEEE 30th International Conference on Performance Computing and Communications Conference* (*IPCCC*). IEEE, Orlando, FL, 2011, pp. 1–8.
12. A. Ephremides, J.E. Wieselthier, and D.J. Baker. A design concept for reliable mobile radio networks with frequency hopping signaling. *Proceedings of the IEEE*, 75(1):56–73, 1987.
13. S. Guha and S. Khuller. Approximation algorithms for connected dominating sets. *Algorithmica*, 20(4):374–387, 1998.
14. M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman & Co., New York, 1990.
15. J. Schleich, G. Danoy, P. Bouvry, and L. Thi Hoai An. Blackbone2: An efficient deterministic algorithm for creating 2-connected m-dominating set-based backbones in ad hoc networks. In *Proceedings of the Seventh ACM International Symposium on Mobility Management and Wireless Access*, Tenerife, Canary Islands, Spain, 2009, pp. 91–98.

16. D. Kim, W. Wang, X. Li, Z. Zhang, and W. Wu. A new constant factor approximation for computing 3-connected m-dominating sets in homogeneous wireless networks. In *2010 Proceedings of IEEE INFOCOM*. IEEE, San Diego, CA, 2010, pp. 1–9.
17. F. Dai and J. Wu. On constructing k-connected m-dominating set in wireless ad hoc and sensor networks. *Journal of Parallel and Distributed Computing*, 66(7):947–958, 2006.
18. L. Ding, X. Gao, W. Wu, W. Lee, X. Zhu, and D.Z. Du. Distributed construction of connected dominating sets with minimum routing cost in wireless networks. In *2010 IEEE 30th International Conference on Distributed Computing Systems* (*ICDCS*). IEEE, Genoa, Italy, 2010, pp. 448–457.
19. D. Kim, Y. Wu, Y. Li, F. Zou, and D.Z. Du. Constructing minimum connected dominating sets with bounded diameters in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 20(2):147–157, 2009.
20. T.N. Nguyen and DT Huynh. Energy-efficient connected d-hop dominating sets in wireless sensor networks. In *IEEE International Conference on Pervasive Computing and Communications*, *2009* (*PerCom'09*), Galveston, Texas, 2009, pp. 1–6.
21. M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, Santa Clara, California, 2004, pp. 517–526.
22. G. Zhou, T. He, S. Krishnamurthy, and J.A. Stankovic. Impact of radio irregularity on wireless sensor networks. In *Proceedings of the Second International Conference on Mobile Systems*, *Applications*, *and Services*. ACM, Boston, Massachusetts, 2004, pp. 125–138.
23. A. Cerpa, J.L. Wong, M. Potkonjak, and D. Estrin. Temporal properties of low power wireless links: Modeling and implications on multi-hop routing. In *Proceedings of the Sixth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Urbana-Champaign, Illinois, 2005, pp. 414–425.
24. A. Cerpa, J.L. Wong, L. Kuang, M. Potkonjak, and D. Estrin. Statistical model of lossy links in wireless sensor networks. In *Fourth International Symposium on Information Processing in Sensor Networks*, *2005* (*IPSN'05*). IEEE, Los Angeles, CA, 2005, pp. 81–88.
25. Y. Liu, Q. Zhang, and L.M. Ni. Opportunity-based topology control in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 21(3):405–416, 2010.
26. B.N. Clark, C.J. Colbourn, and D.S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1):165–177, 1990.
27. S. Guha and S. Khuller. Approximation algorithms for connected dominating sets. *Algorithms*, 20(4):179–193, 1996.
28. D. Du and X. Hu. *Steiner Tree Problems in Computer Communication Networks*. World Scientific Publishing Company Inc., Hackensack, NJ, 2008.
29. L. Ruan, H. Du, X. Jia, W. Wu, Y. Li, and K.I. Ko. A greedy approximation for minimum connected dominating sets. *Theoretical Computer Science*, 329(1):325–330, 2004.
30. J. Wu, M. Gao, and I. Stojmenovic. On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks. In *International Conference on Parallel Processing*. IEEE, Valencia, Spain, 2001, pp. 346–354.
31. F. Dai and J. Wu. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 15(10):908–920, 2004.
32. D. Zhou, M.T. Sun, and T.H. Lai. A timer-based protocol for connected dominating set construction in IEEE 802.11 multihop mobile ad hoc networks. In *Proceedings of Symposium on Applications and the Internet*. IEEE, Trento, Italy, 2005, pp. 2–8.
33. K. Sakai, F. Shen, K.M. Kim, M.T. Sun, and H. Okada. Multi-initiator connected dominating set construction for mobile ad hoc networks. In *IEEE International Conference on Communications*, *2008* (*ICC'08*). IEEE, Bejing, China, 2008, pp. 2431–2436.
34. P.J. Wan, K.M. Alzoubi, and O. Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. *Mobile Networks and Applications*, 9(2):141–149, 2004.

35. W. Wu, H. Du, X. Jia, Y. Li, and S.C.H. Huang. Minimum connected dominating sets and maximal independent sets in unit disk graphs. *Theoretical Computer Science*, 352(1):1–7, 2006.
36. P.J. Wan, L. Wang, and F. Yao. Two-phased approximation algorithms for minimum CDS in wireless ad hoc networks. In *The 28th International Conference on Distributed Computing Systems*. IEEE, Bejing, China, 2008, pp. 337–344.
37. M. Li, P.J. Wan, and F. Yao. Tighter approximation bounds for minimum CDS in wireless ad hoc networks. *The 20th International Symposium on Algorithms and Computation,* Hawaii, *USA,* 699–709, 2009.
38. X. Li, X. Gao, and W. Wu. A better theoretical bound to approximate connected dominating set in unit disk graph. *The 3rd International conference on Wireless Algorithms, Systems, and Applications,* Dallas, TX, 162–175, 2008.
39. X. Gao, Y. Wang, X. Li, and W. Wu. Analysis on theoretical bounds for approximating dominating set problems. *Discrete Mathematics*, *Algorithms and Applications*, 1(1):71–84, 2009.
40. R. Misra and C. Mandal. Rotation of CDS via connected domatic partition in ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 8(4):488–499, 2009.
41. F. Wang, M.T. Thai, and D.Z. Du. On the construction of 2-connected virtual backbone in wireless networks. *IEEE Transactions on Wireless Communications*, 8(3):1230–1237, 2009.
42. D. Mandala, X. Du, F. Dai, and C. You. Load balance and energy efficient data gathering in wireless sensor networks. *Wireless Communications and Mobile Computing*, 8(5):645–659, 2008.
43. M. Song, Y. Zhao, J. Wang, and E.K. Park. A high throughput load balance algorithm for multichannel wireless sensor networks. In *IEEE International Conference on Communications*, Dresden, Germany, 2009, pp. 1–5.
44. Y. Deng and Y. Hu. A load balance clustering algorithm for heterogeneous wireless sensor networks. In *International Conference on E-Product, E-Service and E-Entertainment*. IEEE, 2010, pp. 1–4.
45. G. Cao and F. Yu. The analysis of load balance for wireless sensor network using compressive sensing. In *IEEE 14th International Conference on Computational Science and Engineering* (*CSE*), Dalian, China, 2011, pp. 100–105.
46. M. Zhao and Y. Yang. A framework for mobile data gathering with load balanced clustering and MIMO uploading. In *Proceedings of INFOCOM*. IEEE, Shanghai, China, 2011, pp. 2759–2767.
47. M. Kalantari, M. Haghpanahi, and M. Shayman. A p-norm flow optimization problem in dense wireless sensor networks. In *INFOCOM: The 27th Conference on Computer Communications*. IEEE, Phoenix, AZ, 2008, pp. 341–345.
48. S.R. Kulkarni and P. Viswanath. A deterministic approach to throughput scaling in wireless networks. *IEEE Transactions on Information Theory*, 50(6):1041–1049, 2004.
49. S. Lin, J. Zhang, G. Zhou, L. Gu, J.A. Stankovic, and T. He. ATPC: Adaptive transmission power control for wireless sensor networks. In *Proceedings of the Fourth International Conference on Embedded Networked Sensor Systems*. ACM, Boulder, Colorado, 2006, pp. 223–236.
50. D. Son, B. Krishnamachari, and J. Heidemann. Experimental study of concurrent transmission in wireless sensor networks. In *Proceedings of the Fourth International Conference on Embedded Networked Sensor Systems*. ACM, Boulder, Colorado, 2006, pp. 237–250.
51. C.A. Coello Coello. Evolutionary multi-objective optimization: A historical view of the field. *Computational Intelligence Magazine*, 1(1):28–36, 2006.
52. J. He, S. Ji, M. Yan, Y. Pan, and Y. Li. Genetic-algorithm-based construction of load-balanced CDSs in wireless sensor networks. In *Military Communications*, Baltimore, MD, 2011, pp. 667–672.
53. A.J. Nebro, F. Luna, E. Alba, B. Dorronsoro, J.J. Durillo, and A. Beham. ABYSS: Adapting scatter search to multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 12(4):439–457, 2008.
54. T. Back, D.B. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Computation*. IOP Publishing Ltd., Oxford University Press, New York, NY, 1997.

Appendix A: List of Sensor Networking Standards, Organizations, and Protocols

John R. Vacca
Techwrite

6LoWPAN (http://www.6lowpan.net): A distorted acronym for IPv6 over low-power wireless personal area networks. Presently, it is a standard based on the IETF RCF 4944 and designed to be used over 802.15.4 chips and radios.

Automation Standards Compliance Institute (ASCI) (https://www.isa.org): A nonprofit organization incorporated by ISA in 2006 to provide a home for certification, conformance, and compliance assessment activities in the automation arena.

Bluetooth Smart Energy (http://www.bluetooth.com/Pages/Smart-Energy-Tech.aspx): Addresses the needs for wireless connections of sensors and actuators in the residence. It focuses on the needs of manufacturers of consumer devices in the Smart Grid environment.

Bluetooth Special Interest Group (SIG) (http://www.bluetooth.com/Pages/15-Years-of-Bluetooth-Technology.aspx): The body that oversees the development of Bluetooth standards and the licensing of the Bluetooth technologies and trademarks to manufacturers.

C Standard Library (http://www.tutorialspoint.com/c_standard_library/index.htm): The C standard library is the standard library for the C programming language, as specified in the ANSI C standard. There are no other dependencies aside from libc, the C standard library. Diffie–Hellman and AES encryption are included in the core executable and the kernel module, respectively. The core executable, kernel module, and runtime loadable module have all been written in C with no floating-point math for maximum performance.

DASH7 Alliance (http://www.dash7.org/): A nonprofit industry consortium that promotes the ISO 18000-7 standard for wireless sensor networking.

DASH7 Mode 2 (http://www.dash7.org/): A specification with a free and open-source license.

Device Language Message Specification (DLMS) User Association (http://www.dlms.com): The DLMS User Association is a nonprofit organization, located in Geneva, Switzerland. Its mission is to develop, promote, and maintain the DLMS/COSEM specification. It provides an information exchange forum for users, manufacturers and system providers, test houses, and standardization bodies. It also

provides a conformance testing and certification scheme for metering equipment implementing the specification. The DLMS UA is formally liaisoned with IEC TC 13 WG 14.

Digi Mesh (http://www.digi.com): Designed to meet the need for very-low-power sensor networks where battery-powered routers are required.

European Smart Metering Industry Group (ESMIG) (http://www.esmig.eu): The ESMIG is the European industry association that provides knowledge and expertise on Smart metering and related communications at a European level.

IEEE 802.15.4 (http://ieee802.org/15/pub/TG4.html): An industrial wireless standard.

International Society of Automation (**ISA**) (https://www.isa.org): A nonprofit technical society for engineers, technicians, businesspeople, educators, and students, who work, study, or are interested in industrial automation and pursuits related to it, such as instrumentation.

IPv6 (Internet Protocol Version 6) (http://www.fcc.gov/guides/internet-protocol-version-6-ipv6): This is the latest version of the popular Internet Protocol (IP). With Version 6, the IP addresses structure, routing, and class of service change. IPv6 is part of the TCP/IP suite of protocols sponsored by the Internet Engineering Task Force (IETF).

ISA-18.2 Alarm Management (https://www.isa.org): Provides requirements and recommendations for the activities of the alarm management life cycle.

ISA-88 Batch (https://www.isa.org): The definitive document regarding batch process control.

ISA-95 Enterprise-Control System Integration (https://www.isa.org): Provides a framework for exchanging manufacturing data between enterprise and manufacturing systems as well as between different manufacturing operations management (MOM) systems.

ISA 99 Cyber Security (https://www.isa.org): A multipart group of standards and associated technical reports that address the issue of cyber security in industrial automation and control systems.

ISA100 Committee (https://www.isa.org): Addresses wireless manufacturing and control systems in the areas of the environment in which the wireless technology is deployed and technology and life cycle for wireless equipment and systems.

ISA 100 Standard (https://www.isa.org): Global short-, medium-, and long-term technology needs and solutions are incorporated on a nonexclusive technology basis with no bias toward or against a particular wireless technology.

ISA 100 Wireless (https://www.isa.org): Application of the technology include field sensors used for monitoring, control, alarm, and shutdown that can be vertically integrated from field to business systems; wireless technology whose uses include real-time field-to-business systems; and, across all industries, fluid processing, material processing, and discrete parts manufacturing environments.

ISA100 Wireless Compliance Institute (http://www.isa100wci.org/): Functions as an operational group within the ASCI. ASCI bylaws share the open constructs of ISA, while accounting for compliance organization requirements. Operating the ISA100 Wireless Compliance Institute within ASCI allows the organization to efficiently leverage the organizational infrastructure of ASCI.

ISA 100.11a (http://www.isa.org/isa100): A wireless networking technology standard developed by the International Society of Automation (ISA).

ISA-101 HMI (https://www.isa.org): Includes every facet of HMI design and operation, such as menu hierarchies, screen navigation conventions, graphics and color conventions, dynamic elements, alarming conventions, security methods and electronic signature attributes, interfaces with background programming and historical databases, popup conventions, help screens and methods used to work with alarms, program object interfaces, and configuration interfaces to databases, servers, and networks.

ISA-103 Field Device Tools (https://www.isa.org): Defines the interfaces for both the vertical and the horizontal data flows (called function control and data access), in the framework of a client–server

architecture, so it will allow application software and configuration tools to interact with field devices in a unified way, while hiding the manufacturer-specific interaction with devices or subsystems in a software module.

ISA-104 Electronic Device Description Language (https://www.isa.org): An integral technology for field digital protocols enabling configuration/setup, diagnostics, and monitoring of devices from different manufacturers.

ISA-106 Procedural Automation (https://www.isa.org): The ISA106 committee is continuing to produce a standard addressing the automation of procedures in the continuous process industries.

ISA-108 Intelligent Device Management (https://www.isa.org): Defines standards related to best practices and work processes for design, operation, and use of diagnostic and other information provided by intelligent field devices in the process industries.

ISO 18000-7 standard (http://www.iso.org/): Standard for wireless sensor networking.

M2M (machine-to-machine) module (http://m2m.com/thread/2445): Identifies the space enveloping the collection of devices, services, and the value chain required to allow the interconnection of electronic apparatus, typically but not always wirelessly. Another characteristic of M2M communication is that this interconnection enables primarily automated communication between distant, remote machines and one or more layers of central management applications. It provides for real-time monitoring and control without the need for human intervention.

Observations and Measurements Schema (O&M) (http://www.ec-gis.org): Standard models and XML schema for encoding observations and measurements from a sensor, both archived and real-time.

OGC Sensor Web Enablement Standards (www.sensorsportal.com): Implements sensor webs using the OGC's vendor-neutral interoperability framework for web-based discovery, access, control, integration, and visualization of online sensors and sensor data. The OGC Sensor Web Enablement Standards are composed of the following: Sensor Model Language (SensorML), Transducer Markup Language (TransducerML or TML), Observations and Measurements Schema (O&M), Sensor Planning Service (SPS), and Sensor Registries.

Open standard interfaces (http://www.rebreatherworld.com): The interface implements wildly accepted open standards for both technical and content aspects of the interface.

Point-to-multipoint network protocols (http://www.comtechefdata.com): They tend to use the modern air interfaces of either frequency-hopping spread spectrum (FHSS) or direct sequence spread spectrum (DSSS, 802.15.4).

Sensor Model Language (SensorML) (http://www.opengeospatial.org): Standard models and XML schema for describing sensor systems and processes; provides information needed for the discovery of sensors, location of sensor observations, processing of low-level sensor observations, and listing of sensor system operations that can be invoked by a client process.

Sensor Planning Service (SPS) (http://www.52north.org): Standard web service interface for requesting user-driven acquisitions and observations. This is the intermediary between a client and a sensor collection management environment.

Sensor Registries (http://www.google.it/patents/WO2007079279A2?cl=en): These enable publishing and discovery of sensors and observed values and are implementations of the OpenGIS Catalogue Service Implementation Specification, which has many applications beyond sensor webs.

Standards-compliant wireless operation (http://www.focus.ti.com): Most devices have gravitated to using either an IEEE 802.15.4-compliant wireless channel or an IEEE 802.11b/g-compliant channel.

Transducer Markup Language (TransducerML or TML) (http://www.ogcnetwork.net): Conceptual model and XML schema for describing transducers and supporting real-time streaming of data to and from sensor systems.

TSMP (time-synchronized mesh protocol) (http://www.linear.com): This mesh protocol uses time slots to allocate parts of the spectrum for communication between two nodes.

WiFi (http://www.wisegeek.org): Most often refers to the 802.11 family of standards, ranging from 802.11 and 802.11a all the way through 802.11s.

Wireless HART (http://www.hartcomm.org): Uses the time-synchronized mesh protocol (TSMP). It is also built on top of 802.15.4 DSSS, but it adds a more deliberate frequency-hopping algorithm.

ZigBee Alliance (http://www.zigbee.org/): A global ecosystem of organizations creating wireless solutions for use in energy management, residential, commercial, and consumer applications.

ZigBee PRO (ZigBee 2007) (http://www.zigbee.org): ZigBee is built on top of 802.15.4 using DSSS in the 2.4 GHz band.

ZigBee Smart Energy 1.1 or higher standard implementation (http://www.zigbee.org/Standards/ZigBeeSmartEnergy/Overview.aspx): The standard for interoperable products that monitors, controls, informs, and automates the delivery and use of energy and water. A stack, for example, complies with the ZigBee 2007 specification, supporting both ZigBee and ZigBee PRO feature sets on various hardware platforms, including the CC2530 SoC, ultralow-power MSP430 microcontroller plus CC2520 transceiver, and Stellaris Cortex-M microcontroller plus CC2520 transceiver.

Appendix B: Glossary

John R. Vacca
Techwrite

Accelerometer: A sensor or transducer for measuring acceleration.
Ambient Sensor: See Ambient Temperature Sensor.
Ambient Temperature: Temperature of the air surrounding a component.
Ambient Temperature Sensor: Temperature sensor used to measure the temperature of the air that surrounds a component (the ambient temperature).
Analog Temperature Sensor: Temperature sensor with a continuous analog voltage or current output that is related, usually linearly, to the measured temperature.
Baseline: The electrical signal from a sensor when no measured variable is present.
CCD: Charge-coupled device. One of the two main types of image sensors used in digital cameras.
CompoNet: A four-wire, industrial bus with a master–slave architecture.
Current-Mode Controller: A DC–DC switching regulator that regulates its output voltage by varying the peak inductor current on a cycle-by-cycle basis to output a regulated voltage despite variations in load current and input voltage.
Current-Sense Amplifier: An amplifier that measures current by measuring the voltage drop across a resistor placed in the current path.
Current Sensing: See Current-Sense Amplifier.
Current Sensor: See Current-Sense Amplifier.
Differential Remote Output Sensing: Uses a Kelvin connection at a remote location to sense the output voltage and better control the voltage at that point.
DSSP: Digital-sensor signal processor.
Fan Controller—PWM: An integrated circuit that varies the speed and airflow of a cooling fan using a pulse-width-modulated (PWM) voltage in response to temperature or system commands.
High Side: An element connected between the supply and the load.
Integrated Temperature Sensor: See Local Temperature Sensor.
IO-Link: A 24 V, three-wire, half-duplex, point-to-point sensor and actuator communication interface.
JFET: A junction field-effect transistor (JUGFET) is a FET in which the gate is created by reverse-biased junction (as opposed to the MOSFET that creates a junction via a field generated by conductive gate, separated from the gate region by a thin insulator).
Junction Diode Sensor: The use of a PN junction on a silicon die for determining die temperature.
Junction FET: See JFET.
Junction Temp Sensor: See Remote Temperature Sensor.
Junction Temperature Sensor: See Remote Temperature Sensor.

Local Temperature: The temperature measured on the die of the temperature-measuring integrated circuit.

Local Temperature Sensor: An element or function of an integrated circuit that measures its own die temperature.

Low Side: An element connected between the load and ground.

MEMS: The acronym for Micro Electronic Mechanical Systems, or microelectromechanical systems. Systems that combine mechanical and electrical components and are fabricated using semiconductor fabrication techniques.

Noxious Fumes: A combination of inert and corrosive gases usually associated with exhaust fumes or industrial by-products gases that can cause corrosive effects on temperature and pressure sensors when exposed.

PLC: A programmable logic controller (PLC, or programmable controller) is a ruggedized, microprocessor-based system that provides factory or plant automation by monitoring sensors and controlling actuators in real time.

PN Temperature Sensor: See Junction Diode Sensor.

PWM: A method for using pulse width to encode or modulate a signal.

PWM Fan Control: See Fan Controller—PWM.

PWM Temperature Sensor: Temperature sensor with digital, logic-level output.

Remote Diode: A diode or diode-connected bipolar transistor used as a temperature-sensing element, often integrated onto an integrated circuit whose temperature is to be measured.

Remote Temp Sensor: See Remote Diode.

Remote Temperature: Temperature at a location other than at the die of the temperature-measuring integrated circuit.

Remote Temperature Sensor: A remotely located PN junction used as a temperature-sensing device, usually located on an integrated circuit other than the one doing the measurement.

Response Time: The time for a sensor to respond from no load to a step change in load.

Shock Sensor: An acceleration sensor, generally a piezoelectric type, that can measure high acceleration but cannot measure static g-forces.

Temp: See Temperature.

Temperature: The average kinetic energy of the atoms or molecules of a body or substance, perceived as warmth or coldness.

Temp Sensor: See Analog Temperature Sensor.

Temperature Sensor: Temperature sensor that uses an external diode-connected transistor as the sensing element to measure temperatures external to the sensor (e.g., on a circuit board or on the die of a CPU).

Thermocouple: A temperature sensor formed by the junction of two dissimilar metals.

Transducer Electronic Data Sheet: A Transducer Electronic Data Sheet (TEDS) is a method for plug-and-play sensor and transducer hookup in which the sensor's calibration information is stored within the device and downloaded to the master controller when requested.

Transistor Sensor: See Remote Temperature Sensor.

Transistor Temperature Sensor: See Junction Diode Sensor.

TSSM: Temperature sensor and system monitor.

Voltage Output Temperature Sensor: See PWM Temperature Sensor.

Voltage Temperature Sensor: See Analog Temperature Sensor.

Wireless Sensor Network: Wireless sensor network (WSN) is a network of RF transceivers, sensors, machine controllers, microcontrollers, and user interface devices with at least two nodes communicating by means of wireless transmissions.

WSN: See Wireless Sensor Network.

Index

A

Absolute congestion impact (ACI), 19-6
Absorbing material, 18-22
Accelerometer
  automated sensing task using, 3-8
  sensors, 3-3
Acceptance/rejection methods, 13-7
Accuracy, 14-3, 20-1, 20-3, 20-9–20-12
ACK message, 12-13
Acoustics
  plane wave, 16-2
  reverberation, 14-7
  sensor unit, 20-2–20-4
  wavefield, 16-2
  wave processes, 14-3
Acoustics-based gunfire location system, 14-1
Active illumination, resolution analysis, 17-9–17-10
Active reader active tag (ARAT) system, 9-5
Active reader passive tag (ARPT) system, 9-5
Actual load, 7-9
Adaptive QoS-aware frame concatenation mechanism (AQCM), 4-9
ADCs, *see* Analog-to-digital converters (ADCs)
Addition-based CDS algorithm, 21-5
Ad hoc network, 13-1
Advertising, mobile crowd sensing applications, 3-3
Affinity analysis, data mining, 11-3
Agent-based architecture, WSNs, 5-12–5-13
AIDA protocol, *see* Application-independent data aggregation (AIDA) protocol
Algorithm
  A* shortest path with repulsion, 19-8–19-9
  DTA, 19-15
  FBkSP rerouting, 19-13
  rerouting process, 19-6, 19-13–19-14
American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), 15-5
Amorphous silicon-based technologies, 18-21
Amsterdam Smart City, 15-13
Analog-to-digital converters (ADCs), 1-8
Android's BroadcastReceiver service, 3-7
AP, *see* Authentication preamble (AP)
Application-independent data aggregation (AIDA) protocol, 4-9
AQCM, *see* Adaptive QoS-aware frame concatenation mechanism (AQCM)
ARAT system, *see* Active reader active tag (ARAT) system
ARMA model, *see* Autoregressive moving average (ARMA) model
ARPT system, *see* Active reader passive tag (ARPT) system
Atmospheric transmission windows, 18-2
Attenuation, 20-4
Authentication preamble (AP)
  out-of-sync, 12-10
    ACK message, 12-13
    handshake model, 12-12–12-13
    out-of-SYN message, 12-12–12-13
    overview, 12-13–12-16
    during recovery, 12-11
    SYN message, 12-12–12-13
  PHY-layer security solutions
    IEEE 802.15.4e standard format, 12-5
    method, 12-4–12-5
    protocol uses, 12-6–12-8
    window, 12-6
Authorized link set, 21-23–21-4
Autocorrelation
  analysis, 14-10
  function, 20-5
Automation, 1-2
Automation Standards Compliance Institute (ASCI), A-1
Autoregressive (AR) models, 11-7
Autoregressive moving average (ARMA) model, 11-7–11-8
Avalanche photo diode (APD), 18-3
Average travel time
  average number of reroutings, 19-18–19-19
  compliance rate, 19-22
  CPU time, 19-19–19-21
  penetration rate, 19-22–19-24
  traffic density, 19-22

B

BAE systems, 18-21
Barcelona, smart city projects, 15-12
Bayesian framework
  Cholesky decomposition, 6-15
  communications resource efficiency, 6-12–6-14
  computation and memory resource efficiency, 6-10–6-12
  distributed estimation, 6-2
  distributed position estimation, 6-18–6-19
  fusion node processing, 6-7–6-8, 6-13–6-14
  in-network processing, 6-9
  local state estimation, 6-3–6-4
  nonlinear state estimation, 6-14–6-18
  optimal distributed state estimation, 6-6–6-10
  simulation results and discussion, 6-19–6-22
  state estimate fusion, 6-5–6-6
  time-varying network connectivity, 6-2
Bearing angle vector, 20-2, 20-10
Bee-JamA protocol, 19-2
Big data, 5-6, 11-11
Biometric sensors, 8-1–8-2
  and authentication, 8-2–8-4
  banking and finance industry, 8-2
  BioBeats, 8-14
  challenges and issues, 8-11
  consumer electronics and mobile devices, 8-8–8-11
  cost, 8-13
  decision-making stage, 8-3–8-4
  Droid phones, 8-11
  ear identification, 8-7–8-8
  enrollment, 8-3
  face biometric sensors, 8-6
  financial information, 8-3
  fingerprint sensors, 8-1–8-6
  hand geometry, 8-7
  healthcare industry, 8-13
  interoperability, 8-12–8-13
  iris recognition methods, 8-7
  keystroke dynamics, 8-7
  matching/testing, 8-3
  multimodal, 8-8
  quality, 8-12–8-13
  RecognizeMe app, 8-11
  Samsung Galaxy SIII phone, 8-11
  sensor-embedded steering wheels, 8-14
  US-VISIT6 program, 8-12
  voiceprint biometric methods, 8-7
Blind source separation (BSS), 10-8–10-11
Bluetooth
  false location claim, 3-12–3-13
  radio, automated sensing task using, 3-8
  WSN, 2-7
Bluetooth low energy (BLE), 2-7
Bluetooth Smart Energy, A-1
Bluetooth Special Interest Group (SIG), A-1
Breadth-first search (BFS), 19-5
BSS, *see* Blind source separation (BSS)
Building automation and control networks (BACnet)
  analog input object, 15-5–15-6
  data communication protocol, 15-5
  network communication, 15-5
  properties, 15-6
  standard set of objects, defining, 15-5
Building management system (BMS), 15-3

C

Calibration, 1-11
CANs, *see* Controller area networks (CANs)
Carbon nano tube (CNT), 18-1
  based bolometer characteristics, 18-22
  electrical response determination, 18-25–18-27
  multiwall, 18-27
  single-wall, 18-27
  thermal response determination
    heat flow equation, 18-22–18-23
    internal vibrational energy, determination, 18-23
    net IR radiation power absorbed, 18-24
    temperature map of bolometer, 18-24–18-25
    thermal diffusion coefficient, 18-23–18-24
CarTel project, 19-2
CatchUp, data aggregation, 4-5
CCACK scheme, *see* Cumulative coded ACK (CCACK) scheme
Centered aperture
  monostatic case, 17-11
  resolution analysis, 17-8–17-9
CerfBoard, 2-3
CerfCube, 2-3
Chemical vapor deposition (CVD), 18-6
Choquet integral (CI), nonlinear regression, 11-9–11-11
Client–server communication, 15-6
Clustered correlated failures, 13-13
Clustering
  data mining, 11-3
  data stream, 11-5
CMIS problem, *see* Connected MIS (CMIS) problem
CMOS processes, *see* Complementary metal–oxide–semiconductor (CMOS) processes
Collision avoidance, 20-1, 20-11–20-12
Commercial buildings, 15-2–15-3
Communication protocols
  BACnet, 15-5–15-6
  Ethernet, 15-5
  IEEE 802.11 (Wi-Fi), 15-8–15-9
  Internet protocol, 15-5
  LonWorks, 15-6

6LoWPA, 15-7–15-8
Modbus protocol, 15-6
ZigBee (*see* ZigBee)
Communications, wireless sensor network, 2-7–2-8
Complementary metal–oxide–semiconductor (CMOS) processes, 1-8
Complex waveforms (CW), 16-3
Compression-based methods, 11-6
Compressions, time series sensor streams, 11-7
Computer intelligence algorithms, 14-10
Connected DS (CDS), 21-1; *see also* Load-balanced CDS (LBCDS)
centralized algorithms, 21-4–21-5
distributed algorithms, 21-5
other algorithms, 21-5
Connected MIS (CMIS) problem, 7-3
definition, 7-8
NP-hard problem, 7-9
Consumer electronics, biometric sensors, 8-8–8-11
Continuous white noise acceleration model, 6-19
Controller area networks (CANs), 4-9
Conventional method, 16-4, 20-3, 20-10
COPE, wireless mesh networks, 4-10
Correlated failure in WSNs
example, 13-2
level of interaction parameters, 13-6
MRF-based model
average number of failed nodes, 13-10–13-13
covariance matrices, 13-8
effect of independent *vs*., 13-10, 13-12
failure pattern, 13-11–13-14
Gibbs sampler, 13-8
network connectivity effect, 13-10
53-node WSN topology, 13-8–13-9
statistical analysis, 13-8–13-10
T-test, 13-10–13-11
multilayer, 13-6–13-7
nodes
data exchange, 13-1
failures, 13-1–13-2
form, 13-1
sampling
Gibbs sampler, 13-7
Metropolis sample, 13-8
sink node, 13-8
topology using graph theory
Gibbs random fields, 13-4–13-5
Markov random fields, 13-4
neighborhood system, 13-3
node neighborhood, 13-3
Correlation analysis, 14-9
Crowdsourcing, 3-2
C Standard Library, A-1
Cumulative coded ACK (CCACK) scheme, 4-11

D

DASH7 Alliance, A-1
DASH7 Mode 2, A-1
Data acquisition, 16-1–16-3, 16-8, 16-11
Data aggregation, 7-1
communication mode
comparison, 21-17
network lifetime, 21-18–21-19
p-norm values, 21-17
SD of residual energy, 21-18–21-19
simulation environment, 21-17
energy efficiency, 4-2–4-3
example, 4-2
latency, 4-4–4-5, 7-5
schedule, 7-5
Data aggregation trees (DATs)
construction problem, 7-3
dense sensor deployment, 7-2
imbalanced, 7-3
potential/actual load, 7-4
shortest-path-based, 7-2
Data center sensors, 1-4–1-6
Data exploration, 11-3
Data fusion
decision-based, 4-6
DFuse, 4-7
distance-based, 4-7
distributed systems, 4-5
example, 4-6
two-tier system, 4-6–4-7
value-based, 4-6
Data gathering process, 7-1
Data integrity, WSNs, 5-6
Data management, 11-2
Data mining
affinity analysis, 11-3
association rules, 11-3
challenges, 11-4
classification, 11-2–11-3
clustering, 11-3
exploration/reduction, 11-3
overview, 11-2–11-4
prediction, 11-3
sensor networks, 11-4
challenges, 11-11
data stream classification, 11-5–11-6
data stream clustering, 11-5
frequent pattern mining, 11-6
steps, 11-4
visualization methods, 11-3–11-4
Data purification, 5-6
Data reduction, 11-3
Data reliability issues, 3-4–3-5
Data stream
classification, 11-5–11-6
clustering, 11-5

Data visualization, 11-3–11-4
DATs, *see* Data aggregation trees (DATs)
Decision-based data fusion, 4-6
Deep ultraviolet (DUV), 18-12
Defense Advanced Research Projects Agency (DARPA), 14-1
Denial-of-service (DoS) attack, 9-8
  classification, 12-2
  WSN, 12-3
Detector array technology, 18-1
Deterministic network model (DNM), 7-2, 21-3
Device Language Message Specification (DLMS) User Association, A-1
DFuse, 4-7
Digi Mesh, A-2
Dijkstra algorithm, 19-6
Dimension reduction
  definition, 11-3
  time series sensor streams, 11-7
Direction of arrival (DOA), 10-6–10-7
Distance-based data fusion, 4-7
Distributed algorithms, 11-9
Distributed sensor networks (DSNs), 5-1
DNM, *see* Deterministic network model (DNM)
DOA, *see* Direction of arrival (DOA)
Dominating set (DS), 21-1
Dominating tree (DT), 21-29
Dominator partition, 21-23
DoS attack, *see* Denial-of-service (DoS) attack
Double-integration method, 20-1, 20-4–20-7
DPLC, *see* Dynamic packet length control (DPLC)
Driver rerouting system
  back-end server infrastructure, 19-3
  congestion prediction, 19-5
  smartphone-based vehicular networks, 19-3
  system overview, 19-3–19-4
  traffic data representation and estimation, 19-4
  traffic guidance system, 19-3–19-4
  vehicles selection (*see* Rerouted vehicles)
DRS Technologies, 18-21
DSNs, *see* Distributed sensor networks (DSNs)
DS property constraint, 7-10
Dublin, smart city projects, 15-13
Dubuque, Iowa, smart city projects, 15-13
Dye-sensitized solar cell (DSSC), 18-10
Dynamic feedback (DYN) scheme, 4-9
Dynamic packet length control (DPLC), 4-9–4-10
Dynamic shortest path (DSP)
  advantage, 19-7
  A* shortest path with repulsion, 19-7
    algorithm, 19-8–19-9
    definition, 19-7–19-8
    example, 19-9–19-10
  proactive approach, 19-7
  risk of switching congestion, 19-7
Dynamic traffic assignment (DTA), 19-3, 19-14–19-15

E

EADAT algorithm, *see* Energy-aware distributed heuristic (EADAT) algorithm
ECD, *see* Efficient code dissemination protocol (ECD)
Eco-city, smart city project, 15-12
EENCR, *see* Energy-efficient network-coding-based routing protocol (EENCR)
Efficient code dissemination protocol (ECD), 4-12–4-13
EIF algorithm, *see* Equivalent information fusion (EIF) algorithm
EKF, *see* Extended KF (EKF)
Electronic Product Code (EPC) standard, 9-5–9-6
Energy-aware distributed heuristic (EADAT) algorithm, 4-3
Energy-efficient aggregation scheduling, 7-4–7-5
Energy-efficient network-coding-based routing protocol (EENCR), 4-11
Energy estimation, 14-9
Energy harvesting technology, 2-8–2-10
Energy-latency trade-off, data gathering, 4-4–4-5
Ensemble-based classification, 11-6
Entropy balanced k-shortest paths (EBkSP), 19-10–19-11
Environment
  mobile crowd sensing, 3-4
  sensors, 1-3
EPC standard, *see* Electronic Product Code (EPC) standard
Equivalent information fusion (EIF) algorithm, 6-11–6-12, 6-22
Ethernet, 15-5
European Intelligent Building Group, 15-1
European Smart Metering Industry Group (ESMIG), A-2
Even energy dissipation protocol (EEDP), 21-5
Exhaustion attacks
  energy-efficient solution, 12-2
  network, 12-3
  node, 12-3
Expected allocation probability (EAP), 21-10–21-12
Expected number of transmissions (ETX), 4-4
Extended KF (EKF)
  linearization, 6-15, 6-18
  node localization algorithm, 6-20
  nonlinear state estimation, 6-14
  probability distributions, 6-17
  time-varying node positions, 6-19
Extension of active systems, 20-11

F

Face biometric sensors, 8-6
False acceptance rate (FAR), 8-4
False rejection rate (FRR), 8-4

Field-programmable gate array (FPGA), 2-5–2-6
Filtering
process, 17-3
time series sensor streams, 11-7
Fingerprint sensors, 8-1–8-3
capacitive sensors, 8-4
consumer electronics and mobile devices, 8-9–8-11
live finger detector, 8-5
live scan, 8-4
minutiae, 8-4
optical sensors, 8-4–8-5
pressure sensing scanners, 8-5
RF type sensors, 8-5
static, 8-6
swipe sensor, 8-6
thermal sensor, 8-5–8-6
ultrasonic scanners, 8-5
Finite impulse response (FIR) filters, 10-5
Fitter operator, 21-8–21-9
Fixed aggregation scheme (FIX), 4-9
Flexible manufacturing systems, 1-12
Flow-balanced k-shortest paths, 19-11–19-13
Fluid dynamics process, 14-4
Focal plane arrays (FPAs), 18-1
Forecasting, time series sensor streams, 11-7–11-8
Fourier transform, 14-11
Frame format position, PHY-layer security solutions, 12-5–12-6
Frequency-modulated continuous-wave (FMCW)
imaging technique, 17-13–17-16
signaling, 16-1, 20-12
Frequent pattern mining, 11-6
Full-function devices (FFDs), 15-7
Full-scale field test, 16-7

G

GaN substrates, 18-5
Generalized cross-correlation, 14-11
Geolocation, *see* Unmanned underwater vehicle (UUV)
Gibbs distribution, 13-4–13-5, 13-7–13-8
Gibbs potential, 13-4–13-5
Gibbs random fields, 13-4–13-5
Gibbs sampler, 13-7
Glasgow, 15-13
Global positioning system (GPS) sensors, 3-8, 20-2
Google Maps, 19-2
Government e-services, 15-12
GPS-less radiolocation, 6-2
Greenshield's model, 19-4
Ground-penetrating radar (GPR) imaging, 16-1, 16-5–16-7
Growth Codes, 4-13
Gunshot detection system
acoustic signature
components, 14-3
muzzle blast wave (*see* Muzzle blast wave)
roadmap, 14-3
basic flow, 14-8–14-9
discriminating gunshot noise from background noise, 14-8–14-10
electroacoustic sensor networks
coverage, 14-8
transducer specifications, 14-7–14-8
transmission channels, 14-8
estimation of gunshot origin, 14-10–14-12
against firearm-related violence and crime, 14-2–14-3
implementation, 14-2
nonreported fire events, 14-3
schematic diagram, 14-2
SENTRI system, 14-2
ShotSpotter, 14-2

H

Hammersley–Clifford theorem, 13-4–13-5
Hand geometry, 8-7
Hardware abstraction architecture (HAA), 2-11
Hardware adaptation layer (HAL), 2-11
Hardware interface layer (HIL), 2-11
Hardware presentation layer (HPL), 2-11
Hash-based message authentication code (HMAC), 12-4
HDFT, *see* High-definition fiber tractography (HDFT)
Health care
applications, 11-8
innovation, 1-8
mobile crowd sensing applications, 3-3
Heat flow equation, 18-22–18-23
Heating, ventilation, and air conditioning (HVAC) systems, 1-3–1-4, 15-2–15-3
Heuristic function, 19-7–19-8
High-definition fiber tractography (HDFT), 1-9
HMAC, *see* Hash-based message authentication code (HMAC)
Hop neighborhood, 21-21
Huang's approach, latency, 7-5
HVAC systems, *see* Heating, ventilation, and air conditioning (HVAC) systems
Hybrid cell (HC), 18-10

I

IBM ThinkPad, 8-9–8-10
ICA method, *see* Independent component analysis (ICA) method
IEEE 802.11 (Wi-Fi)
mesh network, 15-8
SEP2, 15-9

IEEE 802.15.4, 2-7–2-8, 5-7, A-2
IEEE 802.15.4b, 5-9
Illumination waveform, resolution analysis, 17-9–17-10
ILR, *see* Improving location reliability (ILR)
Image reconstruction algorithm
  description, 17-2–17-3
  step-frequency FMCW system, 17-3–17-5
  superposition of range profiles, 17-5–17-6
Imaging system
  applications, 16-1
  development, 16-1
Imbalanced dominate allocation scheme, 21-10
Implanted medical devices, 1-8–1-9
Improving location reliability (ILR)
  action, 3-9
  adversarial model, 3-8–3-9
  assumptions, 3-8
  battery consumption, 3-19–3-20
  design, 3-9
  evaluating, 3-13
  fake photos
    location and, 3-17–3-18
    user earnings and, 3-16–3-17
  malicious user, 3-18
  node density, 3-15–3-16
  photo selection phase, 3-10–3-11
  reputation score, 3-18–3-19
  security analysis, 3-11–3-12
  transitive trust, 3-11
  user's survey results, 3-19
  validation process, 3-11
In-car navigation system, 19-4
Independent component analysis (ICA) method, 10-8
Independent failure patterns, 13-13
Information and communication technologies (ICT), 15-1
Infrared (IR)
  applications, 18-1
  bands, 18-2
  CNT-based bolometer (*see* Carbon nano tube (CNT))
  defining, 18-2
Inheritance population initialization (IPI), 21-28
In-network data aggregation technique, 7-1
In-network data processing, WSNs, 5-5–5-6, 5-8
INP, *see* Integer nonlinear programming (INP)
INRX service, 19-2
INSIGHT, WSNs, 5-10
Integer linear programming (ILP) formulation, 7-16–7-17
Integer nonlinear programming (INP)
  application-dependent
    data aggregation, 4-2–4-5
    data fusion, 4-5–4-8
    in WSNs, 4-1
  application-independent
    network coding, 4-10–4-13
    packet packing, 4-8–4-10
  LBMIS problem, 7-10
  research challenges
    joint optimization, 4-14
    systematic modeling and complexity analysis, 4-14
    theoretical foundations, 4-14–4-15
Intel Berkeley Research lab WSN, 13-8–13-9
Intelligent building, 15-1
Intel PXA255 processor, 2-2
Intel SA-1111 companion chip, 2-2
Intel XScale™ microprocessor, 2-2
International Biometric Group (IBG), 8-12
International Society of Automation (ISA), A-2
Internet Engineering Task Force (IETF), 15-7
Internet of Things (IoT), 1-1–1-2
  manufacturing and production, 1-9
  protocol stack, 15-11
Internet protocol (IP), 4-9, 15-5
Internet Protocol Version 6 (IPv6), A-2
Internet routing protocols, 19-2
Interrogator, *see* Radio frequency identification device (RFID), reader
Inverse cumulative distribution function (Inverse CDF), 13-7
IoT, *see* Internet of Things (IoT)
IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN), 2-8, 15-7–15-8, A-1
Iris recognition methods, 8-7
ISA-18.2 Alarm Management, A-2
ISA-88 Batch, A-2
ISA-95 Enterprise-Control System Integration, A-2
ISA 99 Cyber Security, A-2
ISA 100 Committee, A-2
ISA 100 Standard, A-2
ISA 100 Wireless, A-2
ISA 100 Wireless Compliance Institute, A-2
ISA 100.11a, A-2
ISA-101 HMI, A-2
ISA-103 Field Device Tools, A-2
ISA-104 Electronic Device Description Language, A-2
ISA-106 Procedural Automation, A-3
ISA-108 Intelligent Device Management, A-3
ISO 18000-7 standard, A-3
IS property constraint, 7-10
IT equipment, 15-3

J

JamBayes project, 19-2
Joint tactical information distribution system (JTIDS) program, 6-2

K

Kalman filter (KF), 6-2
Keystroke dynamics, biometric sensors, 8-7

L

LBDAT, *see* Load-balanced DAT (LBDAT)
LBPNA, 7-9
  approximation algorithm, 7-17–7-20
  leaf nodes, ILP formulation, 7-16–7-17
  for non–leaf nodes, 7-15–7-16
LBS, *see* Location-based services (LBS)
LBVBP-MOGA
  chromosomes, 21-27–21-28
  convergence analysis, 21-33–21-34
  design, 21-27
  dominating tree
    features, 21-30
    illustration, 21-30
    parent chromosomes selection, 21-29
    sibling chain, 21-30
  fitness function, 21-28–21-29
  genetic algorithm overview, 21-26–21-27
  genetic operations
    crossover operation, 21-30–21-32
    dominatee mutation, 21-32
    pseudocode, 21-32–21-33
  MOGA overview, 21-26–21-27
  multi objective problems, 21-26
  Pareto dominance, 21-26
  Pareto optimality, 21-26
  population initialization, 21-28
  step by step, 21-27
LEACH protocol, *see* Low-energy adaptive clustering hierarchy (LEACH) protocol
Lebesgue integration, 11-10–11-11
Level of service (LOS), 19-5
Lighting, HVAC system and, 15-3
Linear programming (LP) formulation, 7-5
LINK protocol, 3-12
Lisbon, 15-13
Load-balanced allocation of dominatee (LBAD)
  algorithm description
    allocation system, 21-12
    centralized algorithm, 21-13
    distributed algorithm, 21-13–21-15
    stochastic dominatee allocation, 21-12
  analysis, 21-15–21-16
  definition, 21-7
  terminologies, 21-9
    expected allocation probability, 21-10–21-12
    valid degree, 21-10
Load-balanced CDS (LBCDS), 21-2
  approximate algorithm, 21-8
  definition, 21-7
  example illustration, 21-8–21-9
Load-balanced DAT (LBDAT), 7-2–7-3
  construction problem, 7-4
  performance evaluation, 7-20–7-23
Load-balanced data aggregation tree, 7-16–7-20
Load-balanced maximal independent set (LBMIS)
  approximation algorithm, 7-11–7-14
  connecting, 7-14–7-15
  connector set, 7-3
  formulation, 7-10–7-11
  problem, 7-4
    definition, 7-7–7-8
    as INP, 7-10
Load-balanced virtual backbone (LBVB), 21-2–21-3
  cluster-based data gathering, 21-5–21-6
  DNM
    data aggregation communication mode, 21-17–21-19
    data collection communication mode, 21-19–21-20
    LBAD (*see* Load-balanced allocation of dominatee (LBAD))
    LBCDS (*see* Load-balanced CDS (LBCDS))
    network model, 21-6
    p-norm to load balance measure, 21-6–21-7
  PNM
    actual traffic load, 21-24
    assumptions, 21-21
    authorized link set, 21-23–21-24
    dominator partition, 21-23
    hop neighborhood, defined, 21-21
    LBVBP in probabilistic WSNs, 21-25
    LBVBP-MOGA (*see* LBVBP-MOGA)
    network lifetime, 21-34–21-35
    network model, 21-21
    one-hop neighborhood, defined, 21-21
    partition *p*-norm, 21-24–21-25
    p-norm, defining, 21-22
    potential traffic load, 21-22
    simulation environment, 21-34–21-35
    VB *p*-norm, defining, 21-22–21-23
Load-balance factor, 7-2
Load-balancing group clustering (LBGC), 21-6
Local area networks (LANs), 15-5
Localized power-efficient data aggregation protocol (L-PEDAP), 7-4–7-5
Local spatial frequency, 17-7
Local specifications, 13-4, 13-6, 13-14
Location-based services (LBS), 1-9
London, smart city projects, 15-13
Long wavelength infrared (LWIR), 18-2
LonTalk protocol, 15-6
LonWorks (local operating network), 15-6
Lossy links, 21-3
Low-energy adaptive clustering hierarchy (LEACH) protocol, 4-2
Low-power and lossy networks (LLNs), 15-7
Low-rate wireless personal area networks (LR-WPANs), 5-9

L-PEDAP, *see* Localized power-efficient data aggregation protocol (L-PEDAP)
LP formulation, *see* Linear programming (LP) formulation

M

Machine-to-machine (M2M) module, A-3
MAC-independent opportunistic routing protocol (MORE), 4-11
Malicious attacks, 12-3
MANNA management system, 5-11
Marketing, mobile crowd sensing applications, 3-3
Markov random fields (MRF), 13-4; *see also* Correlated failure in WSNs
Masdar City, UAE, smart city projects, 15-12
Matched filter, 20-3, 20-5–20-6, 20-8, 20-10
Maximal independent set (MIS)
  based algorithm, 21-5
  NP-hard problem, 7-9
Maximum-lifetime aggregation scheduling, 7-5
Maximum-lifetime data gathering with aggregation (MLDA) problem, 4-3, 7-5
Maximum power point tracking (MPPT), 2-9
mCrowd, 3-3
McSense
  android application, 3-6
  photo tasks, 3-10
  prototype implementation, 3-7
  system architecture and processes, 3-5
    background services, 3-7
    life cycle, 3-6–3-7
    new sensing tasks, 3-6
    user registration, 3-6
  user study and tasks
    automated resources, 3-8
    automated sensing task, 3-8
    manual photo sensing task, 3-7
MDST, *see* Minimum degree spanning tree (MDST)
Medical care innovation, 1-8
Medium access control (MAC) protocol, 4-3
Medusa, 3-3–3-4
MEID, *see* Mobile equipment identifier (MEID)
Metal-organic chemical vapor deposition (MOCVD), 18-5
Metamaterials, 18-2
MICA2 node, 2-4
Microcontroller, 2-4, 2-6
Microphone
  arrays, 10-2–10-3
    acoustic source localization, 10-5–10-8
    blind source separation, 10-8–10-11
    signal processing, 10-3–10-5
  beam steering, 10-3
  classification, 10-2
  delay-and-sum beamformer, 10-5
  direction of arrival, 10-6–10-7
  directivity pattern, 10-4
  finite impulse response filters, 10-5
  ICA algorithm, 10-10
  independent component estimation, 10-10
  invention, 10-1
  non-Gaussianity, 10-9
  signal, 14-2, 14-8, 14-11
  time series, 10-9
Microsoft's Bing, 19-2
Microwave imaging, 17-13–17-16
Midwave infrared (MWIR), 18-2
Minimum degree spanning tree (MDST), 7-4
Minimum-latency aggregation schedule (MLAS), 7-5
Minimum routing cost CDS (MOC-CDS), 21-2
Minimum-sized CDS (MCDS), 21-1
MLDA problem, *see* Maximum-lifetime data gathering with aggregation (MLDA) problem
Mobile crowd sensing
  applications, 3-2–3-4
  cities, 3-2
  citizen journalism, 3-4
  data reliability issues, 3-4–3-5
  false location claims, 3-14–3-15
  field study
    battery consumption, 3-19–3-20
    experimental evaluation, 3-12–3-14
    location and fake photos, 3-17–3-18
    malicious user, 3-18
    reputation score, 3-18–3-19
    task price on data quality, 3-13–3-14
    user earnings and fake photos, 3-16–3-17
    user's survey results, 3-19
  ILR, 3-8–3-12
  McSense, 3-5–3-8
  platforms, 3-3–3-4
  simulations, 3-14–3-16
Mobile devices, biometric sensors, 8-8–8-11
Mobile equipment identifier (MEID), 3-6
Mobile Millennium Project, 19-2
Mobile sensing applications, 3-2
Modbus protocol, 15-6
Modulation–demodulation process, 17-10
Moisture monitoring, 1-6–1-7
Monostatic case, resolution analysis, 17-10–17-11
MORE, *see* MAC-independent opportunistic routing protocol (MORE)
MT-Deluge, 4-12
Multicore technology, WSNs, 5-13–5-15
Multimodal biometric sensors, 8-8
Multi objective problems (MOPs), 21-26
Multiple-stage discrimination process, 14-9
Multiwall carbon nanotubes (MWCNT), 18-27
Muzzle blast wave
  directional attributes, 14-4–14-6
  outdoor propagation, 14-5–14-7
  shooter location estimations, 14-3–14-4

theoretical Friedlander wave form, 14-4–14-5
two-phase explosion, 14-4
velocity of several small arms, 14-4

N

Nanogenerators (NGs), 18-10
Nanostructures, microbolometer, 18-2
Nanowires (NWs), 18-5
Narrow-waist sensor net protocol, 5-11
NC, *see* Network coding (NC)
Near infrared (NIR), 18-2
Neighborhood system, 13-3
Nericell project, 19-2
Network coding (NC)
code dissemination and reprogramming, 4-12–4-13
network protection, 4-13
opportunistic routing, 4-10–4-12
Network exhaustion attack, 12-3
Network-layer diversification, WSNs, 5-6
Neuron chip, 15-6
New Songdo, South Korea, smart city project, 15-12
Neyman–Pearson (N–P) test, 4-6
Node exhaustion attack, 12-3
Node failures, 13-1–13-2
Node neighborhood, 13-3
Non-intended packets, nonmalicious *vs*. malicious, 12-3
Nonlinear regression, Choquet integral, 11-9–11-11

O

Observations and Measurements Schema (O&M), A-3
OGC Sensor Web Enablement Standards, A-3
Oil well imaging, 16-10–16-11
On-demand (OD)
classification, 11-6
scheme, 4-9
One-hop neighborhood, 21-21
One-stage algorithms, 21-5
OOSM fusion, *see* Out-of-sequence measurement (OOSM) fusion
Open standard interfaces, A-3
Operating systems, wireless sensor network, 2-10–2-12
Optical composite imaging
oil well imaging, 16-10–16-11
three-dimensional endoscopy, 16-12–16-13
3D terrains from image sequences, 16-11–16-12
Out-of-sequence measurement (OOSM) fusion, 6-9
Out-of-SYN message, 12-12–12-13

P

Packet packing
AQCM, 4-9
DPLC, 4-9–4-10
example, 4-8
FIX scheme, 4-9
joint optimization, 4-10
tPack, 4-10
Parallel guessing algorithm, 11-5
Parent node assignment (PNA), 7-2
for leaf node, 7-8
for non–leaf node, 7-8–7-9
Pareto-optimal front, 21-26
Pareto-optimal set, 21-26
Passive infrared (IR) sensors, 15-4
Passive reader active tag (PRAT) system, 9-5
PCA, *see* Principal component analysis (PCA)
Pearl River Tower, 15-9
PEDAP, *see* Power-efficient data-gathering and aggregation protocol (PEDAP)
PeerTIS, 19-2
PEGASIS, *see* Power-efficient gathering in sensor information systems (PEGASIS)
PEIR, 3-3
Percolation theory, 13-2, 13-14
Personal area networks (PANs), 15–6
Photoluminescence (PL), 18-5
close-up of device, 18-8, 18-10
LED lamp, 18-8
on–off switching characteristics, 18-8–18-9
pixel array detector device, 18-8, 18-10
on p-Si, GaN, and SiO_2 substrates, 18-8–18-9
3D optical fiber–based HC, 18-10–10-11
3D piezotronic transistor array, 18-10–18-11
PHY-layer security solutions
authentication preamble
IEEE 802.15.4e standard format, 12-5
method, 12-4–12-5
protocol uses, 12-6–12-8
window, 12-6
comparative table, 12-8–12-10
complete security suite, 12-8
frame format position, 12-5–12-6
Piezoelectric harvesting, 2-9–2-10
Piezoelectric NG, 18-10
PIF algorithm, *see* Predicted information fusion (PIF) algorithm
Platform, definition, 2-1
PLC, *see* Programmable logic controller (PLC)
Plug-load devices, 15-3
PNA, *see* Parent node assignment (PNA)
PNM, *see* Probabilistic network model (PNM)
Point-to-multipoint network protocols, A-3
Posteriori probability, 6-3–6-4
Potential load, 7-7
Potential traffic load, 21-22
Power-efficient data-gathering and aggregation protocol (PEDAP), 4-3, 7-4
Power-efficient gathering in sensor information systems (PEGASIS), 4-3
Power management, WSN, 2-12, 5-5
Power unit, wireless sensor network, 2-8–2-10

PRAT system, *see* Passive reader active tag (PRAT) system
Predicted information fusion (PIF) algorithm, 6-10–6-11, 6-22
Prediction, data mining, 11-3
PreFace with Camera API, 8-6
Principal component analysis (PCA), 11-7
Probabilistic network model (PNM), 7-2, 21-3–21-4
Probabilistic WSNs, 7-2
Probing waveforms, 16-1
Programmable logic controller (PLC), 1-10, 15-6
ProNCP protocol, 4-13
Protocol, AP, 12-6–12-8
Pulse-echo model, 17-2

Q

Quality of service (QoS)
  joint optimization, 4-14
  systematic modeling and complexity analysis, 4-14
Quantum dot superlattice (QDSL), 18-17
Quantum efficiency (QE), 18-5

R

Radio frequency (RF) communication, 1-4
Radio frequency identification device (RFID)
  concerns, 9-8–9-9
  controlling access, 9-6
  control unit, 9-5
  implementations, 9-6–9-7
  interface, 1-9
  logo to stop, 9-8
  reader, 9-1, 9-4–9-5
  retail store using, 9-7
  standards, 9-5–9-6
  tag
    active tags, 9-3
    aspects, 9-3
    bar codes, 9-1–9-2
    classes, 9-3–9-4
    cryptography, 9-4
    frequency signals, 9-2
    passive tags, 9-3
    semiactive tags, 9-3
Random k-shortest paths (RkSP), 19-10
Random linear coding (RLC), 4-10
Range profiles, superposition, 17-5–17-6
Range resolution, 17-5
Rateless Deluge, 4-12
Rayleigh resolution limit, 17-8
Received signal strength indicator (RSSI) measurements, 6-2
RecognizeMe app, 8-11
Reconfigurable device, microcontroller and, 2-6
Reduced-function devices (RFDs), 15-7
Reduction, *see* Data reduction
Region-based connectivity, 13-2
Rerouted vehicles
  alternative route computation and assignment, 19-6
  dynamic shortest path (*see* Dynamic shortest path (DSP))
  dynamic traffic assignment, 19-14–19-15
  evolution
    average travel time, 19-17–19-24
    experimental settings, 19-15–19-17
  level of service, 19-5
  multiple shortest path strategies, 19-6
    entropy balanced k-shortest paths, 19-10–19-11
    flow-balanced k-shortest paths, 19-11–19-13
    random k-shortest paths, 19-10
  parameter L (level), 19-5
  process, 19-5, 19-13–19–14
  ranking, 19-5–19-6
  single shortest path strategies, 19-6
Resolution analysis
  active illumination, 17-9–17-10
  case of centered aperture, 17-8–17-9, 17-11
  description, 17-6–17-7
  monostatic case, 17-10–17-11
  step-frequency FMCW systems, 17-11–17-13
Reversed system, 20-10–20-11
Revolvers, 14-3–14-4
ReXOR, 4-12
RF communication, *see* Radio frequency (RF) communication
RFID, *see* Radio frequency identification device (RFID)
Rio de Janeiro, smart city projects, 15-13
Rise-time derivation, 14-9
RLC, *see* Random linear coding (RLC)
Road transportation, mobile crowd sensing applications, 3-3
RSSI measurements, *see* Received signal strength indicator (RSSI) measurements

S

Sapphire, 18-5–18-8, 18-13–18-15
SAS, *see* Sequential aggregation scheduling (SAS)
Scalability, sensor data, 11-1
Scanning electron microscopy (SEM), 18-5
Scanning laser acoustic microscope (SLAM), 16-2
Scanning tomographic acoustic microscopy (STAM)
  acoustic modality, 16-2
  conventional SLAM system
    prototype system, 16-2
    subsurface range finder, 16-2
    two subsurface layer, 16-2–16-3
  estimation of tissue hardening, 16-4
  experiment of penny imaging, 16-3–16-4
  first and second layer image, 16-3
  multiple-frequency data, 16-3

objective, 16-2
research program, 16-4
subsurface layer image, 16-2
Seattle launched a High-Performance Building program, 15-9
SECURES, *see* System for the Effective Control of Urban Environment Security (SECURES)
Security suite, 12-8
Sensing, wireless sensor network, 2-7
Sensor data
analytics, 11-2
mining, 11-2
processing, 11-2
scalability, 11-1
Sensor Enabled Neural Threat Recognition and Identification (SENTRI) system, 14-2, 14-8
Sensor Model Language (SensorML), A-3
Sensor networking
data center sensors, 1-4–1-6
data mining, 11-4
challenges, 11-11
data stream classification, 11-5–11-6
data stream clustering, 11-5
frequent pattern mining, 11-6
definition, 11-1
deployments, 1-12
environmental, 1-3
fields using, 1-2–1-3
function, 1-1
hardware architecture and application, 1-11
HVAC systems, 1-3–1-4
location-based services, 1-9
manufacturing and production, 1-9
plastic curtains data center, 1-4, 1-6
spray foam insulation, 1-6–1-8
vision sensing, 1-10–1-12
wireless network, 1-1–1-2
Sensor Planning Service (SPS), A-3
Sensor Registries, A-3
Sensors
energy sensing, 15-4–15-5
environmental sensing, 15-4
occupancy and user context, 15-4
smart meter, 15-5
Sequential aggregation scheduling (SAS), 7-5
Shortwave infrared (SWIR) detector technologies, 18-2
IR imaging advantage, 18-18
$In_xGa_{1-x}As$ detector array development, 18-18–18-21
Shotguns, 14-3
ShotSpotter, 14-2, 14-8
Signal processing, microphone arrays, 10-3–10-5
Signal to Noise Ratio (SNR), 18-4
Si micro electro mechanical system (Si MEMS), 18-1
Simulation of urban mobility (SUMO), 19-15–19-16
Simulations, mobile crowd sensing, 3-14–3-16
Single-objective problems (SOPs), 21-27
Single-wall carbon nanotubes (SWCNT), 18-27
Sink node constraint, 7-10
SiO_2 substrates, 18-5
Sliding window model, 11-6
Smart buildings; *see also* Intelligent building
building automation, 15-2
challenges, 15-9
characteristics and technologies
automated control, 15-3
communication protocol (*see* Communication protocols)
data driven, 15-3
integration of systems, 15-3–15-4
sensors, 15-4–15-5
example projects, 15-9
motivation for change, 15-2–15-3
notion, 15-1
smart grid, 15-2
survey of definition, 15-1
Smart city
definition, 15-1–15-2
example projects, 15-12–15-13
issues, 15-9
operations, 15-2
smart grid, 15-2
urban population trends, 15-10
vision
data, 15-11
dimensions, 15-11
infrastructure, 15-11
services, 15-12
users, 15-12
Smart Energy Profile 2.0 (SEP2), 15-9
Smart grid, 15-2
Smart meter, 15-5, 15-14
Smartphones, mobile sensors, 3-1–3-2
Smart Santander project, 15-13
Social networks, 19-2
Social sensing, 11-8–11-9
SoC platform, architecture, 2-4–2-5
Solar power harvesting, 2-9
SPC, *see* Statistical process control (SPC)
Spectral analysis, 14-9
Spray foam insulation, 1-6–1-8
SSP, *see* Sufficient statistics packet (SSP)
Stability, 20-1, 20-11–20-12
Standard network variable types (SNVTs), 15-6
Standards-compliant wireless operation, A-3
Stargate, 2-2
State machine, 2-5
Statistical process control (SPC), 1-10
Stefan–Boltzmann law, 18-24
Steiner tree technique, 21-5
Step-frequency FMCW systems, 17-1
image reconstruction algorithm, 17-3–17-5
resolution analysis, 17-11–17-13
Stochastic user equilibrium (SUE), 19-15

Strain Gated Vertical Piezo-Electric Transistor (SGVPT) array, 18-12
StreetSmart, 19-2
Subtraction-based algorithm, 21-5
Sufficient statistics packet (SSP)
  communication delay, 6-10
  delayed, 6-11–6-12
  formation, 6-13–6-14
  fusion, 6-8
  generation, 6-7
  reprocessing, 6-9–6-11
SUKF
  vs. delayed PIF and EIF, 6-21
  linearization, 6-17–6-18
  node localization algorithm, 6-20
  range measurement standard deviation, 6-20–6-21
  time-varying node positions, 6-19
Superposition, range profiles, 17-5–17-6
Supervised learning algorithms, 11-3–11-4
Surface street estimation, 19-2
SYN message, 12-12–12-13
Synthetic-aperture sonar (SAS) imaging, 16-5
System for the Effective Control of Urban Environment Security (SECURES), 14-1–14-2, 14-8

T

Target acquisition, 14-1
TelosB, 2-3
Temperature coefficient of resistance (TCR), 18-21
Temperature-dependent electrical characteristics, 18-22
Terahertz imaging
  hardware setup, 16-9
  operates in reflection mode, 16-7
  penetration capability, 16-8
  resolution capability, 16-8
  subsurface image, 16-10
  system diagram, 16-9
Three-dimensional endoscopy, 16-12–16-13
Tiered architectures, 1-4
Time delay of arrival (TDOA), 14-11
Time-domain analysis, 14-9
Time series sensor streams, 11-6
  compressions and filtering, 11-7
  dimensionality reduction, 11-7
  forecasting, 11-7–11-8
Time-synchronized mesh protocol (TSMP), A-3
TinyOS, 2-11–2-12
Tmote Sky, 4-12
Tomographic imaging system
  ground-penetrating radar imaging, 16-1, 16-5–16-7
  STAM (*see* Scanning tomographic acoustic microscopy)
  synthetic-aperture sonar imaging, 16-5
  terahertz imaging (*see* Terahertz imaging)
TPMs, *see* Trusted Platform Modules (TPMs)
TraCI, 19-15–19-16
Traffic jam alerts, mobile crowd sensing, 3-4
TrafficView, 19-2
Transducer, 20-2
Transducer Markup Language (TML), A-3
Transient shape analysis, 14-9
Transmission mode waveforms, 16-3
Tree-based CDS algorithm, 21-5
Tree-based topology, 7-1–7-2, 7-4
Trusted Platform Modules (TPMs)
  mobile crowd sensing, 3-4
  trusted hardware, 3-12
Two-stage algorithms, 21-5

U

Ultrasound
  data-acquisition hardware, 17-14
  transceiver array prototype, 17-13
Ultraviolet (UV)
  applications
    GaN UVAPD development (*see* Ultra Violet Avalanche Photodiodes (UVAPDs))
    passive and active UV systems design, 18-4
    systems performance, 18-5
    ZnO/MgZnO nanostructures (*see* ZnO NW arrays)
  band, 18-2–18-3
  defense and commercial applications, 18-2
Ultra Violet Avalanche Photodiodes (UVAPDs)
  alloy composition of AlGaN, relationship between, 18-12–18-13
  GaN/AlGaN, 18-12–18-15
Ultra wideband (UWB), 2-7–2-8
Ultra-wideband networks (UWBNs), 4-9
Unclustered correlated failures, 13-13
Unit disk graph (UDG) model, 21-3
Unmanned underwater vehicle (UUV)
  acoustic sensor unit, 20-2–20-4
  bearing angle estimation
    after nonlinear conversion, 20-9–20-10
    before/after normalization, 20-9
    with conventional method, 20-3
  double-integration method, 20-4–20-7
  extension of active systems, 20-11
  reversed system, 20-10–20-11
Unscented KF (UKF)
  nonlinear state estimation, 6-14
  probability distributions, 6-17
  radiolocation, 6-15
  statistical linearization, 6-15
Unsupervised learning algorithms, 11-3–11-4
US Intelligent Building Institute, 15-1
UWBNs, *see* Ultra-wideband networks (UWBNs)

V

Valid degree (VD), 21-10
Value-based data fusion, 4-6
Vehicle tracking applications, 11-8
Vehicular ad hoc networks (VANETs), 4-5
Vehicular sensor networks
  congestion avoidance
    distributed systems, 19-2
    driver rerouting system for (*see* Driver rerouting system)
    dynamic traffic assignment, 19-3
    real-time traffic guidance system, 19-3
    road traffic routing protocol, 19-2
    traffic equilibrium principle, 19-3
    vehicle probe data, 19-2
    witness web-based services/applications, 19-2
  GPS receivers and navigation systems, 19-1
  IEEE 802.11p, 19-1
  road safety
    abnormal driver behavior, 19-2
    highway cooperative collision avoidance, 19-1–19-2
Vehicular systems, mobile sensors, 3-1–3-2
Very fast decision tree (VFDT) method, 11-6
Video surveillance, 14-2
VigilNet, 5-9–5-10
Virtual backbone (VB), 21-1–21-2
Visible–NIR detector technologies
  IR imaging advantage, 18-15
  $Si_{1-x}Ge_x$ (SiGe) detector arrays, 18-15–18-17
Vision sensing, 1-10–1-12
Visualization, 1-12
Voiceprint biometric sensors, 8-7
VOx technologies, 18-21

W

Wavefield detection, 16-1
Web-based services, 19-4
Wideband pulses, 16-1
WiFi, A-3
Wireless, access units, 1-7
WirelessHART, 2-8, A-3
Wireless local area networks (WLANs), 4-9
Wireless personal area network (WPAN), 12-2
Wireless sensor networks (WSNs)
  applications, 13-1
  architecture, 2-1–2-5
  changing side length, 7-20–7-21
  characteristics, 21-1
  communications, 2-7–2-8
  connected DSs, 21-1
    centralized algorithms, 21-4–21-5
    distributed algorithms, 21-5
    MOC-CDS, 21-5
  correlated failures in (*see* Correlated failure in WSNs)
  decentralized information filter, 6-2
  deterministic/probabilistic, 21-3–21-4
  DoS attacks, 12-3
  energy-efficient aggregation scheduling, 7-4–7-5
  geolocation, 6-2
  heterogeneous architectures, 5-14
  hierarchal architecture
    abstract region, 5-14
    adaptability, 5-8
    agent-based architecture, 5-12–5-13
    configurable applications, 5-8
    cross-layer, 5-5, 5-7–5-8
    data-centric architecture, 5-9
    data integrity, 5-6
    distributed architectures, 5-8
    existing, 5-9–5-13
    hierarchical, 5-2
    information exchange service, 5-12
    in-network data processing, 5-5–5-6, 5-8
    issues, 5-4–5-5
    layered-architecture and modularity, 5-7
    military applications, 5-1
    motivation, 5-3–5-4
    multicore device, 5-13–5-15
    narrow-waist sensor net protocol, 5-11
    network-layer diversification, 5-6
    protocols, 5-5
    resources allocation and management, 5-5
    security and privacy, 5-6
    self-securing and self-healing architecture, 5-14
    standardization, 5-6–5-7
    topologies, 5-3
  information graphs, 6-1–6-2
  INP in, 4-1
  INSIGHT, 5-10
  integrated architecture, 5-14–5-15
  LBVB problem in (*see* Load-balanced virtual backbone (LBVB))
  load balancing application in, 21-5–21-6
  MANNA management system, 5-11
  maximum-lifetime aggregation scheduling, 7-5
  minimum-latency aggregation scheduling, 7-5
  network model
    h-hop neighborhood, 7-7
    1-hop neighborhood, 7-6
  node transmission range, 7-21–7-22
  non–leaf nodes, 7-22–7-23
  operating systems, 2-10–2-12
  PHY-layer security solutions
    authentication preamble (*see* Authentication preamble (AP))
    comparative table, 12-8–12-10
    complete security suite, 12-8
    frame format position, 12-5–12-6

platform, 2-1
power unit, 2-8–2-10
probabilistic, 7-3
problem definition, 7-7–7-9
reconfigurable, 2-5–2-6
sensing, 2-7
simulation environment, 7-20
state of art, 12-3–12-4
technology, 12-1
transitional region phenomenon, 7-2
transmission success ratio, 7-2
tree-based topology, 7-1–7-2, 7-4
VigilNet, 5-9–5-10
virtual backbone, 21-1–21-2
ZigBee, 15-7
WLANs, *see* Wireless local area networks (WLANs)
WPAN, *see* Wireless personal area network (WPAN)
WSNs, *see* Wireless sensor networks (WSNs)

X

X-ray diffraction (XRD), 18-5
XScale, 2-2

Y

Yule–Walker equation, 11-7–11-8

Z

ZigBee
Alliance, A-3
full-function devices, 15-7
Home Automation profile, 15-7
network topologies, 15-7
PRO, A-3
protocol stack, 15-6–15-7
reduced-function devices, 15-7
Smart Energy 1.1 or higher standard implementation, A-4
WSN, 2-7
ZnO NW arrays
characterization on various substrates
PL measurements (*see* Photoluminescence (PL))
SEM images, 18-6–18-7
XRD pattern, 18-6–18-8
ZnO epilayer quality, 18-6
design and performance of 3D HC, 18-10–18-11
dye-sensitized, 18-10
high UV sensitivity and low visible sensitivity, 18-5–18-6
as nanogenerator, 18-10
photoresponse switching properties, 18-5