Wireless Sensor and Actuator Networks

Algorithms and Protocols for Scalable Coordination and Data Communication

Edited by

Amiya Nayak and Ivan Stojmenovic
Wireless Sensor and Actuator Networks
## Contents

Preface ix  
Contributors xv  

1. Applications, Models, Problems, and Solution Strategies  
   Hai Liu, Amiya Nayak, and Ivan Stojmenovic

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1. Wireless Sensors</td>
<td>1</td>
</tr>
<tr>
<td>1.2. Single-Hop Wireless Sensor Networks</td>
<td>2</td>
</tr>
<tr>
<td>1.3. Multihop Wireless Sensor Networks</td>
<td>3</td>
</tr>
<tr>
<td>1.4. Event-Driven, Periodic, and On-Demand Reporting</td>
<td>4</td>
</tr>
<tr>
<td>1.5. Unit Disk Graph Modeling, Hop Count Metric, and Probabilistic</td>
<td>7</td>
</tr>
<tr>
<td>Reception</td>
<td></td>
</tr>
<tr>
<td>1.6. Adjustable Transmission Range and Power Metric</td>
<td>9</td>
</tr>
<tr>
<td>1.7. Cost Metrics</td>
<td>10</td>
</tr>
<tr>
<td>1.8. Sleep and Active State Modeling</td>
<td>11</td>
</tr>
<tr>
<td>1.9. Architectures for Wireless Sensor and Actuator Networks</td>
<td>12</td>
</tr>
<tr>
<td>1.10. Simple Models and Application of Wireless Sensor and Actuator</td>
<td>15</td>
</tr>
<tr>
<td>Networks</td>
<td></td>
</tr>
<tr>
<td>1.11. Generating Connected Wireless Sensor and Actuator Networks</td>
<td>17</td>
</tr>
<tr>
<td>1.13. Problems at Physical, MAC, and Transport Layers</td>
<td>19</td>
</tr>
<tr>
<td>1.14. Problems at the Network Layer</td>
<td>22</td>
</tr>
<tr>
<td>1.15. Localized Protocols as the Solution Framework</td>
<td>25</td>
</tr>
<tr>
<td>1.16. Implementation of Sensor Motes</td>
<td>27</td>
</tr>
<tr>
<td>1.17. Experiments On Test Beds</td>
<td>28</td>
</tr>
<tr>
<td>1.18. Experiences with the Development of Sensor Network Systems</td>
<td>29</td>
</tr>
<tr>
<td>References</td>
<td>30</td>
</tr>
</tbody>
</table>
## 2. Energy-Efficient Backbones and Broadcasting in Sensor and Actuator Networks

*Hai Liu, Amiya Nayak, and Ivan Stojmenovic*

2.1. Backbones 33  
2.2. Grid Partitioning-Based Backbones 35  
2.3. Clustering-Based Backbones 36  
2.4. Connected Dominating Sets as Backbones 38  
2.5. Overview of Broadcasting Techniques 48  
2.6. Physical Layer-Based Flooding, Neighbor Detection and Route Discovery 58  
2.7. Parameterless Broadcasting for Delay Tolerant-Networks 59  
2.8. Backbones and Broadcasting in Sensor–Actuator Networks 61  
2.9. RNG and LMST 64  
2.10. Minimal Energy Broadcasting 66  
References 70  

## 3. Sensor Area Coverage

*Hai Liu, Amiya Nayak, and Ivan Stojmenovic*

3.1. Problems, Models, and Assumptions 75  
3.2. Coverage and Connectivity Criteria 78  
3.3. Area-Dominating Set Based Sensor Area Coverage Algorithm 81  
3.4. Asynchronous Sensor Area Coverage 83  
3.5. Synchronous Sensor Area Coverage 85  
3.6. Multicoverage By Sensors 88  
3.7. Physical Layer-Based Sensing, Protocols, and Case Studies 89  
3.8. Operation Range Assignment in WSANs 90  
References 91  

## 4. Geographic Routing in Wireless Sensor and Actuator Networks

*Hai Liu, Amiya Nayak, and Ivan Stojmenovic*

4.1. Flooding-Based Routing and Georouting in Sensor Networks 96  
4.2. Greedy, Projection, and Direction-Based Routing 97  
4.3. Applications of Cost to Progress Ratio Framework to Georouting 100  
4.4. Memorization-Based Georouting with Guaranteed Delivery 103  
4.5. Guaranteed Delivery without Memorization 105  
4.6. Beaconless Georouting 114  
4.7. Georouting with Virtual and Tree Coordinates 117  
4.8. Georouting in Sensor and Actuator Networks 118  
4.9. Link Quality Metric in Sensor and Actuator Networks 119  
4.10. Physical Layer Aspects and Case Studies of Georouting 120  
References 122
5. Multicasting, Geocasting, and Anycasting in Sensor and Actuator Networks

Arnaud Casteigts, Amiya Nayak, and Ivan Stojmenovic

5.1. Multicasting 127
5.2. Geocasting with Guaranteed Delivery 134
5.3. Rate-Based Multicasting 143
5.4. Anycasting with Guaranteed Delivery 147
References 150

6. Sink Mobility in Wireless Sensor Networks

Xu Li, Amiya Nayak, and Ivan Stojmenovic

6.1. Introduction 153
6.2. Energy Hole Problem 155
6.3. Energy Efficiency by Sink Mobility 160
6.4. Sink Mobility in Delay-Tolerant Networks 162
6.5. Sink Mobility in Real-Time Networks 172
References 181

7. Topology Control in Sensor, Actuator, and Mobile Robot Networks 185

Arnaud Casteigts, Amiya Nayak, and Ivan Stojmenovic

7.1. Introduction 185
7.2. General Approaches In Static Sensor Networks 186
7.3. The Minimum Spanning Tree 187
7.4. Data Aggregation 189
7.5. Spanning Trees in Uncontrolled Dynamic Topologies 193
7.6. Detection of Critical Nodes and Links 195
7.7. Biconnected Robot Team Movement for Sensor Deployment 197
7.8. Augmentation Algorithm for Robot Self Deployment 198
7.9. Biconnectivity From Connectivity without Additional Constraints 200
7.10. Biconnectivity from Connectivity with Additional Constraints 203
References 206

8. Location Service in Sensor and Mobile Actuator Networks 209

Xu Li, Amiya Nayak, and Ivan Stojmenovic

8.1. Introduction 209
8.2. Classification of Location Services 210
8.3. Location Update Policies 212
8.4. Flooding-Based Algorithms 212
8.5. Quorum-Based Algorithms 219
Contents

8.6. Home-Based Approaches 225
References 229

9. Coordination in Sensor, Actuator, and Robot Networks 233

Hai Liu, Veljko Malbasa, Ivan Mezei, Amiya Nayak, and Ivan Stojmenovic

9.1. Sensor-Actuator Coordination 233
9.2. Task Assignment in Multirobot Systems 236
9.3. Selecting Best Robot(s) when Communication Cost is Negligible 238
9.4. Selecting Best Robot(s) with Nonnegligible Communication Costs 240
9.5. Dynamic Task Assignment 244
9.6. Deploying Sensors to Improve Connectivity 245
9.7. Fault-Tolerant Semipassive Coordination Among Actuators 247
9.8. Dispersion of Autonomous Mobile Robots 248
9.9. Distributed Boundary Coverage by Robots 249
9.10. Clustering Robot Swarms 250
9.11. Robot Teams for Exploration and Mapping 251
9.13. Flying Robots 258
References 259

10. Sensor Placement in Sensor and Actuator Networks 263

Xu Li, Amiya Nayak, David Simplot-Ryl, and Ivan Stojmenovic

10.1. Introduction 263
10.2. Movement-Assisted Sensor Placement 264
10.3. Mobile Sensor Migration 265
10.4. Sensor Placement by Actuators 266
10.5. Coverage Maintenance by Actuators 271
10.6. Sensor Self-Deployment 272
10.7. Sensor Relocation 287
References 292

Index 295
Preface

Traditional and existing sensor and actuator networks use wired communications, whereas, wireless sensors provide radically new communication and networking paradigms, and myriad new applications. They have small size, low battery capacity, non-renewable power supply, small processing power, limited buffer capacity (thus routing tables, if used at all, must be small), a low-power radio, and lack unique identifiers. Sensors may measure distance, direction, speed, humidity, wind speed, soil makeup, temperature, chemicals, light, vibrations, motion, seismic data, acoustic data, strain, torque, load, pressure, and so on. These nodes are autonomous devices with integrated sensing, processing, and communication capabilities. Nodes in a sensor network are generally densely deployed. Thousands of sensors may be placed, mostly at random, either very close or inside a phenomenon to be studied. Once deployed, the sensors are expected to self-configure into an operational wireless network, and must work unattended. The limited energy budget at the individual sensor level implies that in order to ensure longevity, the transmission range of individual sensors needs to be restricted. In turn, this implies that wireless sensor networks must be multi-hop ones.

Current research and implementation efforts are mostly oriented toward a traditional scenario with stationary sensors and a single static sink that collects information from sensors where the sink is directly connected to the user (or task manager). However, the latest research has unearthed the practically unsolvable problem of uneven energy distribution and energy holes in this scenario. Therefore, generalized scenarios have been considered, such as sensor networks with multiple stationary sinks, single mobile sink, or multiple mobile sinks. Mobile sensors have also been discussed. And, we envision adding actuators to the network. The difference between sinks and actuators is that actuators are able to act on the environment; mobile actuators may additionally act on the sensors. Actuators may also perform the roles of sinks, or both sinks and actuators may coexist in a given network. We will now elaborate on aspects of actuation.

Sensor–actuator networks are heterogeneous networks that comprise networked sensor and actuator nodes which communicate among each other using wireless links to perform distributed sensing and actuation tasks. Actuators (called
also actors) are resource-rich, potentially mobile, and are involved in taking decisions and performing appropriate actions on themselves (e.g. controlled movement), on sensors (such as activating sensors, moving or replacing a sensor), and/or in the environment (e.g. turn on their own a water sprinkler to stop the fire). Sensor–actuator networks are expected to operate autonomously in unattended environments. They may be directly connected (using, for instance, web infrastructure) and responsive to a user (task manager) who controls the network via sinks. One or more actuator(s) may also play the role of sink(s). In fact, sinks can be treated as special kinds of actuators, although a better interpretation might be to associate them with base stations that communicate directly with the user.

Since the actuating task is a more complicated and energy-consuming activity than the sensing task, actuators are resource-rich nodes equipped with better processing capabilities, higher transmission powers, and longer battery life. Moreover, depending on the application, there may be a need to rapidly respond to sensor input. Therefore, the issue of real-time communication is very important since actions are performed on the environment after the sensing occurs. In addition, while the number of sensor nodes deployed to achieve a specific application objective may be in the order of hundreds or thousands, such a dense deployment is not necessary for actuator nodes due to the different coverage requirements and physical interaction methods of acting a task. Hence, the number of actuators is much less than that of sensors.

The goal of this book is to present a fault-tolerant, reliable, low latency, and energy-aware framework for wireless sensor and actuator networks, so that the ultimate goal of their applications (protecting critical infrastructures, enabling timely emergency responses, and environment monitoring) can be fulfilled. Future sensor–actuator networks will be more heterogeneous and radically distributed, potentially with millions of nodes. They may respond to multiple tasks, to multiple and potentially mobile sinks and/or actuators, and multiple sensor networks may be integrated into a single network. There are algorithmic challenges in the rapidly emerging field of future heterogeneous super-networks where sensor networks will be integrated into wired and/or wireless infrastructure. Challenges of such wide-area sensor systems include scalability, robustness, manageability, and actuation. Having this futuristic vision in mind, this book will provide a protocol framework at the network layer, namely data communication and coordination issues. While being general, the framework should generate optimal solutions when applied/mapped in any specific emerging application; that is, the very same protocols may be applied in scenarios ranging from a simple scenario with one fixed actuator to the envisioned super-networks. To achieve such ambitious goals, several primary criteria for protocol design must be followed: energy consumption, localized design, reliability, parameterless behavior, and simplicity.

This book is problem-oriented, with each chapter discussing computing and communication problems and solutions that arise in rapidly emerging wireless sensor and actuator networks. The main direction of the book is to review various algorithms and protocols that have been developed in the area with emphasis on the most recent ones. The book is intended to cover a wide range of recognized
problems in sensor–actuator networks, striking a balance between theoretical and practical coverage. The theoretical contributions are limited to the scenarios and solutions that are believed to have practical relevance. This book is unique in addressing sensor and actuator/actor networking in a comprehensive manner, covering all the aspects, and providing up-to-date information. It is an appropriate and timely forum, where industry, operators, and academics from several different areas can learn more about current trends and become aware of the possible architectures of sensor and actuator networks, their advantages, and their limits in future commercial, social, and educational applications.

This book is intended for researchers and graduate students in computer science and electrical engineering, and researchers and developers in the telecommunication industry. It is directed at those who are looking for a reference resource in sensor and actuator networking and those who want to get a global view of this area.

The book is based on a number of stand-alone chapters that together cover the subject matter in a fully comprehensive manner. As a result of the exponential growth in the number of studies, publications, conferences, and journals on sensor networks, a number of graduate courses fully or partially concentrating on sensor networks have emerged recently. It is expected that this book will act as a supplemental textbook for such graduate courses. It can be also used as a stand-alone textbook for a course specifically on wireless sensor and actuator networks. The chapters cover subjects describing state-of-the-art approaches and surveying the existing important solutions. They provide readable but informative content, with appropriate illustrations, figures, and examples. A number of chapters also provide some problems and exercises for use in graduate courses.

The book content addresses the dynamic nature of wireless sensor and actuator networks. Due to frequent node addition and deletion from networks (changes between active and sleeping periods, done to conserve energy, are one of the contributors to this dynamic), and possible node movement, the algorithms that can be potentially used in real equipments must be localized and must have minimal communication overhead. The overhead should consider both the construction and its maintenance for the structure used in solutions and ongoing protocols. We believe that this is the only approach that will eventually lead to the design of protocols for real applications. We will explain now our design principles and the priorities given to the coverage of topics in this book.

A scalable solution is one that performs well in a large network. Sensor networks may have hundreds or thousands of nodes. Priority is given to protocols that perform well for small networks, and perform significantly better for large networks (more precisely, are still working as opposed to crashing when other methods are applied). In order to achieve scalability, new design paradigms must be applied. The main paradigm shift is to apply localized schemes as opposed to most existing protocols that require global information. In a localized algorithm, each node makes protocol decisions solely based on knowledge about its local neighbors. In addition, the goal is to provide protocols that will minimize
the number of messages between nodes, because bandwidth and power are limited. Protocols should use a small constant number of messages, often even none beyond preprocessing ‘hello’ messages. Localized message-limited protocols provide scalable solutions. Typical local information to be considered is one-hop or two-hop neighborhood information (information about direct neighbors and possibly the neighbors of neighbors). Nonlocalized distributed algorithms, on the other hand, typically require global network knowledge, including information about the existence of every edge in the graph. The maintenance of global network information, in the presence of mobility or changes between sleep and active periods, imposes a huge communication overhead, which is not affordable for bandwidth- and power-limited nodes. In addition to being localized, protocols are also required to be simple, easy to understand and implement, and to have good average case performance. Efficient solutions often require position information. It was widely recognized that sensor networks can function properly only if reasonably accurate position information is provided to the nodes.

**BRIEF OUTLINE OF THIS BOOK**

This book consists of 10 chapters. It begins with an introductory chapter that describes various scenarios where sensor and actuator networks may be applied, problems at physical, medium access, network, and transport layers, and various application layer tools for enabling applications. It argues for the use of localized algorithms, and discusses the generation of sensor and actuator networks for simulation purposes.

Chapter 2 discusses backbones as subsets of sensors or actuators that suffice for performing basic data communication operations. They are applied for energy-efficient data dissemination tasks. The goal is to minimize the number of re-broadcasts while attempting to deliver messages to all sensors or actuators. Neighbor detection and route discovery algorithms that consider a realistic physical layer are described. An adaptive broadcasting protocol without parameters, suitable for delay-tolerant networks, is further discussed. We also survey existing solutions for the minimal energy broadcasting problem where nodes can adjust their transmission powers.

Sensor networks normally have redundancy for sensing coverage. Some sensors are allowed to sleep while preserving network functionality. Sensors should decide which of them should be active and monitor an area, and which of them may sleep and become active at a later time. Sensor area coverage problem has been considered for both the unit disk graph– and physical layer–based sensing models in Chapter 3. Actuators may similarly run a protocol to decide about their service areas, releasing some of them from their particular duty. Operational range assignment for both sensor and actuators nodes is also discussed.

Chapter 4 surveys existing flooding-based and position-based routing schemes. It also describes a general cost-to-progress ratio-based approach for designing routing protocols under a variety of metrics, such as hop count, power, remaining energy, delay, and others. Chapter 4 also describes routing
with guaranteed delivery for unit disk graphs and ideal MAC layer based on the application of the Gabriel graph, a localized planar and connected structure. Solutions are expanded toward beaconless behavior, where nodes are not aware of their neighborhood. Georouting with virtual coordinates is based on hop distances to some landmarks. This chapter also discusses physical layer aspects of georouting, routing in sensor–actuator networks, and load balancing issues in routing.

Chapter 5 reviews the scenarios where a given message is sent from a single source (sensor) to possibly several destinations (actuators). These scenarios can be subdivided into multicasting, geocasting, multiratecasting, and anycasting. In multicasting, a given message must be routed from one node to a number of destinations whose locations may be arbitrary and spread over the network. Geocasting destinations are all nodes located in a given geographical area. Multiratecasting is a generalization of multicasting, where regular messages are sent from a source to several destinations, possibly at a different rate for each destination. Finally, in an anycasting scenario, a source must send a message to any node, preferably only one, among a given set of destinations. Each of these scenarios corresponds to a typical use case in sensor and actuator networks.

Data gathering aims to collect sensor readings from sensory fields at predefined sinks or actuators (without aggregating at intermediate nodes) for analysis and processing. Research has shown that sensors near a data sink deplete their battery power faster than those far apart, due to their heavy overhead of relaying messages. Nonuniform energy consumption causes degraded network performance and shortens network lifetime. Recently, sink mobility has been exploited to reduce and balance energy expenditure among sensors. The effectiveness has been demonstrated both by theoretical analysis and by experimental study. In Chapter 6, we investigate the theoretical aspects of the uneven energy depletion phenomenon around a sink/actuator, and address the problem of energy-efficient data gathering by mobile sinks/actuators. We present taxonomy and a comprehensive survey of the state of the art on the topic.

The efficiency of many sensor network algorithms depends on characteristics of the underlying connectivity, such as the length and density of links. The number and nature of links that are to be used among all potentially available links can be controlled. Topology control can be achieved by modifying the transmission radii, selecting a given subset of the links, or moving some nodes (if such functionality is available). Chapter 7 reviews some of these problems and related solutions, applicable to the context of sensor and actuator networks. Spanning structures and minimum weight connectivity are applied for power-efficient and delay-bounded data aggregation. Detection of critical nodes and links aims to provide fault tolerance to the applications. Some recent and prospective works considering biconnectivity of mobile sensors/actuators and related deployment of sensors, augmentation, area and point coverage are discussed.

In the location service problem, mobile actuators send location update messages, while stationary sensors send search messages to learn the latest position of actuators. The task is to minimize combined update and search message cost,
while maximizing the success rate of finding a target actuator and subsequently routing to it. In the literature, many location service algorithms have been proposed for mobile ad hoc networks, and they can be directly applied to sensor and mobile actuator networks. Chapter 8 reviews research efforts on this topic.

Chapter 9 surveys the existing representative work in both, sensor–actuator and actuator–actuator coordination. Sensor–actuator coordination deals with establishing data paths between sensors and actuators, and can be used for sensor deployment. Actuator–actuator coordination includes robot coordination for sensor placement, dynamic task allocation, selecting the best robot to respond to reported events, robot dispersion, boundary coverage, and fault-tolerant response. In coordinated actuator movement problems, actuators are moved to desired locations to save energy in long-term communication tasks where the traffic is sufficiently regular and large in volume to warrant nodes expending energy for moving. Chapter 9 also reviews recent developments in coordination among flying robots.

Coverage is the functional basis of any sensor network. The impact on coverage from stochastic node dropping and inevitable node failure, coupled with controlled node mobility, gives rise to the problem of movement-assisted sensor placement in wireless sensor and actuator networks (WSAN). One or more actuators may carry sensors, and drop them at a proper position, while moving around, in the region of interest (ROI) to construct desired coverage. Mobile sensors may change their original placement so as to improve existing coverage. Emerging coverage holes are to be covered by idle sensors. Actuators may place spare sensors according to certain energy optimality criteria. If sensors are mobile, they can relocate themselves to fill holes. Chapter 10 comprehensively reviews existing solutions to the sensor placement problem in WSAN.

**ACKNOWLEDGMENTS**

The editors are grateful to all the authors for their valuable contribution to this book. The editors are thankful to Michael Christian from Wiley for his timely and professional cooperation, and for his decisive support of this project.

We hope that the readers will find this book informative and worth reading. Comments received by readers will be greatly appreciated.

Amiya Nayak and Ivan Stoimenović
SITE, University of Ottawa,
Ottawa, Ontario, Canada
Contributors

Arnaud Casteigts  School of Information Technology and Engineering, University of Ottawa, 800 King Edward, Ottawa, Ontario K1N 6N5, Canada
Xu Li  School of Information Technology and Engineering, University of Ottawa, 800 King Edward, Ottawa, Ontario K1N 6N5, Canada
Hai Liu  Hong Kong Baptist University, Hong Kong, P.R. China
Veljko Malbasa  Faculty of Technical Sciences, University of Novi Sad, Serbia
Ivan Mezei  Faculty of Technical Sciences, University of Novi Sad, Serbia
Amiya Nayak  School of Information Technology and Engineering, University of Ottawa, 800 King Edward, Ottawa, Ontario K1N 6N5, Canada
David Simplot-Ryl  CNRS/INRIA/University of Lille 1, France
Ivan Stojmenovic  School of Information Technology and Engineering, University of Ottawa, 800 King Edward, Ottawa, Ontario K1N 6N5, Canada
Chapter 1

Applications, Models, Problems, and Solution Strategies

Hai Liu\(^1\), Amiya Nayak\(^2\), and Ivan Stojmenovic\(^2\)

\(^1\)Hong Kong Baptist University, Hong Kong, P.R. China
\(^2\)School of Information Technology and Engineering, University of Ottawa, Ottawa, Ontario, Canada K1N 6N5

Abstract

This introductory chapter describes various applications, scenarios, and models of wireless sensor and actuator networks. Problems at the physical, medium access, network, and transport layers as well as various tools needed to enable their functioning are identified. Various assumptions and metrics used in simulations and protocol descriptions are discussed. The chapter then describes ways of generating sensor and actuator networks based on widely accepted unit disk graph models. Finally, this chapter discusses solution approaches arising in sensor networks and advocates the use of localized protocols, where individual sensors and actuators make their decisions based on local knowledge.

1.1 WIRELESS SENSORS

We will elaborate first on wireless sensors; then on wireless sensor networks (WSNs) (with a single sink), their properties, models, and application types. Afterwards, we will add actuators to the model and discuss various combinations of sensors and actuators that can form a heterogeneous network with different levels of complexity.
Recent technological advances have enabled the development of small-sized (a few cubic centimeters), low-cost, low-power, and multifunctional sensor devices. There are different types of sensors. Sensors are normally specialized, but sometimes a few capabilities may be available in a single sensor. They may measure distance, direction, speed, humidity, wind speed, soil makeup, temperature, chemical composition, light, vibration, motion, seismic activity, acoustic properties, strain, torque, load, pressure, and so on.

Traditionally, sensors are attached to the environment and their measurements are sent to a base station (BS) with wired communication. There exists a large body of knowledge on such models and applications of sensors which have been studied by a huge community of researchers. During the last decade, a new vision of sensor nodes as autonomous devices with integrated sensing, processing, and communication capabilities has emerged. Attaching antenna for receiving signals and a transmitter enables wireless communication of sensors. Sensors also have a small processor and a small memory for coding and decoding signals, as well as for running simple communication protocols. They differ in their battery capacity; for example, some of them run on small batteries and last a day, whereas others have larger batteries attached that let them last up to a month with continuous operation. In some applications, a renewable power supply such as a solar panel is used. Further, some sensors are embedded into other devices and draw their required energy from them. Such sensors do not have energy limitations in their functioning. In some scenarios, sensors could be provided with a wireless single-hop access to infrastructure networks such as the Internet.

For some applications, sensors may be of a large size, especially if they are protected by boxes or lifted to a height that improves their communication and protection level. When collected data is not time critical, sensors may function in isolation. For example, seismological data or bird presence detected acoustically can simply be collected in the local sensor memory and downloaded when visited by humans. This book concentrates, however, on scenarios involving networks of wireless sensors.

### 1.2 SINGLE-HOP WIRELESS SENSOR NETWORKS

The majority of the existing applications for “wireless” sensors rely on a single-hop wireless network to reach a BS for further processing of the measured phenomena. That is, sensor measurements are sent directly, using a wireless medium, from sensor to BS. Most of these applications rely on sensors that are embedded into a different device. Also, the majority of applications for embedded sensors rely on single-hop wireless communication. For example, small sensors can be embedded into a traffic surveillance system to monitor traffic on congested roads or be used to monitor hot spots in a region or building.

Health care is one of the primary applications for wireless networks composed of embedded sensors. Sensors can be embedded into watches which, when attached to patients, monitor and analyze data such as pulse and blood pressure. In case of potential health risks, individual sensors send alarm messages to
1.3 Multihop Wireless Sensor Networks

Nodes in the WSNs are generally randomly and densely deployed. For example, thousands of sensor nodes may be dropped from airplanes to monitor an interested area. Once deployed, these sensors are expected to self-configure into a wireless network. Since the energy budget of an individual sensor is very limited, the transmission range of sensors is also restricted. Thus, WSNs usually operate in a

a nearby control center via one-hop wireless communication. As these sensors are battery powered, they can benefit from intelligent sensor management that provides energy efficiency as well as quality of service (QoS) control.

For example, a wireless motion analysis sensor for stroke patient rehabilitation was studied in John et al. (2005). Wearable sensor motes with armbands were attached to stroke patients to monitor their limb movements and muscle activity during rehabilitation exercises. The sensor board consisted of a three-axis accelerometer, a gyroscope, and various electromyogram (EMG) sensors. It was able to capture a rich set of motion data used for studying the effects of various rehabilitation exercises on the patient population. The collected data was transmitted to a data acquisition or control center, such as a laptop or PC, via one-hop wireless communication. The system architecture is shown in Figure 1.1 below.

Some large-scale sensor networks may also be single-hop in terms of wireless communication needed for reporting. For example, the sink, or several sinks, could be mobile and move around the network. This would allow them to get close enough to the sensors so that report collecting could be done in a single hop. In other examples, embedded sensors could move toward a fixed sink. For example, sensors can be embedded into sea mammals to trace their locations over time. When a sea mammal approaches a fixed BS, reports can be downloaded.

1.3 MULTIHOP WIRELESS SENSOR NETWORKS

Figure 1.1 Monitoring limb movement in stroke patient rehabilitation.
multihop fashion. A large number of sensor devices can be organized in a multi-hop fashion to provide unlimited potential to “sense” the physical world. Reports from individual sensors are sent to other sensors, where they can be combined with other sensor readings or simply retransmitted to other sensors until a sink node that is capable of communicating with a user is reached. Therefore, individual sensor readings may need several wireless hops to reach a BS. Such WSNs have received significant attention in recent years. A WSN usually consists of a large number of low-cost and low-energy sensor nodes, which can be deployed on the ground, in the air, in vehicles, or inside buildings. Nodes in WSNs sense data, find routes, and forward sensing data to a sink or BS that is usually far away from the data source. Since sensors usually have a small size, low-battery capacity, nonrenewable power supply, limited processing ability, small buffer capacity, and a low-powered radio, WSNs pose new challenges to both industrial and academic communities.

Applications for WSNs have been envisioned for a wide range of areas. These include, but are not limited to, the following: environment monitoring (e.g., traffic, habitat, security, etc.), infrastructure protection (e.g., power grids and water distribution), fire prevention, agriculture, health care, chemical plumes tracking, building monitoring or control, warehouse management, smart transportation, and context-aware computing (e.g., smart homes and responsive environments) as well as industrial sensing, diagnostics and process control, biomedical sensor engineering, water and waste management, military applications, and so on.

Most of the scenarios considered contain a single sink (also called base station), which is normally static. The sink in a WSN collects information from sensors and then analyzes and processes the information for specific applications. The sink could be connected to the Internet via wireless or wired communications such that a remote user is able to inquire about data via the Internet (at any time or from anywhere). Single sink scenarios, or scenarios with multiple fixed or mobile sinks, have also been explored in literature. Sensors in WSNs are usually static. However, they can be mobile when attached to robots, soldiers, or vehicles.

1.4 Event-Driven, Periodic, and On-Demand Reporting

There are three types of applications for WSNs and each has its corresponding data communication modes: event-driven, periodic, and on-demand reporting. In the event-driven mode, sensors report the sensing data to the sink once a specified event (e.g., fire) has been detected. In the periodic reporting (or time-driven) mode, sensor nodes gather information from the environment at predetermined times and periodically send the data to the sink. In the on-demand (or query-driven) mode, users decide when to gather data. They send instructions to the WSN indicating that they wish to receive data and then wait for the required type of data to be sent in the requested format. Users may even specify the future reporting periods; subsequent reports would then be sent in periodic reporting mode.
Target or event detection and tracking is a typical example of applications in event-driven reporting. Its purpose is to detect, classify, and locate specific targets or events, as well as track the targets or events over a specified region. Once there is an event or a target emerging in the area, the sensor nodes around the target or event gather the required information and report back to the sink. One characteristic of event-driven reporting is its real-time requirement. This means that data transmission latency is one of the key problems in these applications.

Targets can be divided into two categories: targets in the first category are individual objects that usually have a small size when compared to the sensing area of the network. These targets emit noise, light, and seismic waves, such that nearby sensor nodes are able to detect and track them. A typical example is to deploy a sensor network to detect troops, such as tanks and soldiers, in a battlefield. Once a tank moves into a specific area, information on the tank such as its location and speed, will be gathered by the sensor nodes and reported to the BS via multihop communication. The targets in the second category are continuous objects, which spread in the sensing area of the network. An example is the use of WSNs to detect and track diffused poison gas or chemical/biochemical liquids.

Figure 1.2 shows a typical scenario of event-driven reporting in WSNs. Sensor nodes are deployed in the sensor field to form a wireless network. Once there is an event in the monitoring field, such as a fire, a nearby sensor node, say A, will detect the fire if the sensed temperature exceeds a predefined threshold. Then A either starts the routing process (reactive) or uses the route in its routing table (proactive, e.g., A-B-C-D-E), to report information of the event to the sink. The sink may then take appropriate actions immediately or store the data in the database for future statistical use.

Periodic reporting is different from event-driven reporting. Data gathered in periodic reporting does not require urgent delivery to the sink. Further, the data in the event-driven reporting usually comes from sensors in the vicinity of a target or event, whereas the data in periodical reporting is normally gathered from sensor nodes throughout the sensor field.

Sensors report to the sink by applying data gathering and data aggregation operations. Data gathering refers to forwarding the measured data to the sink.
without further changes on the way toward sink. This is normally achieved via a routing task, that is, sending a message from a sender node (sensor) to a destination node (sink), using other sensors to forward the report. However, data collected by sensor nodes might be redundant, correlated, and/or inconsistent with data from other sensors. Data aggregation is used to combine data coming from different sensor nodes. This eliminates redundancy and minimizes the number of transmissions.

A general approach employed in data gathering and data aggregation is to construct a spanning tree which is rooted at the sink and connects all sensor nodes in the network. If one node fails, the topology will be reorganized into a new topology. Tree maintenance is usually an energy-demanding operation.

In data gathering or aggregation, data from each sensor is forwarded to the sink along the spanning tree. This is illustrated in Figure 1.3 where a WSN is deployed for agricultural applications. A large number of sensor nodes are scattered throughout a field to monitor the temperature, light levels, and soil moisture. The sink, located in the house, queries the sensors, which configure themselves. The reporting tree is constructed in the process and rooted at the sink. Data is periodically collected from all sensors in the field and sent to the sink. In data gathering operations, individual data from each sensor is forwarded along the tree without being combined with measurements from other sensors. Data aggregation may be applied too, for example, combining the readings from all of the sensors inside a zone and submitting a combined report via data gathering.

Figure 1.3  Data gathering or aggregation in agricultural applications.
operations along the tree hop-by-hop toward the root. For instance, information on soil conditions in different zones of the same field might be needed to apply uneven amounts of fertilizer. Sometimes a single report from the whole field would suffice such as information on the current temperature. In this case, upon receiving the data from each child node in the tree, a sensor node aggregates the data with its own before delivery to its parent node in the constructed tree.

The traditional view of large-sized static sensor networks with one fixed sink has been challenged by their theoretical and simulation analysis discovering some bottlenecks in their performance. For example, it was reported that while using the same transmission range for sensors optimizes energy per report (without data aggregation), it also creates energy holes around the sink while the periphery is left with almost full energy (Olariu et al., 2006). Moreover, data aggregation is often impossible. For example, sensors monitoring movements do not generate the same reports and sink instructions are also not aggregated. Therefore, the problems do not seem to have a resolution unless the model itself is changed: It should be either small scale (e.g., up to hundred nodes) or involve multiple sinks, mobile sinks, mobile sensors, and so on. However, this in turn complicates network layer protocols.

### 1.5 UNIT DISK GRAPH MODELING, HOP COUNT METRIC, AND PROBABILISTIC RECEPTION

Multihop wireless communication in networks of equal devices applying same and fixed transmission radii (i.e., a homogeneous network), has a simple modeling that is an excellent and extremely useful simplification of the complex physical layer. In the unit disk graph (UDG), two nodes communicate if and only if the distance between them is at most $R$, where $R$ is the transmission radius which is equal for all nodes. A UDG is therefore determined by the positions of nodes and a fixed common transmission range $R$. To illustrate this, if we use $R/2$ as the radius of the disk of each node, two nodes are connected if and only if their corresponding disks intersect. An example of a UDG is shown in Figure 1.4 below. Unit disk graphs successfully model WSNs, wireless ad hoc networks (used in rescue, conference, and battlefield scenarios), vehicular network communications, and wireless networks of actuators (to be defined shortly). In combined networks, such as sensor and actuator networks, they can model communication of each component network separately by using different transmission radii for them.

![Figure 1.4](image-url) An example of a unit disk graph (with radius $R/2$).
Hop count can be used as a metric for routing in UDG if each node applies the same and fixed transmission power. It is defined as the number of hops from one node to another. Hop count between two adjacent nodes is 1. In Figure 1.4, the hop count between node A and node B is 4. In homogeneous networks, where nodes do not adjust transmission radius, the route with the smallest hop count from a source to a destination guarantees the minimal energy cost and the lowest transmission latency (assuming that the delay at each node is the same).

Although the protocols at the network layer are mostly designed with ideal UDG assumptions, experiments are normally carried out on simulators that implement more realistic physical and medium access control (MAC) layers. The UDG model is not realistic since variations in the received signal strengths are not considered. In fact, it has been pointed out that the impact of signal strength fluctuations is sometimes more significant than the impact of node mobility (Stojmenovic et al., 2005a). Therefore, nondeterministic radio fluctuations cannot be ignored when designing robust protocols for sensor and ad hoc networks. In addition to distance, the received signal strength also depends on other factors such as environment and transmission medium.

Existing physical layer models, such as the combined Friis and two-ray ground model (Nadeem and Agrawala, 2004) and the lognormal shadowing model (Stojmenovic et al., 2005b), require nodes to estimate the probability of receiving a bit or a packet based on either signal strength, distance between nodes, or merely by deriving statistics from a number of bits or packets recently sent between two nodes. The realistic physical model normally uses a function to represent the packet reception probability. For instance, the packet reception probability $p(x)$ in the shadowing model (Stojmenovic et al., 2005b) depends on the length of the packet, and the distance $x$ between two nodes. Suppose $R$ is the distance so that the packet reception probability is $p(R) = 0.5$, the function $p(x)$ may have approximately the following values: $p(0) = 1$, $p(0.1R) \approx 1$, $p(0.5R) \approx 0.9$, $p(R) = 0.5$, $p(1.5R) \approx 0.25$, and $p(2R) = 0$. The values give a sufficient intuition on how to design physical layer–aware routing protocols. If a fixed signal-to-noise ratio (SNR) is assumed then the function $p(x)$ looks like the graph in Figure 1.5 (Kuruvila et al., 2005). In this example, the probability for successful transmission at distance $d = 30$ is more than 0.95. If $d = 41$, the probability for successful transmission is around 0.5. This means that approximately half of the transmissions are successful. If $d = 50$, the probability for successful transmission decreases to around 0.05. Two nodes can still communicate as long as they make a sufficient number of attempts.

At the physical layer, the hop count metric may not properly reflect the real cost involved in a route. For example, suppose there are many long edges in the shortest path, in terms of hop count. Many retransmissions may be required between adjacent nodes on these long edges due to low probability of packet reception. Thus, the expected hop count (EHC) should be used instead. Expected hop count is defined as the expected number of messages between the sender and the receiver, including retransmission, acknowledgments and so on. Extended hop
1.6. Adjustable Transmission Range and Power Metric

Sensor nodes have the capacity to adjust their transmission ranges without incurring any significant cost for the adjustment. A common transmission radius is normally preferred because medium access protocols currently considered to be de facto standards, such as Zigbee, require it for proper functioning. However,
finding a minimal common transmission radius is a nontrivial problem, especially for its maintenance. A possible compromise is that each sensor is made aware of its neighbors by using “hello” messages. But when the neighbor for forwarding is decided, the transmission power could be adjusted to reflect the distance. Here, we still assume the UDG modeling with adjusted transmission radius. In reality, there is an impact of the realistic physical layer at critical transmission distances, which will be discussed later.

A simple power consumption model is introduced in Rodoplu and Meng (1999). The total power needed to transmit and receive a message between two nodes at distance $d$ is proportional to $d^\alpha + c$, where $\alpha$ is the signal strength attenuation factor which is normally between 2 and 5 depending on the transmission medium and the environment, and $c$ is a contact that accounts for signal processing at the transmitter and receiver, as well as the minimal power to receive a signal properly. This model has been restated in Heinzelman et al. (2000). The energy consumption per bit is calculated as follows: 

$$\text{power} = E_{\text{trans}} + \beta d^\alpha + E_{\text{rec}},$$

where $E_{\text{trans}}$ and $E_{\text{rec}}$ are distance-independent terms which represent the overhead of the transmitter electronics and receiver electronics, respectively. For simplicity, $E_{\text{trans}}$ and $E_{\text{rec}}$ are often assumed to be the same (Chen et al., 2004), and $c$ from Rodoplu and Meng (1999) model is proportional to $E_{\text{trans}} + E_{\text{rec}}$. $\beta d^\alpha$ is the distance-dependent term that represents the power consumption required to transmit one bit from a sender to a receiver over distance $d$. If a message contains $k$ bits, the power consumption is then normally multiplied by $k$.

When the transmission range of the nodes is adjustable, power metric is used to measure the optimality of a routing algorithm in different ways. The simplest way is to measure power consumption at each hop and look for a route that minimizes the total power consumption (the sum of powers consumed at each hop). However, some nodes may be centrally positioned and used on many paths. Their energies can be depleted while the energies of some peripheral nodes may remain close to their maximum. The lifetime of a network may be measured in several ways, including the moment the first node spends all its energy or network partitioning (the moment a particular sensor is not able to deliver its report to the sink because of energy holes in the network coming from sensors left with no energy). Thus, the minimum energy metric routing may not maximize the network’s lifetime since some sensors may suffer early failure. An alternative is to maximize the lifetime of the network.

### 1.7 COST METRICS

A variety of metrics and their combinations can be used to design and evaluate communication protocols for WSNs. We have discussed so far hop count and power consumption metrics. A convenient metric that can be used to avoid nodes with low remaining energy on a routing path is called reluctance (Stojmenovic and Lin, 2001b). The remaining energy $g$ at a sensor node can be normalized in the interval $(0,1)$. The resistance $f$ is then $f = 1/g$, meaning that the reluctance becomes huge when a sensor is close to depletion.
Some applications of WSNs, where real-time or multimedia data are involved in communications, require a guarantee on QoS metrics such as delay, throughput, and bandwidth. For example, sensor networks for fire detection require short latency to transmit emergency data to the sink. QoS routing is usually performed through resource reservation in individual nodes along the route.

In the sequel, the term cost will often be used to denote one of the mentioned metrics or a newly designed metric which is often a combination of several existing metrics. One example of a combined metric is power * reluctance, which can be used in designing routing paths to balance between finding routes with a low total sum of power metrics on route and also avoiding nodes with low remaining energy. The cost metric is therefore often used to find a trade-off among these parameters.

As another example, a conditional max–min battery capacity routing is studied in Toh (2001). If there is at least one route such that the residual energy of each node is greater than a specified threshold, the minimum energy metric is chosen. Otherwise, the route that maximizes the minimum residual energy is selected. In this algorithm, there exists a hidden cost for finding routes needed to elect the best one. The sender node needs global network knowledge to gather it, which requires communication overhead not accounted for in the selected metric of route efficiency. This point will be further discussed in the Section 1.15.

1.8 SLEEP AND ACTIVE STATE MODELING

Energy consumption is one of the key problems in WSNs. Several energy consumption models have been studied in the literature. The following discussion and the graph (Fig. 1.6) are based on the study by Barrenetxea et al. (2008). The graph shows energy consumption of a TinyNode sensor mote in different states. The experiment shows that the calculation of the sensor node’s receiving costs depends on the assumption of the node’s status. If it assumes that the radios of the nodes are always on, the energy consumption of the receiving costs is negligible since the cost for receiving packets has been included in the cost for keeping the radios on. More precisely, the energy consumption is equal to 2 mA when the radio is off but is equal to 16 mA when the radio is on for reception. This means that it takes about eight times more energy for listening compared to sleeping state. The total energy consumption for receiving depends on how long the radios need to be on to receive an incoming packet. Using the example in Figure 1.6 below, we suppose that transmitting a packet at 15 dB consuming 60 mA takes 5 ms. Receiving the packet takes at least 5 ms. However, it is not possible for the node to turn on exactly at the time the packet is sent. That is, to receive the packet, the node should turn on its radio for more than 5 ms (according to the used protocol). In Figure 1.6, the energy consumption of the radio is 15 mA. Therefore, if the total time the radio is on is more than 20 ms, the energy consumption of receiving a packet is more than the energy consumption of transmitting a packet.
Figure 1.6 (a) CPU off; (b) CPU on; (c) radio on; (d) sending a packet at 0dB; (e) sending a packet at 5dB; (f) sending a packet at 10dB; (g) sending a packet at 15dB.

1.9 ARCHITECTURES FOR WIRELESS SENSOR AND ACTUATOR NETWORKS

Although WSNs have been employed in many applications, such as environment monitoring and health care, there are an increasing number of applications that require the use of actuators along with sensors. This occurs when the network system needs to interact with the physical system or the environment via actuators (also called actors). From the engineering aspect, an actuator is a transducer that accepts a signal and converts it to a physical action. Actuators transform an input signal into an action upon the environment. Typical examples of actuators are robots, electrical motors, and humans. Traditional sensor and actuator networks use wired communications among themselves; these networks have been well studied. The advent of small, intelligent, low-energy, and low-cost wireless sensing and actuation devices has the potential to significantly expand existing applications of wired sensor actuator networks. Wireless sensor actuator networks (WSANs) are emerging as the next generation of WSNs. The major difference between WSANs and WSNs is that WSANs are capable of changing the environment and physical world while WSNs cannot. Wireless sensor actuator networks are envisioned for applications that include disaster relief operations, intelligent buildings, home automation, smart spaces, pervasive computing systems, cyber-physical systems and nuclear, biological, and chemical attack detection (Xia et al., 2007).

A WSAN usually consists of a group of sensor nodes that are used to gather information from the environment, and actuator nodes that are used to change the behavior of the environment. There are wireless links between the sensor