Computer Networks 52 (2008) 2292–2330

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Wireless sensor network survey

Jennifer Yick, Biswanath Mukherjee, Dipak Ghosal *

Department of Computer Science, University of California, Davis, CA 95616, United States

ARTICLE INFO

Article history:
Received 20 March 2007
Received in revised form 3 April 2008
Accepted 7 April 2008
Available online 14 April 2008

Responsible Editor: E. Ekici

Keywords:
Wireless sensor network
Protocols
Sensor network services
Sensor network deployment
Survey

ABSTRACT

A wireless sensor network (WSN) has important applications such as remote environmental monitoring and target tracking. This has been enabled by the availability, particularly in recent years, of sensors that are smaller, cheaper, and intelligent. These sensors are equipped with wireless interfaces with which they can communicate with one another to form a network. The design of a WSN depends significantly on the application, and it must consider factors such as the environment, the application's design objectives, cost, hardware, and system constraints. The goal of our survey is to present a comprehensive review of the recent literature since the publication of [I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, IEEE Communications Magazine, 2002]. Following a top-down approach, we give an overview of several new applications and then review the literature on various aspects of WSNs. We classify the problems into three different categories: (1) internal platform and underlying operating system, (2) communication protocol stack, and (3) network services, provisioning, and deployment. We review the major development in these three categories and outline new challenges.

1. Introduction

Wireless sensor networks (WSNs) have gained worldwide attention in recent years, particularly with the proliferation in Micro-Electro-Mechanical Systems (MEMS) technology which has facilitated the development of smart sensors. These sensors are small, with limited processing and computing resources, and they are inexpensive compared to traditional sensors. These sensor nodes can sense, measure, and gather information from the environment and, based on some local decision process, they can transmit the sensed data to the user.

Smart sensor nodes are low power devices equipped with one or more sensors, a processor, memory, a power supply, a radio, and an actuator.[1] A variety of mechanical, thermal, biological, chemical, optical, and magnetic sensors may be attached to the sensor node to measure properties of the environment. Since the sensor nodes have limited memory and are typically deployed in difficult-to-access locations, a radio is implemented for wireless communication to transfer the data to a base station (e.g., a laptop, a personal handheld device, or an access point to a fixed infrastructure). Battery is the main power source in a sensor node. Secondary power supply that harvests power from the environment such as solar panels may be added to the node depending on the appropriateness of the environment where the sensor will be deployed. Depending on the application and the type of sensors used, actuators may be incorporated in the sensors.

A WSN typically has little or no infrastructure. It consists of a number of sensor nodes (few tens to thousands) working together to monitor a region to obtain data about the environment. There are two types of WSNs: structured and unstructured. An unstructured WSN is one that contains a dense collection of sensor nodes. Sensor nodes may be deployed in an ad hoc manner[2] into the field. Once

* Corresponding author. Tel.: +1 530 754 9251; fax: +1 530 752 4767.
E-mail addresses: yick@cs.ucdavis.edu (J. Yick), mukherje@cs.ucdavis.edu (B. Mukherjee), ghosal@cs.ucdavis.edu (D. Ghosal).

[1] An actuator is an electro-mechanical device that can be used to control different components in a system. In a sensor node, actuators can actuate different sensing devices, adjust sensor parameters, move the sensor, or monitor power in the sensor node.

[2] In ad hoc deployment, sensor nodes may be randomly placed into the field.

1389-1286/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2008.04.002

deployed, the network is left unattended to perform monitoring and reporting functions. In an unstructured WSN, network maintenance such as managing connectivity and detecting failures is difficult since there are so many nodes. In a structured WSN, all or some of the sensor nodes are deployed in a pre-planned manner.[3] The advantage of a structured network is that fewer nodes can be deployed with lower network maintenance and management cost. Fewer nodes can be deployed now since nodes are placed at specific locations to provide coverage while ad hoc deployment can have uncovered regions.

WSNs have great potential for many applications in scenarios such as military target tracking and surveillance [2,3], natural disaster relief [4], biomedical health monitoring [5,6], and hazardous environment exploration and seismic sensing [7]. In military target tracking and surveillance, a WSN can assist in intrusion detection and identification. Specific examples include spatially-correlated and coordinated troop and tank movements. With natural disasters, sensor nodes can sense and detect the environment to forecast disasters before they occur. In biomedical applications, surgical implants of sensors can help monitor a patient's health. For seismic sensing, ad hoc deployment of sensors along the volcanic area can detect the development of earthquakes and eruptions.

Unlike traditional networks, a WSN has its own design and resource constraints. Resource constraints include a limited amount of energy, short communication range, low bandwidth, and limited processing and storage in each node. Design constraints are application dependent and are based on the monitored environment. The environment plays a key role in determining the size of the network, the deployment scheme, and the network topology. The size of the network varies with the monitored environment. For indoor environments, fewer nodes are required to form a network in a limited space whereas outdoor environments may require more nodes to cover a larger area. An ad hoc deployment is preferred over pre-planned deployment when the environment is inaccessible by humans or when the network is composed of hundreds to thousands of nodes. Obstructions in the environment can also limit communication between nodes, which in turn affects the network connectivity (or topology).

Research in WSNs aims to meet the above constraints by introducing new design concepts, creating or improving existing protocols, building new applications, and developing new algorithms. In this study, we present a top-down approach to survey different protocols and algorithms proposed in recent years. Our work differs from other surveys as follows:

- While our survey is similar to [1], our focus has been to survey the more recent literature.
- We address the issues in a WSN both at the individual sensor node level as well as a group level.
- We survey the current provisioning, management and control issues in WSNs. These include issues such as localization, coverage, synchronization, network security, and data aggregation and compression.
- We compare and contrast the various types of wireless sensor networks.
- Finally, we provide a summary of the current sensor technologies.

The remainder of this paper is organized as follows: Section 2 gives an overview of the key issues in a WSN. Section 3 compares the different types of sensor networks. Section 4 discusses several applications of WSNs. Section 5 presents issues in operating system support, supporting standards, storage, and physical testbed. Section 6 summarizes the control and management issues. Section 7 classifies and compares the proposed physical layer, data-link layer, network layer, and transport layer protocols. Section 8 concludes this paper. Appendix A compares the existing types of WSNs. Appendix B summarizes the sensor technologies. Appendix C compares sensor applications with the protocol stack.

2. Overview of key issues

Current state-of-the-art sensor technology provides a solution to design and develop many types of wireless sensor applications. A summary of existing sensor technologies is provided in Appendix A. Available sensors in the market include generic (multi-purpose) nodes and gateway (bridge) nodes. A generic (multi-purpose) sensor node's task is to take measurements from the monitored environment. It may be equipped with a variety of devices which can measure various physical attributes such as light, temperature, humidity, barometric pressure, velocity, acceleration, acoustics, magnetic field, etc. Gateway (bridge) nodes gather data from generic sensors and relay them to the base station. Gateway nodes have higher processing capability, battery power, and transmission (radio) range. A combination of generic and gateway nodes is typically deployed to form a WSN.

To enable wireless sensor applications using sensor technologies, the range of tasks can be broadly classified into three groups as shown in Fig. 1. The first group is the system. Each sensor node is an individual system. In order to support different application software on a sensor system, development of new platforms, operating systems, and storage schemes are needed. The second group is communication protocols, which enable communication between the application and sensors. They also enable communication between the sensor nodes. The last group is services which are developed to enhance the application and to improve system performance and network efficiency.

From application requirements and network management perspectives, it is important that sensor nodes are capable of self-organizing themselves. That is, the sensor nodes can organize themselves into a network and subsequently are able to control and manage themselves efficiently. As sensor nodes are limited in power, processing capacity, and storage, new communication protocols and management services are needed to fulfil these requirements.

[3] In pre-planned deployment, sensor nodes are pre-determined to be placed at fixed locations.

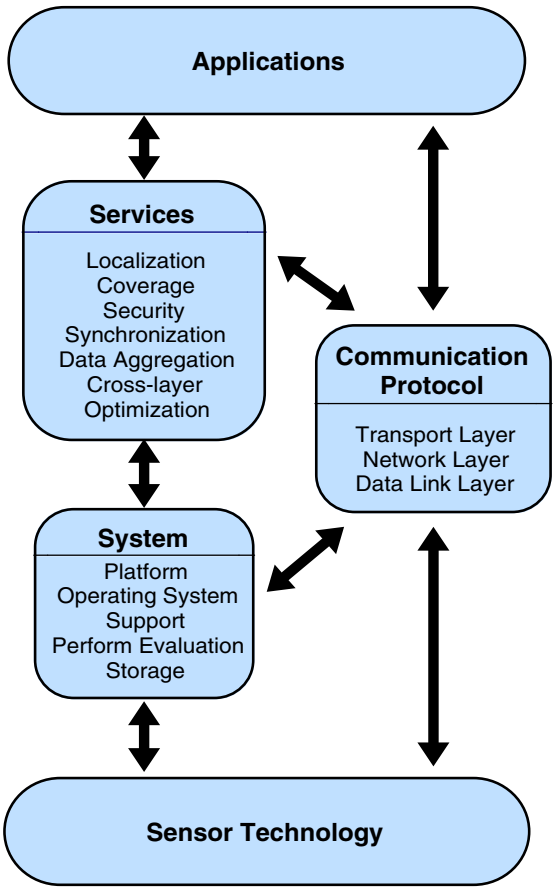

Fig. 1. Broad classification of various issues in a WSN.

The communication protocol consists of five standard protocol layers for packet switching: application layer, transport layer, network layer, data-link layer, and physical layer. In this survey, we study how protocols at different layers address network dynamics and energy efficiency. Functions such as localization, coverage, storage, synchronization, security, and data aggregation and compression are explored as sensor network services.

Implementation of protocols at different layers in the protocol stack can significantly affect energy consumption, end-to-end delay, and system efficiency. It is important to optimize communication and minimize energy usage. Traditional networking protocols do not work well in a WSN since they are not designed to meet these requirements. Hence, new energy-efficient protocols have been proposed for all layers of the protocol stack. These protocols employ cross-layer optimization by supporting interactions across the protocol layers. Specifically, protocol state information at a particular layer is shared across all the layers to meet the specific requirements of the WSN.

As sensor nodes operate on limited battery power, energy usage is a very important concern in a WSN; and there has been significant research focus that revolves around harvesting and minimizing energy. When a sensor node is depleted of energy, it will die and disconnect from the network which can significantly impact the performance of the application. Sensor network lifetime depends on the number of active nodes and connectivity of the network, so energy must be used efficiently in order to maximize the network lifetime.

Energy harvesting involves nodes replenishing its energy from an energy source. Potential energy sources include solar cells [8,9], vibration [10], fuel cells, acoustic noise, and a mobile supplier [11]. In terms of harvesting energy from the environment [12], solar cell is the current mature technique that harvest energy from light. There is also work in using a mobile energy supplier such as a robot to replenish energy. The robots would be responsible in charging themselves with energy and then delivering energy to the nodes.

Energy conservation in a WSN maximizes network lifetime and is addressed through efficient reliable wireless communication, intelligent sensor placement to achieve adequate coverage, security and efficient storage management, and through data aggregation and data compression. The above approaches aim to satisfy both the energy constraint and provide quality of service (QoS)[4] for the application. For reliable communication, services such as congestion control, active buffer monitoring, acknowledgements, and packet-loss recovery are necessary to guarantee reliable packet delivery. Communication strength is dependent on the placement of sensor nodes. Sparse sensor placement may result in long-range transmission and higher energy usage while dense sensor placement may result in short-range transmission and less energy consumption. Coverage is interrelated to sensor placement. The total number of sensors in the network and their placement determine the degree of network coverage. Depending on the application, a higher degree of coverage may be required to increase the accuracy of the sensed data. In this survey, we review new protocols and algorithms developed in these areas.

3. Types of sensor networks

Current WSNs are deployed on land, underground, and underwater. Depending on the environment, a sensor network faces different challenges and constraints. There are five types of WSNs: terrestrial WSN, underground WSN, underwater WSN, multi-media WSN, and mobile WSN (see Appendix B).

Terrestrial WSNs [1] typically consist of hundreds to thousands of inexpensive wireless sensor nodes deployed in a given area, either in an ad hoc or in a pre-planned manner. In ad hoc deployment, sensor nodes can be dropped from a plane and randomly placed into the target area. In pre-planned deployment, there is grid placement, optimal placement [13], 2-d and 3-d placement [14,15] models.

In a terrestrial WSN, reliable communication in a dense environment is very important. Terrestrial sensor nodes must be able to effectively communicate data back to the base station. While battery power is limited and may not

[4] QoS defines parameters such as end-to-end delay which must be guaranteed to an application/user.

be rechargeable, terrestrial sensor nodes however can be equipped with a secondary power source such as solar cells. In any case, it is important for sensor nodes to conserve energy. For a terrestrial WSN, energy can be conserved with multi-hop optimal routing, short transmission range, in-network data aggregation, eliminating data redundancy, minimizing delays, and using low duty-cycle operations.

Underground WSNs [16,17] consist of a number of sensor nodes buried underground or in a cave or mine used to monitor underground conditions. Additional sink nodes are located above ground to relay information from the sensor nodes to the base station. An underground WSN is more expensive than a terrestrial WSN in terms of equipment, deployment, and maintenance. Underground sensor nodes are expensive because appropriate equipment parts must be selected to ensure reliable communication through soil, rocks, water, and other mineral contents. The underground environment makes wireless communication a challenge due to signal losses and high levels of attenuation. Unlike terrestrial WSNs, the deployment of an underground WSN requires careful planning and energy and cost considerations. Energy is an important concern in underground WSNs. Like terrestrial WSN, underground sensor nodes are equipped with a limited battery power and once deployed into the ground, it is difficult to recharge or replace a sensor node's battery. As before, a key objective is to conserve energy in order to increase the lifetime of network which can be achieved by implementing efficient communication protocol.

Underwater WSNs [18,19] consist of a number of sensor nodes and vehicles deployed underwater. As opposite to terrestrial WSNs, underwater sensor nodes are more expensive and fewer sensor nodes are deployed. Autonomous underwater vehicles are used for exploration or gathering data from sensor nodes. Compared to a dense deployment of sensor nodes in a terrestrial WSN, a sparse deployment of sensor nodes is placed underwater. Typical underwater wireless communications are established through transmission of acoustic waves. A challenge in underwater acoustic communication is the limited bandwidth, long propagation delay, and signal fading issue. Another challenge is sensor node failure due to environmental conditions. Underwater sensor nodes must be able to self-configure and adapt to harsh ocean environment. Underwater sensor nodes are equipped with a limited battery which cannot be replaced or recharged. The issue of energy conservation for underwater WSNs involves developing efficient underwater communication and networking techniques.

Multi-media WSNs [20] have been proposed to enable monitoring and tracking of events in the form of multimedia such as video, audio, and imaging. Multi-media WSNs consist of a number of low cost sensor nodes equipped with cameras and microphones. These sensor nodes interconnect with each other over a wireless connection for data retrieval, process, correlation, and compression. Multi-media sensor nodes are deployed in a pre-planned manner into the environment to guarantee coverage. Challenges in multi-media WSN include high bandwidth demand, high energy consumption, quality of service (QoS) provisioning, data processing and compressing techniques, and cross-layer design. Multi-media content such as a video stream requires high bandwidth in order for the content to be delivered. As a result, high data rate leads to high energy consumption. Transmission techniques that support high bandwidth and low energy consumption have to be developed. QoS provisioning is a challenging task in a multi-media WSN due to the variable delay and variable channel capacity. It is important that a certain level of QoS must be achieved for reliable content delivery. In-network processing, filtering, and compression can significantly improve network performance in terms of filtering and extracting redundant information and merging contents. Similarly, cross-layer interaction among the layers can improve the processing and the delivery process.

Mobile WSNs consist of a collection of sensor nodes that can move on their own and interact with the physical environment. Mobile nodes have the ability sense, compute, and communicate like static nodes. A key difference is mobile nodes have the ability to reposition and organize itself in the network. A mobile WSN can start off with some initial deployment and nodes can then spread out to gather information. Information gathered by a mobile node can be communicated to another mobile node when they are within range of each other. Another key difference is data distribution. In a static WSN, data can be distributed using fixed routing or flooding while dynamic routing is used in a mobile WSN. Challenges in mobile WSN include deployment, localization, self-organization, navigation and control, coverage, energy, maintenance, and data process.

Mobile WSN applications include but are not limited to environment monitoring, target tracking, search and rescue, and real-time monitoring of hazardous material. For environmental monitoring in disaster areas, manual deployment might not be possible. With mobile sensor nodes, they can move to areas of events after deployment to provide the required coverage. In military surveillance and tracking, mobile sensor nodes can collaborate and make decisions based on the target. Mobile sensor nodes can achieve a higher degree of coverage and connectivity compared to static sensor nodes. In the presence of obstacles in the field, mobile sensor nodes can plan ahead and move appropriately to obstructed regions to increase target exposure.

4. Applications

WSN applications can be classified into two categories: monitoring and tracking (see Fig. 2). Monitoring applications include indoor/outdoor environmental monitoring, health and wellness monitoring, power monitoring, inventory location monitoring, factory and process automation, and seismic and structural monitoring. Tracking applications include tracking objects, animals, humans, and vehicles. While there are many different applications, below we describe a few example applications that have been deployed and tested in the real environment.

PinPtr [2] is an experimental counter-sniper system developed to detect and locate shooters. The system uti-

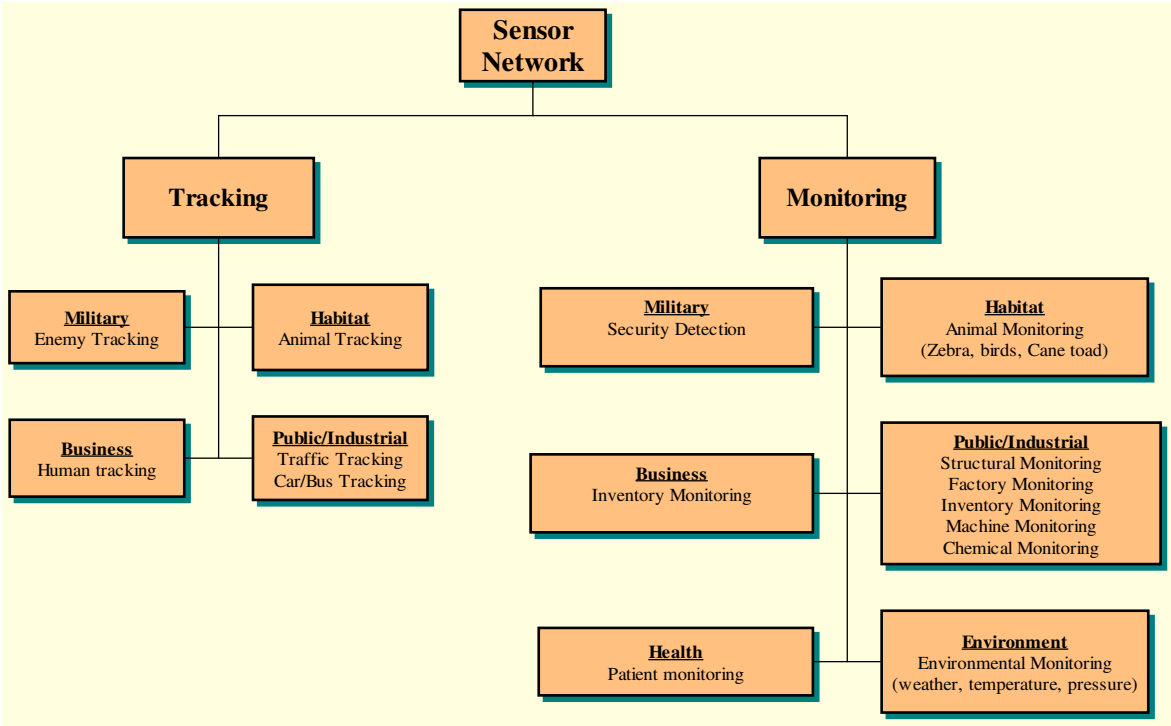

Fig. 2. Overview of sensor applications.

lizes a dense deployment of sensors to detect and measure the time of arrival of muzzle blasts and shock waves from a shot. Sensors route their measurements to a base station (e.g., a laptop or PDA) to compute the shooter's location.

Sensors in the PinPtr system are second-generation Mica2 motes connected to a multi-purpose acoustic sensor board. Each multi-purpose acoustic sensor board is designed with three acoustic channels and a Xilinx Spartan II FPGA. Mica2 motes run on a TinyOS [21] operating system platform that handles task scheduling, radio communication, time, I/O processing, etc. Middleware services developed on TinyOS that are exploited in this application include time synchronization, message routing with data aggregation, and localization.

Macroscope of redwood [22] is a case study of a WSN that monitors and records the redwood trees in Sonoma, California. Each sensor node measures air temperature, relative humidity, and photo-synthetically-active solar radiation. Sensor nodes are placed at different heights of the tree. Plant biologists track changes of spatial gradients in the microclimate around a redwood tree and validate their biological theories.

Semiconductor plants and oil tanker application reported in [23] focus on preventive equipment maintenance using vibration signatures gathered by sensors to predict equipment failure. Based on application requirements and site survey, the architecture of the network is developed to meet application data needs. Two experiments were carried out: the first was in a semiconductor fabrication plant and the second on an onboard oil tanker in the North Sea. The goal was to reliably validate the requirements for industrial environments and evaluate the effect of the sensor network architecture. The study also analyzed the impact of platform characteristics on the architecture and performance of real deployment.

Underwater monitoring study in [24] developed a platform for underwater sensor networks to be used for long-term monitoring of coral reefs and fisheries. The sensor network consists of static and mobile underwater sensor nodes. The nodes communicate via point-to-point links using high speed optical communications. Nodes broadcast using an acoustic protocol integrated in the TinyOS protocol stack. They have a variety of sensing devices, including temperature and pressure sensing devices and cameras. Mobile nodes can locate and move above the static nodes to collect data and perform network maintenance functions for deployment, re-location, and recovery. The challenges of deploying sensors in an underwater environment were some key lessons from this study.

MAX [25] is a system for human-centric search of the physical world. MAX allows people to search and locate physical objects when they are needed. It provides location information reference to identifiable landmarks rather than precise coordinates. MAX was designed with the objectives of privacy, efficient search of a tagged object, and human-centric operation. MAX uses a hierarchical architecture that requires objects to be tagged, sub-stations as landmarks, and base-station computers to locate the object. Tags on objects can be marked as private or public which is searchable by the public or owner only. MAX is designed for low energy and minimal-delay queries. The implementation of MAX was demonstrated using

Crossbow motes where trials were conducted in a room of physical objects.

Connection-less sensor-based tracking system using witness (*CenWits*) [26] is a search-and-rescue system designed, implemented, and evaluated using Berkeley Mica2 sensor motes. The system uses several small radio frequencies (RF)-based sensors and a small number of storage and processing devices. CenWits is not a continuously-connected network. It is designed for intermittent network connectivity. It is comprised of mobile sensors worn by subjects (people), access points that collect information from these sensors and GPS receivers, and location points to provide location information to the sensors. A subject will use the GPS receivers and location points to determine its current location. The key concept is the use of witnesses to convey a subject's movement and location information to the outside world. The goal of CenWits is to determine an approximate small area where search-and-rescue efforts can be concentrated.

Cyclops [27] is a small camera device that bridges the gap between computationally-constrained sensor nodes and complimentary metal-oxide semiconductor (CMOS) imagers. This work provides sensor technology with CMOS imaging. With CMOS imaging, humans can (1) exploit a different perspective of the physical world which cannot be seen by human vision, and (2) identify their importance. Cyclops attempts to interface between a camera module and a lightweight sensor node. Cyclops contains programmable logic and memory circuits with high speed data transfer. It contains a micro-controller to interface with the outside world. Cyclops is useful in a number of applications that require high speed processing or high resolution images.

WSN in a petroleum facility [28] can reduce cost and improve efficiency. The design of this network is focused on the data rate and latency requirement of the plant. The network consists of four sensor node and an actuator node. The sensor nodes are based on T-mote sky devices [29]. Two AGN1200 pre-802.11N Series MIMO access points [30] are used to create an 802.11b 2.4 GHz wireless local area network. In this multi-hop WSN, the T-mote sky devices send their radio packets to the base station which is forwarded to a crossbow stargate gateway. The crossbow stargate gateway translates the radio packets and sends it along the Ethernet MIMO to a single board TS-3300 computer [31]. The single board TS-3300 computer outputs the sensor data to the distributed control system. The distributed control system can also submit changes to the actuator. In this study, results of network performance, RSSI and LQI measurement and noise were gathered. Results show that the effect of latency and environmental noise can significantly affect the performance of a WSN placed in an industrial environment.

Volcanic monitoring [32] with WSN can help accelerate the deployment, installation, and maintenance process. WSN equipments are smaller, lighter, and consume less power. The challenges of a WSN application for volcanic data collection include reliable event detection, efficient data collection, high data rates, and sparse deployment of nodes. Given these challenges, a network consists of 16 sensor nodes was deployed on Volcàn Reventador in northern Ecuador. Each sensor node is a T-mote sky device [29] equipped with an external omni-directional antenna, a seismometer, a microphone, and a custom hardware interface board. Of the 16 sensor nodes, 14 sensor nodes are equipped with a single axis Geospace Industrial GS-11 Geophone with corner frequency of 4.5 Hz while the other two sensor nodes carried triaxial Geospace Industries GS-1 seismometers with corner frequencies of 1 Hz. The custom hardware interface board was designed with four Texas Instruments AD7710 analog-to-digital converters to integrate with the T-mote sky devices. Each sensor node draws power from a pair of alkaline D cell batteries. Sensor nodes are placed approximately 200–400 m apart from each other. Nodes relay data via multi-hop routing to a gateway node. The gateway node connected to a long-distance FreeWave radio modem transmits the collected data to the base station. During network operation, each sensor node samples two or four channels of seismoacoustic data at 100 Hz. The data is stored in local flash memory. When an interesting event occurs, the node will route a message to the base station. If multiple nodes report the same event, then data is collected from the nodes in a round-robin fashion. When data collection is completed, the nodes return to sampling and storing sensor data locally.

In the 19 days of deployment, the network observed 230 eruptions and other volcanic events. About 61% of the data was retrieved from the network due to short outages in the network from software component failure and power outage. Overall, the system performed well in this study.

Health monitoring applications [33] using WSN can improve the existing health care and patient monitoring. Five prototype designs have been developed for applications such as infant monitoring, alerting the deaf, blood pressure monitoring and tracking, and fire-fighter vital sign monitoring. The prototypes used two types of motes: T-mote sky devices [29] and SHIMMER (Intel Digital Health Group's Sensing Health with Intelligence, Modularity, Mobility, and Experimental Re-usability).

Because many infant die from sudden infant death syndrome (SIDS) each year, Sleep Safe is designed for monitoring an infant while they sleep. It detects the sleeping position of an infant and alerts the parent when the infant is lying on its stomach. Sleep Safe consists of two sensor motes. One SHIMMER mote is attached to an infant's clothing while a T-mote is connected to base station computer. The SHIMMER node has a three-axis accelerometer for sensing the infant's position relative to gravity. The SHIMMER node periodically sends packets to the base station for processing. Based on the size of the sensing window and the threshold set by the user, the data is processed to determine if the infant is on their back.

Baby Glove prototype is designed to monitor vitals. Baby Glove is a swaddling baby wrap with sensors that can monitor an infant's temperature, hydration, and pulse rate. A SHIMMER mote is connected to the swaddling wrap to transmit the data to the T-mote connected to the base station. Like Sleep Safe, an alert is sent to the parent if the analyzed data exceeds the health settings.

FireLine is a wireless heart rate sensing system. It is used to monitor a fire fighter's heart rate in real-time to detect any abnormality and stress. FireLine consist of a T-mote, a custom made heart rate sensor board, and three re-usable electrodes. All these components are embedded into a shirt that a fire fighter will wear underneath all his protective gears. The readings are taken from the T-mote is then transfer to another T-mote connected to the base station. If the fire fighter's heart rate is increasing too high, an alert is sent.

Heart@Home is a wireless blood pressure monitor and tracking system. Heart@Home uses a SHIMMER mote located inside a wrist cuff which is connected to a pressure sensor. A user's blood pressure and heart rate is computed using the oscillometric method. The SHIMMER mote records the reading and sends it to the T-mote connected to the user's computer. A software application processes the data and provides a graph of the user's blood pressure and heart rate over time.

LISTSENse enables the hearing impaired to be informed of the audible information in their environment. A user carries the base station T-mote with him. The base station T-mote consists of a vibrator and LEDs. Transmitter motes are place near objects (e.g., smoke alarm and doorbell) that can be heard. Transmitter motes consist of an omni-directional condenser microphone. They periodically sample the microphone signal at a rate of 20 Hz. If the signal is greater than the reference signal, an encrypted activation message is sent to the user. The base station T-mote receiving the message actives the vibrator and its LED lights to warn the user. The user must press the acknowledge button to deactivate the alert.

ZebraNet [9] system is a mobile wireless sensor network used to track animal migrations. ZebraNet is composed of sensor nodes built into the zebra's collar. The node consists of a 16-bit TI microcontroller, 4 Mbits off-chip flash memory, a 900 MHz radio, and a GPS unit. Positional readings are taking using the GPS and sent multi-hop across zebras to the base station. The goal is to accurately log each zebra's position and use them for analysis. A total of 6–10 zebra collars were deployed at the Sweetwaters game reserve in central Kenya to study the effects and reliability of the collar and to collect movement data. After deployment, the biologists observed that the collared zebras were affected by the collars. They observed additional head shakes from those zebra in the first week. After the first week, the collared zebra show no difference than the uncollared zebra. A set of movement data was also collected during this study. From the data, the biologists can better understand the zebra movements during the day and night.

Open research issues

The enabling applications provide some key attributes that determine the driving force behind WSN research. Existing applications such as environmental monitoring, health monitoring, industrial monitoring, and military tracking have application-specific characteristics and requirements. These application-specific characteristics and requirements coupled with today's technology lead to different hardware platforms and software development. A variety of hardware platforms and technology have been developed over the years; however, more experimental work is necessary to make these applications more reliable and robust in the real world. Appendix C compares the application with the protocol stack.

WSNs have the potential to enhance and change the way people interact with technology and the world. The direction of future WSNs lies in identifying real business and industry needs. Interactions between research and development are necessary to bridge the gap between existing technology and the development of business solutions. Applying sensor technology to industrial applications will improve business processes as well as open up more problems for researchers.

5. Internal sensor system

For a sensor to operate in a wireless sensor network, there are several internal system issues that need to be addressed through the system platform and operating system (OS) support. In addition, supporting standards, storage, and physical testbeds are reviewed in the following subsections.

5.1. System platform and OS support

Current WSN platforms are built to support a wide range of sensors. Products that offer sensors and sensor nodes have different radio components, processors, and storage. It is a challenge to integrate multiple sensors on a WSN platform since sensor hardware is different and processing raw data can be a problem with limited resources in the sensor node. System software such as the OS must be designed to support these sensor platforms. Research in this area involves designing platforms that support automatic management, optimizing network longevity, and distributed programming. Below we discuss two platforms: a Bluetooth-based sensor system [34] and a detection-and-classification system [35].

Bluetooth-based sensor networks [20] reported a study to determine if a Bluetooth-based sensor node is viable for a WSN. Typical radio components used in a WSN are based on fixed frequencies where sensor nodes within communication range compete for a shared channel to transmit data. But Bluetooth is based on spread-spectrum transmission where separate channels are used to transmit data.

The Bluetooth-based devices used in the experiments are BTnodes developed by ETH Zurich [36]. A stripped-down version of the Bluetooth stack for TinyOS was designed and ported into the BTnodes. In order to support a multi-hop network, each BTnode is equipped with two radios: one configured to operate as a master and the other as a slave. The master radio can support up to seven connections while the slave radio looks for another node to connect to. Because Bluetooth is connection oriented, a master and slave connection must be established before data is exchanged. When a new node joins the network, its slave radio is first enabled. The new node tries to connect itself with the rest of the network. When the new node finds a node to connect to as its slave, it turns on the master radio to accept connections from nodes that are not yet connected to the network. If the new node fails to connect to other nodes in

its vicinity due to the maximum number of connections being reached at the other nodes, it re-connects to the first node it had contacted in the network. With the second request, the master radio in that node will drop one of its slave node connections and accept the connection from the new node. The disconnected node will find another node in its vicinity to connect. The network topology formed by this procedure is a connected tree.

Experimental results indicate that Bluetooth-based sensor networks using BTnodes are suitable for applications that are active over a limited time period with a few unpredictable traffic bursts. BTnodes can achieve high throughput; however, they consume a lot of energy even when idle. Connection maintenance is expensive and dual radios are needed to support multi-hop routing. Hence, Bluetooth can only serve as an alternative to broadcast radios.

Detection-and-classification system developed in VigilNet [35] can detect and classify vehicles, persons, and persons carrying ferrous objects. It targets objects with a maximum velocity error of 15%. The VigilNet surveillance system consists of 200 sensor nodes which are deployed in a pre-planned manner into the environment. Their locations are assigned at the time they are deployed. Each sensor node is equipped with a magnetometer, a motion sensor, and a microphone.

A hierarchical architecture was designed for this system in order to distribute sensing and computation tasks to different levels of the system. The hierarchical architecture is comprised of four tiers: sensor-level, node-level, group-level, and base-level. The lowest level, the sensor-level, deals with the individual sensor and its sensing algorithm to detect and classify objects. Once the sensing algorithm has processed the sensor data, the classification result is sent to the next level, namely the node-level. At the node-level, classification deals with the fusion of various sensor data obtained by the individual nodes. The node-level sensing algorithm relays the sensor data from each sensor and forms node-level classification results. Both the sensor-level and node-level classification functions reside on the node itself. The next level is the group-level. This level of classification is performed by a group of nodes. A set of nodes is organized in a group, and a group leader is elected to perform group-level classification. The input to the group-level classification is the node-level classification results of the aggregated attributes. At group-level classification, group leaders can accomplish more advanced tasks and gain better knowledge of the location of the targets. The highest level is the base-level classification. At this level, the results from the group-level classification are transmitted via multi-hop to the base station. The base-level classification algorithm finalizes the results collected and reduces false positives among the reported results.

VigilNet was deployed and tested in an outdoor site. The system was able to accurately detect targets and reduce false negatives with a dense deployment of sensor nodes.

5.2. Standards

Wireless sensor standards have been developed with the key design requirement for low power consumption. The standard defines the functions and protocols necessary for sensor nodes to interface with a variety of networks. Some of these standards include IEEE 802.15.4 [37], ZigBee [38,39], WirelessHART [40,41], ISA100.11 [42], IETF 6LoWPAN [43–45], IEEE 802.15.3 [46], Wibree [47]. The following paragraphs describes these standards in more detail.

IEEE 802.15.4: IEEE 802.15.4 [37] is the proposed standard for low rate wireless personal area networks (LR-WPAN's). IEEE 802.15.4 focuses on low cost of deployment, low complexity, and low power consumption. IEEE 802.15.4 is designed for wireless sensor applications that require short range communication to maximize battery life. The standard allows the formation of the star and peer-to-peer topology for communication between network devices. Devices in the star topology communicate with a central controller while in the peer-to-peer topology ad hoc and self-configuring networks can be formed. IEEE 802.15.4 devices are designed to support the physical and data-link layer protocols. The physical layer supports 868/915 MHz low bands and 2.4 GHz high bands. The MAC layer controls access to the radio channel using the CSMA-CA mechanism. The MAC layer is also responsible for validating frames, frame delivery, network interface, network synchronization, device association, and secure services. Wireless sensor applications using IEEE 802.15.4 include residential, industrial, and environment monitoring, control and automation.

ZigBee [38,39] defines the higher layer communication protocols built on the IEEE 802.15.4 standards for LR-PANs. ZigBee is a simple, low cost, and low power wireless communication technology used in embedded applications. ZigBee devices can form mesh networks connecting hundreds to thousands of devices together. ZigBee devices use very little power and can operate on a cell battery for many years. There are three types of ZigBee devices: ZigBee coordinator, ZigBee router, and ZigBee end device. ZigBee coordinator initiates network formation, stores information, and can bridge networks together. ZigBee routers link groups of devices together and provide multi-hop communication across devices. ZigBee end device consists of the sensors, actuators, and controllers that collects data and communicates only with the router or the coordinator. The ZigBee standard was publicly available as of June 2005.

WirelessHART: The WirelessHART [40,41] standard provides a wireless network communication protocol for process measurement and control applications. The standard is based on IEEE 802.15.4 for low power 2.4 GHz operation. WirelessHART is compatible with all existing devices, tools, and systems. WirelessHART is reliable, secure, and energy efficient. It supports mesh networking, channel hopping, and time-synchronized messaging. Network communication is secure with encryption, verification, authentication, and key management. Power management options enable the wireless devices to be more energy efficient. WirelessHART is designed to support mesh, star, and combined network topologies. A WirelessHART network consists of wireless field devices, gateways, process automation controller, host applications, and network manager. Wireless field devices are connected to process or plant equipment. Gateways enable the communication be-

tween the wireless field devices and the host applications. The process automation controller serves as a single controller for continuous process. The network manager configures the network and schedule communication between devices. It also manages the routing and network traffic. The network manager can be integrated into the gateway, host application, or process automation controller. WirelessHART standards were released to the industry in September 2007 and will soon be available in commercial products.

ISA100.11a: ISA100.11a [42] standard is designed for low data rate wireless monitoring and process automation applications. It defines the specifications for the OSI layer, security, and system management. The standard focuses on low energy consumption, scalability, infrastructure, robustness, and interoperability with other wireless devices. ISA100.11a networks use only 2.4 GHz radio and channel hopping to increase reliability and minimize interference. It offers both meshing and star network topologies. ISA100.11a also provides simple, flexible, and scaleable security functionality.

6LoWPAN: IPv6-based Low power Wireless Personal Area Networks [43–45] enables IPv6 packets communication over an IEEE 802.15.4 based network. Low power device can communicate directly with IP devices using IP-based protocols. Using 6LoWPAN, low power devices have all the benefits of IP communication and management. 6LoWPAN standard provides an adaptation layer, new packet format, and address management. Because IPv6 packet sizes are much larger than the frame size of IEEE 802.15.4, an adaptation layer is used. The adaptation layer carries out the functionality for header compression. With header compression, smaller packets are created to fit into an IEEE 802.15.4 frame size. Address management mechanism handles the forming of device addresses for communication. 6LoWPAN is designed for applications with low data rate devices that requires Internet communication.

IEEE 802.15.3: IEEE 802.15.3 [46] is a physical and MAC layer standard for high data rate WPAN. It is designed to support real-time multi-media streaming of video and music. IEEE 802.15.3 operates on a 2.4 GHz radio and has data rates starting from 11 Mbps to 55 Mbps. The standard uses time division multiple access (TDMA) to ensure quality of service. It supports both synchronous and asynchronous data transfer and addresses power consumption, data rate scalability, and frequency performance. The standard is used in devices such as wireless speakers, portable video electronics, and wireless connectivity for gaming, cordless phones, printers, and televisions.

Wibree: Wibree [47] is a wireless communication technology designed for low power consumption, short-range communication, and low cost devices. Wibree allows the communication between small battery-powered devices and Bluetooth devices. Small battery powered devices include watches, wireless keyboard, and sports sensors which connect to host devices such as personal computer or cellular phones. Wibree operates on 2.4 GHz and has a data rate of 1 Mbps. The linking distance between the devices is 5–10 m. Wibree is designed to work with Bluetooth. Bluetooth with Wibree makes the devices smaller and more energy-efficient. Bluetooth–Wibree utilizes the existing Bluetooth RF and enables ultra-low power consumption. Wibree was released publicly in October 2006.

5.3. Storage

Conventional approaches in WSNs require that data be transferred from sensor nodes to a centralized base station because storage is limited in sensor nodes. Techniques such as aggregation and compression reduce the amount of data transferred, thereby reducing communication and energy costs. These techniques are important for real-time or event-based applications, but they may not suffice. Applications that operate on a query-and-collect approach will selectively decide which data are important to collect. Optimizing sensor storage becomes important in this case when massive data is stored over time.

Given that storage space is limited and communication is expensive, a storage model is necessary to satisfy storage constraints and query requirements. In this subsection, we evaluate several storage methods in terms of design goals, assumptions, operation models, and performance.

GEM: Graph EMbedding (GEM) [48] provides an infrastructure for routing and data-centric storage for sensor networks. The idea of graph embedding works in two steps. The first step is choosing a labelled guest graph for routing and data-centric storage. The second step is to embed the guest graph onto the actual sensor topology. Each sensor node in this network is given an identifier and a label encoded with its position. Each sensor node needs only to know the labels of its neighbors. To support data-centric storage in GEM, each data item has a name that can be mapped to a label and stored at different nodes. When a client requests data, it sends a query with the data's name into the network. The node that has the data will route the data back to the requested. GEM enables node-to-node routing by using a lookup mechanism to find a node's current label. If two nodes need to communicate, the sender node must first retrieve the label of the receiving node. A lookup request message is sent by the sender to the receiver. Upon receiving the lookup request, the receiver retrieves the label in a distributed hash table. Once the sender node has the receiver's label, it can send messages to the receiver.

To demonstrate how GEM is applied to a sensor network, the virtual polar coordinate space (VPCS) was developed in this study. In VPCS, a ring-tree graph is embedded into the network topology. Each sensor node is assigned a level which is the number of hops from the root node. Each node is also assigned a virtual angle range which identifies the node within that level. The virtual angle range is a subset of its parent's virtual angle range. Children of a node may not have overlapping angle ranges. The virtual polar coordinate routing (VPCR) algorithm is built on top of VPCS to route a message from a node to another. VPCR utilizes polar coordinates for efficient routing. Each node has a label defined by a space in a VPCS. VPCR is greedy because it forwards packets closer to the destination angle range. Packet forwarding is accomplished by checking for nearby 2-hop neighbor nodes which have an angle range that is closer to the destination angle than the current node's angle range. If so, VPCR forwards the packet to that node.

Each node is required to store state information about its neighbors. VPCR makes routing more efficient by routing with cross-links in the ringed tree. Experimental results show that VPCR is efficient in both energy usage and routing.

TSAR: Two-tier sensor storage architecture (TSAR) [49] uses interval skip graphs to employ a multi-resolution ordered distributed index structure for efficient support of spatio-temporal and value queries. Sensor nodes send concise identifying information (or metadata) to a nearby proxy. Proxies interact with one another to construct a distributed index of the metadata reported by the sensors and an index of the associated data stored at the sensors. The index provides a logical view of the distributed data. The index is used to pinpoint all data from the corresponding sensors. Actual data remains in the sensor nodes. TSAR reduces energy overhead at sensor nodes by using the proxies for queries and low cost transmission of metadata to the proxies. There are four main contributions: (1) novel distributed index structure based on interval skip graphs, (2) each sensor's local archive to store data in flash memory, (3) a prototype of TSAR on a multi-tier testbed, and (4) a detailed evaluation of TSAR. Experimental results show feasibility and low energy latency of the distributed storage architecture in a multi-tier sensor network.

Multi-resolution storage: Multi-resolution storage system [50] provides storage and long-term querying of the data for data-intensive applications. Multi-resolution storage uses in-network wavelet-based summaries to store data in a spatially- and hierarchically-decomposed distributed storage structure. The storage system architecture is divided into three parts: (1) wavelet process to construct multi-resolution summaries, (2) drill-down query process to reduce search cost, and (3) a data-aging scheme to discard summaries. In the first part, the wavelet process uses a summarizing technique that provides data compression for spatio-temporal data sets. Wavelet construction has two phases: temporal summarization phase and spatial summarization phase. The first phase requires each node to compress the time-series data by exploiting temporal redundancy in the signal. The second phase constructs a hierarchical grid-based overlay. At each level, data is compressed more in a spatial scale. At the highest level, one or a few nodes contain an overall summary of all the data in the network.

The second part of the system architecture is the drill-down query process to reduce the cost of search. Drill-down queries are inserted at the highest level of the hierarchy and use a coarse summary as a hint to indicate which region in the network will most likely contain the response to the query. The query is forwarded to nodes that store summaries of these regions. The query is routed from one sub-region to the next till it reaches the lowest level of the hierarchy or when there are enough results in the intermediate nodes. The drill-down query process is very efficient in that it can obtain query results in a few steps.

Lastly, old data must be discarded in order to create space to store new data. To determine how old is the data in the network, each data is given an age that specifies the amount of time that the summary has been stored. Two data-aging schemes are proposed: a training-based algorithm and a greedy algorithm. The training algorithm operates on a limited training set of data. During the training period, aging parameters are extracted from a training set. The training set is typically data sensed during system deployment. A weighted cumulative error is computed from different queries. The cumulative error is fed into an optimization function to evaluate aging parameters for different summaries. For the greedy algorithm, there are no prior data sets to determine the aging parameters. It assigns weights to summaries according to expected importance of each resolution toward drill-down queries. The goal of the aging schemes is to provide data management and enhance the query process. Results show that both schemes perform within 2% of the optimal scheme, but the training scheme performed better than the greedy scheme.

5.4. Testbeds

A WSN testbed is consists of sensor nodes deployed in a controlled environment. It is designed to support experimental research in a real-world setting. It provides researchers a way to test their protocols, algorithms, network issues and applications. Experiments can easily be configured, run, and monitored remotely. Experiments can also be repeated to produce the same results for analysis. The following paragraphs describe several WSN testbeds in more detail.

ORBIT: Open access research testbed for next-generation wireless networks (ORBIT) [51] consists of 64 remotely accessible sensor nodes placed indoor with $\sim$1 m spacing apart. Each ORBIT radio node consists of a 1-GHz VIA C3 processor, two wireless PCI 802.11a/b/g interface, two ethernet ports, and an integrated chassis manager. Users can log on remotely to set up their experiment. ORBIT can be used to test new applications, measure system performance, run cross-layer experiments, and test new protocols and algorithms.

MoteLab: MoteLab [52] is a web-based WSN testbed consisting of a set of MicaZ motes [53] connected to a central server. The central server handles scheduling, re-programming and data logging of the motes. A user can log onto a web interface to create and schedule experiments. The goal of MoteLab is to allow users to evaluate WSN applications without manually re-programming and re-deploying the nodes into the physical environment. The users can retrieve data through the web interface and interact with individual nodes. MoteLab consists of the following software components: a SQL database, web interface, DB logger, and job daemon. The SQL database stores all the information needed for the test-bed operation. The web interface uses PHP to generate the web contents for the users to access. The DB logger is connected to each node to receive messages and store them in the SQL database. The job daemon is responsible for re-programming each node and starting and stopping system components. MoteLab have been used to study newly developed protocols, signal strength analysis, and cluster analysis.

Emulab: Emulab [54] is a remotely accessible mobile and wireless sensor testbed. The testbed consists of Acroname robots carrying an XScale based Startgate small computer and 900 Hz Mica2 mote [53]. The robots operate on

battery power which last up to 3 hours and uses 802.11b for communication. The radios are set to 900 MHz. The robot's motion and steering comes from two drive wheels that operate at a maximum rate of 2 m/s. There are six infrared proximity sensors on all sides of the robot to detect obstructions. Users can create experiments through a web interface and schedule events to control the robots movement. Emulab can be used to study network topologies, mobility effects on protocols, test algorithms, and mobile applications.

5.5. Diagnostics and debugging support

In order to guarantee the success of the sensor network in the real environment, it is important to have a diagnostic and debugging system that can measure and monitor the sensor node performance of the overall network. Studies that deal with handling various types of hardware and software failures help extend the life of each sensor which in turn help increase the sensor network lifetime. In addition to failures, addressing methods to enhance communication performance can make the system more efficient. In the following subsections, we first describe a tool call Sympathy [26] that detects and localizes failures. We then discuss the study reported in [55] which analyzes packet delivery performance at the physical and the medium access control (MAC) layers.

Sympathy: Sympathy [26] is a diagnosis tool for detecting and debugging failures in sensor networks. It is specifically designed for data-collection applications where nodes periodically send data back to a centralized base station or sink. Sympathy detects failures in a system by selecting metrics such as connectivity, data flow, node's neighbor and next hops. Connectivity metrics provide connectivity information from every node in the network. Sympathy collects every node's current routing table with information for next hop and path quality. Flow metrics provide the network's traffic load as well as its connectivity. Sympathy collects packet level information transmitted and received from each node. In addition, Sympathy also maintains information for packets transmitted from the sink to the nodes. Based on these metrics, Sympathy detects when nodes are not delivering sufficient data to the sink and locates the cause of the failure.

Sympathy can identify three types of failures: self, path, and sink. In self failure, the node itself has failed due to a crash, re-boot, bug in software code, or connectivity issue. In path failure, a node along the path fails, causing other nodes to fail or there are collisions along the path. In sink (i.e., base station) failure, the whole network appears to be failing when it is the sink that has failed. Failure at the sink may be due to bad sink placement, changes in the environment after deployment, and connectivity issues.

In Sympathy, the sink/base station runs the necessary software to detect and localize the failure. Localizing a failure is a four-stage process. In the first stage, the sink collects metrics from the sensor nodes in the system. Upon receiving a packet, Sympathy looks for failures by analyzing the received metrics and running tests to determine the cause. Common causes include a node crashing or rebooting, no route to the base station/sink, or the request never reaching the node. In these cases, Sympathy identifies the type of failure and reports it to the user. Hence, collecting information about each node allows Sympathy to detect failures more quickly.

Analysis of data packet delivery: the work in [55] studied packet delivery performance of a sensor network at the physical and MAC layers. At the physical layer, the work in [55] studies the performance of packet delivery under different transmit powers and physical-layer encoding. At the MAC layer, different MAC layer mechanisms such as carrier sensing and link-layer re-transmission are used to measure the efficiency of packet delivery. Up to 60 Mica motes were used to measure packet delivery under three different environmental settings: an office building, a habitat with moderate foliage, and an open parking lot. Under these settings, results show that both physical and MAC layers contribute to the packet-delivery performance, which is defined as the fraction of packets not successfully received by the receiver within a time window.

At the physical layer, traffic is generated by one node at one end of the line transmitting one packet per second. Packet-delivery performance is measured with the MAC layer disabled under different environments, coding schemes, and transmission settings. Results show that at least 20% of the nodes had at least 10% packet loss and at least 10% of the nodes had greater than 30% packet loss. Spatial characteristics show the existence of a gray area for some nodes. Nodes that are a certain distance from the sender have uniformly high packet reception rate. Beyond this distance is a gray area in which the reception rate changes dramatically. Receiving nodes in this gray area are likely to experience either 90% successful reception or less than 50% reception rate. The gray area defined for an office building and open parking lot is one-third of the total communication range while for habitat setting, it is one-fifth.

At the MAC layer, experiments vary in topology, environment, and traffic pattern. Packet losses in this case are largely due to lost transmissions. Under light load, nearly 50% of the links have an efficiency of 70% or higher. Under heavy load, nearly 50% of the links have efficiency less than 20%. Depending on the load, between 50% and 80% of the communication energy is used for repairing lost transmissions. Packet-delivery performance can be greatly improved by adding a simple set of mechanisms such as topology control to discard neighbors with asymmetric links.

Open research issues

The design of a WSN platform must deal with challenges in energy efficiency, cost, and application requirements. It requires the optimization of both the hardware and software to make a WSN efficient. Hardware includes using low cost tiny sensor nodes while software addresses issues such as network lifetime, robustness, self-organization, security, fault tolerance, and middleware. Application requirements vary in terms of computation, storage, and user interface and consequently there is no single platform that can be applied to all applications. Existing platforms discussed here include a Bluetooth-based sensor system

[34] and a detection-and-classification system [35]. Future work in this area entails examining a more practical platform solution for problems in new applications.

Storage capacity in low-end sensor nodes is limited. Rather than sending large amounts of raw data to the base station, a local sensor node's storage space is used as a distributed database to which queries can send to retrieve data. Existing approaches [48–50] present data structures that can efficiently manage and store the data. Nevertheless, energy-efficient storage data structure is still an open area of research that requires optimizing various types of database queries both with respect to performance and energy efficiency.

Performance studies provide valuable information for developing tools and solutions to improve system performance. Critical factors that influence system performance include scalability, communication, protocols at different layers, failures, and network management. Scalability issues can degrade system performance. Communication protocols are still trying to achieve a reasonable throughput when the size of the network increases. Optimizing and analyzing protocols at different layers can improve system performance and determine their benefits and limitations. Sensor nodes can fail at any time due to hardware, software, or communication reasons. It is important that there are services to handle these failures before and after they occur. Development of network management tools enables monitoring of system performance and configuring of sensor nodes.

6. Network services

Sensor provisioning, management, and control services are developed to coordinate and manage sensor nodes. They enhance the overall performance of the network in terms of power, task distribution, and resource usage. Provisioning properly allocates resources such as power and bandwidth to maximize utilization. In provisioning, there is coverage and localization. Coverage in a WSN needs to guarantee that the monitored region is completely covered with a high degree of reliability. Coverage is important because it affects the number of sensors to be deployed, the placement of these sensors, connectivity, and energy. Localization is the process by which a sensor node tries to determine its own location after deployment. Management and control services play a key role in WSNs as they provide support to middleware services such as security, synchronization, data compression and aggregation, cross-layer optimization, etc. In this section, we study provisioning, control, and management services based on their objectives. A brief summary of each plane is described in each of the sections below.

6.1. Localization

In WSNs, sensor nodes that are deployed into the environment in an ad hoc manner do not have prior knowledge of their location. The problem of determining the node's location (position) is referred to as localization. Existing localization methods include global positioning system (GPS), beacon (or anchor) nodes, and proximity-based localization. Equipping the sensor nodes with a GPS receiver is a simple solution to the problem. However, such a GPS-based system may not work when the sensors are deployed in an environment with obstructions such as dense foliage areas. The beacon (anchor) method makes use of beacon (anchor) nodes, which know their own position, to help sensors determine their position. This method has its shortcoming. It does not scale well in large networks and problems may arise due to environmental conditions. Proximity-based localization makes use of neighbor nodes to determine their position and then act as beacons for other nodes. Below we review some of the key localization techniques that differ from the above methods.

Moore's algorithm: Ref. [56] presents a distributed localization algorithm for location estimation without the use of GPS or fixed beacon (anchor) nodes. A key feature of this algorithm is the use of a robust quadrilateral. A robust quadrilateral is a fully-connected quadrilateral whose four sub-triangles are robust. Localization based on robust quadrilateral can be adjusted to support noisy measurements and it correctly localizes each node with a high probability.

This algorithm has three phases: cluster localization phase, cluster optimization phase, and cluster transformation phase. In the first phase, each node becomes the center of a cluster and measures the distance of its one-hop neighbors. The information gathered is broadcasted. For each cluster, each node computes the complete set of robust quadrilaterals and finds the largest sub-graph of overlapping robust quadrilaterals. Position estimations for a local coordinate system are computed for as many nodes as possible using the overlap graph using a breadth-first search. The second phase is an optimization phase that can be omitted. Position estimations are refined using numerical optimization such as spring relaxation or the Newton–Raphson method. The last phase computes the transformation between local coordinate system of connected clusters. The transformation computes the rotation, translation, and possible reflection that best aligns the nodes of two local coordinate systems.

There is, however, one drawback to this system. Under conditions of low node connectivity and high measurement noise, the algorithm may not be able to localize some nodes.

RIPS: The work in [57] proposes a localization system called Radio Interferometric Positioning System (RIPS) which utilizes two radio transmitters to create an interference signal. Two radio transmitters are placed at different locations and set at slightly different radio frequencies to provide ranging information for localization. At least two receivers are needed to calculate the phase offset of the observed signals. The relative phase offset is a function of the relative positions between the two transmitters and the receivers, and the carrier frequency. By measuring the relative phase offset, one can analyze and determine the relative locations of the two receivers or the location of the radio source if the receiver locations are known.

Spotlight: Spotlight [58] is a system that achieves high accuracy of localization without the use of expensive hardware like other localization systems. Spotlight uses an

asymmetric architecture where computation resides on a single Spotlight device. The Spotlight device uses a steerable laser light source which illuminates the sensor nodes that are placed in a known terrain. The main idea of the Spotlight localization system is to generate controlled events in the field where the sensor nodes are deployed. An event can be defined as a lighted sensor area. Using time events perceived by a sensor node and spatio-temporal properties of the generated events, spatial information regarding the sensor node can be inferred. Results show that Spotlight is more accurate than other range-based localization schemes and much more effective for long-range localization problems. The cost of localization is low since only one single device is necessary to localize the network.

Secure localization: Secure localization [59] focuses on securing the localization process. The goal is to prevent malicious beacon nodes from providing false location to sensors. Sensors rely on beacon information to compute their position. To prevent the localization process from being compromised, the following security requirement must be satisfied. Sensors must only accept information from authenticated beacon nodes. Sensors should only use information that has not been tampered. Sensors should be able to request location information at anytime. Upon a location request, information exchange must take place immediately and not at a later time. Neither a source's nor sensor's location should be disclosed at any time to prevent malicious nodes from taking over a location in the network. If any one of these requirements is breached, the localization process is compromised.

Some of the existing secure location techniques include SeRloc [60], Beacon Suite [61], DRBTS [62], SPINE [63], and ROPE [64]. SeRloc uses a set of locator nodes equipped with directional antennas to provide sensors with location information. Each locator transmits a different beacon at each antenna sector. An attacker would have to impersonate several locators to compromise the localization process. While SeRloc prevents attackers from compromising the localization process, beacon suite identifies the malicious beacon nodes. Beacon nodes serve two purposes: (1) provide location information to sensor nodes, and (2) detect malicious beacon signals. To detect malicious beacon signals, a beacon can request location information from another beacon in order to observe its behaviour. When a beacon node determines that the beacon that it's observing is misbehaving, it reports the beacon to the base station. A similar approach called distributed reputation and trust-based security (DRBTS) protocol identifies malicious information by enabling beacon node monitoring. Beacon nodes monitor each other and provide information to the sensor nodes. Sensor nodes can choose to accept a beacon's information based on votes from their common neighbors. Using this voting approach, sensor nodes can determine the trustworthy beacons within their range. It is demonstrated through simulation the robustness and effectiveness of DRBTS in large networks.

A centralized approach, secure positioning in sensor network (SPINE) is based on verifiable multi-lateration. SPINE bounds each sensor to at least three reference points within its range in order to compute its position. SPINE effectively prevents against nodes from lying about its position. Like SeRLoc, ROPE uses a set of locators to provide location information to the sensor nodes. Each sensor shares a pairwise key with every locator. Prior to data collection, ROPE provides a location verification mechanism to verify the locations of the sensors.

MAL: Mobile-assisted localization (MAL) [65] utilizes a mobile user (a human or robot) to assist in collecting distance information between itself and static sensor nodes for node localization. In node localization, a minimum number of distance samples must be collected before a node's coordinates can be computed. The goal is to re-construct the position of the nodes given a graph with measured distance edges. In MAL, a mobile user explores the sensor region and incrementally builds a localization graph between the mobile's various positions and the static sensor nodes. The number of measurements required by the mobile is linear to the number of static sensor nodes. When the required number measurement to build a rigid graph is obtained, an anchor-free localization (AFL) algorithm is run to compute the node's coordinate. AFL first computes the initial coordinate assignment of all the nodes using only node connectivity information. AFL then uses a non-linear optimization procedure to reduce the sum of squared distance errors between the node's actual distance and the distance of the current coordinate assignment. Simulation results show that MAL performs better in large mobile coverage areas. The estimated distance error decreases with the increasing number of nodes.

6.2. Synchronization

Time synchronization in a wireless sensor network is important for routing and power conservation. The lack of time accuracy can significantly reduce the network's lifetime. Global time synchronization allows the nodes to cooperate and transmit data in a scheduled manner. Energy is conserved when there are less collisions and retransmissions. In addition, energy is saved when nodes are duty-cycled.[5] Existing time synchronization protocols aim to accurately estimate time uncertainty and synchronize each node's local clock in the network. In the following subsection, we briefly review a few of these protocols.

Uncertainty-driven approach: Ref. [66] proposes an uncertainty-driven approach to duty-cycling by modelling long-term clock drifts between nodes to minimize the duty-cycling overheads. This approach uses long-term empirical measurements to evaluate and analyze three key parameters that influence long-term synchronization. The parameters are synchronizing rate, history of past synchronization beacons, and the estimation scheme. By measuring these parameters, one can design a rate-adaptive, energy-efficient, long-term time synchronization algorithm, called the rate-adaptive time synchronization (RATS) protocol. RATS's objective is to maximize the synchronization sampling period while bounding the prediction error within the user-defined error bound. During

[5] Sensor nodes are duty-cycled to save energy. In duty-cycle, the sensor node would periodically turn its radio off to save energy and on to participate in network communication.

runtime, RATS repeatedly computes the synchronization sampling period and the prediction error. To keep the prediction error within the user-defined error bound, the multiplicative increase and multiplicative decrease (MIMD) strategy is used to adapt the sampling rate and minimize energy usage. MIMD is simple and can adapt to system changes and environmental conditions. If the predicted error is below the lower threshold, the sampling period is increased multiplicatively. If the prediction error is above the upper threshold, the sampling period is decreased multiplicatively. The sampling period remains the same when the prediction error is between the two thresholds. Results show that the protocol is able to reduce energy consumption and provide synchronization precision for different applications.

Lucarelli's algorithm: This scheme [67] considers synchronization with bi-directional nearest-neighbor coupling. Nodes in the network converge to a synchronized state based on local communication topology. Each sensor node contains a state variable x_i that increase from 0 to 1. When x_i reaches 1, the sensor node emits a pulse signal and resets to 0. Each sensor node fires periodically at a fixed rate. A node which hears its neighbor's signal would increment its state variable x_i by the amount of $\varepsilon g(x_i)$ where ε is a small coupling constant and $g(x_i)$ is a positive value between [0, 1]. It is guaranteed that, over time, the nodes will converge to synchronicity.

Reachback firefly algorithm (RFA): RFA [68] is a decentralized synchronicity algorithm implemented on TinyOS-based motes. RFA accounts for sensor network communication effects such as message loss and delays. The algorithm is based on a mathematical model proposed in [69], explaining how neurons and fireflies spontaneously synchronize. The firefly synchronization is robust and adapts to changes such as losses, adding nodes, and link changes. The algorithm works such that each node in the network acts as an oscillator with a fixed time period T. Each node has an internal time t which it increments till T. At time T, the node will fire a signal and reset the internal clock t back to zero. Neighboring nodes that observe the firing will shorten their own time to fire. The time to shorten is determined by a function called the firing function and a small constant ε. After some time, nodes in the network will synchronize to a common phase and firing pulse.

Unlike other algorithms, RFA aims to resolve related wireless communication issues. Three of these issues are (1) estimating the delay of a message before it is sent, (2) handling messages from a previous time period, and (3) handling wireless contention. RFA uses the MAC layer to record the time delay between when a node fires and when the message is transmitted. With the time-delay information, a node receiving the firing message can determine the actual time the firing message was sent by subtracting the MAC-layer time delay from the reception time of the message. RFA uses the reachback response to handle delayed messages from a previous time. When a node hears a neighbor fire, it places the message in a queue until time $t = T$ before it retrieves the messages from the queue. After processing the messages, the node makes an overall increment of t. With reachback response, a node is always reacting to information that is one time period old. Lastly, RFA avoids repeated collisions by adding a random transmission delay to the node-firing messages at the application level. After a node fires, it waits for a grace period before processing the queued messages. Results show that RFA is able to achieve synchronicity and deal with communication latencies at the same time.

Timing-sync protocol for sensor network (TPSN): TPSN [70] provides time synchronization for every sensor node in the network. TPSN is based on a conventional sender–receiver synchronization approach. TPSN has two phases, a level discovery phase and synchronization phase. In the first phase, the algorithm creates a hierarchical topology in the network. Every sensor node is assigned a level in the hierarchical structure. A sensor node at level i can communicate with at least one sensor node at level i-1. Only one sensor node is assigned with level 0 which is called the root node. The root node is responsible for initiating the second phase once the hierarchical structure has been established. In the second phase, each sensor node tries to synchronize with a sensor node that is one level lower than them. Eventually, the sensor nodes will synchronize with the root node. When the root node is synchronized, the whole network is then time synchronized.

During the synchronization phase, packet collisions may occur. When collisions occur, nodes will timeout for a random time and re-transmit. This process continues until a two-way message exchange has been completed. Over time, sensor nodes may die off. When a sensor node is cannot find any neighbors that is one level lower than it, it will broadcast a level request message so that it can be assign a new level in the hierarchy. This is assuming that the network is still connected and has at least one neighbor node that is higher than the sensor node. If the root node dies, the nodes at level 1 will run a leader election algorithm to elect a new leader. When a new leader is elected, TPSN is run again with the level discovery phase. The performance of TPSN was compared against the reference broadcast synchronization (RBS) [71] approach which is based on a receiver–receiver synchronization. Results show that TPSN is two times better than RBS.

Clock-sampling mutual network synchronization (CSMNS): CSMNS [72] is a distributed and autonomous network synchronization approach. CSMNS does not depend on a centralized node to synchronize time nor does it depend on special circuitry to send continuous pulses. It is a non-hierarchical approach that supports single and multi-hop communication. It exchanges timing information with IEEE 802.11 periodic beacon transmission. In a network of N nodes, each with a clock that has a different time-drift coefficient and initial time, the main goal of CSMNS is to synchronize all the clocks and minimize the relative time drift of the time process. Each node in the network contends to send its time process in periodic beacon transmission. Upon receiving a beacon transmission, the node computes the difference between the time stamp of the received beacon and the time stamp of the local node for the correction factor. The node then set its clock to the value of the adjusted time stamp if it is later than its own. An extension of CSMNS called CSMNS-RMN reduces the number of nodes contending to send a beacon

at every target beacon transmission time. Every node will contend to send its beacon within the contention window. If a node receives a beacon before sending its own, it will not contending to send its beacon. After a while, a single node called the rotating master (RM) node will win the contention. All nodes have equal opportunity to be a RM node. Using this approach, there is significant energy saving from reduce beacon collision.

Time synchronization (TSync): TSync [73] is an accurate, lightweight, flexible, and comprehensive time solution for WSNs. TSync uses multi-channel radios for frequency diversity to reduce packet collisions and interferences. By reducing the number of collisions, the variance in round-trip delay decreases as a result improves the accuracy in time estimation. TSync consists a pull and push mechanism. The pull mechanism is an individual-based time request (ITR) protocol. ITR allows each sensor node to independently synchronize itself with the surrounding environment. A sensor node first sends a query message on the control channel to get a clock channel for time synchronization. The query message travels upstream until it reaches a reference node, i.e., base station. The reference node sends an acknowledgement message back to the specified the clock channel. All nodes along the path switch to the specified clock channel. The sensor node then sends a synchronization request on the specified clock channel to the reference node. The reference node then sends back the time to the sensor node. The push mechanism is a hierarchy referencing time synchronization (HRTS) protocol. HRTS enables a reference node to synchronize multiple sensor nodes. In HRTS, the reference node initiates the synchronize process by broadcasting a beacon on the control channel. A sensor node specified by the reference node sends a reply to the reference node. The reference node calculates the clock offset and broadcasts it to all its surrounding sensor nodes. The surrounding sensor nodes synchronize themselves and repeat this process with their neighboring nodes away from the reference node. Both ITR and HRTS achieve different accuracy and can be parameterized to suit a given application.

Global synchronization: Li and Rus [74] discuss three methods to global synchronization: all-node-based method, cluster-based method, and fully localized diffusion-based method. The all-node-based method routes a message along a specified cycle path and synchronize all the nodes along the path. An initiating node sends a message along the cycle. Each node receiving the message records its local time and order in the cycle. When the initiating node receives its message, the initiating node sends another message to the nodes providing information of the starting and ending time of the last message. Each node then adjusts its local time with the computed clock error. In the cluster-based method, the network is synchronized using a hierarchical approach. The sensor nodes are first organized into clusters where they adjust their clocks according to the cluster head's clock. The cluster heads are then synchronized using the all-node-based method. The fully localized diffusion-based method achieves global synchronization by averaging all clock readings and adjusting each clock in the network to the average time. A sensor node in the network that has a high clock value sends its time to all the neighbor nodes and then decreases its local clock time. Nodes that have a low clock value reads the time and increasing its clock value. After a number of diffusion rounds, each sensor node will have the same clock value.

Synchronization protocol classification: Sundararaman et al. [75] have classified synchronization protocols based on two kinds of features: application-dependent features and synchronization issues. Application dependent features are classified into single-hop vs. multi-hop networks, stationary vs. mobile networks, and MAC layer-based vs. standard-based approach. Synchronization issue involves sensors adjusting their local clocks to a common time scale. Options proposed to resolve these issues include master–slave synchronization, peer-to-peer synchronization, clock correction, untethered clocks, internal synchronization, external synchronization, probabilistic synchronization, deterministic synchronization, sender-to-receiver synchronization, and receiver-to-receiver synchronization. Master–slave synchronization assigns one node in the network to be the master and the rest to be slaves. The slave node synchronizes its local clock with the master node. In peer-to-peer synchronization, nodes communicate directly with each other to exchange time information until the network is synchronized. Clock correction is a method in which nodes in the network either instantaneously or continually corrects its local clock to keep the entire network synchronized. Untethered clock achieves common time without synchronization. In this approach, timestamps are exchanged between nodes and compared to achieve a global time scale. Internal synchronization is based on a global time to minimize the local clock offset, whereas external synchronization uses a standard source of time such as the universal coordinated time (UTC). Another method called probabilistic synchronization guarantees that the failure probability can be bounded while deterministic synchronization guarantees a deterministic upper bound on the clock offset. In sender-to-receiver synchronization, the sender sends its timestamp to the receiver. The receiver then synchronizes its time with the sender's timestamp and computes the message delay. In receiver-to-receiver synchronization, receivers receive the same broadcast message and exchange the timestamp at which they received the broadcast message. Each receiver then computes the offset based on the difference in receive times.

6.3. Coverage

Given a WSN, the problem of determining sensor coverage for a designated area is important when evaluating the WSN's effectiveness. The quality of monitoring in a WSN is dependent on the application. Applications such as target tracking may require a higher degree of coverage to track the target accurately while applications such as environmental or habitat monitoring can tolerate a lower degree of coverage. A higher degree of coverage requires multiple sensors monitoring the same location to produce more reliable results. Existing research focuses on coverage in the context of energy conservation. Some have proposed techniques to select the minimal set of active nodes to be

awake to maintain coverage. Others have proposed sensor deployment strategies for distributed detection in large-scale sensor networks. In the following subsection, we describe several of these protocols.

Coverage configuration protocol (CCP): CCP [76] is a decentralized protocol that configures the network to provide a specific degree of coverage. During runtime, CCP can change the degree of coverage in the network when requested by the application. In CCP, a node can be in one of three states: sleep, active, and listen. In the sleep state, the node turns off its radio until the sleep timer expires, and then it enters the listen state. In the listen state, the node collects hello messages from its neighbors and executes the Ks-coverage eligibility algorithm. The Ks-coverage eligibility algorithm determines whether a node is eligible to switch states. If every location within a node's coverage range is not Ks-covered by other active nodes, the node will be eligible to become active, else it will go back to sleep. In active mode, the node periodically updates its sensing neighbor table and executes the Ks-coverage eligibility algorithm to determine if it will remain active.

CCP is integrated with SPAN [77] to provide both coverage and connectivity. In CPP with SPAN, connectivity in the network is guaranteed if the communication range is less than twice the sensing range. The eligibility rule for CCP and SPAN are combined. The eligibility rule states that, in order for inactive nodes to become active, they must be eligible in either the SPAN or the CCP eligibility rule. An active node will withdraw if it does not satisfy either the SPAN or the CCP eligibility rule.

Minimal and maximal exposure path algorithms: Ref. [78] evaluates network coverage using the minimal and maximal exposure path method. The minimal exposure path is defined as the path between two given points such that total sensor exposure along the path is minimized whereas the maximal exposure path is the path where total sensor exposure along the path is maximized.

This work first defined the closed-form solution for the minimal exposure path using a single sensor. The single sensor model measures a sensor's sensitivity to an object in the sensor field. The sensor's sensitivity is defined by a function that is inversely proportional to distance between the sensor and the object. As the object moves closer to the sensor, the higher is the sensor's sensitivity. The minimal exposure path in this case is solved analytically by the method of variational calculus. For multiple sensors in the network, a grid-based approximation algorithm solves the minimal exposure path. The algorithm utilizes the Voronoi cell concept to determine the largest sensor exposure value at any given point and returns the path which is within the bounded error of the minimal exposure path.

To determine the maximal exposure path, the authors proved that the problem is NP-hard and generated approximate solutions. The proposed heuristic methods are random path, shortest path, best point, and adjusted best point. The random-path heuristic finds a path using shortest-path nodes and random nodes. Shortest-path nodes are selected with some percentage while random nodes are selected to increase the chance of exposure. As a result, the path found will contain a reasonable total exposure. The shortest-path heuristic model uses the shortest path between two points as the maximal exposure path. The result here may give the shortest path; however, it is not the optimal solution. The best-point heuristic superimposes a grid over the sensor field. Using an ellipse as the search space, it finds the shortest path that connects the starting and ending points with each grid point. Two shortest paths that share the same grid points are combined to compute the total exposure. The path that contains the highest exposure will be the maximal exposure path. The adjusted best-point heuristic improves upon the best-point heuristic by selecting paths that contain multiple shortest paths. In addition, it adjusts the path by adding, moving, and deleting nodes to increase path exposure. The authors have shown that the adjusted best-point algorithm outperforms all the other heuristics.

Differentiated Surveillance Service Protocol: Ref. [79] proposes a differentiated surveillance service protocol which provides different degrees of sensing coverage in a WSN. The protocol is an extension of an adaptive energy-efficient sensing coverage scheme. In this protocol, sensor nodes are static and know only their own location. Each node is either in sleeping or working mode. Sensor nodes go through two phases: initialization and sensing. In the initialization phase, a sensor node determines its own location and synchronizes time with its neighbors. After the initialization phase, it enters the sensing phase where a working schedule is set up. The sensing phase divides time into rounds of equal duration. A working schedule is set up to determine when a node should remain awake or go to sleep. The working schedule for each node is a four tuple $(T, Ref, T_{\text{front}}, T_{\text{end}})$ where T is the duration of each round, *Ref* is a random time reference point, T_{front} is the duration of time before the reference point, and T_{end} is the duration of time after the reference point. The number of rounds is defined by i. A node wakes up at time $(T \times i + \text{Ref} - T_{\text{front}})$ and sleeps at time $(T \times i + \text{Ref} + T_{\text{end}})$. The value of T is constant and pre-determined across all nodes. The reference time value is uniformly chosen between $[0, T)$. T_{front} and T_{end} are computed based on nearby-neighbor reference points. This is to guarantee that the area is covered by at least one node. In order to provide differentiated surveillance service, the values of T_{front} and T_{end} can either be increased or decreased proportionally. By increasing these parameters, nodes will be awake for a longer period of time, thereby increasing the degree of sensing coverage, whereas decreasing the time on these parameters will decrease coverage.

6.4. Compression and aggregation

Both data compression and aggregation reduce communication cost and increase reliability of data transfer. Data compression and aggregation are necessary for WSN applications which have large amount of data to send across the network. Depending on the importance of the data, one method may be better than the other. Data-compression techniques involve compressing the size of the data before transmission. Decompression of the data occurs at the base station. In data compression, it is important that no information is lost and individual data readings are retained.

For data aggregation, data is collected from multiple sensors and combined together to transmit to the base station. In this case, aggregated data is more important than individual readings. This method is often used in a cluster-based approach. Each of these techniques addresses the issue of energy, robustness, scalability, accuracy, and efficiency.

Synopsis diffusion: Synopsis diffusion [80] provides a framework for in-network aggregation and best-effort multi-path routing. Synopsis diffusion enables energy savings, robustness to different topologies, and improved data accuracy. Accuracy and reliability of data are achieved by controlling the level of redundancy in message routing. By adapting message redundancy with network conditions, energy consumption is reduced.

Synopsis diffusion performs in-network aggregation using three functions: synopsis generation, synopsis fusion, and synopsis evaluation. The synopsis generation function takes a sensor reading and creates a synopsis to represent the data. The synopsis fusion function takes two synopses and creates one new synopsis. The synopsis evaluation function translates a synopsis to its final answer. The synopsis diffusion algorithm consists of two phases: distribution and aggregation. During the distribution phase, a query node floods the network with query messages. In the aggregation phase, each node uses the fusion function to merge its local synopses with the received synopses. The query node receiving the fused data translates it using the evaluation function. Synopsis diffusion supports message redundancy detection by using a set of order-and-duplicate-insensitive (ODI) synopses generation and fusion functions. The synopsis diffusion framework is topology independent and can be applied to any topology structure. Results show that synopsis diffusion reduces error in loss conditions and addresses node failures. Lastly, synopsis diffusion also improves upon energy consumption.

q-Digest: A novel data structure, called q-digest (Quantile Digest) [81], aims to capture the distribution of sensor data in an energy-efficient manner and provide error guarantees. A q-digest is a subset of a complete tree which contains only nodes with significant data values. The q-digest encodes information about the distribution of sensor values. The size of the q-digest is determined by a compression parameter k. Each node in the q-digest must satisfy two digest properties. The first property states that only leaf nodes may have high data values. The second property states that there should not be a node and its children with low data values. If the data value is small, a child's data value is merged into its parent's data value to achieve compression.

q-Digest has the following properties: error-memory tradeoff, confidence factor, and multiple queries. The q-digest framework allows a user to specify the message size and error tradeoff. q-Digest adapts these values by staying within the specified bound and providing error guarantees. To provide the best possible error guarantees, the error for each particular q-digest structure is computed. This computed error is known as the confidence factor. The confidence factor ensures that error in any query is bounded, else it is discarded. q-Digest supports a variety of queries such as average query, median query, and histogram query. Each query is initiated by the base station. For each query, nodes of the q-digest are traversed and information is reported back to the base station. Results indicated that q-digest can accurately preserve information and approximate queries using limited memory and power.

6.5. Security

A WSN is vulnerable to threats and risks. An adversary can compromise a sensor node, alter the integrity of the data, eavesdrop on messages, inject fake messages, and waste network resource. Unlike wired networks, wireless nodes broadcast their messages to the medium. Hence, the issue of security must be addressed in WSNs.

There are constraints in incorporating security into a WSN such as limitations in storage, communication, computation, and processing capabilities. Designing security protocols requires understanding of these limitations and achieving acceptable performance with security measures to meet the needs of an application. Below we review several security proposals at different layers of the protocol stack.

Decentralized key-exchange protocol: This protocol [82] guarantees the confidentiality of a key exchange even if an attacker has compromised some nodes in the network. The objective of the protocol is to minimize resource consumption on the individual devices in terms of memory requirements, CPU usage, and network traffic. The protocol guarantees the secrecy of a key exchange as long as there is less than s subverted nodes. The protocol uses s node-disjoint paths in an s-connected graph to distribute key shares. The nodes will use these key shares to generate a session key. If a key graph contains s node-disjoint paths between the source and destination, the source will randomly generates s key shares $k_1 \dots k_s$ of identical length and sends them over the s node-disjoint paths to the destination. On each link of the path, the key share is encrypted and integrity protected with the existing share key for this link. Once key share is established, the attacker cannot recover data without access to all the key shares. Simulation results show that the network traffic grows linearly during key establishment.

LKE: Location-aware key establishment (LKE) [83] is resilient against node capture attacks in large-scale sensor networks. LKE requires only a small amount of space to store keying information. LKE consist of four phases: pre-distribution phase, node self-configuration phase, polynomial share-distribution phase, and pairwise key-establishment phase. In the pre-distribution phase, all sensors are programmed and configured the same before deployment. A sensor's role and position is configured after deployment in the node self-configuration phase. Sensors determine their position based on a localization technique. Using the location information, each sensor differentiates itself as either a worker or a service node. Service nodes are self-elected. They are in charge of key space generation and key information distribution. If a sensor is not a service node, it is a worker node. Worker nodes get their key information from the service nodes in order to communicate with other nodes in the network. The polynomial

share-distribution phase securely disseminates the polynomial share information to the worker nodes in three steps. The first step is the key space advertisement where the service node broadcasts its location, and public key information to the worker nodes. The second step is secure channel establishment. A worker node, which receives the server message, checks its validity to prevent false information. For each valid announcement that the server node receives, a computationally-asymmetric channel based on Rabin's cryptosystem [84] is established. Both the service node and the worker node agree on a shared key. With the shared key, the service node encrypts the computed location-aware polynomial share and transmits it to the worker node in the last step. LKE employs an efficient pairwise key-establishment scheme for node communication. Two sensors sharing a common key space based on their location information can communicate with a common key. If two sensors do not share any key space, intermediate nodes are exploited for path key establishment. LKE is resilient against node capture attacks as long as no more than t sensor nodes are captured with the same key space. When more than t sensor nodes are compromised, all secure links within the key space are compromised. Results show that LKE requires low storage overhead in worker nodes and provides resilience against attacks.

TinySec: TinySec [85] uses link-layer security architecture to guarantee message authenticity, integrity, and confidentiality. Message authenticity is the ability to detect false messages and reject them. Similar to message authenticity is message integrity, the detection of a tampered message. TinySec provides message authenticity and integrity by including a message authentication code (MAC) with each packet. The MAC is a cryptographically-secure checksum of a message. The MAC is computed using a share secret key between the sender and the receiver. The sender computes the MAC of a packet using its secret key. The packet and the MAC are sent to the receiver. The receiver sharing the same secret key re-computes the MAC value of the message and compares it against the MAC received. If they are the same, the packet is accepted, else it is dropped. If an adversary alters the message during transit, he/she would not be able to re-compute the MAC value. Hence, the receiver will reject the message. Message confidentiality keeps information safe from unauthorized members. In this case, the encryption mechanism should achieve semantic security. Semantic security implies that adversaries cannot learn any property of the message even if they have obtained the message. TinySec achieves semantic security by using a unique initialization vector (IV) as a side input to the encryption algorithm. The purpose of IV is to add variation to the encryption process when there is little variation in the message set. The receiver must use IV to decrypt messages. Using IV, adversaries will not be able to determine the contents of messages simply by looking at its encryption.

TinySec supports authentication encryption (TinySec-AE) and authentication only (TinySec-Auth) modes of operation. With TinySec-AE, the data payload is encrypted and the packet is authenticated using the MAC. With TinySec-Auth, only authentication is performed on the packet with a MAC. TinySec utilizes cipher block chaining (CBC) for data encryption. CBC is used together with non-repeating IV to provide strong confidentiality guarantees.

Open research issues

Provisioning, management, and control services are needed to sustain network connectivity and maintain operations. Provisioning services such as localization and coverage can improve network performance. Efficient algorithms can reduce the cost of localization while sensor nodes are able to self-organize and identify themselves in some spatially coordinated system. Localization has been studied extensively to minimize energy, cost, and localization errors. The problem of energy conservation while maintaining a desired coverage has also been studied. Coverage efficiency depends on the number of active nodes. The more active nodes there are in the network, the higher is the degree of coverage. Coverage protocols should meet different levels of coverage requirements and be energy efficient. Existing solutions have investigated different degrees of coverage along with network connectivity. Future research and development should continue to focus on optimizing coverage for better energy conservation.

Management and control services include synchronization, data aggregation and compression, security, and cross-layer optimization. In a dense WSN, there is a need for network-wide time synchronization. Time synchronization eliminates event collision, energy wastage, and non-uniform updates. Proposed time synchronization protocols aim to synchronize local node clocks in the network and reduce energy overhead. Continuing research should focus on minimizing uncertainty errors over long periods of time and dealing with precision.

With large amounts of data generated over time, the cost of transferring all of the sensed data to the base station is expensive. Data compression and aggregation techniques aid in reducing the amount of data to be transferred. The development of various compression and aggregation scheme for event-based or continuous data collection network is a challenging research topic.

For security monitoring in a WSN, secure protocols have to monitor, detect, and respond to attacks with uninterrupted service. Many proposed secure protocols are for the network layer and data-link layer. Malicious attacks can occur at any layer in the protocol stack. Secure monitoring for different layers of the protocol stack need to be explored. Cross-layer secure monitoring is another challenging area for research.

7. Communication protocol

The development of a reliable and energy-efficient protocol stack is important for supporting various WSN applications. Depending on the application, a network may consist of hundreds to thousands of nodes. Each sensor node uses the protocol stack to communicate with one another and to the sink. Hence, the protocol stack must be energy efficient in terms of communication and be able to work efficiently across multiple sensor nodes. We review the various energy-efficient protocols proposed for the transport layer, network layer, and data-link layer, and their cross-layer interactions in the following subsections.

7.1. Transport layer

The transport layer ensures the reliability and quality of data at the source and the sink. Transport layer protocols in WSNs should support multiple applications, variable reliability, packet-loss recovery, and congestion control mechanism. The development of a transport layer protocol should be generic and independent of the application. It should provide variable packet reliability for different applications. Each WSN application can tolerate different levels of packet loss. Packet loss may be due to bad radio communication, congestion, packet collision, full memory capacity, and node failures. Any packet loss can result in wasted energy and degraded quality of service (QoS) in data delivery. Detection of packet loss and correctly recovering missing packets can improve throughput and energy expenditure.

There are two approaches for packet recovery: hop-by-hop and end-to-end. Hop-by-hop retransmission requires that an intermediate node cache the packet information in its memory. This method is more energy efficient since retransmission distance is shorter. For end-to-end retransmission, the source caches all the packet information and performs retransmission when there is a packet loss. End-to-end retransmission allows for variable reliability whereas hop-by-hop retransmission performs better when reliability requirements are high.

A congestion control mechanism monitors and detects congestion, thereby conserving energy. Before congestion occurs, the source is notified to reduce its sending rate. Congestion control helps reduce retransmission and prevents sensor buffer overrun. As in packet-loss recovery, there are two approaches to congestion control: hop-by-hop and end-to-end. Hop-by-hop mechanism requires every node along the path to monitor buffer overflows. Hop-by-hop mechanism lessens congestion at a faster rate than the end-to-end mechanism. When congestion is detected by a sensor node, all nodes along the path change their behaviour. End-to-end mechanism relies on the end nodes to detect congestion. Congestion is flagged when timeout or redundant acknowledgements are received.

There are tradeoffs between hop-by-hop and end-to-end approaches for packet-loss recovery and congestion control mechanism. Depending on the type, reliability, and time-sensitivity of the application, one approach may be better than the other. Existing transport layer protocols in WSNs attempt to address the above design issues.

Sensor transmission control protocol (STCP): STCP [86] is a reliable transport layer protocol that provides variable reliability, congestion detection and avoidance, and support of multiple applications in the same network. Functionalities of STCP are executed at the base station. The base station is assumed to have high processing capability, storage, and power to communicate with all the nodes in the network.

A source node must transmit a single session initiation packet to the base station before sending data. The session initiation packet contains information about the number of flows from the node, the type of data flow, transmission rate, and required reliability. The sensor node must wait for an acknowledgement from the base station before transmitting data. For continuous data flows, the base station estimates the time of arrival of each packet from each source. If a packet is not received by the base station within a given period of time, the base station determines whether the current required reliability is met. Reliability is a measure of the fraction of packets that are successfully received. If current reliability goes below the required level, the base station sends out a negative acknowledgement (NACK) to the source node for retransmission. Each source node stores its transmitted packets in a buffer. When the buffer reaches a threshold, it is cleared.

For event-driven flows, the source node computes the reliability of the packet reaching the base station. If the computed value is more than the required reliability, the node will not buffer the packet to save storage space. The base station sends out positive acknowledgement (ACK) for each packet received from a source node. When an ACK reaches the source node, the corresponding transmitted packet is deleted from the buffer. Every sensor node maintains two thresholds in its buffer: low and high thresholds. When the buffer reaches the lower threshold, the congestion bit is set with a certain probability. Once the buffer reaches the higher threshold, the congestion bit is set for all packets. The congestion bit is a flag informing the base station to either notify the source to reduce its transmission rate or re-route packets along a different path.

Price-oriented reliable transport protocol (PORT): PORT [87] minimizes energy consumption, achieves the necessary level of reliability, and provides a congestion-avoidance mechanism. PORT minimizes energy consumed by avoiding high communication cost. End-to-end communication cost is the measure of the amount of energy consumed to deliver a packet from the source to the base station (sink). To achieve the necessary level of reliability and minimize energy, the source's reporting rate is dynamically adjusted in a bias manner. PORT provides an in-network congestion mechanism to alleviate traffic dynamically.

PORT differs from other transport protocols in that its view of reliability is not a ratio of the total incoming packet rate to the desire incoming rate, but the assurance that the sink obtains enough information on the phenomenon of interest. When a phenomenon of interest occurs, nodes closer to the phenomenon will contain more information and less error. PORT adapts bias packet reporting rate of the sensor nodes to increase the sink's information regarding the phenomenon. PORT provides two mechanisms that ensure this reliability. The first is a dynamic source report rate feedback mechanism to allow the sink to adjust the reporting rate of each data source. Each packet sent by the source is encapsulated with its node price. Node price is the total number of transmission attempts made before a successful packet is delivered from the source to the sink. It is a metric used to evaluate the energy cost of the communication. The sink adjusts the reporting rate of each source based on the source's node price and the information provided about the physical phenomenon. Feedback from the sink is sent to the sources along the reverse path. The second mechanism provides the sink with end-to-end communication cost information from the source to the sink.

End-to-end communication cost is used to alleviate congestion. When congestion occurs, communication cost increases with respect to packet loss. The sink uses the communication cost information to slow down the reporting rate of the appropriate source and increase the reporting rate of other sources that have lower communication cost since reliability must be maintained.

GARUDA: GARUDA [88] is a reliable downstream data delivery transport protocol for WSNs. It addresses the problem of reliable data transfer from the sink to the sensors. Reliability is defined in four categories: (1) guarantee delivery to the entire field, (2) guarantee delivery to a subregion of sensors, (3) guarantee delivery to a minimal set of sensors to cover the sensing region, and (4) guarantee delivery to a probabilistic subset of sensors.

GARUDA's design is a loss-recovery core infrastructure and a two-stage NACK-based recovery process. The core infrastructure is constructed using the first packet delivery method. The first packet delivery method guarantees first packet delivery using a Wait-for-First-Packet (WFP) pulse. WFP pulse is a small finite series of short duration pulses sent periodically by the sink. Sensor nodes within the transmission range of the sink will receive this pulse and wait for the transmission of the first packet. The first packet delivery determines the hop-count from the sink to the node. Nodes along the path can become candidates for the core. A core candidate elects itself to be a core node if it has not heard from neighboring core nodes. In this manner, all core nodes are elected in the network. An elected core node must then connect itself to at least one upstream core node.

GARUDA uses an out-of-order forwarding strategy to overcome the problem of under-utilization in the event of packet losses. Out-of-order forwarding allows subsequent packet to be forwarded even when a packet is lost. GARUDA uses a two-stage loss-recovery process. The first stage involves core nodes recovering the packet. When a core node receives an out-of-sequence packet, it sends a request to an upstream core node notifying that there are missing packets. The upstream core node receiving that message will respond with a unicast retransmission of the available requested packet. The second stage is the non-core recovery phase, which involves non-core nodes requesting retransmission from the core nodes. A non-core node listens on all retransmissions from its core node and waits for completion before sending its own retransmission request.

Delay sensitive transport (DST): DST protocol [89] addresses the issue of congestion control, reliability, and timely packet delivery. DST has two components: a reliable event transport mechanism and a real-time event transport mechanism. Reliable event transport mechanism measures the observed delay-constrained event reliability against the desired delay-constrained event reliability to determine if appropriate action is needed to ensure the desire reliability level for event-to-sink communication. The observed delay-constrained event reliability is defined as the number of packet received within a certain delay bound at the sink over a specified interval. The desired delay-constrained event reliability is the minimum number of data packets required for the event to be a reliable detection. If the observed delay-constrained event reliability is greater than the desire delay-constrained event reliability, the event is considered to be reliable. Otherwise, the report rate of the sensors must be increased to assure that the desired reliability level is met. DST also assures reliable and timely event detection within the event-to-sink delay bound. The real-time event transport mechanism uses this event-to-sink delay bound delay to achieve the application specific objectives. The event-to-sink delay is a measure of the event transport delay and event process delay. Event transport delay is the time between the event occurring and when the sink receives it. Event process delay is the processing delay at the sink. For congestion detection, DST measures buffer overflow at each node and computes the average node delay. Upon congestion, sensor nodes inform the sink of the congestion situation. The sink in response would adjust the reporting rate of the sensors. Simulation experiments show that DST achieves reliability and timely event detection with minimum energy consumption and latency.

Pump slowly, fetch quickly (PSFQ): PSFQ [90] is a reliable transport protocol that is scalable and robust. The goals of PSFQ are to guarantee data segment delivery, minimize the number of transmission for lost detection and data recovery operation, to operate in harsh environments, and to provide a loose delay bound for data delivery. PSFQ operates in three functions: pump operation, fetch operation, and report operation. The pump operation controls the rate at which data packets are passed along into the network. The pump operation is based on a simple scheduling scheme which used two timers, $T_{\min}$ and $T_{\max}$. A node must wait at least $T_{\min}$ before transmitting a packet. By waiting at least $T_{\min}$, a node is given the opportunity to recover missing packets and reduce redundant broadcasts. $T_{\max}$ is used as a loose upper delay bound for when all packets should be received. The fetch operation is called when there is a gap in the sequence number between the packets received. The fetch operation requests a retransmission of the lost packet from the neighboring nodes. If multiple packets are lost in a bursty event, a single fetch would be sent to retrieve the packets. Lastly, the report operation provides a feedback status to the users. A status report message travels from the farthest target node in the network to the requesting user. Along the path, each node appends its report message in an aggregated manner into the original message. Results show that PSFQ outperforms the idealized scalable reliable multi-cast (SRM-I) [5] in terms of tolerance, communication overhead, and delivery latency.

Event-to-sink reliable transport (ESRT): ESRT protocol [91] is developed for reliable event detection with minimum energy expenditure. ESRT uses a congestion control mechanism to reduce energy consumption while maintaining the desired reliability level at the sink. ESRT algorithm is run mainly at the sink. The sink computes the reliability factor and reporting frequency at each interval. The reliability factor is a measure of the data packets received from the source nodes to the sink. The computed reliability factor is compared against an application-defined desired reliability. If the computed reliability is greater than the desired reliability, ESRT would reduce the

reporting frequency of the source nodes. If the computed reliability is lower than the desired reliability, ESRT would increase the reporting frequency of the source nodes to achieve the desired reliability. At each interval, the sink broadcasts the new reporting frequency to source nodes in the network. Upon receiving the information, source nodes adjust their reporting rate. In ESRT, the congestion control mechanism is based on monitoring the routing buffer of each sensor nodes. A sensor node whose buffer overflows is an indication of congestion. Upon experiencing congestion, the sensor node sets the congestion notification bit flag in its outgoing packets. The sink receiving these packets along with the computed reliability factor determines the state of the network and acts accordingly. Simulation results show that ESRT is able to attain the desired reliability level with minimum energy expenditure under different network states with random and dynamic topologies.

Congestion detection and avoidance (CODA): CODA [92] is an energy-efficient congestion control scheme that can quickly mitigate congestion, once detected. CODA has three components: congestion detection, hop-by-hop backpressure, and multi-source regulation. CODA detects congestion by monitoring buffer occupancy and measuring channel load. Monitoring of the buffer size requires only a small amount of overhead and processing. When buffer occupancy is high, sensors listen to the local channel load conditions to detect congestion. Once congestion is detected, the sensor node broadcasts a suppression message to its neighbors and makes adjustment to prevent congestion downstream. It also broadcasts a backpressure message upstream to the source. Each upstream node receiving the backpressure message determines whether or not to propagate the message. Depending on the congestion policy, a node can prevent further congestion build up by dropping the incoming data packets or adjust their sending rate. In the event of a persistent congestion, CODA uses a closed-loop multi-source regulation method to assert congestion control over multiple sources from the sink. When the source node event rate is less than some fraction of the maximum theoretical throughput of the channel, the source regulates its own rate. Upon exceeding this value, the source node is most likely to be contributing to congestion. In this case, the source enters sink regulation. The sink sends a message to the source with a pre-defined event rate that the sink computed. When congestion is relieved, the sensor node would then regulate itself again without the sink. Results show that CODA can improvement performance and reduce energy usage.

Open research issues

The main goal of a transport layer protocol in a WSN is to achieve reliable data transport by inspecting the network state for congestion and reliability. Table 1 summarizes and compares the transport protocols reviewed. STCP, DST, ESRT and PORT address the problem of congestion control and reliability guarantee from the sensors to the sink while GARUDA and PSFQ examine only the problem of reliability from the sink to the sensors. STCP, GARUDA, and PSFQ incorporate packet recovery. STCP provides end-to-end reliability guarantee and uses ACK/NACK for loss detection and notification. PORT also provides end-to-end reliability guarantee but does not recover lost packets. GARUDA, however, uses hop-by-hop reliability guarantee and NACKs for packet recovery.

Although many transport layer protocols have been proposed for WSNs, there are still several open research problems such as cross-layer optimization, fairness, and congestion control with active queue management. The transport layer can benefit from cross-layer interactions. Cross-layer interactions can improve the performance of the transport protocol by selecting better paths for retransmission and getting error reports from the link layer. There is some research in this area; however, cross-layer

Table 1
Comparison of transport layer protocols for WSNs

		STCP	PORT	GARUDA	CODA	DST	PSFQ	ESRT
Congestion	Congestion control	Yes	Yes	No	Yes	Yes	No	Yes
	Congestion detection	Buffer size	Node price and link-loss rates	–	buffer size and channel load	Buffer size and average node delay calculation	–	Buffer size
	Congestion mitigation	Traffic redirection or end-to-end rate adjustment	Traffic redirection or end-to-end rate adjustment	–	Drop packets or adjust sending rate at each node	End-to-end rate adjustment	–	End-to-end rate adjustment
Reliability	Direction	Sensor to sink	Sensor to sink	Sink to sensor	Sensor to sink	Sensor to sink	Sensor to sink	Sensor to sink
	Reliability measure	Packet reliability	Event information reliability	Packet and destination reliability	–	Event reliability	Packet reliability	Event reliability
	End-to-end/Hop-by-hop	End-to-end	–	Hop-by-hop	–	End-to-end	Hop-by-hop	End-to-end
	Packet recovery	Yes	No	Yes	–	No	Yes	No
	Cache	Yes	–	Yes	–	–	Yes	–
	ACK/NACK	ACK, NACK	–	NACK	ACK	–	NACK	–
Energy conservation		Yes	Yes	Yes	Yes	Yes	–	Yes

optimization across multiple layers needs to be explored more. Current congestion control mechanisms focus on monitoring of the channels and dynamically adjusting the data rate of the source when congestion occurs. There is no active monitoring of the queue to avoid congestion. Incorporating active queue management with congestion control may further reduce packet loss and increase throughput. The transport protocol should guarantee fairness among sensor nodes. One solution to this problem is to assign packets with priority. The problem of guaranteeing fairness in a frequently-changing topology has not been extensively explored.

7.2. Network layer

The network layer handles routing of data across the network from the source to the destination. Routing protocols in WSNs differs from traditional routing protocols in several ways. For one, sensor nodes do not have Internet protocol (IP) addresses, so IP-based routing protocols cannot be used in a WSN. The design of network protocols in a WSN needs to be scalable. It should easily manage communication among many nodes and propagate sensor data to the base station. The protocol should meet network resource constraints such as limited energy, communication bandwidth, memory, and computation capabilities. By meeting these constraints, a sensor network's lifetime can be prolonged. Lastly, the protocol should address issues of efficiency, fault tolerance, fairness, and security. A few representative approaches are described below.

Geographical routing: Geographical routing [93] uses a greedy forwarding mechanism to forward a packet from the source to the destination. This approach forwards packet by choosing neighbors which are closest to the destination. It assumes that the network is sufficiently dense, nodes know their own location and their neighbors' locations, and multi-hop forwarding is reliable.

Several novel forwarding strategies are proposed to improve the performance of geographic routing. These forwarding strategies can be divided into two categories: distance-based and reception-based. For distance-based forwarding, a node only knows the distance of its neighbors while in reception-based forwarding the packet reception rates of its neighbors are also known. Distance-based forwarding consists of the original greedy forwarding and distanced-based blacklisting. In original greedy forwarding, each node forwards packets to the neighbor closest to the destination based on a minimum reception rate. A minimum reception rate must be met before two nodes can become neighbors. Original greedy forwarding selects neighbors with highest distance, independent of the reception rate. In distance-based blacklisting, each node blacklists neighbors that are above a certain distance threshold from itself. The blacklist distance threshold is set as a fraction of a nominal radio range. Packets are forwarded to the neighbor closest to the destination from those neighbors at a distance less than the threshold from the current forwarding node.

Four reception-based forwarding schemes are proposed:

1. *Absolute reception-based blacklisting*: Each node blacklists all neighbors that have a reception rate below a certain threshold. Only neighbors closest to the destination with a reception rate above the threshold will receive the packet for forwarding.
2. *Relative reception-based blacklisting*: A node blacklists a different set of neighbors for each new destination. Blacklisting of neighbors depend on the node's ranking within a set of neighbors. A node's ranking depends on its distance to the destination and the reception rate. Relative reception-based blacklisting prevents all neighbors to be blacklisted as in absolute reception-based blacklisting.
3. *Best reception neighbor*: Best reception neighbor forwards packets to neighbors with the highest reception rate from the neighbors that are closer to the destination.
4. *Best reception rate and distance*: Best reception rate and distance is based on the product of the reception rate and the distance. The node computes this product value for all neighbors that are close to the destination. The neighbor with the highest product value will be chosen.

Results show that reception-based forwarding strategies are more efficient than distance-based strategies. Relative reception-based blacklisting achieves higher delivery rates than absolute reception-based blacklisting. Overall, the new geographical forwarding strategies are better in terms of energy and minimizing route disconnection than the original greedy forwarding approach.

Anchor location service (ALS): ALS protocol [94] is a grid-based protocol that supports location-based routing between multiple moving sources and destinations. The sources and destinations are all sensor nodes in the network. ALS first constructs a predefined geographical grid structure for the network. Before deployment, each sensor node contains information regarding the size of each grid cell and the base line coordinates. Sensor nodes are randomly deployed and they obtain their own location using an existing positioning mechanism, e.g., GPS. Knowing its location, the sensor determines which grid square it falls into and decides whether it will become a grid node. Sensors that become grid nodes establish connections with neighboring grid nodes.

Multiple destination nodes may exist in the network. Each destination node selects a nearby grid node to be its sink agent. The sink agent is responsible for distributing location information of the destination node using an anchor system. The anchor system is made up of a set of grid nodes, called anchors, which act as location servers. When an event occurs, a sensor node becomes a source node that transmits data to a destination node. The source node first registers itself to the nearest grid node that becomes the source agent. The source agent queries the anchor system to locate the destination node and reports the information to the source node. Upon receiving the sink agent's information, the source sends data packets to the sink agent using a location-based routing protocol.

Secure routing (SecRout): SecRout protocol [95] guarantees secure packet delivery from the source to the sink. SecRout employs a two-level cluster-based approach to se-

cure the network. The lower level contains sensors or cluster members while the upper level contains cluster heads. In the self-organization phase, sensor nodes are divided into clusters. Each cluster contains a cluster head. Sensor nodes communicate to the sink (or base station) via cluster heads. Data is first sent from the sensors to the cluster head. The cluster head aggregates the data from its members and sends it to the sink (or base station).

For secure packet delivery, SecRout uses symmetric cryptography to secure packets along the path. Each sensor node is given a unique identity (ID) and a preloaded key (KEY). The ID identifies the node and the KEY is used to secure messages sent to the sink. The sink is assumed to be a high power node with high memory and computation capability. The sink also knows about network topology and all sensor node information. A table containing each node's ID and KEY pair is maintained by the sink. It is assumed that the sink cannot be compromised and can be trusted.

Secure data transfer starts with a sensor node encrypting its data packet using a cluster key. The cluster key is generated by the cluster head during the self-organizing phase and is shared among sensor nodes within the cluster. Upon receiving the encrypted information, the cluster head verifies the data using its cluster key. If the verification succeeds, the cluster head will decrypt the data. The cluster head collects data from all its members and then aggregates the data to form a new data packet. The new data packet will be encrypted with the cluster head's preloaded key and sent to the sink via multi-hop routing. The sink receiving the packet again verifies the authenticity of the packet. If verification succeeds, it will decrypt the packet and store the information.

SecRout guarantees that packets will reach the sink even if malicious nodes exist in the route. Routing packets and data packets contain only partial path information such as the next-hop neighbor. Each sensor node maintains a routing table containing partial routing path (previous and next node) to the sink. When a node is compromised, it will not be able to obtain information about the traversed intermediate nodes. SecRout provides route maintenance to update the routing table and trigger new route discovery when it detects a malicious node.

Secure cell relay (SCR): SCR [96] routing protocol is designed to provide resistance against security attacks. SCR is a cluster-based algorithm where nodes form a cluster (cell) based on their locations. SCR differs from other cluster-based algorithm in that there is no cluster head election. An active node becomes the relay node (cluster head) based on its remaining energy. In SCR, the entire network is divided into equal-sized square cells. Each sensor is statically aware of its own location and that of the base stations. It is assumed that the base stations can be trusted while sensors can be compromised.

Before deployment, each sensor node and the base station share a common global key, K_G, used for initial neighbor discovery and handshake phase communication. It is assumed that, before deployment, all sensor nodes and the base station are synchronized. SCR uses symmetric encryption to secure packets. After deployment, the base station encrypts its location information with K_G and floods it to all sensor nodes. In the neighbor discovery phase, sensor nodes discover their neighbors via a three-way handshake protocol, which establishes the shared secret keys between neighbor nodes. After establishing a table of shared secret keys of all neighbors, each sensor node destroys K_G and uses the shared secret key for future communication with its neighbors.

Based on the location of the source and the sink, a routing path is formed through a series of cells in the direction from source to sink. SCR routing provides two or more backup paths determined by the source. When an adversary attacks a node, the backup paths will be used to forward packets. SCR provides defence against the following attacks: Sybil, wormhole and sinkhole, selective forwarding, and hello flood. In Sybil attack, the adversary can pretend to be a node. Since shared keys are known only between the neighboring nodes, the attack will fail. In wormhole and sinkhole attack, an adversary can broadcast a new route, tunnel, or fake link to the base station. Nodes that receive this new route will not use it because they route only through the routing cell path. In selective forwarding, the adversary pretends to be a relay node in a cell and drops some packets while forwarding others. To prevent this attack, a node in a cell can only become a relay node for a user-defined number of times. After this number, the source will have to set up another path to route around this node. In addition, the sink will be notified that the current relay node will no longer be a relay node. In hello flood attack, the adversary node tries to establish a unidirectional link with a sensor node. This will not work since a sensor node uses three-way handshake to establish its neighbors and the shared secret keys.

Open research issues

Many routing protocols have been proposed for routing data in sensor networks. Table 2 summaries the characteristics of routing protocols covered in this survey. Important considerations for these routing protocols are energy efficiency and traffic flows. In this review, two categories of routing approaches are explored: location-based routing and cluster-based routing. Location-based routing considers node location to route data. Cluster-based routing employs cluster heads to do data aggregation and relay the information to the base station. A comparison of security routing protocol is also included in the table. A security routing protocol strives to meet security requirements to guarantee secure delivery of the data from the source to the destination.

Future research issues should address security, QoS, and node mobility. Experimental studies regarding security applied to different routing protocols in WSNs should be examined. There is little research in QoS routing in sensor networks. QoS guarantees end-to-end delay and energy-efficient routing. In applications where sensor nodes are mobile, new routing protocols are needed to handle frequent topology changes and reliable delivery.

7.3. Data-link layer

The data-link layer is concerned with the data transfer between two nodes that share the same link. Since the

Table 2
Comparison of network layer protocols for WSNs

Description	Geographical routing	ALS	SecRout	SCR
Routing type	Location-based	Location-based	Cluster-based	Cluster-based
Scalability	Fair	Good	Good	Good
Synchronization	No	No	Yes	Yes
Data cache	No	No	Yes	–
Data aggregation	No	No	Yes	Yes
Computation overhead	Neighbor selection/ blacklisting	Each anchor processes sink location information	Data aggregation, encrypting and decrypting packets	Encrypting and decrypting packets
Communication Overhead	Neighbor discovery	Network and anchor system setup, and sink query process	Setup and maintaining clusters	Setup cells, neighbor discovery and three-way handshake
Data security	No	No	Yes	Yes
Energy requirement	Not specified	Not specified	High power base station	High power base station

underlying network is wireless, for effective data transfer, there is a need for medium access control and management. The MAC protocol design should have the following attributes: energy efficiency, scalable to node density, frame synchronization, fairness, bandwidth utilization, flow control, and error control for data communication.

Error detection and correction services are offered at the data-link layer as well as the transport layer. One of the widely used error-detection technique is cyclic redundancy check (CRC) [97]. CRC operates as follows in WSN. The sender and receiver must first agree on a fixed data block size before transmission. The sender splits a packet from the network layer into data blocks which will be reassembled at the receiver. An 8-bit CRC can be used for error detection. The blocks containing the data and the CRC bits are packaged into a frame. Each frame is sent to the receiver. Upon receiving the frame, the receiver identifies whether the data block contains error. If there are error blocks, the receiver will initiate the recovery process to retrieve those error blocks after receiving a certain number of frames.

Recovery techniques in WSN include automatic repeat request (ARQ) [97], forward error correction (FEC) [98], hybrid ARQ (HARQ) [99], simple packet combining (SPaC) [100], and multi-radio diversity (MRD) [101]. ARQ uses acknowledgement and timeout to provide explicit feedback to the sender. The feedback can be in the form of a positive acknowledgement (ACK) or a negative acknowledgement (NACK). The sender receiving a NACK or timing out will retransmit the data frame. A limitation to ARQ is that it is limited to frame error detection. An entire frame has to be retransmitted if there is a single bit error. FEC, on the other hand, decreases the number of retransmissions. The sender adds some more amount of redundant data into each message so that the receiver can detect and correct errors. The advantage of FEC is that retransmission is reduced and the wait time for sending an acknowledgement and retransmitting the data can be avoided. Hybrid ARQ is a variation of the ARQ method. In hybrid ARQ, both ARQ and FEC are combined. There are two types of Hybrid ARQ schemes: type-I and type-II. Type-I includes both the error detection and error correction bits in every transmission packet and using a correction code to correct error. Type-II transmits either the error detection bits or the FEC information along with the data. If an error is detected in the first packet, it will wait for the second packet which contains the FEC parities and error detection to correct the error. If errors still exist, the packets are combined to error correct itself. SPaC and MRD perform error correction by combining corrupted packets and using HARQ. SPaC buffers corrupted packet at the receiver and waits for retransmission. Rather than retransmitting the original packet, the sender transmits parity bits. Upon receiving the retransmission packet, the receiver performs packet combining to recover the errors. MRD uses two techniques to recover from error. The first technique is frame combining with multiple erroneous frames together in the attempt to avoid re-transmission. The second technique is the request-for-acknowledgement (RFA) scheme to recover the packet.

The design of the MAC protocol in a WSN is subject to various constraints such as energy, topology, and network changes. Minimizing energy to extend the network lifetime is its primary goal. The design of the MAC protocol should prevent energy wastage due to packet collisions, overhearing, excessive retransmissions, control overheads, and idle listening. It should also adapt to topology and network changes efficiently. A wide range of MAC protocols have been proposed to achieve high channel utilization, collision avoidance, and energy efficiency. We review some of the representative approaches below.

TRAMA: Ref. [102] proposed a TRaffic-Adaptive Medium Access protocol (TRAMA) to increase channel utilization in an energy-efficient manner. TRAMA attains energy efficiency by avoiding collisions and switching to an idle state when there are no transmissions. To avoid collisions, TRAMA adapts its transmission schedule according to traffic information patterns. TRAMA assumes a single, time-slotted channel for data and control signal transmissions. Time is divided into sections of random-access and scheduled-access periods. TRAMA supports unicast, multi-cast, and broadcast traffic.

TRAMA consist of three components: (1) Neighbor Protocol (NP), (2) Schedule Exchange Protocol (SEP), and (3) Adaptive Election Algorithm (AEA). In TRAMA, nodes start in random-access mode where each node transmits at random slots. Nodes can join the network at random access periods. During this period, NP sends out small signalling packets to gather neighbor updates. If there are no updates, signalling packets are sent as keep-alive beacons. The signalling packets are used to maintain connectivity between neighbors. A node deletes a neighbor node from its table if

it does not hear from the neighbor within a certain period of time.

The second component, SEP, sets up the traffic-based schedule. The schedule captures the traffic window for which the node can transmit. During scheduled-access period, the node periodically broadcasts its schedule information to its one-hop neighbors. Each node generates a schedule by computing the schedule interval. The schedule interval represents the number of slots that the node can announce its schedule information to the neighbors. The schedule is sent along with every data packet. The *ChangeOver* slot is the last slot in the current schedule interval. All nodes listen during the *ChangeOver* slot to synchronize their schedule.

The last component, AEA, determines the state of the node. For energy efficiency, nodes are switched to sleep most of the time. A TRAMA node determines the current state (transmit, receive, or sleep) that it should be in based on the node's priority within the two-hop neighbor and the one-hop neighbor schedules. A node computes its priority value using a pseudo-random hash function during each time slot. Both priority and schedule information are used to determine if a node will become a transmitter or receiver for that time slot while others will switch into sleep mode. Nodes selected to transmit can give up their slots for re-use if they do not have data to send.

TRAMA guarantees delivery and energy efficiency with the expense of packet delays. It achieves high throughput and avoids collision.

B-MAC protocol: Unlike TRAMA, the Berkeley media access control (B-MAC) [103] is a reconfigurable carrier-sense multiple access (CSMA) protocol that achieves low power processing, collision avoidance, and high channel utilization. B-MAC optimizes system performance by employing an adaptive preamble sampling scheme. A set of adaptive bi-directional interfaces is used to reconfigure the protocol based on the network load.

B-MAC contains the following functionality: clear channel assessment (CCA) and packet back off, link-layer acknowledgement, and low power listening (LPL). For collision avoidance, B-MAC utilizes CCA to determine if the channel is clear. CCA is an outlier algorithm that searches for outliers in the received sample signals. An outlier exists if the channel energy is significantly below the noise floor. During the channel sampling period, if an outlier is found, the channel is clear, else the channel is busy. In case of a busy channel, packet backoff is used. Backoff time is either initially defined or randomly chosen.

B-MAC supports link-layer acknowledgement for unicast packets. When the receiver receives a packet, an acknowledgement packet is sent to the sender. To reduce power consumption, B-MAC employs an adaptive preamble sampling scheme called LPL. LPL performs periodic channel sampling by cycling through awake and sleep periods. In the awake period, the node's radio is turned on to check for activities in the channel using CCA. If activities are detected, it will remain awake to receive the incoming packet. Once it receives the packet, it will go back to sleep. Idle listening occurs when the node is awake but there is no activity in the channel. A timeout will force the node to go back to sleep.

All B-MAC functionality such as acknowledgements, CCA, and backoff can be changed through a set of adaptive bi-directional interfaces. By enabling or disabling B-MAC functionality, the throughput and energy consumption of a node can change.

Z-MAC protocol: In comparison to B-MAC, Z-MAC [100] is a hybrid MAC protocol that combines the strength of the TDMA and CSMA while offsetting their weaknesses. Z-MAC achieves high channel utilization and low latency under high contention. It reduces collisions between two-hop neighbors at a low cost. Z-MAC is robust to dynamic topology changes and time synchronization failures which commonly occur in the network. Z-MAC uses CSMA as the baseline MAC scheme and a TDMA schedule to enhance contention resolution. This design results in high initial overhead which is amortized over a long period of network operation and eventually improves the throughput and energy efficiency. The protocol uses an efficient and scalable channel scheduling algorithm for channel re-use and slot assignment. Unlike TDMA, a node may transmit during any time slot. It will always perform carrier sensing and transmit a packet when the channel is clear. An owner of the slot will have higher priority over non-owners to access the channel. The goal is to allow the re-use of a slot when the slot owner is not transmitting data. By mixing CSMA and TDMA, Z-MAC becomes more robust to timing failures, time-varying channel conditions, slot-assignment failures, and topology changes. Performance results show that Z-MAC is better than B-MAC under medium to high contention. Under low contention, B-MAC is slightly better in terms of energy.

Low power reservation-based MAC protocol: The low power reservation-based MAC protocol [104] addresses the issue of energy conservation and adaptation to traffic. To address the issue of energy conservation, the reservation-based MAC protocol uses a clustered hierarchical network and a TDMA-like frame structure. In a clustered hierarchy network, nodes organize themselves into clusters and contend for the role of cluster head in each cluster. Cluster head nodes are responsible for synchronizing all the nodes in their cluster to a TDMA schedule. The TDMA-like frame structure has contention-based slot reservation, schedule establishment, and slotted data transmission. Unlike traditional TDMA with fixed frame size, the protocol adapts the TDMA frame size according to the probability of successful data transmission. According to the protocol, the cluster head increases the frame size if the number of failures exceeds a predetermined value. On the contrary, if the number of failure is small, it will decrease the frame size. By adapting the frame size, the probability of success of packet transmission is increased. Nodes are able to effectively transmit at a higher data rate as a result of increasing throughput. In terms of energy dissipation, adaptive frame size shows significant energy savings due to less collisions and high probability of success.

Low power distributed MAC protocol: [105] presents a low power distributed MAC protocol which combines CSMA/CA and multi-channel spread spectrum techniques. For a given frequency band, the band is partition into multiple channels. A channel and code is assigned to each node in the network. The channel and code assigned must be unique across each node's two-hop neighbors. The primary

goal is to avoid collisions and minimize energy wastage. The protocol also incorporates a new wake up radio operation to save energy. Two radios are used, a low power wake-up radio and a normal data radio. The low power radio monitors the network and triggers the normal radio to wake up when there is data to transmit or receive. The normal radio switches between active and sleep modes. Results show that energy consumption for channel monitoring is almost negligible and average energy consumption is significantly reduced.

Spatial correlation-based collaborative MAC (CC-MAC): CC-MAC protocol [106] exploits the spatial correlation of the data at the MAC layer to regulate and prevent redundant transmissions. CC-MAC has two components: event MAC (E-MAC) and network MAC (N-MAC). E-MAC filters out the correlated data packets while N-MAC prioritizes the routing packets. CC-MAC protocol is implemented into each sensor node. In a WSN, the E-MAC protocol forms correlation regions to filter out correlated event information. In each region, a single representative sensor node is selected to transmit its data while all other sensor nodes back off for a specified period. At the end of each period, all nodes in that region except the representative sensor node go into a contention phase to get elected as the new representative. With E-MAC protocol filtering out correlated packets, the N-MAC protocol routes the packet to the sink using a priority-based method. Route-through packets are given higher precedence over newly-generated packets. Routing nodes use a backoff procedure to avoid collisions between multiple route-through packets transmitting at the same time. In terms of performance, CC-MAC protocol shows significant savings in energy, latency, and packet drop rate.

Open research issues

Table 3 compares the MAC protocols reviewed above. Both TRAMA and Z-MAC require a random access period and a schedule exchange period. In addition, time synchronization must be achieved in the network. In comparison with other contention-based protocols, TRAMA has higher delay and is suited for applications that are not time sensitive. B-MAC and Z-MAC both adapt well to topology changes while TRAMA does not. B-MAC has higher throughput under low contention environment while Z-MAC performs better in high contention environments. Low power reservation-based MAC, low power distributed MAC, and TRAMA minimize energy with sleep cycles when nodes do not have data to transmit or receive. CC-MAC, on the other hand, filters correlated information and prioritizes packets.

Although various MAC protocols have been proposed, there is possible future work for system performance optimization. Cross-layer optimization is an area that needs to be explored more extensively. Cross-layer interaction can reduce packet overhead on each of the layers, thereby reducing energy consumption. Interaction with the MAC layer can provide other layers with congestion control information and enhance route selection. Many existing MAC protocols address performance studies of static sensor nodes, but there is still a lack of literature for comparing these protocols in a mobile network. By enhancing the MAC protocol, one can significantly improve communication reliability and energy efficiency.

7.4. Physical layer

The physical layer provides an interface for transmitting bit streams over the physical-communication medium. It is responsible for interacting with the MAC layer, performing transmission and reception, and modulation. The interaction between the physical layer and MAC layer is an important issue. Error rate at the physical layer is high and time-varying in a wireless environment. The MAC layer interacts

Table 3
Overview of a representative set of link-layer protocols

Attributes	TRAMA	B-MAC	Z-MAC	Low power reservation-based MAC	Low power distributed MAC	CC-MAC
Channel access mode	Time-slotted random and scheduled access	Clear channel assessment (CCA)	Time-slotted random and scheduled access	Time-slotted contention based slot reservation	Multi-channel access	Time-slotted contention based slot reservation
Time synchronization	Yes	No	Yes	Yes	No	No
Protocol type	TDMA/CSMA	CSMA	TDMA/CSMA	TDMA	CSMA/CA	CSMA/CA
Protocol specifics	Achieves adequate throughput and fairness through transmitter-election algorithm and channel re-use	Bi-directional interface for reconfiguration of system services to optimize performance	Exploits the strengths of TDMA and CSMA while offsetting their weaknesses	Increases the probability of success in packet transmission by adapting to traffic requirements to maximize data throughput	Combines CSMA and spread spectrum techniques to achieve higher power efficiency and bandwidth	Filters out correlated data and ensures prioritization of packets to the sink which results in achieving higher network performance
Energy conservation	Schedule sleep intervals and turn radio off when idle, collision avoidance scheduling	Low power listening (LPL) time for energy efficiency	Low power listening (LPL) time for energy efficiency	Nodes sleep and wake up based on assigned data slot	Power saving mode with low power wake up radio for channel listening and normal radio for data transmission	Dropping highly correlated information packet to reduce energy use in transmission

with the physical layer to detect and correct errors. Other interactions include sharing of the transmission and channel information with the MAC layer to achieve higher performance and resource utilization.

For a WSN, minimizing energy consumption and maximizing network lifetime starts at the physical layer. At the physical layer, energy is used in operating radio circuitry and bit stream transmission. Energy used to run the radio circuitry is fixed whereas the energy spent to transmit the data can vary based on channel loss, interference, and transmission distance. There is a tradeoff between transmission power and error. Proper selection of the transmission power is needed to minimize energy loss and for the network to operate more efficiently. Modulation schemes are needed to transmit data over a wireless channel. Different modulation schemes have been developed to achieve the highest probability of successful transmission under different conditions. Energy-efficient modulation schemes should minimize both transmission and circuit energy. Recent research studies include physical-layer requirements, low power radio design, power-aware transmission schemes, and modulation schemes.

The physical layer must be designed with consideration of WSN requirements. Ref. [107] discusses the physical-layer requirements with a focus on digital communication and existing hardware technology. Digital communication with the radio must be small in size since the sensor nodes are small. The radio must also be cheap since hundreds to thousands of sensor nodes may be deployed. The re-use of radio for sensing and communication can significantly reduce cost and energy. With respect to energy, the radio must be low power. Important considerations must be made when determining whether to use existing hardware. Depending on the characteristic of the WSN, there are tradeoffs among radios in terms of energy, data rate, error, transmission distance, and reliability.

Interference, synchronization, and multi-casting are other requirements that must be considered at the physical layer. If sensor nodes are densely deployed, signal interference among the sensor nodes may be inevitable. Each sensor node can lower its transmission power to reduce interference; however, synchronization among the sensor nodes is needed. There must be synchronization between the link and physical layers and among sensor nodes. With synchronization, communication interference can be minimized. Lastly, radios with the ability to multi-cast are useful for transmitting data to multiple sensor nodes at the same time. Only the intended sensor nodes should receive the information.

7.4.1. Bandwidth choices

In WSNs, there are three classes of physical-layer technologies based on bandwidth: narrow band, spread-spectrum, and ultra-wideband. Narrow band uses radio bandwidth that is on the order of symbol rate. Narrow band focuses on bandwidth efficiency. Bandwidth efficiency is the measure of the data rate over the bandwidth. In spread-spectrum, the narrow signal is spread into a wideband signal. The spreading function used to determine the bandwidth is independent of the message. Spread-spectrum has the ability to reduce power and still communicate effectively. It is more robust to interference and multi-path channel impairment. Compared to spread-spectrum, ultra-wideband employs larger bandwidth, on the order of gigahertz, compared to the typical spread-spectrum systems. Ultra-wideband spreads its signal over the large bandwidth such that the interference to other radios is negligible. Like spread-spectrum, ultra-wideband can communicate with low power.

Ref. [66] shows that spread-spectrum technologies meet WSN requirements better than narrow-band technology. Narrow band optimizes on bandwidth efficiency while both spread spectrum and ultra-wideband tradeoff bandwidth with energy savings. Narrow-band systems are less robust to interference compared to spread-spectrum systems. Depending on the type of spread spectrum, synchronization can be good because of auto-correlation properties of the pseudo-random sequence. As for multi-cast, narrow-band systems are not designed to perform this task. Spread-spectrum systems, on the other hand, can achieve this with the appropriate pseudo-random codes. Ultra-wideband has many attractive features, but compared to spread spectrum, it has its challenges and issues. More studies are required to better understand ultra-wideband.

7.4.2. Radio architecture

Reducing energy consumption at the physical layer requires low power operations. Energy consumption at the physical layer is due to circuitry energy and transmission energy. The transmitter and receiver require energy to run their circuitry. To start a transmitter, a significant amount of time and energy is required. Energy at startup in some cases can be higher than the energy required for an actual transmission. For a transmitter that switches between the sleep to active state, a fast startup transmitter architecture is needed to minimize both energy and time.

FN frequency synthesizer with $\Sigma\Delta$ modulator: Ref. [108] proposes a transmitter architecture based on a fractional-N (FN) frequency synthesizer with $\Sigma\Delta$ modulator. The architecture achieves fast startup time and data rate by increasing the loop bandwidth. Each noise source from the synthesizer and the modulator goes through different loop characteristics. By adjusting the loop bandwidth, power consumption can be reduced.

WiseNet: Other radio architectures such as WiseNet [109] also seek to reduce power consumption with low voltage operations. WiseNet employs a dedicated duty-cycle radio and a low power MAC protocol design (WiseMAC) to lower its power consumption. To optimize the startup time and save energy, the system wakes up the different transceiver blocks in a sequence. The lower-power baseband blocks wake up before the radio frequency (RF) circuits. Startup time varies inversely with the frequency of operation.

7.4.3. Modulation schemes

The modulation scheme used by a radio can impact the energy consumption of a node. Energy-efficient modulation schemes are needed to reduce energy consumption.

Binary and M-ary modulation: In [108,110], a comparison is drawn between binary modulation and multi-level (M-ary) modulation. M-ary modulation transmits symbols from a set of M distinct waveforms while binary modulation

uses two distinct waveforms. For M-ary modulation, $\log_2 M$ bits are sent per sample. It is shown that M-ary modulation is more energy efficient than binary modulation when the startup time is short and the RF output power is small. In another comparison, for a large value of M, M-ary frequency shift keying (M-FSK) is more energy efficient compared to M-ary phase shift keying (M-PSK) and M-ary quandrature amplitude modulation (M-QAM) when M is greater than eight. For small M, M-FSK is not as energy efficient because more RF power is required to achieve the same bit-error-rate performance as M-PSK and M-QAM. However, for large M, the SNR required for M-FSK grows slowly, making it very energy efficient. Compared to M-PSK and M-QAM, the SNR grows very fast when M is large. Hence, M-FSK is a better solution for energy savings.

Modulation optimization: In [111], a detailed analysis of the tradeoff in transmission energy, circuit energy, transmission time, and constellation size for both uncoded and coded M-QAM and M-FSK is studied. For an uncoded system, optimizing transmission time and modulation parameters can increase energy savings. It is shown that up to eighty percent energy savings is achievable when the system is optimized. In terms of uncoded M-QAM and M-FSK, uncoded M-QAM is more bandwidth and energy efficient compared to uncoded M-FSK for short-range applications. Uncoded M-FSK, however, can be used in power-limited applications because it requires less transmitting power compared to M-QAM. For a coded system, coding has benefits which vary with transmission distance and the modulation scheme. For a coded M-QAM system, coding increases energy efficiency and transmission distance. A coded M-FSK system, on the other hand, can reduce energy consumption only when the distance is large.

Energy-per-useful-bit metric: Ref. [112] proposes an energy-per-useful-bit metric (EPUB) to optimize and compare different physical layers in a WSN. EPUB enables a comparison between different physical layers which have similar network scenarios, same channel model, average transmission distance, bit-error rate, and MAC scheme. EPUB is a function of synchronization cost and relative usage cost of the transmitter and receiver. In order to optimize the physical layer, EPUB must be reduced. Modulation scheme, carrier frequency, and data rate were examined to determine the tradeoffs for EPUB reduction. Results show that increasing the data rate, lowering the carrier frequency, and using simple modulation can significantly reduced EPUB.

Open research issues

The physical layer in a WSN must be energy efficient. The physical-layer design starts with the design of the radio. The design or selection of a radio is very important because the radio can impact the performance of the other protocol layers. An energy-efficient radio should consume the lowest possible energy required to properly its function and communicate. Minimizing the energy consumption at the physical layer requires that the circuitry energy and transmission energy be optimized. Circuitry energy can be minimized with the reduction of wakeup and startup times. The shorter the wakeup and startup duration is, the lower is the amount of energy consumed. Modulation schemes have been proposed to reduce the energy for transmitting each bit. Table 4 summarizes the physical-layer issues.

Future work entails new innovations in low power radio design with emerging technologies, exploring ultra-wideband techniques as an alternative for communication, creating simple modulation schemes to reduce synchronization and energy cost, determining the optimal transmission power, and building more energy-efficient protocols and algorithms.

7.5. Cross-layer interactions

The cross-layered approach in WSN is more effective and energy efficient than in traditional layered approach. While traditional layered approach endures more transfer overhead, cross-layered approach minimizes these overhead by having data shared among layers. In the cross-layered approach, the protocol stack is treated as a system and not individual layers, independent of each other. Layers share information from the system. The development of various protocols and services in a cross-layered approach is optimized and improved as a whole. Various design solutions are proposed to explore the benefits of a cross-layer approach. Below are these proposals.

Table 4
Overview of physical-layer issues

Design requirements	Solutions	Main concept
Bandwidth choices	Narrow band, Spread-spectrum, Ultra-wideband	Spread-spectrum is preferred over narrow band because of its ability to reduce power, communicate effectively, and more robustness to interference and multi-channel impairment. Ultra-wideband is an alternate solution to spread-spectrum
Radio architecture	FN frequency synthesizer with ΣΔ modulator, WiseNet	Fast startup radio architectures minimize both energy and time. FN frequency synthesizer with ΣΔ modulator achieves fast startup time and data rate by adjusting the loop bandwidth. WiseNet achieves low energy consumption by using a duty-cycled radio with a low power MAC protocol
Modulation Schemes	Binary modulation, M-FSK modulation, M-PSK modulation, M-QAM modulation	Multi-level modulation achieves more energy efficiency than binary modulation when the startup time is short. M-FSK is more efficient compared to M-PSK and M-QAM when $M > 8$

Unifying sensornet protocol (SP): Ref. [113] proposed a unified SP which provides shared neighbor management and a message pool. The protocol runs on a single link-layer technology over a broad range of devices and supports a variety of network protocols while not losing efficiency. The protocol allows network level protocols to choose their neighbors wisely based on the information available at the link layer. This abstraction layer promotes cooperation across the link and network layers to utilize the limited resource efficiently. Experiments using this protocol were carried out using two types of radio technology: B-MAC on micas and IEEE 802.15.4 on Telos. Measurements from these implemented protocols show that performance is not sacrificed with the SP abstraction. In addition, there are benefits in using a common link abstraction.

EYES MAC and source routing protocol: EYES MAC protocol [114] exploits the benefits of cross-layer interaction between the network and data-link layers. In this design, the MAC protocol monitors topology changes, node and communication failures, and power duty-cycling. The MAC protocol shares this information with the EYES Source Routing (ESR) protocol. ESR utilizes this information to assist in route setup and maintenance. In a dynamic network, EYES MAC protocol can save energy by efficiently re-establishing routes and minimizing flooding to the network.

The MAC protocol in this design has three modes of operation: active, passive, and dormant. In the active mode, the protocol forwards messages to the destination and accepts messages from the passive mode. In the passive mode, the protocol keeps track of active nodes that forward their data and inform them of network-wide messages. In the dormant mode, the node is in low power mode to save energy and does not transmit messages. The MAC protocol is TDMA based where each node is assigned one time slot and has control over it. A time slot is divided into three sections: communication request (CR), traffic control (TC), and the data section. The CR section allows other nodes to make request to control the current time slot. The TC section allows the owner of the time slot to transmit a TC message. A TC message contains synchronization and control information. In addition, it indicates when the communication in the data section will take place. The data section is when data is transmitted.

Nodes gain knowledge of the local topology information from received neighboring TC messages. This information is collected and shared with the ESR protocol. ESR is a dynamic, self-starting, multi-hop routing protocol. It contains three phases: route setup, route maintenance, and route re-establishment. In route setup, the source floods the network with route request to the destination. The destination receives this message and replies back on the fastest route to the source. Routing decision is made locally to reduce routing overhead. In route maintenance, the MAC protocol detects when connections break and notifies ESR to perform route recovery. This detection and recovery strategy effectively reduces flooding overhead and shortens route re-establishment time.

EYES MAC protocol was compared against sensor-MAC (SMAC) [115] and dynamic source routing (DSR) [116] in terms of energy and network lifetime. EYES MAC protocol outperforms SMAC and DSR when nodes are mobile. In a mobile network, there may be frequent route updates due to route breakage. EYES MAC protocol minimizes on overhead in routing and route re-establishment by utilizing the information from the MAC protocol. SMAC and DSR perform better when nodes are static and when routes are established only once.

Joint routing, MAC, and Link optimization approach: Ref. [117] is a cross-layer design between the link, MAC, and routing layer to minimize overall energy consumption across all nodes. In this design, nodes can be in three modes: active, sleep, and transient. Using a variable-length TDMA scheme, nodes become active in their assigned time slot. In active mode, nodes transmit their data and go back to sleep mode to save energy. When the node wakes up from sleep, it enters the transient mode before switching to the active mode. To maximize a node's lifetime, link adaptation is introduced where each node in the network adapts its transmission rate. Along with link adaptation, optimal routing and scheduling is used to compute total energy consumption. The problem of minimizing the overall energy consumption in the network is formulated into a linear programming (LP) problem. Relaxation methods are used to refine the results. Results show that multi-hop routing with link adaptation and scheduling is more energy efficient than single-hop routing without link adaptation. Link adaptation reduces the transmission time in relaying nodes thereby reducing energy consumption.

Unified cross-layer protocol: The unified cross-layer protocol [118] combines the functionalities of the transport, network and medium access protocols into a single module. The unified cross-layer module (XLM) achieves energy efficiency and reliable event communication. XLM is built upon the concept of letting a node decide whether it wants to participate in communication or not. A node makes its decision based an initiative determination procedure. The initiative determination procedure is a binary operation based on a set of four conditions. All four conditions must be satisfied for a node to participate. The first condition is to ensure that the link is reliable for communication. The received SNR of the packet must be above some specified threshold for communication. The second and third conditions ensure that there will be no congestion at the node. The node must be able to relay the data where the incoming traffic rate must not be greater than the outgoing traffic rate. In addition, the node must not experience any buffer overflow. The last condition ensures that the node has enough energy to participate. The remaining energy of a node must be above the minimum energy threshold. If all four conditions are satisfied, the node will participate in communication. Using this XLM concept, nodes will operate in duty-cycle fashion to save energy. After the initiative determination procedure, the node forwards the packet by performing the receiver contention operation. The receiver contention operation is based on the receiver-based routing [119,120] which uses the routing level of each node to decide its priority in sending the packet. Simulation results show that XLM outperforms the tradition layered protocol in network performance and communication efficiency.

Jointly-optimal congestion-control and power-control (JOCP) algorithm: Refs. [121,122] propose a jointly-optimal

Table 5
Overview of a representative set of cross-layer protocols

Cross-layer designs	Layer interaction	Main concept
Unifying sensornet protocol (SP)	Network and MAC layer	SP provides a unified interface to a range of data-link and physical-layer technologies while supporting a variety of network protocols. With SP, multiple network protocols can coexist and work efficiently in the same network.
EYES MAC and routing protocol	Network and MAC layer	EYES MAC protocol shares topological information with the source routing protocol to assist in route setup and maintenance. Route re-establishment can be accomplished with energy efficient saving.
Routing, MAC, and Link Optimization	Network, MAC, and link layer	Routing, MAC, and link optimization focuses on minimizing the network's overall energy consumption. An energy-efficient joint routing schedule along with a link adaptation scheme can maximize the network lifetime. In addition, a variable-length TDMA scheme can minimize the energy consumption across the network.
Unified Cross-layer design	Transport, Network, and MAC layer	A unified cross-layer model (XLM) is used to achieve efficient and reliable event communication with minimum energy expenditure. XLM combines the functionality of the transport, network, and MAC all into one. XLM lets the node decide when to participate in communication. XLM handles contention, local congestion control, and duty-cycling to achieve reliability and efficient communication.
Joint physical, MAC, and routing layer	Physical, MAC, and routing layer	A computational algorithm for cross-layer optimization computes the solutions to increase network lifetime and bandwidth efficiency. The optimizing problem computes the optimal transmission power, data rates, and link schedule.
JOCP	Transport and physical layer	JOCP increase the end-to-end throughput and energy efficiency by optimally joining end-to-end congestion control with per-link power control.

congestion-control and power-control (JOCP) algorithm which aims to increase throughput and minimize energy consumption. JOCP optimizes end-to-end congestion control and per-link-basis power control. JOCP algorithm is composed of four simultaneous updates run at each time slot until convergence. The first update involves each intermediate node updating its weighted queuing delay. The second update involves the source node measuring the total delay and updating its TCP window size and source rate. The third update involves each transmitting node calculating the received messages and using a flooding protocol to pass it to other nodes. The fourth update involves each transmitting node adjusting its power based on the messages received and its queuing delays. Results show that a cross-layer interaction between the physical and transport layer can enhance performance.

Joint physical, MAC, and routing layer: Ref. [123] presents a computational algorithm for cross-layer optimization. The problem of computing the transmission power, data rates, and link schedule is formulated into an optimization problem. The goal is to maximize the network lifetime. The optimization problem has the following constraints: (1) flow conservation, (2) rate constraints, (3) energy conservation, and (4) range constraints. The first constraint ensures that the flow in the network must be balanced on each time slot. The second constraint bounds the maximum data rate of each link. The third constraint ensures that the energy consumed over time is less than or equal to the initial energy. The last constraint ensures that the transmission power of a node must be less than or equal to the maximum transmission power. Using these constraints, the computed solution gives the optimal transmission power and rate over each link. The link schedule is solved using a mixed integer convex optimization program. Results show that using a joint optimal design can increase the lifetime of the network.

Open research issues

Cross-layer designs improve performance and optimize interaction between layers. Cross-layer design considers the sharing of information across layers. For instance, a MAC protocol shares topology information with the network protocol to assist in route setup and maintenance. Such information can be shared directly between the two protocols. Proposed cross-layer designs have focused on the physical, data-link, network, and transport layers. Table 5 provides a summary of the cross-layer protocols. Future research in cross-layer design can focus on collaboration between all the layers to achieve higher energy saving, network performance, and extend network lifetime.

8. Conclusion

Unlike other networks, WSNs are designed for specific applications. Applications include, but are not limited to, environmental monitoring, industrial machine monitoring, surveillance systems, and military target tracking (see Fig. 2). Each application differs in features and requirements. To support this diversity of applications, the development of new communication protocols, algorithms, designs, and services are needed.

We have surveyed in this paper issues on three different categories: (1) internal platform and underlying operating system, (2) communication protocol stack, and (3) network services, provisioning, and deployment issues. We have summarized and compared different proposed designs, algorithms, protocols, and services. Moreover, we have highlighted possible improvements and research in each area. There are still many issues to be resolved around WSN applications such as communication architectures, security, and management. By solving these issues, we can close the gap between technology and application.

Acknowledgement

We gratefully acknowledge the helpful comments from the reviewers, which have improved the paper very significantly.

Appendix A

The following table shows a sample of the companies that provide wireless sensor technology solutions. There are other companies offering similar solutions, however the information below is representative of that available in the public domain at the time of this writing

Company	Sensor technologies	Technology description	Sensor application
Crossbow	Motes: Mica2, Mica2Dot, MicaZ Gateway nodes: Stargate, MIB600 Interface board: MIB600 Ethernet, MIB510 Serial, MIB520 USB	Mica2 and Mica2Dot are both based on the Atmel ATmega128L microcontroller boards with 4 kB of EEPROM and 128 kB of Flash. MicaZ provides new capabilities to the Mica family with 250 kbps high data rate radio. MICAz supports both IEEE802.15.4 and ZigBee. Connector for external peripherals such as light, pressure, barometric, temperature, acoustic, acceleration/seismic, magnetic, and other sensor board. Stargate uses the Intel PXA255 processor with 32 MB Flash and 64 MB SDRAM. Stargate is compatible with Mica family motes	Seismic structural monitoring, indoor/outdoor environmental monitoring, security protection and surveillance monitoring, inventory monitoring, health monitoring
Moteiv	Motes: T-mote sky Gateway nodes: T-mote Connect	Low power operation with MSP430 F1611 microcontroller containing 10 kB of RAM and 48 kB of Flash. T-mote enables drop in placement and USB connection to the host computer. Integrated on board are humidity, temperature and light sensors. Radio range is 50 m indoor and 125 m outdoor. Gateway nodes provide a bridge to Ethernet. Each gateway node supports for up to two T-mote wireless modules	Indoor and outdoor monitoring applications
Dust networks	SmartMesh-XT motes: M1030, M2030, M2135 SmartMesh-XT manager: PM1230, PM2030, PM2130	SmartMesh-XT motes are reliable and low power nodes. Battery life in these motes can last 5–10 years. These motes have nine digital I/O, serial ports and seven analog output ports SmartMesh-XT manager manages and provides QoS functions to the motes. A manager node can manage up to 250 motes. Radio range for these devices indoor is 10–30 m and >100 m outdoor	Building automation monitoring, industrial process monitoring, and security and defense monitoring

Millennial net	MeshScape 916 MHz and 2.4 MHz: mesh node, mesh gate, end nodes Mesh485: mesh sub-based router, mesh router, and mesh bridges	MeshScape nodes are low power nodes that have sleep and wakeup modes. They route data and provide a backup route in case of congestion. They are configurable through mesh gates. MeshScape gate acts as a portal to monitor network performance as well as providing network configurations. End nodes are integrated with sensors and actuators to capture data. Mesh485 sub-base routers are attached to sensors to collect data and send to the bridge. Mesh485 bridge provides an interface between the end system controller and the sub-base router. Mesh485 routers are used to increase distance between nodes and the bridge	Building monitoring and industrial process monitoring
Sensicast	Sensicast EMS and RTD nodes Sensicast gateway bridge, mesh router	Sensicast EMS node provides real-time temperature and humidity readings. Battery life up to 3 years where users adjust power duty-cycle. Sensicast RTD nodes are wireless temperature devices. Battery life is about 1.5–2 years. Sensicast gateway bridge nodes manage and monitor the network. Sensicast Gateway bridges communicate through RS232, USB or Ethernet. Sensicast mesh routers are repeaters in the network to ensure reliability in the network	Industrial monitoring of temperature and energy

Appendix B

The following table shows a comparison between the different types of wireless sensor networks:

	Terrestrial WSN	Underground WSN	Underwater WSN	Multi-media WSN	Mobile WSN
Definition	A network consists of hundreds to thousands of sensor nodes deployed on land	A network consists of wireless sensor nodes deployed in caves or mines or underground	A network consists of wireless sensor and vehicles deployed into the ocean environment	A network consists of wireless sensor devices that have the ability to store, process, and retrieve multi-media data such as video, audio, and images	A network consists of mobile sensor nodes that have the ability to move
Challenges	– In-network data aggregation to improve performance across communication, energy cost, and delay – Minimizing energy cost – Reduce the amount of data communication – Finding the optimal route – Distributing energy consumption – Maintaining network connectivity – Eliminating redundancy	– Expensive deployment maintenance, and equipment cost – Threats to device such as the environment and animals – Battery power cannot easily be replaced – Topology challenges with pre-planned deployment – High levels of attenuation and signal loss in communication	– Expensive underwater sensors – Hardware failure due to environment effects (e.g., corrosion) – Battery power cannot easily be replaced – Sparse deployment – Limited bandwidth – Long propagation delay, high latency, and fading problems	– In-network processing, filtering, and compressing of multi-media content – High energy consumption and bandwidth demand – Deployment based on multi-media equipment coverage – Flexible architecture to support different applications – Must integrate various wireless technologies – QoS provisioning is very difficult due to link capacity and delays – Effective cross-layer design	– Navigating and controlling mobile nodes – Must self-organized – Localization with mobility – Minimize energy cost – Maintaining network connectivity – In-network data processing – Data distribution – Mobility management – Minimize energy usage in locomotion – Maintain adequate sensing coverage
Applications	– Environmental sensing and monitoring – Industrial monitoring – Surface explorations	– Agriculture monitoring – Landscape management – Underground structural monitoring – Underground environment monitoring of soil, water or mineral – Military border monitoring	– Pollution monitoring – Undersea surveillance and exploration – Disaster prevention monitoring – Seismic monitoring – Equipment monitoring – Underwater robotics	– Enhancement to existing WSN applications such as tracking and monitoring	– Environmental monitoring – Habitat monitoring – Military surveillance – Target tracking – Underwater monitoring – Search and rescue

Appendix C

The following table shows a comparison between the applications and the communication protocols

Application	Application characteristic	Transport layer	Network layer	Data-link layer	Physical layer
PinPtr [2]	– can tolerate multiple sensor failures – provides good coverage and high accuracy – services use include synchronization, data aggregation, and localization – can tolerate small latency		– multi-hop routing	–	– mica mote with a multi-channel transceiver is used – acoustic signal detection of events
Macroscope in the redwood [22]	– robust system – supports data aggregation – synchronization – position of the sensor can largely affect the result of the data – lacks the ability to detect failures in the network	–	– multi-hop robust routing (MinRoute)	–	– duty-cycling for power conservation
Semiconductor plants and oil tanker [23]	– Self configuration – Security – Maintainability and log network lifetime – Must achieve adequate coverage – Eliminate interference – Must be energy efficient to meet battery lifetime – Centralized protocol used for power management – Must be fault tolerant – Higher data rate using Intel motes compare to Mica2 motes	–	– Single-destination-DSDV routing	– End-to-end reliable bulk transfer protocol	– Assessing RF coverage and identifying interferences – Duty-cycling for power conservation
Underwater monitoring [24]	– Localization of an autonomous vehicle under water is very difficult – Reliability of hardware, software, and data transfer is very important	–	– Direct connection, single hop	– CRC checksum for error detection – Packet-by-packet delivery or groups of packet delivery depending on the communication link	– Ultrasonic and optical communication are used. – Advantage of acoustic communication is long-range communication, can be used for localization – Disadvantage of acoustic communication is the limited bandwidth, long propagation delay, and signal fading issue – Advantage of optimal communication has high data rate due to high frequency signal but cost less than ultrasonic – Disadvantage of optical communication is short-range communication and line-of-sight operations – Pulse position modulation

(*continued on next page*)

Appendix C (*continued*)

Application	Application characteristic	Transport layer	Network layer	Data-link layer	Physical layer
MAX [25]	– Human-centric operation – System has to be robust to reconfiguration – Must be secured from unauthorized access – Minimum delay – Efficiency in energy, band-width, and memory – Hierarchical architecture – Tradeoff between computation, processing and storage	–	– Two-hop connection from the tags to the sub-station and from the sub-station to the base station	– Polling protocol is used between the base station and sub-stations to minimize delay and collision	– Radio frequency with received signal strength indicator (RSSI) is used. (Chipcon's CC1000 radio transceiver)
CenWits [26]	– Provides adaptive tradeoff between memory and power consumption – Issue: limited memory available sensor motes – Size of stored information can be very large – Information must be organized and processed efficiently – Power management with group set communication	–	– Direct connection to another sensor or access point	–	– MICA2 sensor nodes equipped with a GPS receiver and RF transmitter – GPS has its limitation when used in the canyons and rainy forest Cenwits addresses this by incorporating location point – Transmission beaconing adapts the users speed for power management
Cyclops [27]	– High speed data transfer and processing – High resolution images – Power consumption is minimal – Delay sensitive – Several power states to minimize energy		– Direct connection		
Petroleum facility [28]	– Reliable data rate – Minimum latency – Accurate end-to-end signal – Minimize energy	–	– Multi-hop routing	–	–
Volcano monitoring [32]	– Sparse deployment – Linear configuration – High data rates – High battery consumption – Must have reliable data transmission – Time must be synchronized – Java-based GUI for network monitoring	–	– Multi-hop routing (MultiHopLQI)	–	–
Health care monitoring [33]	– infant monitoring – alert the deaf – blood pressure monitoring and tracking – vital sign monitoring	–	– Direct connection	–	– T-mote and SHIMMER motes equipped with pressure and temperature sensors, and microphone
ZebraNet [9]	– Mobile sensor nodes must accurately log positions – Communication latency is not important – Sparse system – Flash memory to store data – High power consumption – Conserve power by duty-cycling node and GPS – Use rechargeable battery with a solar array	–	– Flooding	–	– Duty-cycle to conserve energy

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, IEEE Communications Magazine 40 (8) (2002) 104–112.

[2] G. Simon, M. Maroti, A. Ledeczi, G. Balogh, B. Kusy, A. Nadas, G. Pap, J. Sallai, K. Frampton, Sensor network-based countersniper system, in: Proceedings of the Second International Conference on Embedded Networked Sensor Systems (Sensys), Baltimore, MD, 2004.

[3] J. Yick, B. Mukherjee, D. Ghosal, Analysis of a Prediction-based Mobility Adaptive Tracking Algorithm, in: Proceedings of the IEEE Second International Conference on Broadband Networks (BROADNETS), Boston, 2005.

[4] M. Castillo-Effen, D.H. Quintela, R. Jordan, W. Westhoff, W. Moreno, Wireless sensor networks for flash-flood alerting, in: Proceedings of the Fifth IEEE International Caracas Conference on Devices, Circuits, and Systems, Dominican Republic, 2004.

[5] T. Gao, D. Greenspan, M. Welsh, R.R. Juang, A. Alm, Vital signs monitoring and patient tracking over a wireless network, in: Proceedings of the 27th IEEE EMBS Annual International Conference, 2005.

[6] K. Lorincz, D. Malan, T.R.F. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, S. Moulton, Sensor networks for emergency response: challenges and opportunities, Pervasive Computing for First Response (Special Issue), IEEE Pervasive Computing, October–December 2004.

[7] G. Wener-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, M. Walsh, Deploying a wireless sensor network on an active volcano, Data-Driven Applications in Sensor Networks (Special Issue), IEEE Internet Computing, March/April 2006.

[8] V. Raghunathan, A. Kansai, J. Hse, J. Friedman, M. Srivastava, Design considerations for solar energy harvesting wireless embedded systems, in: Proceedings of the IPSN, 2005, 457–462.

[9] P. Zhang, C.M. Sadler, S.A. Lyon, M. Martonosi, Hardware design experiences in ZebraNet, in: Proceedings of the SenSys'04, Baltimore, MD, 2004.

[10] S. Roundy, J.M. Rabaey, P.K. Wright, Energy Scavenging for Wireless Sensor Networks, Springer-Verlag, New York, LLC, 2004.

[11] M. Rahimi, H. Shah, G.S. Sukhatme, J. Heideman, D. Estrin, Studying the feasibility of energy harvesting in mobile sensor network, in: Proceedings of the IEEE ICRA, 2003, pp. 19–24.

[12] A. Kansai, M.B. Srivastava, An environmental energy harvesting framework for sensor networks, in: Proceedings of the International Symposium on Low Power Electronics and Design, 2003, pp. 481–486.

[13] S. Toumpis, T. Tassiulas, Optimal deployment of large wireless sensor networks, IEEE Transactions on Information Theory 52 (2006) 2935–2953.

[14] J. Yick, G. Pasternack, B. Mukherjee, D. Ghosal, Placement of network services in sensor networks, Self-Organization Routing and Information, Integration in Wireless Sensor Networks (Special Issue) in International Journal of Wireless and Mobile Computing (IJWMC) 1 (2006) 101–112.

[15] D. Pompili, T. Melodia, I.F. Akyildiz, Deployment analysis in underwater acoustic wireless sensor networks, in: WUWNet, Los Angeles, CA, 2006.

[16] I.F. Akyildiz, E.P. Stuntebeck, Wireless underground sensor networks: research challenges, Ad-Hoc Networks 4 (2006) 669–686.

[17] M. Li, Y. Liu, Underground structure monitoring with wireless sensor networks, in: Proceedings of the IPSN, Cambridge, MA, 2007.

[18] I.F. Akyildiz, D. Pompili, T. Melodia, Challenges for efficient communication in underwater acoustic sensor networks, ACM Sigbed Review 1 (2) (2004) 3–8.

[19] J. Heidemann, Y. Li, A. Syed, J. Wills, W. Ye, Underwater sensor networking: research challenges and potential applications, in: Proceedings of the Technical Report ISI-TR-2005-603, USC/Information Sciences Institute, 2005.

[20] I.F. Akyildiz, T. Melodia, K.R. Chowdhury, A survey on wireless multimedia sensor networks, Computer Networks Elsevier 51 (2007) 921–960.

[21] D. Gay, P. Levis, R.v. Behren, The nesC language: a holistic approach to networked embedded systems, in: Proceedings of the PLDI, San Diego, CA, 2003.

[22] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, W. Hong, A macroscope in the redwoods, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[23] L. Krishnamurthy, R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, N. Kushalmager, L. Nachman, M. Yarvis, Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the North Sea, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[24] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, P. Corke, Data collection, storage, retrieval with an underwater sensor network, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[25] K.K. Yap, V. Srinivasan, M. Motani, MAX: Human-centric search of the physical world, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[26] J.H. Huang, S. Amjad, S. Mishra, CenWits: A sensor-based loosely coupled search and rescue system using witnesses, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[27] M. Rahimi, R. Baer, O.I. Iroezi, J.C. Garcia, J. Warrior, D. Estrin, M. Srivastava, Cyclops: in situ image sensing and interpretation in wireless sensor networks, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[28] I. Johnstone, J. Nicholson, B. Shehazad, J. Slipp, Experiences from a wireless sensor network deployment in a petroleum environment, in: IWCMC, Honolulu, Hawaii, 2007.

[29] Moteiv, <http://moteiv.com>.

[30] SOHOware Inc., <http://www.sohoware.com>.

[31] Technologic Systems, <http://www.embeddedarm.com>.

[32] G. Werner-Allen, K. Lorincz, M. Welsh, O. Marcillo, J. Johnson, M. Ruiz, J. Lees, Deploying a wireless sensor network on an active volcano, IEEE Internet Computing 10 (2006) 18–25.

[33] C.R. Baker, K. Armijo, S. Belka, M. Benhabib, V. Bhargava, N. Burkhart, A.D. Minassians, G. Dervisoglu, L. Gutnik, M.B. Haick, C. Ho, M. Koplow, J. Mangold, S. Robinson, M. Rosa, M. Schwartz, C. Sims, H. Stoffregen, A. Waterbury, E.S. Leland, T. Pering, P.K. Wright, Wireless sensor networks for home health care, in: AINAW, Ontario, Canada, 2007.

[34] M. Leopold, M.B. Dydensborg, P. Bonnet, Bluetooth and sensor networks: a reality check, in: Proceedings of the Sensys'03, Los Angeles, CA, 2003.

[35] L. Gu, D. Jia, P. Vicaire, T. Yan, L. Luo, A. Tirumala, Q. Cao, T. he, J.A. Stankovic, T. Abdelzaher, B.H. Krogh, Lightweight detection and classification for wireless sensor networks in realistic environment, in: Proceedings of the Sensys'05, Los Angeles, CA, 2005.

[36] S.Y. Cheung, S.C. Ergen, P. Varaiya, Traffic surveillance with wireless magnetic sensors, in: Proceedings of the 12th ITS World Congress, San Francisco, CA, 2005.

[37] I. Howitt, J.A. Gutierrez, IEEE802.15.4 low rate-wireless personal area network coexistence issues, Wireless Communications and Networking 3 (2003) 1481–1486.

[38] ZigBee: wireless control that simply works, <http://www.zigbee.org>.

[39] ZigBee Standards Overview, <http://www.freescale.com/webapp/sps/site/overview.jsp?nodeId=01J4Fs25657725>.

[40] HART – The Logical Wireless Solution, <http://www.hartcomm2.org/hart_protocol/wireless_hart/hart_the_logical_solution.html>.

[41] Draft standard: What's in the April'07 WirelessHART specification, <http://www.controleng.com/article/CA6427951.html>.

[42] ISA100.11a, <http://www.isa.org/MSTemplate.cfm?MicrositeID=1134&CommitteeID=6891>.

[43] 6LoWPAN, <http://6lowpan.net/>.

[44] G. Mulligan, L.W. Group, The 6LoWPAN architecture, in: Proceedings of the EmNets, Cork, Ireland, 2007.

[45] G. Montenegro, N. Kushalnagar, J. Hui, D. Culler, Transmission of IPv6 packets over IEEE 802.15.4 networks, RFC 4944 (2007).

[46] IEEE Standard 802.15.3, Wireless medium access control (MAC) and physical layer (PHY) specifications for high rate wireless person area networks (WPANs), September 2003.

[47] Wibree, <http://www.wibree.com/>.

[48] J. Newsome, D. Song, GEM: Graph EMbedding for routing and data-centric storage in sensor networks without geographic information, in: Proceedings of the Sensys'03, San Diego, CA, 2003.

[49] P. Desnoyers, D. Ganesan, P. Shenoy, TSAR: a two tier sensor storage architecture using interval skip graphs, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[50] D. Ganesan, B. Greenstein, D. Perelyubskiy, D. Estrin, J. Heidemann, An evaluation of multi-resolution storage for sensor networks, in: Proceedings of the Sensys'03, Los Angeles, CA, 2003.

[51] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, M. Singh, Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols, in: Proceedings of the IEEE Wireless Communications and Networking Conference, 2005.
[52] G. Werner-Allen, P. Swieskowski, M. Welsh, MoteLab: a wireless sensor network testbed, in: ISPN, 2005.
[53] Crossbow Technology, <http://www.xbow.com>.
[54] D. Johnson, T. Stack, R. Fish, D.M. Flickinger, L. Stoller, R. Ricci, J. Lepreau, Mobile Emulab: a robotic wireless and sensor network testbed, in: IEEE INFOCOM, 2006.
[55] J. Zhao, R. Govindan, Understanding packet delivery performance in dense wireless sensor networks, in: Proceedings of the First International Conference on Embedded Networked Sensor Systems (Sensys), Los Angeles, CA, 2003.
[56] D. Moore, J. Leonard, D. Rus, S. Teller, Robust distributed network localization with noisy range measurements, in: Proceedings of the Sensys'04, San Diego, CA, 2004.
[57] M. Maroti, B. Kusy, G. Balogh, P. Volgyesi, A. Nadas, K. Molnar, S. Dora, A. Ledeczi, Radio interferometric geolocation, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.
[58] R. Stoleru, T. He, J.A. Stankovic, D. Luebke, A high-accuracy, low-cost localization system for wireless sensor networks, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.
[59] A. Srinivasan, J. Wu, A survey on secure localization in wireless sensor networks, in: B. Furht (Ed.), Wireless and Mobile Communications, CRC Press/Taylor and Francis Group, Boca Raton/London, 2007.
[60] L. Lazos, R. Poovendran, SeRLoc: secure range independent localization for wireless sensor networks, in: First IEEE International Conference on Mobile Ad hoc and Sensor Systems, Fort Lauderdale, FL, 2004.
[61] D. Liu, P. Ning, W. Du, Detecting malicious beacon nodes for secure location discovery in wireless sensor networks, in: Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDC'05), 2005.
[62] A. Srinivasan, J. Teitelbaum, J. Wu, DRBTS: Distributed reputation-based beacon trust system, in: Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06), 2006, pp. 277–283.
[63] S. Capkun, J.-P. Hubaux, Secure positioning of wireless devices with application to sensor networks, in: Proceedings of the IEEE INFOCOM'05, 2005.
[64] L. Lazos, R. Poovendran, S. Capkun, ROPE: robust position estimation in wireless sensor networks, in: Proceedings of the IPSN'05, 2005.
[65] N.B. Priyantha, H. Balakrishnan, E.D. Demaine, S. Teller, Mobile-assisted localization in wireless sensor networks, in: Proceedings of the IEEE INFOCOM, Miami, FL, 2005.
[66] S. Ganeriwal, D. Ganesan, H. Shim, V. Tsiatsis, B. Srivastava, Estimating clock uncertainty for efficient duty-cycling in sensor networks, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.
[67] D. Lucarelli, I.-J. Wang, Decentralized synchronization protocols with nearest neighbor communication, in: Proceedings of the Sensys'04, Baltimore, MD, 2004.
[68] G. Wener-Allen, G. Tewari, A. Patel, M. Welsh, R. Nagpal, Firefly-inspired sensor network synchronicity with realistic radio effects, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.
[69] M. Maroti, B. Kusy, G. Simon, A. Ledeczi, Synchronization of pulse-coupled biological oscillators, SIAM 50 (1990) 1645–1662.
[70] S. Ganeriwal, R. Kumar, M.B. Srivastava, Timing-sync protocol for sensor networks, in: Proceedings of the Sensys'03, Los Angeles, CA, 2003.
[71] J. Elson, L. Girod, D. Estrin, Fine-grained network time synchronization using reference broadcasts, in: Proceedings of the 5th symposium on Operation System Design and Implementation (OSDI 2002), 2002.
[72] C.H. Rentel, T. Kunz, A clock-sampling mutual network synchronization algorithm for wireless ad hoc networks, in: Proceedings of the IEEE Wireless Communications and Networking Conference, 2005.
[73] H. Dai, R. Han, TSync: a lightweight bidirectional time synchronization service for wireless sensor networks, ACM SIGMOBILE, Mobile Computing and Communications Review 8 (January) (2004) 125–139.
[74] Q. Li, D. Rus, Global clock synchronization in sensor networks, in: Proceedings of the INFOCOM, 2004, pp. 564–574.
[75] B. Sundararaman, U. Buy, A.D. Kshemkalyani, Clock synchronization for wireless sensor network: a survey, Ad-Hoc Networks 3 (May) (2005) 281–323.
[76] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, C. Gill, Integrated coverage and connectivity configuration in wireless sensor networks, in: Proceedings of the First International Conference on Embedded Networked Sensor Systems (Sensys), Los Angeles, CA, 2003.
[77] B. Chen, K. Jamieson, H. Balakrishnan, R. Morris, Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks, in: Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom), Rome, Italy, 2001.
[78] G. Veltri, Q. Huang, G. Qu, M. Potkonjak, Minimal and maximal exposure path algorithms for wireless embedded sensor networks, in: Proceedings of the First International Conference on Embedded Networked Sensor Systems (Sensys), Los Angeles, CA, 2003.
[79] T. Yan, T. He, J.A. Stankovic, Differentiated surveillance for sensor networks, in: Proceedings of the First International Conference on Embedded Networked Sensor Systems (Sensys), Los Angeles, CA, 2003.
[80] S. Nath, P.B. Gibbons, S. Seshan, Z.R. Anderson, Synopsis diffusion for robust aggregation in sensor networks, in: Proceedings of the Sensys'04, Baltimore, MD, 2004.
[81] N. Shrivastava, C. Buragohain, D. Agrawal, S. Suri, Medians and beyond: new aggregation techniques for sensor networks, in: Proceedings of the Sensys'04, Baltimore, MD, 2004.
[82] A. Wacker, M. Knoll, T. Heiber, K. Rothermel, A new approach for establishing pairwise keys for securing wireless sensor networks, in: Proceedings of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.
[83] F. Liu, M.J.M. Rivera, X. Cheng, Location-aware key establishment in wireless sensor networks, in: Proceedings of the International Wireless Communications and Mobile Computing Conference (IWCMC), Vancouver, Canada, 2006.
[84] M.O. Rabin, Digitalized signatures and public-key functions as interactable as factorization, Technical Report MIT/LCS/TR-212, Laboratory for Computer Science, MIT, 1979.
[85] C. Karlof, N. Sastry, D. Wagner, TinySec: a link layer security architecture for wireless sensor networks, in: Proceedings of the Sensys'04, Baltimore, MD, 2004.
[86] Y.G. Iyer, S. Gandham, S. Venkatesan, STCP: a generic transport layer protocol for wireless sensor networks, in: Proceedings of the 14th IEEE International Conference on Computer Communications and Networks, San Diego, CA, 2005.
[87] Y. Zhou, M.R. Lyu, PORT: a price-oriented reliable transport protocol for wireless sensor network, in: Proceedings of the 16th IEEE International Symposium on Software Reliability Engineering (ISSRE), Chicago, IL, 2005.
[88] S.-J. Park, R. Vedantham, R. Sivakumar, I.F. Akyildiz, A scalable approach for reliable downstream data delivery in wireless sensor networks, in: Proceedings of the ACM MobiHoc'04, Roppongi, Japan, 2004.
[89] V.C. Gungor, O.B. Akan, DST: Delay sensitive transport in wireless sensor networks, in: Proceedings of the Seventh IEEE International Symposium on Computer Networks (ISCN), 2006, pp. 116–122.
[90] C.-Y. Wan, A.T. Campbell, L. Krishnamurthy, PSFQ: a reliable transport protocol for wireless sensor, in: Proceedings of the 1st ACM International workshop on Wireless Sensor Networks and Applications, 2002, pp. 1–11.
[91] Y. Sankarasubramaniam, O.B. Akan, I.F. Akyilidiz, ESRT: event-to-sink reliable transport in wireless sensor networks, in: Proceedings of the MobiHoc, Annapolis, MD, 2003.
[92] C.-Y. Wan, S.B. Eisenman, A.T. Campbell, CODA: Congestion detection and avoidance in sensor networks, in: Proceedings of the Sensys, 2003.
[93] K. Seada, M. Zuniga, A. Helmy, B. Krishnamachari, Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks, in: Proceedings of the Sensys'04, Baltimore, MD, 2004.
[94] R. Zhang, H. Zhao, M.A. Labrador, The anchor location service (ALS) protocol for large-scale wireless sensor networks, in: Proceedings of the First International on Integrated Internet Ad hoc and Sensor Networks, Nice, France, 2006.
[95] J. Yin, S. Madria, SecRout: a secure routing protocol for sensor networks, in: Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06), Vienna, Austria, 2006.

[96] X. Du, Y. Xiao, H.-H. Chen, Q. Wu, Secure cell relay routing protocol for sensor networks, Wireless Communications and Mobile Computing 6 (2006) 375–391.
[97] R.K. Ganti, P. Jayachandran, H. Luo, T.F. Abdelzaher, Datalink streaming in wireless sensor networks, in: Proceedings of the Sensys'06, Boulder, CO, 2006.
[98] R.E. Blahut, Theory and Practice of Error Control Coding, Addison-Wesley, New York, 1983.
[99] H. Liu, H. Ma, M.E. Zarki, S. Gupta, Error control schemes for networks: an overview, ACM Mobile Networking and Applications (MONET) 2 (1997) 167–182.
[100] H. Dubois-Ferriere, D. Estrin, M. Vetterli, Packet combining in sensor networks, in: Proceedings of the Sensys'05, San Diego, CA, 2005.
[101] A. Miu, H. Balakrishnan, C.E. Koksal, Improving loss resilience with multi-radio diversity in wireless networks, in: Proceedings of the ACM MobiCom, Cologne, Germany, 2005.
[102] V. Rajendran, K. Obraczka, J.J. Garcia-Luna-Aceves, Energy-efficient, collision-free medium access control for wireless sensor networks, in: Proceedings of the First International Conference on Embedded Networked Sensor Systems (Sensys), Los Angeles, CA, 2003.
[103] J. Polastre, J. Hill, D. Culler, Versatile low power media access for wireless sensor networks, in: Proceedings of the Sensys'04, San Diego, CA, 2004.
[104] S. Mishra, A. Nasipuri, An adaptive low power reservation based MAC protocol for wireless sensor networks, in: Proceedings of the IEEE International Conference on Performance Computing and Communications, 2004, pp. 316–329.
[105] C. Guo, L.C. Zhong, J.M. Rabaey, Low power distributed MAC for ad hoc sensor radio networks, in: Proceedings of the IEEE Globecom, 2001, pp. 2944–2948.
[106] M.C. Vuran, I.F. Akyildiz, Spatial correlation-based collaborative medium access control in wireless sensor networks, IEEE/ACM Transactions on Networking 14 (2006) 316–329.
[107] K.D. Wong, Physical layer consideration for wireless sensor networks, in: Proceedings of the IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan, 2004.
[108] A.Y. Wang, S. Cho, C.G. Sodini, A.P. Chandrakasan, Energy efficient modulation and MAC for asymmetric RF microsensor systems, in: Proceedings of the ISLPED'01, Huntington Beach, CA, 2001.
[109] C.C. Enz, A. El-Hoiydi, J.-D. Decotignia, V. Peiris, WiseNET: an ultralow-power wireless sensor network solution, IEEE Computer Society 37 (2004) 62–70.
[110] E. Shih, S.-H. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, A. Chandrakasan, Physical layer driven protocol and algorithm design for energy efficient wireless sensor networks, in: ACM SIGMOBILE, Rome, Italy, 2001.
[111] S. Cui, A.J. Goldsmith, A. Bahai, Energy-constrained modulation optimization, IEEE Transactions on Wireless Communication 4 (2005) 2349–2360.
[112] J. Ammer, J. Rabaey, The energy-per-useful-bit metric for evaluating and optimizing sensor network physical layers, in: Proceedings of the IWWAN'06, 2006.
[113] J. Polastre, J. Hui, P. Levi, J. Zhao, D. Culler, S. Shenker, I. Stoica, A unifying link abstraction for wireless sensor networks, in: Proceedings of the third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.
[114] L.V. Hoesel, T. Nieberg, J. Wu, P.J.M. Havinga, Prolonging the lifetime of wireless sensor networks by cross-layer interaction, IEEE Wireless Communications Magazine 11 (December) (2004) 78–86.
[115] W. Ye, J. Heidemann, D. Estrin, An energy-efficient MAC protocol for wireless sensor network, in: Proceedings of the Infocom, New York, 2002.
[116] D.B. Johnson, D.A. Maltz, Dyanmic source routing, in: T. Imielinski, H.F. Korth (Eds.), Ad Hoc Wireless Networks Mobile Computing, vol. 353, 1996.
[117] S. Cui, R. Madan, A.J. Goldsmith, S. Lall, Joint routing MAC and link layer optimization in sensor networks with energy constraints, in: Proceedings of the IEEE ICC, 2005, pp. 725–729.
[118] I.F. Akyildiz, M.C. Vuran, O.B. Akan, A cross-layer protocol for wireless sensor networks, in: Proceedings of the Conference on Information Science and Systems (CISS), 2006.
[119] P. Skraba, H. Aghajan, A. Bahai, Cross-layer optimization for high density sensor networks: distributed passive routing decisions, in: Proceedings of the Ad-Hoc Now'04, Vancouver, 2004.
[120] M. Zorzi, R. Rao, Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance, IEEE Transactions on Mobile Computing 2 (2003) 337–348.
[121] M. Chiang, Balancing transport and physical layers in wireless multihop networks: jointly optimal congestion control and power control, IEEE Journal on Selected Area in Communications (JSAC) 23 (2005) 104–116.
[122] M. Chiang, To layer or not to layer: balancing transport and physical layers in wireless multihop networks, in: Proceedings of the IEEE INFOCOM, 2004, pp. 2525–2536.
[123] R. Madan, S. Cui, S. Lall, A. Goldsmith, Cross-layer design for lifetime maximization interference-limited wireless sensor networks, IEEE Communications on Wireless Communications 5 (2005) 3142–3152.

Jennifer Yick received the BS Degree in Computer Science and Engineering from the University of California, Davis, in 2001, and the MS Degree in Computer Science from the University of California, Davis, in 2004. She is currently a PhD candidate in the Department of Computer Science at University of California, Davis. Her current research interests include energy conservation, localization, clustering, target tracking and network survivability in wireless sensor networks.

Biswanath Mukherjee (S'82–M'87) received the BTech (Hons) Degree from Indian Institute of Technology, Kharagpur, India in 1980 and the PhD Degree from University of Washington, Seattle in June 1987. At Washington, he held a GTE Teaching Fellowship and a General Electric Foundation Fellowship. In July 1987, he joined the University of California, Davis, where he has been Professor of Computer Science since July 1995 (and currently holds the Child Family Endowed Chair Professorship), and served as Chairman of the Department of Computer Science during from September 1997 to June 2000. He is winner of the 2004 Distinguished Graduate Mentoring Award at UC Davis. Two PhD Dissertations (by Dr. Laxman Sahasrabuddhe and Dr. Keyao Zhu), which were supervised by Professor Mukherjee, were winners of the 2000 and 2004 UC Davis College of Engineering Distinguished Dissertation Awards. To date, he has graduated nearly 25 PhD students, with almost the same number of MS students. Currently, he supervises the research of nearly 20 scholars, mainly PhD students and including visiting research scientists in his laboratory. He is Co-winner of paper awards presented at the 1991 and the 1994 National Computer Security Conferences. He serves or has served on the editorial boards of the IEEE/ACM Transactions on Networking, IEEE Network, ACM/Baltzer Wireless Information Networks (WINET), Journal of High Speed Networks, Photonic Network Communications, Optical Network Magazine, and Optical Switching and Networking. He served as Editor-at-Large for optical Networking and Communications for the IEEE Communications Society; as the Technical Program Chair of the IEEE INFOCOM'96 conference; and as Chairman of the IEEE Communication Society's Optical Networking Technical Committee (ONTC) during 2003–2005. He is Author of the textbook 'Optical WDM Networks' published by Springer in January 2006. Earlier, he Authored the textbook 'Optical Communication Networks' published by McGraw-Hill in 1997, a book which received the Association of American Publishers, Inc.'s 1997 Honorable Mention in Computer Science. He is a Member of the Board of Directors of IPLocks, Inc., a Silicon Valley startup company. He has consulted for and served on the Technical Advisory Board (TAB) of a number of startup companies in optical networking. His current TAB appointments include: Teknovus, Intelligent Fiber Optic Systems, and LookAhead Decisions, Inc. (LDI). His research interests include lightwave networks, network security, and wireless networks.

Dipak Ghosal received the BTech Degree in Electrical Engineering from Indian Institute of Technology, Kanpur (India), in 1983, and MS Degree in Computer Science and Automation from Indian Institute of Science, Bangalore, India, in 1985. He received his PhD Degree in Computer Science from University of Louisiana, in 1988. He is currently a Professor in the Department of Computer Science at the University of California, Davis. His primary research interests are in the areas of high speed and wireless networks with particular emphasis on the impact of new technologies on the network and higher layer protocols and applications. He is also interested in the application of parallel architectures for protocol processing in high speed networks and in the application of distributed computing principles in the design of next generation network architectures and server technologies.