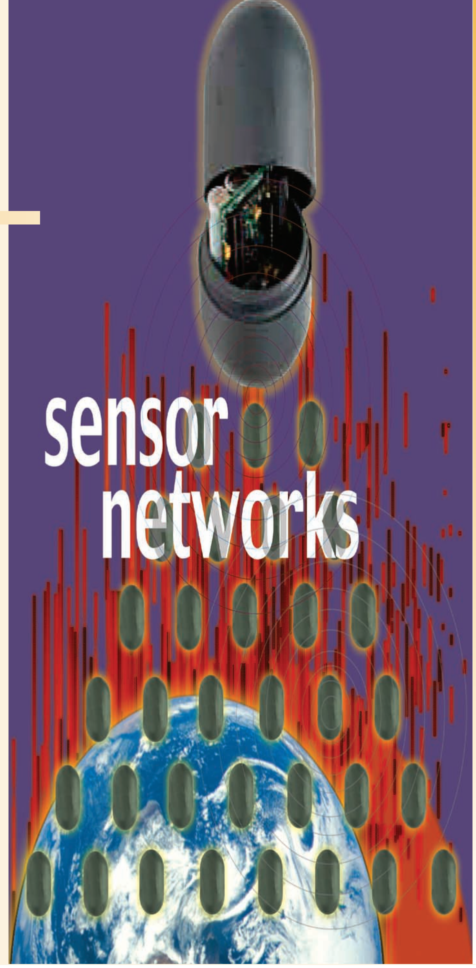
Overview of Sensor Networks



Wireless sensor networks could advance many scientific pursuits while providing a vehicle for enhancing various forms of productivity, including manufacturing, agriculture, construction, and transportation.

David Culler
University of California, Berkeley

Deborah Estrin

Mani Srivastava
University of California, Los Angeles

The advances in science and technology are deeply intertwined. The telescope enables a deeper understanding of astronomy, the microscope brings bacteria into view, and satellites survey the Earth's surface, expanding what we can perceive and measure. Now, we can use computers to visualize, through numerical simulation, physical phenomena we cannot observe through empirical means.

This trend has advanced with the prolonged exponential growth in the underlying semiconductor technology. The number of transistors on a cost-effective chip and, therefore, the processing or storage capacity of that chip, doubles every year or two, following Moore's law. While it has provided ever more computing power, researchers are now applying this technology in ways that enable a new role for computing in science.

A given computing capacity becomes exponentially smaller and cheaper with each passing year. Researchers can use the semiconductor manufacturing techniques that underlie this miniaturization to build radios and exceptionally small mechanical structures that sense fields and forces in the physical world. These inexpensive, low-power communication devices can be deployed throughout a physical space, providing dense sensing close to physical phenomena, processing and communicating this information, and coordinating actions with other nodes. Combining these capabilities with the system software technology that forms the Internet makes it possible to instrument the world with increasing fidelity.

To realize this opportunity, information technology must address a new collection of challenges. The individual devices in a wireless sensor network (WSN) are inherently resource constrained: They have limited processing speed, storage capacity, and communication bandwidth. These devices have substantial processing capability in the aggregate, but not individually, so we must combine their many vantage points on the physical phenomena within the network itself.

In most settings, the network must operate for long periods of time and the nodes are wireless, so the available energy resources—whether batteries, energy harvesting, or both—limit their overall operation. To minimize energy consumption, most of the device's components, including the radio, will likely be turned off most of the time. Because they are so closely coupled to a changing physical world, the nodes forming the network will experience wide variations in connectivity and will be subject to potentially harsh environmental conditions. Their dense deployment generally means that there will be a high degree of interaction between nodes, both positive and negative. Each of these factors further complicates the networking protocols.

Despite these operational factors, deploying and maintaining the nodes must remain inexpensive. Because manually configuring large networks of

0018-9162/04/$20.00 © 2004 IEEE

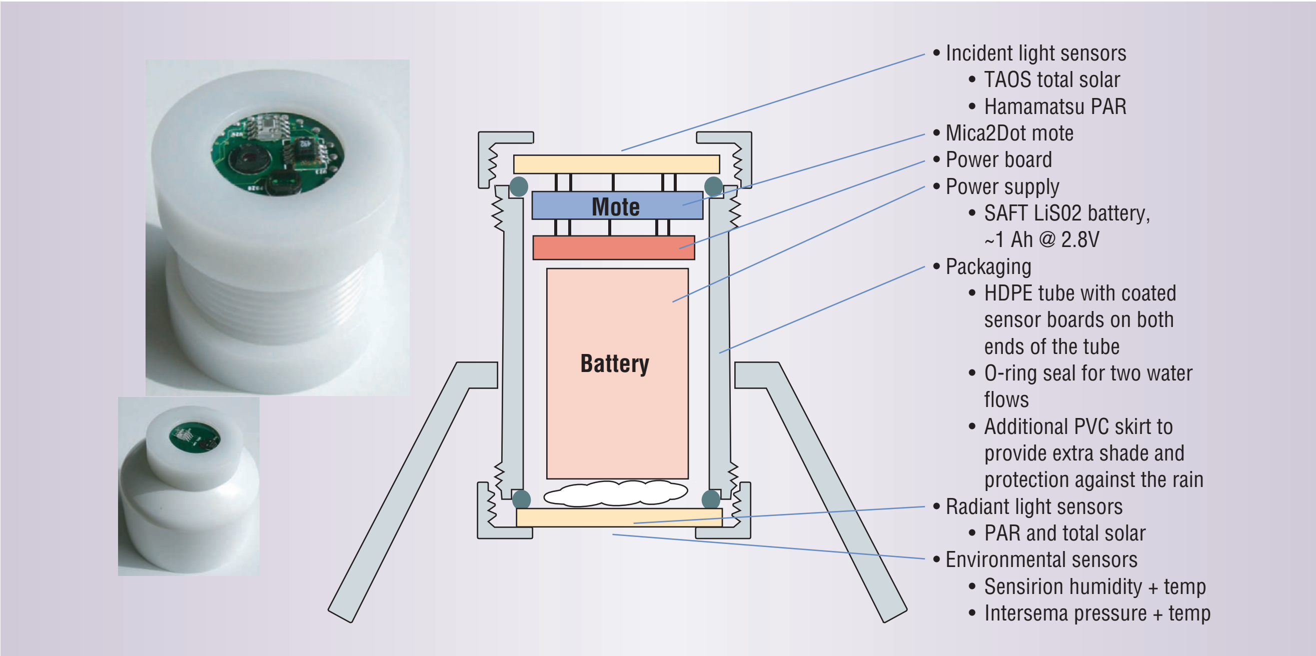


Figure 1. Wireless sensor node for environmental monitoring. An entire wireless microweather station fits in a tube about the size of a film canister.

small devices is impractical, the nodes must organize themselves and provide a means of programming and managing the network as an ensemble, rather than administering individual devices. Overcoming these challenges will let computer technology fill a new role in the progress of science.

SENSOR NETWORK APPLICATIONS

Although computer-based instrumentation has existed for a long time, the density of instrumentation made possible by a shift to mass-produced intelligent sensors and the use of pervasive networking technology gives WSNs a new kind of scope that can be applied to a wide range of uses. These can be roughly differentiated into

- monitoring space,
- monitoring things, and
- monitoring the interactions of things with each other and the encompassing space.

The first category includes environmental and habitat monitoring, precision agriculture, indoor climate control, surveillance, treaty verification, and intelligent alarms. The second includes structural monitoring, ecophysiology, condition-based equipment maintenance, medical diagnostics, and urban terrain mapping. The most dramatic applications involve monitoring complex interactions, including wildlife habitats, disaster management, emergency response, ubiquitous computing environments, asset tracking, healthcare, and manufacturing process flow.

Environmental monitoring

Many initial WSNs have been deployed for environmental monitoring, which involves collecting readings over time across a volume of space large enough to exhibit significant internal variation. Researchers are using WSNs to monitor nesting seabird habitats and microclimate chaparral transects and to conduct analogous studies of contaminant propagation, building comfort, and intrusion detection. One example, monitoring the microclimate throughout the volume of redwood trees, helps form a sample of entire forests.

Redwood trees are so large that entire ecosystems exist within their physical envelope. Climatic factors determine the rate of photosynthesis, water and nutrient transport, and growth patterns. Substantial variations are known to exist over the volume of an individual specimen, and researchers believe that the microclimatic structure varies over regions of the forest. In addition, water transport rates and the scale of respiration may influence the microclimate around a tree, effectively creating its own weather. All these factors influence the habitat dynamics of species existing in and on the tree.

Researchers traditionally have performed ecosystem measurements by hauling a suite of instruments, weighing perhaps 30 lbs, up a tree by a winch attached high in the canopy. Serial cables then hang down to the forest floor, where a data logger collects measurements. The instruments obtain measurements at various elevations at distinct points in time over relatively short intervals.

Figure 1 shows a modern WSN used for environmental monitoring in collaboration with biologist Todd Dawson. An entire wireless weather station fits in a tube about the size of a film canister. On top, two incident-light sensors measure total solar radiation, specifically light and photosynthetically active radiation, the bands at which chlorophyll are sensitive. An identical pair of sensors on the bottom,

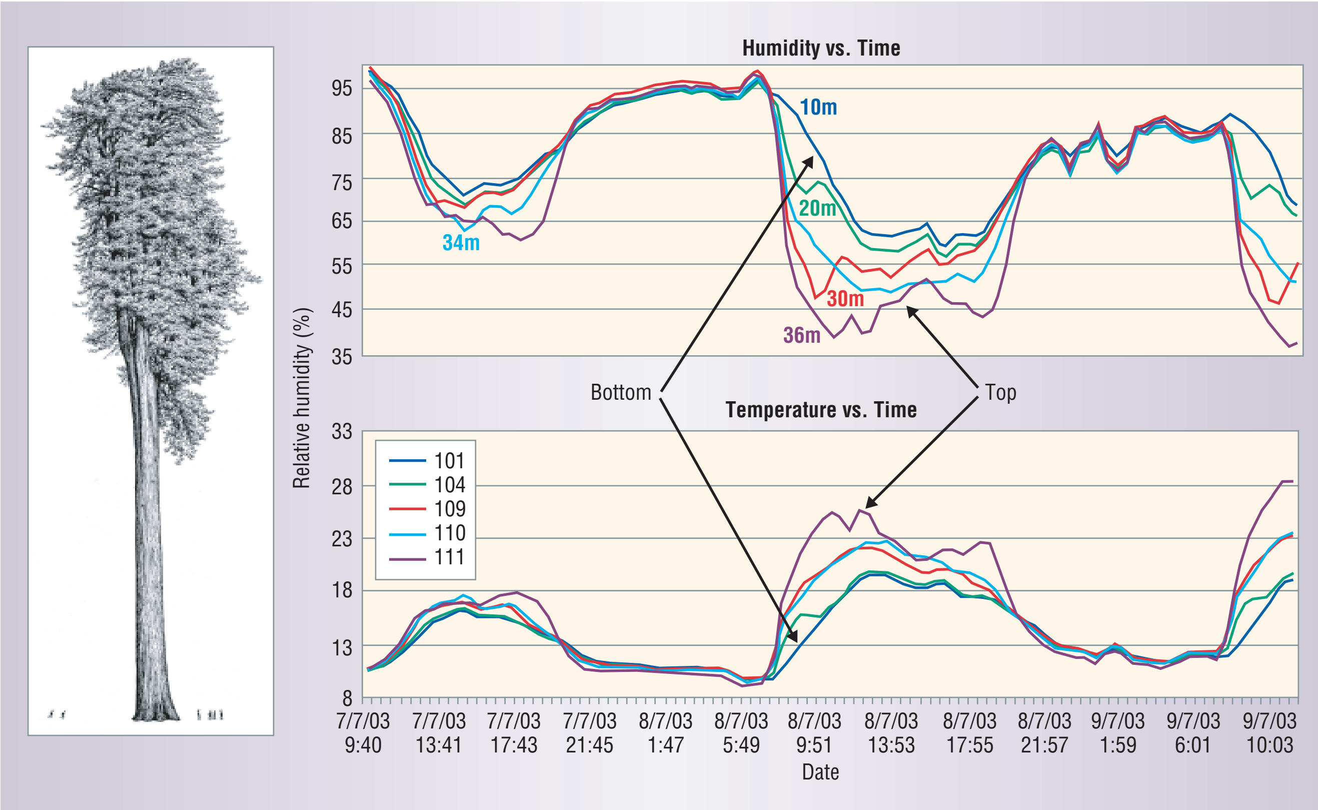


Figure 2. WSN climate data. The WSN samples climate data every five minutes and computes an average temperature at each elevation.

underneath a shade, measure radiant light. In addition, on the bottom there are environmental sensors to monitor relative humidity, barometric pressure, and temperature. The shade keeps rain off these sensors and prevents warming.

The weather-protected center of the tube contains a small computer, data storage, battery, and low-power radio to collect data, process it, and route information among the nodes and to the outside world. This provides a cost-effective means of obtaining simultaneous measurements at many points in the tree, spanning elevation and radial direction over a prolonged period.

Various facets of the forest, such as the center versus the windward, leeward, or sunward edges, can be similarly instrumented using small router nodes or long-distance uplinks placed in stretches of intervening forest to provide connectivity.

Figure 2 shows a temperature profile over three days, collected from 16 nodes at four elevations in a 35-meter study tree. The WSN samples climate data every five minutes and computes an average temperature at each elevation. The measurements show that within the expected daily cycle, the top of the tree experiences much wider climatic variation than the forest floor.

The network data also shows how weather fronts move up and down the tree. The top of the tree warms rapidly as the sun rises. This thermal front moves down the tree as the day warms, decreasing as it progresses. At nightfall, the situation reverses, with the top cooling below the base.

Variations in humidity are even more pronounced than climatic variations, further illustrating the importance of dense instrumentation because the tree moves tremendous volumes of water, which increases the humidity within the canopy. These weather fronts create powerful temperature and moisture gradients that could be instrumental in understanding growth dynamics, water intake, and nutrient transport over such large structures, yet they cannot be observed with sparse instrumentation.

In an initial study, researchers will likely extract all the environmental data from such a network and use it to identify biologically relevant features, such as a temperature front. Later, the network will likely perform the front-tracking algorithm within the tree itself, communicating only relevant statistical summaries, thereby increasing the network lifetime. As the ecophysiological model evolves, the researchers will likely refine the detection and tracking features.

Motion monitoring

Monitoring objects presents different challenges and opportunities. Many applications can be viewed as a form of condition-based maintenance. A physical structure, such as a machine, motor, airplane wing, bridge, or building has typical modes of vibration, acoustic emissions, and response to stimuli. Variations in these behaviors indicate wear, fatigue, or other mechanical changes. For example, a bearing often will squeak and shudder before it seizes up.

> **Using sensor nodes to perform structured analysis requires establishing a highly accurate time frame.**

In addition to the sophisticated instrumentation of the actual wafer processing, a modern semiconductor fabrication plant can have several thousand vibration sensors attached to various pieces of routine machinery. A team of electricians tours the plant with a computing device that attaches to a sensor and logs a sample for a short period. The team then carries these logs back to a central computer, which analyzes them for signs of wear. Months elapse between visits to a particular machine. This scenario applies to a wide range of manufacturing and power generation plants.

WSNs offer an alternative approach: performing local processing at each device and transporting the data continuously to operations staff. Sampling rates are much higher than with environmental monitoring—typically about 100 Hz for vibration analysis and several kilohertz for acoustic analysis. This requires more energetic sampling activity and greater care in how the network performs sampling. Buffering the data requires greater amounts of storage, and the system potentially performs more extensive local processing on data chunks sampled intermittently.

Rather than transmitting large amounts of raw data, the sensor nodes can perform signal analysis, communicating only the modes of vibration or detected anomalies. Sensor nodes can monitor control networks to establish what is active when a sample is taken or even to determine when to sample. Because reducing the cost of obtaining and processing data reduces costs overall, increasing the timeliness of analysis can improve plant performance.

The analysis of structural response in, for example, bridges, buildings, and airframes, places a further requirement to use data collected at different points in the structure in spatial-temporal analysis. This requires establishing a common, highly accurate time frame across nodes. Nodes share time-correlated raw or processed data to perform the structural analysis. For example, the sensor data from one instrument can be used as an input to a model-based analysis at each of several other points in a structure and compared to sensor data at those points. Researchers refine these models by using them iteratively in normal circumstances to detect anomalies.

EMBEDDED NETWORK TECHNOLOGY

WSNs merge a wide range of information technology that spans hardware, systems software, networking, and programming methodologies.

Microprocessors, power, and storage

A sensor network node's hardware consists of a microprocessor, data storage, sensors, analog-to-digital converters (ADCs), a data transceiver, controllers that tie the pieces together, and an energy source. Recently, a new operating point has emerged that suits all these components. As semiconductor circuits become smaller, they consume less power for a given clock frequency and fit in a smaller area. In simple microcontrollers, miniaturization increases efficiency rather than adding functionality, allowing them to operate near one milliwatt while running at about 10 MHz. Most of the circuits can be powered off, so the standby power can be about one microwatt. If such a device is active 1 percent of the time, its average power consumption is just a few microwatts.

This scale of power can be obtained in many ways. Solar cells generate about 10 milliwatts per square centimeter outdoors and 10 to 100 microwatts per square centimeter indoors. Mechanical sources of energy, such as the vibration of windows and air conditioning ducts, can generate about 100 microwatts.

A typical cubic-centimeter battery stores about 1,000 milliamp-hours, so centimeter-scale devices can run almost indefinitely in many environments. However, low-power microprocessors have limited storage, typically less than 10 Kbytes of RAM for data and less than 100 Kbytes of ROM for program storage—or about 10,000 times less storage capacity than a PC has. This limited amount of memory consumes most of the chip area and much of the power budget. Designers typically incorporate larger amounts of flash storage, perhaps a megabyte, on a separate chip.

Microsensors

Sensors give these nodes their eyes and ears. Many materials change their electrical characteristics when subjected to varying environmental conditions. Sensors are manufactured so these changes are predictable over a certain range. For example, a thermistor is a variable resistor that changes smoothly with temperature. An ADC converts the voltage drop into a binary number that a microcontroller can store or process. Photocells and fog detectors work similarly, but they consist of finely interleaved combs separated by a material that uses incident photons or moisture to change resistance.

Many more sophisticated structures have been developed to detect other phenomena. These structures consume a few milliwatts and only need to be turned on a fraction of the time. Extremely efficient

ADCs have been developed so that the sensor subsystem has an energy profile similar to the processor.

Microelectromechanical systems (MEMS) can sense a wide variety of physical phenomena cheaply and efficiently. Researchers can use the processes for etching transistors on silicon to carve out tiny mechanical structures, such as a microscopic springboard within an open cavity. Gravitational forces or acceleration can deflect this cantilevered mass, causing powerful internal forces that cause changes in material properties or delicate alignments, which can be amplified and digitized. Manufacturers used the first major commercial MEMS sensor, the accelerometer, to trigger automotive airbag release.

Whereas high-precision piezoelectric accelerometers cost hundreds of dollars, MEMS provided sufficient precision for a few dollars. Once the devices entered mass production, they could ride the CMOS technology growth of modern chips to become increasingly accurate while remaining inexpensive. A wide variety of MEMS devices can sense various forces, chemical concentrations, and environmental factors.

Microradios

For some time now, manufacturers have added sensors to many appliances, vehicles, and gadgets. The breakthrough comes from communicating sensor readings to other devices, translating the physical world into information—bits—that the devices can transport, store, and process.

Radio components can now be manufactured using conventional CMOS technology, enabling wireless entry devices, pagers, walkie-talkies, cell phones, and wireless local area networks for mobile laptops. However, the amount of energy required to communicate wirelessly increases rapidly with distance. Obstructions—such as people or walls—and interference further attenuate the signal.

Wireless LANs and cell phones consume hundreds of milliwatts and rely on a powerful infrastructure. WSN radios consume about 20 milliwatts, and their range typically is measured in tens of meters.

For small devices to cover long distances, the network must route the information hop by hop through nodes, much as routers move information across the Internet. Even so, communication remains one of the most energy-consuming operations, with each bit costing as much energy as about 1,000 instructions. Thus, WSNs process data within the network wherever possible.

SYSTEMS CHALLENGE

Bridging the gap between the hardware technology's raw potential and the broad range of applications presents a systems challenge. The network must allocate limited hardware to multiple concurrent activities, such as sampling sensors, processing, and streaming data. The potential interconnections between devices must be discovered and information routed effectively from where it is produced to where it is used. There must also be a means of programming the ensemble.

To reduce energy consumption, WSNs process data within the network wherever possible.

TinyOS

Conventional operating systems such as Unix run well on a 32-bit microprocessor at 50 to 100 MHz, with several megabytes of RAM and a gigabyte or more of secondary storage. Today, this can be achieved in a handheld device that runs for several hours on a single charge.

A more typical operating point for WSNs is one year on a pair of AA batteries with a small fraction of these resources. Further, this application focuses on structured interaction with the physical world, rather than on complex human interactivity. The developers of the open source TinyOS tailored it for this application.

The TinyOS provides a framework for assembling application-specific systems that can handle substantial concurrency within limited physical resources. The software components and the underlying operating system support specific event-driven functionality. The lowest-level components abstract the physical hardware and deliver physical interrupts as sanitized, asynchronous events. Each component handles certain events and signals actions in other components, but they never consume processor cycles while waiting for future events.

Each application includes only the components it requires. For example, a small stack of components process sensor readings. The lowest components handle the ADC to obtain raw readings, whereas the higher ones filter and distill data streams for the application.

The network involves a more complex stack in which the lower levels deal with acquiring the radio channel, framing data streams into packets that receiving nodes can recognize. These components also perform error coding and channel scheduling, as well as detecting the arrival of incoming packets and processing them into input buffers.

Higher levels deal with buffer management, authentication, and multiplexing the network

The link layer transmits a structured series of bits that form a packet encoded in the radio signal.

across application components. A typical top-level application might receive and process a stream of filtered sensor readings, then deliver important notifications to the network. A second component would receive such notification messages, maintain a routing structure, and retransmit them along the next hop in a route to a data collection gateway.

Network sensor platforms

Berkeley motes and TinyOS are widely used for exploring systems issues and deploying pilot applications. The microcontroller provides a modest amount of RAM and program storage and contains an internal ADC. A simple frequency-agile radio with roughly the bandwidth of a modem provides the connectivity that developers can use to construct a network. Off-chip flash memory provides storage to hold both the program while it transfers through the network and the data buffering beyond the on-chip RAM. Several sensor boards have been designed for this platform.

The Intel iMote is a recent integrated design that uses a commercial chip with a powerful ARM microprocessor, storage, and radio integrated into a single package. The radio implements the Bluetooth standard, which is becoming widely used in laptops and cell phones. The radio operates at higher bandwidth and has a sophisticated frequency-hopping protocol.

Normally, the ARM processor would be devoted to managing the Bluetooth radio and transferring packets to and from a serial port. In the iMote, however, TinyOS runs directly on the ARM processor, providing a stand-alone system that services various sensors and routes, processes high-level information streams, and manages power consumption.

Most of the lower-level TinyOS components are implemented directly in hardware. These include an extremely low-power ADC and a very efficient radio developed in the Smart Dust project (http://robotics.eecs.berkeley.edu/~pister/SmartDust/). The entire design occupies only 5 square millimeters. It is estimated that at 1 percent active, the chip could run a hundred years on the energy stored in a pair of AA batteries.

On the other end of the spectrum of in situ nodes lie 32-bit processor-based devices such as Stargate, which run traditional operating systems such as Linux that are equipped with longer-range radios such as IEEE 802.11 or with cell phone modems.

This node class will play a critical role in most deployed systems. At a minimum, these nodes will act as gateway points both to retrieve data off the sensor network and to monitor, configure, and task the system. In more sophisticated heterogeneous systems, these nodes will be distributed more widely and used as points of aggregation, data storage, data fusion, and hosts for higher-end sensors. Because they are so much more energy intensive, these nodes operate along with a large battery and some form of recharge such as a solar panel. Alternatively, when wall outlets are available, the nodes can draw power from them.

SELF-ORGANIZED NETWORKS

Wireless communication and instrumentation have long been associated with remote sensing from satellites and missile telemetry—prime examples of wireless links. A network consists of many nodes, each with multiple links connecting to other nodes. Information moves hop by hop along a route from the point of production to the point of use.

In a wired network like the Internet, each router connects to a specific set of other routers, forming a routing graph. In WSNs, each node has a radio that provides a set of communication links to nearby nodes. By exchanging information, nodes can discover their neighbors and perform a distributed algorithm to determine how to route data according to the application's needs. Although physical placement primarily determines connectivity, variables such as obstructions, interference, environmental factors, antenna orientation, and mobility make determining connectivity a priori difficult. Instead, the network discovers and adapts to whatever connectivity is present.

Connectivity

The networking capability of WSNs is built up in layers. The lowest layer controls the physical radio device. Radios are by nature a broadcast medium: When one node transmits, a collection of others can receive the signal unless it is garbled by other transmissions at the same time. To avoid contending for the radio channel, the link layer listens on the channel and transmits only when the channel is clear. It transmits a structured series of bits that form a packet encoded in the radio signal.

When not transmitting, nodes sample the channel and scan for a special symbol at the start of a packet that also lets the receiver align itself with the sender's time. The packet layer manages buffers, schedules packets onto the radio, detects or even corrects errors, handles packet losses, and dispatches packets to system or application components.

Dissemination and data collection

Developers use this basic communication capacity to implement protocols that let the collection of nodes transport and process information and coordinate their activities. A basic capability in such networks involves disseminating information over many nodes. This can be achieved by a flooding protocol in which a root node broadcasts a packet with some identifying information. Receiving nodes retransmit the packet so that more distant nodes can receive it. However, a node can receive different versions of the same message from several neighboring nodes, so the network uses the identifying information to detect and suppress duplicates. Flooding protocols use various techniques to avoid contention and minimize redundant transmissions.

The network uses dissemination to issue commands, convey alarms, and configure and task the network. But it also uses dissemination to establish routes. Each packet identifies the transmitter and its distance from the root. To form a distributed tree, nodes record the identity of a node closer to the root. The network can use this reverse communication tree for data collection by routing data back to the root or for data aggregation by processing data at each level of the tree.

The root can be a gateway to a more powerful network or an aggregation point within the sensor network, as determined by some higher-level task. Often, tree formation and data collection are interwoven. Data can begin following up the tree as soon as a parent node is discovered. Nodes may learn of potential parents by overhearing data messages. The network continually collects statistics to reinforce the best routes.

These communication patterns differ significantly from those found on the Internet, where many client computers open connections to named servers and transfer large streams of data back and forth. In sensor networks, communication is usually performed in the aggregate, and participants are identified by attributes such as physical location or sensor value range. This style of routing has been formulated as *directed diffusion*, a process in which nodes express interest in data by attribute. The nodes flood interest outward to form a routing gradient, and they collect data up the gradient by reinforcing the associated subtree.

Reliability also follows a different pattern. Increasingly, sensor networks will deploy *disruption-tolerant networking* approaches in which they transfer bundles of data reliably, hop by hop, in contrast to the Internet, which sets up an end-to-end connection using byte or packet matching between the original source and destination to determine reliability. The DTN model better suits the variable connectivity that results from dynamic environments and the need to duty-cycle.

In sensor networks, disruption-tolerant networking reliably transfers data bundles hop by hop.

CONSERVING POWER AND BANDWIDTH

Communication, usually the most energy-intensive operation a node performs, must contend for a share of limited bandwidth. The network stack attempts to minimize energy usage, either by eliminating communication or by turning off the radio when no communication needs to occur.

Several approaches are possible. For example, nodes could process data locally and only communicate when they detect an interesting event. This approach would be employed in an intelligent alarm system or an environmental monitoring system that focuses data collection on time or areas of interest.

In many cases, crude low-power sensors trigger higher-powered sensor devices, such as cameras. Performing aggregation within the network can reduce communication. For example, an application might need to determine the average temperature at shaded nodes in a certain geographic region. Selecting the subset of readings of interest could be performed at the tree's leaves, routing their aggregation as data upward, so that each node transmits at most a single packet to provide a statistical summary of its subtree. More sophisticated aggregation could involve detecting distributed regions of interest.

Compression and scheduling also can conserve energy at lower layers. Some protocol overhead is associated with data communication to maintain routing structures, manage contention, and enhance reliability. Sensor networks can avoid explicit protocol messages by piggybacking control information on data messages and by overhearing packets destined for other nodes. They can use prescheduled time to reduce contention and the time the radio remains live. This can be coordinated with the high-level application behavior by, for example, periodic low-rate data sampling. Alternatively, the network could implement energy conservation generically within lower layers by, for example, time division multiple access.

In the spatial dimension, the network can assign specific responsibilities to certain nodes, such as retransmission or aggregation. Finally, the network can reject uninteresting packets by turning off the radio after receiving only a portion. However, because these many optimizations can be mutually

In settings in which general human activity occurs, many potentially interested parties can have varying uses for the data.

conflicting, a rich and growing body of literature employs different combinations of techniques under different application and platform assumptions.

PRIVACY

Dense instrumentation, real-time access, and in-network processing make a qualitative difference in our ability to perceive what is happening throughout large physical structures. In environmental monitoring and condition-based maintenance, the purpose of data collection, the parties responsible for using the data, and the scope of dissemination are clear. The situation becomes much less clear in more casual settings in which more general human activity occurs, such as the home, the workplace, a transportation terminal, or a shopping venue. In these cases, many potentially interested parties can have varying uses for the data.

More detailed sensing—such as occupancy, motion, and even physiological state—further amplifies concerns over proper use and dissemination. Indeed, image data, such as that obtained by the surveillance cameras in pervasive use today, can be viewed as an extremely powerful sensor, but network access and automated analysis are limited. These social factors are an inherent concern with sensor network technology. Fortunately, this area has become an active focus of research while the technology is still in its early stages.

IN THIS ISSUE

The articles in this special issue span novel sensor network applications, embedded network technology, and systems design challenges.

The three projects described in "Sensor Network Applications" demonstrate that deploying sensor networks to monitor the natural environment requires an understanding of earth science combined with sensor, communications, and computer technology.

In "Environmental Sensor Networks," Kirk Martinez and his coauthors from the University of Southampton describe their GlacsWeb project focusing on ongoing research in subglacial bed deformation and discuss the challenges encountered in extracting data gathered by sensor nodes deployed in remote locations.

As a final defense against the potential detonation of a radiological dispersion device (RDD) capable of broadcasting nonfissile but highly radioactive particles over a densely populated area, select traffic choke points in the US have large portal monitoring systems to help detect vehicles transporting illicit isotopes. In "Radiation Detection with Distributed Sensor Networks," Sean M. Brennan and coauthors discuss a project that is being developed at Los Alamos National Laboratory in cooperation with the University of New Mexico to provide a hand-carried DSN system that is less visible, uses less power per detector, and is more suitable for deployment in an urban environment than a portal monitoring system.

Detecting and accurately locating snipers has long been an elusive goal of the armed forces and law enforcement agencies. In "Shooter Localization in Urban Terrain," Akos Ledeczi and coauthors describe PinPtr, a prototype system that provides a novel approach to solving this difficult problem even in a challenging environment such as complex urban terrain. In PinPtr, an ad hoc wireless network of tiny inexpensive acoustic sensors measures both the muzzle blast and the shock wave to accurately determine the shooter's location and the bullet's trajectory. The system automatically classifies the measurements and eliminates those that are erroneous or the result of multipath effects.

"WiseNET: An Ultralow-Power Wireless Sensor Network Solution" by Christian C. Enz and his colleagues from the Swiss Center for Electronics and Microtechnology describes a platform for implementing wireless sensor networks that achieve low-power operation. To optimize overall power consumption, WiseNET uses a codesign approach that exploits the relationship between media access control layer performance and radio transceiver parameters. The system combines a complex system-on-chip sensor node with a dedicated duty-cycled radio and WiseMAC, a MAC-layer protocol designed for low-duty-cycle wireless-sensor networks.

In "The Flock: Wireless Mote Networks in an Undergraduate Curriculum," Bruce Hemingway and his colleagues from the University of Washington explain how they integrated the emerging field of wireless sensor networks into an undergraduate computer engineering curriculum to provide an introduction to embedded systems development. Using TinyOS sensor motes provided a convenient platform for applying concepts taught in a program consisting of three classes that supplement a core curriculum shared with computer science majors: advanced digital design, embedded software, and a capstone design experience.

As the technology that is commercially available today becomes established enough to warrant greater investment, straightforward engineering efforts will yield complete devices with processing, storage, sensing, and communication functions that fit in much less than a cubic centimeter of space and cost just a few dollars.

Looking forward, the technology will likely evolve into a much less distinct and visible form. Instead of being housed in many small devices, these elements will likely become part of the manufacturing process for various materials and objects. These sensors will tend to operate within the ambient energy sources of their intended environment and be placed at key junctures where analysis is most critical. As this vision evolves, so will the need for fundamentally new information technology architectures, from programming languages to signal-processing algorithms.

Over the 50 years of modern computing, we have seen a new class of computer emerge about once a decade, progressing through mainframes, minicomputers, personal computers, and mobile computers. Each successive model relies upon technical advances, especially integration, to make computing available in a form factor not previously possible. Each has ushered in new uses for computer technology. Each succeeding generation is smaller, more plentiful, and more intimately associated with personal activity than the generation that preceded it.

WSNs appear to represent a new class. They follow the trends of size, number, and cost, but have a markedly different function. Rather than being devoted to personal productivity tasks, WSNs make it possible to perceive what takes place in the physical world in ways not previously possible. In addition to offering the potential to advance many scientific pursuits, they also provide a vehicle for enhancing larger forms of productivity, such as manufacturing, agriculture, construction, and transportation. ■

David Culler *is a professor of computer science at the University of California, Berkeley, and founding director of Intel Research, Berkeley. His research interests include sensor networks, parallel computer architectures, and high-performance communication. Culler received a PhD from the Massachusetts Institute of Technology. He is a Senior Member of the IEEE and a Fellow of the ACM. Contact him at culler@eecs.berkeley.edu.*

Deborah Estrin *is the director of the Center for Embedded Networked Sensing and a professor in the Department of Computer Science at the University of California, Los Angeles. Her research interests focus on developing protocols and systems architectures needed to deploy sensor networks for environmental monitoring. Estrin received a PhD in computer science from the Massachusetts Institute of Technology. She is a Fellow of the IEEE, the ACM, and the AAAS. Contact her at destrin@cs.ucla.edu.*

Mani Srivastava *is an associate professor in the Department of Electrical Engineering at the University of California, Los Angeles. His research focuses on architectures, protocols, and algorithms for networked computing, communication, and DSP systems and VLSIs, and on CAD techniques to optimize and synthesize them. Srivastava received a PhD in electrical engineering from the University of California, Berkeley. He is a member of the IEEE and the ACM. Contact him at mbs@ee.ucla.edu.*