SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF COMPUTING

DEPARTMENT OF INFORMATION TECHNOLOGY

UNIT - 1

SITA1501 – WIRELESS SENSOR NETWORKS AND ARCHITECTURE

UNIT - 1 INTRODUCTION AND OVERVIEW OF WIRELESS SENSOR NETWORKS

Syllabus:
Introduction, Brief Historical Survey of Sensor Networks, and Background of Sensor Network Technology, Ah-Hoc Networks, Applications of Wireless Sensor Networks: Sensor and Robots, Reconfigurable Sensor Networks, Highway Monitoring, Military Applications, Civil and Environmental Engineering Applications, Wildfire Instrumentation, Habitat Monitoring, Another Taxonomy of WSN Technology, Basic Sensor Network Architectural Elements, Home Control, Medical Applications.

1.1 Introduction

A sensor network is an
infrastructure comprised of sensing (measuring), computing, and communication elements that gives an administrator the ability to instrument, observe, and react to events and phenomena in a specified environment. The administrator typically is a civil, governmental, commercial, or industrial entity. The environment can be the physical world, a biological system, or an information technology (IT) framework. Network(ed) sensor systems are seen by observers as an important technology that will experience major deployment in the next few years for a plethora of applications, not the least being national security. Typical applications include, but are not limited to, data collection, monitoring, surveillance, and medical telemetry. In addition to sensing, one is often also interested in control and activation.

There are four basic components in a sensor network: (1) an assembly of distributed or localized sensors; (2) an interconnecting network (usually, but not always, wireless-based); (3) a central point of information clustering; and (4) a set of computing resources at the central point (or beyond) to handle data correlation, event trending, status querying, and data mining. In this context, the sensing and computation nodes are considered part of the sensor network; in fact, some of the computing in the wireless case), the developments in IT (such as high-power processors, large random-access memory chips, digital signal processing, and grid computing), coupled with recent engineering advances, are in the aggregate opening the door to a new generation of low-cost sensors and actuators that are capable of achieving high-grade spatial and temporal resolution.

1.2 Background of Sensor Network Technology

Researchers see WSNs as an ‘‘exciting emerging domain of deeply networked systems of low-power wireless motes with a tiny amount of CPU and memory, and large federated networks for high-resolution sensing of the environment’’. Sensors in a WSN have a variety of purposes, functions, and capabilities. The field is now advancing under the push of recent technological advances and the pull of a myriad of potential applications. The radar networks used in air traffic control, the national electrical power grid, and nationwide weather stations deployed over a regular topographic mesh are all examples of early-deployment sensor networks; all of these systems, however, use specialized computers and communication protocols and consequently, are very expensive. Much less expensive WSNs are now being planned for novel applications in physical security, health care, and commerce. Sensor networking is a multidisciplinary area that involves, among others, radio and networking, signal processing, artificial intelligence, data-base management, systems architectures for operator-friendly infrastructure administration, resource optimization, power management algorithms, and platform technology (hardware and software, such as operating systems). The applications, networking principles, and protocols for these systems are just beginning to be developed. The near-ubiquity of the Internet, the advancements in wire-less and wireline communications technologies, the network build-out (particularly in the wireless case), the developments in IT (such as high-power processors, large random-access memory chips, digital signal processing, and grid computing), coupled with recent engineering advances, are in the aggregate opening the door to a new generation of low-cost sensors and actuators that are capable of achieving high-grade spatial and temporal resolution.

The technology for sensing and control includes electric and magnetic field sensors; radio-wave frequency sensors; optical-, electrooptic, and infrared sensors; radars; lasers; location/navigation sensors; seismic and pressure-wave sensors; environmental parameter sensors (e.g., wind, humidity, heat); and biochemical national security–oriented sensors. Today’s sensors can be described as ‘‘smart’’ inexpensive devices equipped with multiple onboard sensing elements; they are low-cost low-power untethered multifunctional nodes that are logically homed to a central sink node.

Sensor devices, or wireless nodes (WNs), are also (sometimes) called motes. A stated commercial goal is to develop complete microelectro-mechanical systems (MEMSs)–based sensor systems at a volume of 1 mm^3. Sensors are internetworked via a series of multihop short-distance low-power wire-less links (particularly within a defined sensor field); they typically utilize the Internet or some other network for long-haul delivery of information to a point (or points) of final data aggregation and analysis. In general, within the sensor field, WSNs employ contention-oriented random-access channel sharing and transmission techniques that are now incorporated in the IEEE 802 family of standards; indeed, these techniques were originally developed in the late 1960s and 1970s expressly for wireless (not cabled) environments and for large sets of dispersed nodes with limited channel-management intelligence. However, other channel-management techniques are also available.

Sensors are typically deployed in a high-density manner and in large quantities: A WSN consists of densely distributed nodes that support sensing, signal processing, embedded computing, and connectivity; sensors are logically linked by self-organizing means (sensors that are deployed in short-hop point-to-point master–slave pair arrangements are also of interest). WNs typically transmit information to collecting (monitoring) stations that aggregate some or all of the information. WSNs have unique characteristics, such as, but not limited to, power constraints and limited battery life for the WNs, redundant data acquisition, low duty cycle, and, many-to-one flows. Consequently, new design methodologies are needed across a set of disciplines including, but not limited to, information trans-port, network and operational management, confidentiality, integrity, availability, and, in-network/local processing. In some cases it is challenging to collect (extract) data from WNs because connectivity to and from the

WNs may be intermittent due to a low-battery status (e.g., if these are dependent on sunlight to recharge) or other WN malfunction. Furthermore, a lightweight protocol stack is desired. Often, a very large number of client units (say 64k or more) need to be supported by the system and by the addressing apparatus.

Sensors span several orders of magnitude in physical size; they (or, at least some of their components) range from nanoscopic-scale devices to mesoscopic-scale devices at one end, and from microscopic-scale devices to macroscopic-scale devices at the other end. Nanoscopic (also known as nanoscale) refers to objects or devices on the order of 1 to 100 nm in diameter; mesoscopic scale refers to objects between 100 and 10,000 nm in diameter; the microscopic scale ranges from 10 to 1000 mm, and the macroscopic scale is at the millimeter-to-meter range. At the low end of the scale, one finds, among others, biological sensors, small passive microsensors (such as Smart Dust), and ‘‘lab-on-a-chip’’ assemblies. At the other end of the scale one finds platforms such as, but not limited to, identity tags, toll collection devices, controllable weather data collection sensors, bioterror-ism sensors, radars, and undersea submarine traffic sensors based on sonars. Some refer to the latest generation of sensors, especially the miniaturized sensors that are directly embedded in some physical infrastructure, as microsensors. A sensor network supports any type of generic sensor; more narrowly, networked micro-sensors are a subset of the general family of sensor networks. Microsensors with onboard processing and wireless interfaces can be utilized to study and monitor a variety of phenomena and environments at close proximity.

Sensors can be simple point elements or can be multipoint detection arrays. Typically, nodes are equipped with one or more application-specific sensors and with on-node signal processing capabilities for extraction and manipulation (pre-processing) of physical environment information. Embedded network sensing refers to the synergistic incorporation of microsensors in structures or environments; embedded sensing enables spatially and temporally dense monitoring of the system under consideration (e.g., an environment, a building, a battlefield). Sensors may be passive and/or be self-powered; farther down the power-consumption chain, some sensors may require relatively low power from a battery or line feed . At the high end of the power-consumption chain, some sensors may require very high power feeds (e.g., for radars).

Sensors facilitate the instrumenting and controlling of factories, offices, homes, vehicles, cities, and the ambiance, especially as commercial off-the-shelf technology becomes available. With sensor network technology (specifically, with embedded networked sensing), ships, aircraft, and buildings can ‘‘self-detect’’ structural faults (e.g., fatigue-induced cracks). Places of public assembly can be instrumented to detect airborne agents such as toxins and to trace the source of the contamination should any be present (this can also be done for ground and underground situations). Earthquake-oriented sensors in buildings can locate potential survivors and can help assess structural damage; tsunami-alerting sensors are useful for nations with extensive coastlines. Sensors also find extensive applicability on the battlefield for reconnaissance and surveillance.

In the next few years, advances in the areas of sensor design and materials that have taken place in the recent past will lead, almost assuredly, to significant reductions in the size, weight, power consumption, and cost of sensors and sensor arrays; these advances will also affect an increase in their spatial and temporal resolution, along with improved measuring accuracy.

Implementations of WSNs have to address a set of technical challenges; how-ever, the move toward standardization will, in due course, minimize a number of these challenges by addressing the issues once and then result in off-the-shelf chip-sets and components. A current research and development (R&D) challenge is to develop low-power communication with low-cost on-node processing and self-organizing connectivity/protocols; another critical challenge is the need for extended temporal operation of the sensing node despite a (typically) limited power supply (and/or battery life). In particular, the architecture of the radio, including the use of low-power circuitry, must be properly selected. In practical terms this implies low power consumption for transmission over low-bandwidth channels and low-power-consumption logic to pre-process and/or compress data. Energy-efficient wireless communications systems are being sought and are typical of WSNs. Low power consumption is a key factor in ensuring long operating hori-zons for non-power-fed systems (some systems can indeed be power-fed and/or rely on other power sources). Power efficiency in WSNs is generally accomplished in three ways:

- Low-duty-cycle operation.
- Local/in-network processing to reduce data volume (and hence transmission time).
- Multihop networking reduces the requirement for long-range transmission since signal path loss is an inverse exponent with range or distance. Each node in the sensor network can act as a repeater, thereby reducing the link range coverage required and, in turn, the transmission power.

Conventional wireless networks are generally designed with link ranges on the order of tens, hundreds, or thousands of miles. The reduced link range and the com-pressed data payload in WSNs result in characteristic link budgets that differ from those of conventional systems. However, the power restrictions, along with the desire for low node cost, give rise to what developers call ‘‘profound design challenges’’. Cooperative signal processing between nodes in proximity may enhance sensitivity and specificity to environmental event detection. New CMOS (complementary metal-oxide semiconductor) chipsets optimized for WSNs are the key to commercialization success and are, in fact, being developed.

Category 1 WSNs (C1WSNs): almost invariably mesh-based systems with multihop radio connectivity among or between WNs, utilizing dynamic routing in both the wireless and wireline portions of the network. Military-theater systems typically belong to this category.

Category 2 WSNs (C2WSNs): point-to-point or multipoint-to-point (star-based) systems generally with single-hop radio connectivity to WNs, utilizing static routing over the wireless network; typically, there will be only one route from the WNs to the companion terrestrial or wireline forwarding node (WNs are pendent nodes). Residential control systems typically belong to this category.

C1WSNs support highly distributed high-node-count applications (e.g., environ-mental monitoring, national security systems); C2WSNs typically support con-fined short-range spaces such as a home, a factory, a building, or the human body. C1WSNs are different in scope and/or reach from evolving wireless C2WSN technology for short-range low-data-rate wireless applications such as RFID (radio-frequency identification) systems, light switches, fire and smoke detectors, thermostats, and, home appliances. C1WSNs tend to deal with large-scale multipoint-to-point systems with massive data flows, whereas C2WSNs tend to focus on short-range point-to-point, source-to-sink applications with uniquely defined transaction-based data flows.

For a number of years, vendors have made use of proprietary technology for collecting performance data from devices. In the early 2000s, sensor device sup-pliers were researching ways of introducing standardization. WNs typically trans-mit small volumes of simple data (e.g., ‘‘Is the temperature at the set level or lower?’’). For within-building applications, designers ruled out Wi-Fi (wireless fidelity, IEEE 802.11b) standards for sensors as being too complex and supporting more bandwidth than is actually needed for typical sensors. Infrared systems require line of sight, which is not always achievable; Bluetooth (IEEE 802.15.1) technology was at first considered a possibility, but it was soon deemed too com-plex and expensive. This opened the door for a new standard IEEE 802.15.4 along with ZigBee (more specifically, ZigBee comprises the software layers above the newly adopted IEEE 802.15.4 standard and supports a plethora of applications). C2WSNs have lower layers of the communication protocol stack (Physical and Media Access Control), which are comparable to that of a personal area network (PAN), defined in the recently developed IEEE 802.15 standard: hence, the utilization of these IEEE standards for C2WSNs. IEEE 802.15.4 operates in the 2.4-GHz industrial, scientific, and medical (ISM) radio band and supports data transmission at rates up to 250 kbps at ranges from 30 to 200 ft. ZigBee/IEEE 802.15.4 is designed to complement wireless technologies such as Bluetooth, Wi-Fi, and ultra-wideband (UWB), and is targeted at commercial point-to-point sensing applications where cabled connections are not possible and where ultralow power and low cost are requirements.

With the emergence of the ZigBee/IEEE 802.15.4 standard, systems are expected to transition to standards-based approaches, allowing sensors to transfer information in a standardized manner. C2WSNs (and C1WSN, for that matter) that operate outside a building and over a broad geographic area may make use of any number of other standardized radio technologies. The (low-data-rate) C2WSN market is expected to grow significantly in the near future: The volume of low-data-rate wireless devices is forecast to be three times the size of Wi-Fi by the turn of the decade, due to the expected deployment of the systems based on the ZigBee/IEEE 802.15.4 standard (industry observers expect the number of ZigBee-compliant nodes to increase from less than 1 million in 2005 to 100 million in 2008).

There is also considerable research in the area of mobile ad hoc networks (MANETs). WSNs are similar to MANETs in some ways; for example, both involve multihop communications. However, the applications and technical requirements for the two systems are significantly different in several respects:

- The typical mode of communication in WSN is from multiple data sources to a data recipient or sink (somewhat like a reverse multicast) rather than communication between a pair of nodes. In other words, sensor nodes use primarily multicast or broadcast communication, whereas most MANETs are based on point-to-point communications.
- In most scenarios (applications) the sensors themselves are not mobile (although the sensed phenomena may be); this implies that the dynamics in the two types of networks are different.
- Because the data being collected by multiple sensors are based on common phenomena, there is potentially a degree of redundancy in the data being communicated by the various sources in WSNs; this is not generally the case in MANETs.
- Because the data being collected by multiple sensors are based on common phenomena, there is potentially some dependency on traffic event generation in WSNs, such that some typical random-access protocol models may be inadequate at the queueing-analysis level; this is generally not the case in MANETs.
- A critical resource constraint in WSNs is energy; this is not always the case in MANETs, where the communicating devices handled by human users can be replaced or recharged relatively often. The scale of WSNs (especially, C1WSNs) and the necessity for unattended operation for periods reaching weeks or months implies that energy resources have to be managed very judiciously. This, in turn, precludes high-data-rate transmission.
- The number of sensor nodes in a sensor network can be several orders of magnitude higher than the nodes in a MANET.

1.3 Basic Sensor Network Architectural Elements

These elements and design principles need to be placed in the context of the C1WSN sensor network environment, which is characterized by many (some-times all) of the following factors: large sensor population (e.g., 64,000 or more client units need to be supported by the system and by the addressing apparatus), large streams of data, incomplete/uncertain data, high potential node failure; high potential link failure (interference), electrical power limitations, processing power limitations, multihop topology, lack of global knowledge about the network, and (often) limited administrative support for the network (C2WSNs have many of these same limitations, but not all). Sensor network developments rely on advances in sensing, communication, and computing (data-handling algorithms, hardware, and software). As noted, to manage scarce WSN resources adequately, routing protocols for WSNs need to be energy-aware. Data-centric routing and in-network processing are important concepts that are associated intrinsically with sensor networks. The end-to-end routing schemes that have been proposed in the literature for mobile ad hoc networks are not appropriate WSNs; data-centric technologies are needed that perform in-network aggregation of data to yield energy-efficient dissemination.

Sensor Types and Technology:

A sensor network is composed of a large number of sensor nodes that are densely deployed. To list just a few venues, sensor nodes may be deployed in an open space; on a battlefield in front of, or beyond, enemy lines; in the interior of industrial machinery; at the bottom of a body of water; in a biologically and/or chemically contaminated field; in a commercial building; in a home; or in or on a human body. A sensor node typically has embedded processing capabilities and onboard storage; the node can have one or more sensors operating in the acoustic, seismic, radio (radar), infrared, optical, magnetic, and chemical or biological domains. The node has communication interfaces, typically wireless links, to neighboring domains. The sensor node also often has location and positioning knowledge that is acquired through a global position-ing system (GPS) or local positioning algorithm. (Note, however, that GPS-based mechanisms may sometimes be too costly and/or the equipment may be too bulky.) Sensor nodes are scattered in a special domain called a sensor field. Each of the distributed sensor nodes typically has the capability to collect data, analyze them, and route them to a (designated) sink point. Figure 1.1 depicts a typical WSN arrangement. Although in many environments all WNs are assumed to have similar functionality, there are cases where one finds a heterogeneous environment in regard to the sensor functionality.

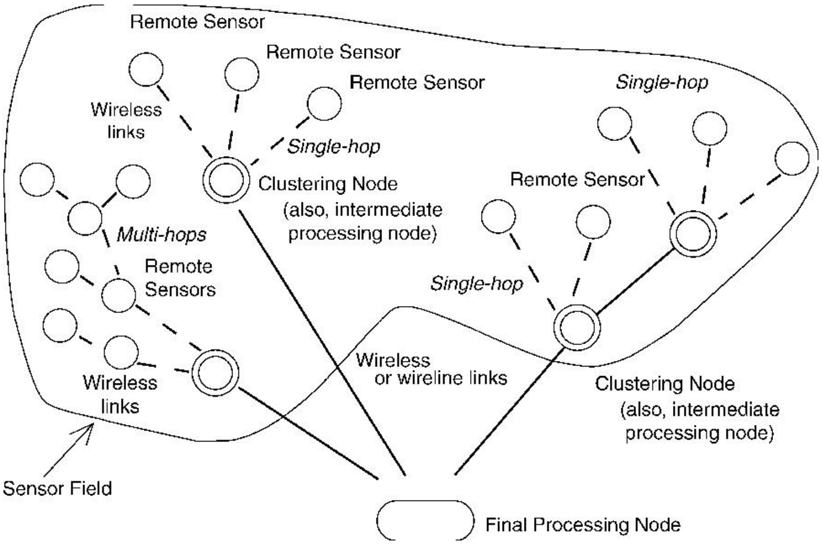


Figure 1.1 Typical sensor network arrangement

The following are important issues pertaining to WSNs (see also Table 1.1): sensor type; sensor placement; sensor power consumption, operating environment, computational/sensing capabilities and signal processing, connectivity, and telemetry or control of remote devices. It is critical to note in this context that node location and fine-grained time (stamping) are essential for proper operation of a sensor network; this is almost the opposite of the prevalent Internet architecture, where server location is immaterial to a large degree and where latency is often not a key consideration or explicit design objective. In sensor networks, fine-grained time synchronization and localization are needed to detect events of interest in the environment under observation. Location needs to be tracked both in local three-dimensional space (e.g., On what floor and in which quadrant is the smoke detected? What is the temperature of the atmosphere at height h?) and over a broader topography, to assess detection levels across a related set (array) of sensors (e.g., What is the wind direction for wind containing contaminated particles at mile-post i, i þ 1, i þ 2, etc., along a busy highway?). Localization is used for functionality such as beamforming for localization of target and events, geographical forwarding, and geographical addressing.

Embedded sensor networks are predicated on three supporting components: embed-ding, networking, and sensing. Embedding implies the incorporation of numerous distributed devices to monitor the physical world and interact with it; the devices are untethered nodes of small form factors that are equipped with a control and communication subsystem. Spatially- and temporally-dense arrangements are com-mon. Networking implies the concept of physical and logical connectivity.

TABLE 1.1 Categorization of Issues Related to Sensors and Their Communication/ Computing Architecture

Sensors	Size: Small [e.g., nanoscale electromechanical systems (MEMS)], medium [e.g., microscale electromechanical systems (MEMS)], and large (e.g., radars, satellites): cubic centimeters to cubic decimeters Mobility: stationary (e.g., seismic sensors), mobile (e.g., on robot vehicles) Type: passive (e.g., acoustic, seismic, video, infrared, magnetic) or active (e.g., radar, ladar)
Operating environment	Monitoring requirement: distributed (e.g., environmental monitoring) or localized (e.g., target tracking) Number of sites: sometimes small, but usually large (especially for C1WSNs) Spatial coverage: dense, spars: C1WSN: low-range multihop or C2WSN: low-range single-hop (point-to-point) Deployment: fixed and planned (e.g., factory networks) or ad hoc (e.g., air-dropped) Environment: benign (factory floor) or adverse (battlefield) Nature: cooperative (e.g., air traffic control) or noncooperative (e.g., military targets) Composition: homogeneous (same types of sensors) or heterogeneous (different types of sensors) Energy availability: constrained (e.g., in small sensors) or unconstrained (e.g., in large sensors)
Communication	Networking: wired (on occasion) or wireless (more common) Bandwidth: high (on occasion) or low (more typical)
Processing architecture	Centralized (all data sent to central site), distributed or in-network (located at sensor or other sides), or hybrid

Logical connectivity has the goal of supporting coordination and other high-level tasks; physical connectivity is typically supported over a wireless radio link. Sensing implies the presence of these capabilities in a tightly coupled environment, typically for the measurement of physical-world parameters. Some of the characteristic features of sensor networks include the following:

- Sensor nodes are densely deployed.
- Sensor nodes are prone to failures.
- The topology of a sensor network changes very frequently.
- Sensor nodes are limited in power, computational capacities, and memory.
- Sensor nodes may not have global identification because of the large amount of overhead and the large number of sensors.

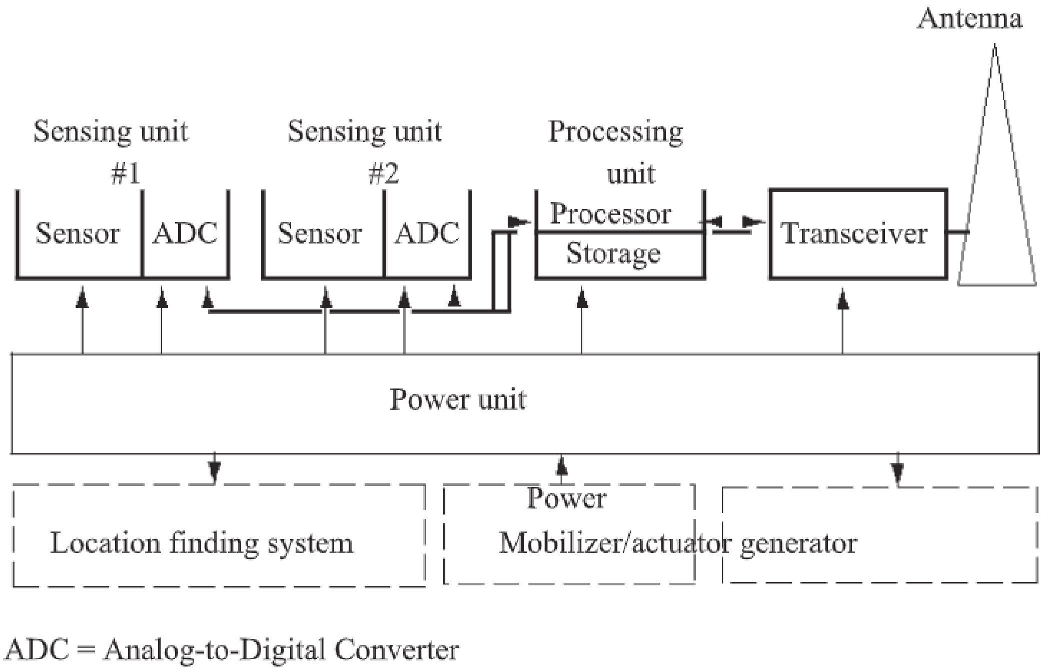


Figure 1.2 Typical sensing node

Power consumption is often an issue that needs to be taken into account as a design constraint. In most instances, communication circuitry and antennas are the primary elements that draw most of the energy. Sensors are either passive or active devices. Passive sensors in element form include seismic-, acoustic-, strain-, humidity-, and temperature-measuring devices. Passive sensors in array form include optical- [visible, infrared 1 micron (mm), infrared 10 mm], and biochemical-measuring devices. Passive sensors tend to be low-energy devices. Active sensors include radar and sonar; these tend to be high-energy systems. The trend is toward VLSI (very large scale integration), integrated optoelectronics, and nanotechnology; work is under way in earnest in the biochemical arena. The components of a (remote) sensing node include (see Figure 1.2) the following:

- A sensing and actuation unit (single element or array)
- A processing unit
- A communication unit
- A power unit
- Other application-dependent units

Figure 1.3 depicts an example on an (ultra)miniature sensor. In addition to (embedded) sensing there is a desire to build, deploy, and manage unattended or untethered embedded control and actuation systems, sometimes called control networks. Such a control system acts on the environment either in a self-autonomous manner or under the telemetry of a remote or centralized node. Key applications require more than just sensing: They need control and actuation. Control refers to some ‘‘minor’’ activity internal to the sensor (e.g., zoom, add an optical filter, rotate an antenna); actuation refers to a ‘‘major’’ activity external to the sensor itself (e.g., open a valve, emit some fluid into the environment, engage a motor to relocate somewhere else). Applications requiring control and/or actuation include transportation, high-tech agriculture, medical monitoring, drug delivery, battlefield interventions, and so on. In addition to standard concerns (e.g., reliability, security), actuation systems also have to take into account factors such as safety.

Software (Operating Systems and Middleware) To support the node operation, it is important to have open-source operating systems designed specifically for WSNs. Such operating systems typically utilize a component-based architecture that enables rapid implementation and innovation while minimizing code size as required by the memory constraints endemic in sensor networks. TinyOS is one such example of a de facto standard, but not the only one. TinyOS’s component library includes network protocols, distributed services, sensor drivers, and data acquisition tools; these can be used as-is or be further refined for a specific application. TinyOS’s event-driven execution model enables fine-grained power management, yet allows the scheduling flexibility made necessary by the un-predictable nature of wireless communication and physical world interfaces. TinyOS has already been ported to over a dozen platforms and numerous sensor boards. A wide community uses TinyOS in simulation to develop and test various algorithms and protocols, and numerous groups are actively contributing code to establish standard interoperable network services.

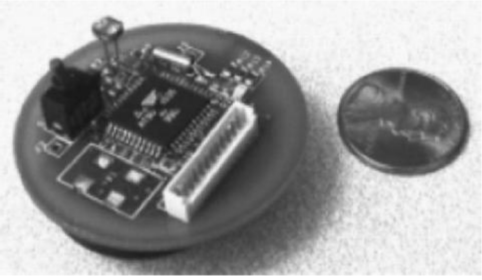


Figure 1.3 Miniature sensor: the MacroMote, developed at UC–Berkeley. (Courtesy of UC–Berkeley.)

Standards for Transport Protocols The goal of WSN engineers is to develop a cost-effective standards-based wireless networking solution that supports low-to-medium data rates, has low power consumption, and guarantees security and reliability. The position of sensor nodes does not have be predetermined, allowing random deployment in inaccessible terrains or dynamic situations; however, this also means that sensor network protocols and algorithms must possess self-organizing capabilities. For military and/or national security applications, sensor devices must be amenable to rapid deployment, the deployment must be supportable in an ad hoc fashion, and the environment is expected to be highly dynamic.

Researchers have developed many new protocols specifically designed for WSNs, where energy awareness is an essential consideration; focus has been given to the routing protocols, since they might differ from traditional networks (depending on the application and network architecture). Networking is an important architectural component of sensor networks, and standards play a major role in this context.

Figure 1.4 depicts a generic protocol stack model that can be utilized to describe the communications apparatus (also see Table 1.2). Table 1.3 shows some typical lower-layer protocols that are in principle applicable to WSNs; overall, a lightweight protocol stack is sought for WSNs. Issues here relate to the following:

- Task management plane
- Mobility management plane
- Power management plane
- Upper layers (communications)
- Transport layer
- Network layer
- Data link layer
- Physical layer

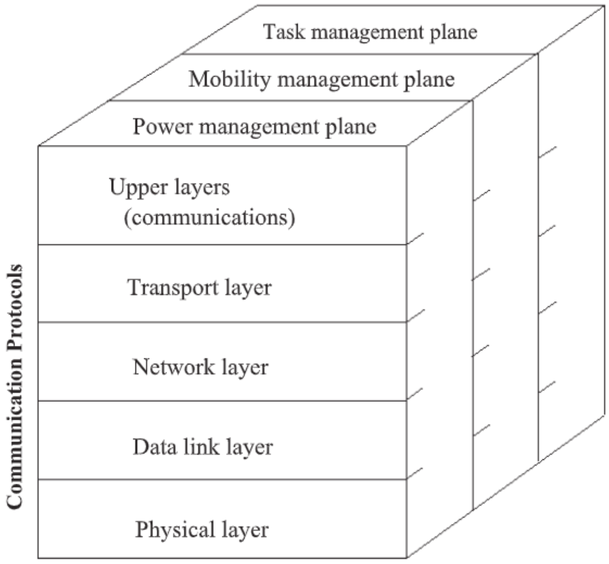


Figure 1.4 Generic protocol stack for sensor networks

TABLE 1.2 Possible WSN Protocol Stack

Upper layers	In-network applications, including application processing, data aggregation, external querying query processing, and external database
Layer 4	Transport, including data dissemination and accumulation, caching, and storage
Layer 3	Networking, including adaptive topology management and topological routing
Layer 2	Link layer (contention): channel sharing (MAC), timing, and locality
Layer 1	Physical medium: communication channel, sensing, actuation, and signal processing

- Physical connectivity and coverage: How can one interconnect dispersed sensors in a cost-effective and reliable manner, and what medium should be used (e.g., wireless channels)?
- Link characteristics and capacity, along with data compression.
- Networking security and communications reliability (including naturally occurring phenomena such as noise impairments, and malicious issues such as attacks, interference, and penetration)
- Physical-, link-, network-, and transport-layer protocols, with an eye to reliable transport, congestion detection and avoidance, and scalable and robust communication.
- Communication mechanisms in what could be an environment with highly correlated and time-dependent arrivals (where many of the queueing assumptions used for system modeling could break down.

Although sensor electronics are becoming inexpensive, observers see the lack of networking standards as a potentially retardant factor in the commercial deployment of sensor networks. Because today there are still numerous proprietary network protocols, manufacturers have created vendor-specific and consequently, expensive products that will not work with products from other manufacturers.

Security:

Security deals with confidentiality (encryption), integrity (e.g., identity management, digital signatures), and availability (protection from denial of service).

Network Design Issues in sensor networks:

Issues relate to reliable transport (possibly including encryption), bandwidth-and power-limited transmission, data-centric routing, in-network processing, and self-configuration. Design factors include operating environment and hardware constraints such as transmission media, radio-frequency integrated circuits, power constraints communications network interfaces; and network architecture and protocols, including network topology and fault tolerance, scalability, self-organization, and mobility.

Sensor networks are generally self-configuring systems. The goal is to be able to adapt to unpredictable situations and states. Static or semi dynamic topologies lend themselves easily to preconfiguration, but highly dynamic environments require self-configuration. In designing a sensor network, one is naturally looking for acceptable accuracy of information (even in the presence of failed nodes and/or links, and possibly conflicting or partial data); low network and computing latency; and optimal resource use (specifically, power and bandwidth). Work is under way to develop techniques that can be employed to deal with these and other pertinent issues, such as how to represent sensor data, how to structure sensor queries, how to adapt to changing node or network conditions, and how to manage a large network environment where nodes have limited network management functionality.

Sensor networks often employ data processing directly in the network itself. Part of the motivation is the potential for large pools of data being generated by the sensors. By utilizing computation close to the source of the data for trending, aver-aging, maxima and minima, or out-of-range activities, one is able to reduce the communication throughput that would otherwise be needed. Intrinsic to this is the development of localized algorithms that support global goals; it follows that forms of collaborative signal processing are desired.

Researchers are looking at new system architectures to manage interactions. Currently, many sensor systems suffer from being one-of-a-kind with piecemeal design approaches. This predicament leads to suboptimal economics, longevity, interoperability, scalability, and robustness. Standards will go a long way to address a number of these concerns. A number of researchers [1.5] are taking the position that the traditional approach and/or protocol suite is not adequate for embedded, energy-constrained, untethered, small-form-factor, unattended systems, because these systems cannot tolerate the communication overhead associated with the rout-ing and naming intrinsic in the Internet suite of protocols. Proponents are making a pitch for special-purpose system functions in place of the general-purpose Internet functionality designed for elastic applications. In effect, resource constraints require a more streamlined and more tightly integrated communications layer than that possible with a TCP–IP or ISO (International Organization for Standardization) stack.

1.4 Brief Historical Survey of Sensor Networks

The history of sensor networks spans four phases, described briefly below.

Phase 1: Cold-War Era Military Sensor Networks During the cold war, extensive acoustic networks were developed in the United States for submarine surveillance; some of these sensors are still being used by the National Oceanographic and Atmospheric Administration (NOAA) to monitor seismic activity in the ocean. Also, networks of air defense radars were deployed to cover North America; to handle this, a battery of Airborne Warning and Control System (AWACS) planes operated as sensors.

Phase 2: Defense Advanced Research Projects Agency Initiatives The major impetus to research on sensor networks took place in the early 1980s with programs sponsored by the Defense Advanced Research Projects Agency (DARPA). The dis-tributed sensor networks (DSN) work aimed at determining if newly developed TCP–IP protocols and ARPAnet’s (the predecessor of the Internet) approach to communication could be used in the context of sensor networks. DSN postulated the existence of many low-cost spatially distributed sensing nodes that were designed to operate in a collaborative manner, yet be autonomous; the goal was for the net-work to route information to the node that can best utilize the information. The DSN program focused on distributed computing, signal processing, and tracking. Technology elements included acoustic sensors, high-level communication protocols, processing and algorithm calculations (e.g., self-location algorithms for sensors), and distributed software (dynamically modifiable distributed systems and language design). Researchers at Carnegie

Mellon University focused on providing a network operating system for flexible transparent access to distributed resources, and researchers at the Massachusetts Institute of Technology focused on knowledge-based signal-processing techniques. Testbeds were developed for tracking multiple targets in a distributed environment; all components in the testbed network were custom built. Ongoing work in the 1980s resulted in the development of a multiple-hypothesis tracking algorithm to address difficult problems involving high target density, missing detections, and false alarms; multiple-hypothesis tracking is now a standard approach to challenging tracking problems.

Phase 3: Military Applications Developed or Deployed in the 1980s and 1990s (These can properly be called first-generation commercial products.) Based on the results generated by the DARPA–DSN research and the testbeds developed, military planners set out in the 1980s and 1990s to adopt sensor network technology, making it a key component of network-centric warfare. An effort was made at the time to start employing commercial off the shelf (COTS) technology and common network interfaces, thereby reducing cost and development time. In traditional warfare environments each platforms ‘‘owns’’ its weapons in a fairly autonomous manner (distinct platforms operate independently). In network-centric warfare, weapon systems are not (necessarily) tightly affiliated with a specific plat-form; instead, through the use of distributed sensors, the weapon systems and plat-forms collaborate with each other over a sensor network, and information is sent to the appropriate node. Sensor networks can improve detection and tracking performance through multiple observations, geometric and phenomenological diversity, extended detection range, and faster response time. An example of network-centric warfare include the cooperative engagement capability, a system that consists of multiple radars collecting data on air targets. Other sensor networks in the military arena include acoustic sensor arrays for antisubmarine warfare, such as the fixed distributed system and the advanced deployable system, and autonomous ground sensor systems such as the remote battlefield sensor system and the tactical remote sensor system.

Phase 4: Present-Day Sensor Network Research (These can properly be called second-generation commercial products.) Advances in computing and communication that have taken place in the late 1990s and early 2000s have resulted in a new generation of sensor network technology. Evolving sensor networks represent a significant improvement over traditional sensors. Inexpensive compact sensors based on a number of high-density technologies, including MEMS and (in the next few years) nanoscale electromechanical systems (NEMS), are appearing. Standardization is a key to wide-scale deployment of any technology, including WSN (e.g., Internet–Web, MPEG-4 digital video, wireless cellular, VoIP). Advances in IEEE 802.11a/b/g-based wireless networking and other wireless systems such as Bluetooth, ZigBee,[9] and WiMax are now facilitating reliable and ubiquitous connectivity. Inexpensive processors that have low power-consumption requirements make possible the deployment of sensors for a plethora of applications. Commercially-focused efforts are now directed at defining mesh, peer-to-peer, and cluster-tree network topologies with data security features and interoperable application profiles. Table 1.4 summarizes these generations of commercial pro-ducts and alludes to a next-generation (third-generation) set of products.

1.5 Ad hoc Networks:

An ad hoc network is one that is spontaneously formed when devices connect and communicate with each other. The term ad hoc is a Latin word that literally means "for this," implying improvised or impromptu.

Ad hoc networks are mostly wireless local area networks (LANs). The devices communicate with each other directly instead of relying on a base station or access points as in wireless LANs for data transfer co-ordination (fig 1.5). Each device participates in routing activity, by determining the route using the routing algorithm and forwarding data to other devices via this route.

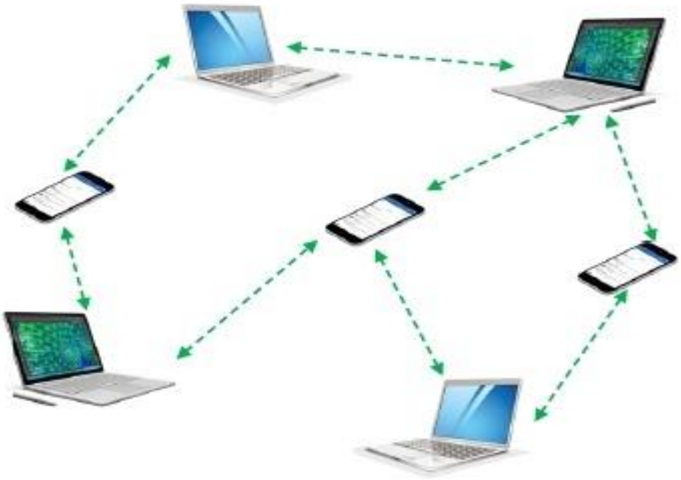
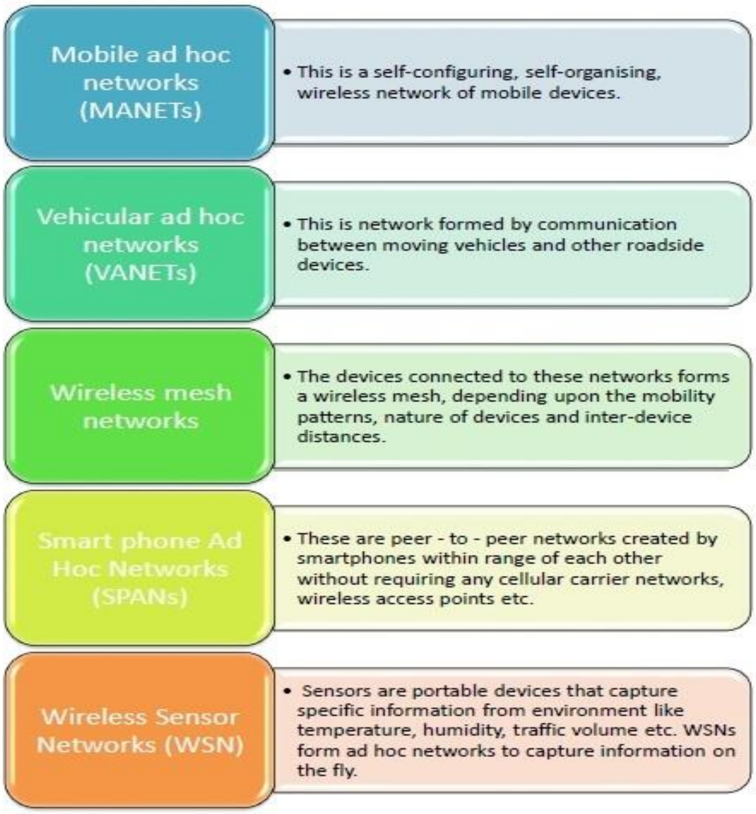


Figure 1.5 Ad hoc Networks

- *Classifications of Ad Hoc Networks*

Ad hoc networks can be classified into several types depending upon the nature of their applications. The most prominent ad hoc networks that are commonly incorporated are illustrated in the diagram below (fig 1.6).



Mobile ad hoc networks (MANETs)
- This is a self-configuring, self-organising, wireless network of mobile devices.

Vehicular ad hoc networks (VANETs)
- This is network formed by communication between moving vehicles and other roadside devices.

Wireless mesh networks
- The devices connected to these networks forms a wireless mesh, depending upon the mobility patterns, nature of devices and inter-device distances.

Smart phone Ad Hoc Networks (SPANs)
- These are peer - to - peer networks created by smartphones within range of each other without requiring any cellular carrier networks, wireless access points etc.

Wireless Sensor Networks (WSN)
- Sensors are portable devices that capture specific information from environment like temperature, humidity, traffic volume etc. WSNs form ad hoc networks to capture information on the fly.

Figure 1.6 Types of Ad hoc Networks

1.6 Applications of Wireless Sensor Networks

Various applications of WSNs are currently either already in mature use or still in infant stages of development. WSN applications are classified according to the nature of their use into various categories are: home control, medical applications, sensor and robots, habitat monitoring, wildfire instrumentation, civil and environmental engineering applications, highway monitoring military applications.

- **Home Control**

Home control applications provide control, conservation, convenience, and safety, as follows (see Figure 1.7):

- Sensing applications facilitate flexible management of lighting, heating, and cooling systems from anywhere in the home.
- Sensing applications automate control of multiple home systems to improve conservation, convenience, and safety.
- Sensing applications capture highly detailed electric, water, and gas utility usage data.

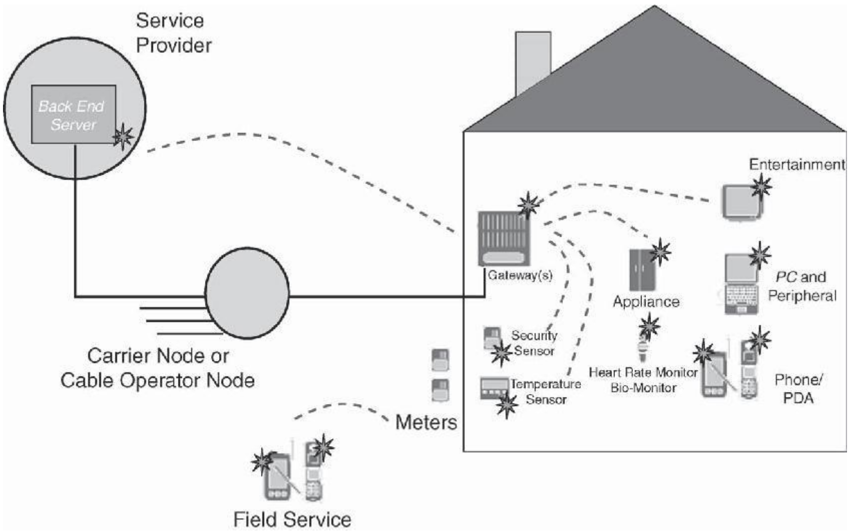


Figure 1.7 Home control applications

- Sensing applications embed intelligence to optimize consumption of natural resources.
- Sensing applications enable the installation, upgrading, and networking of a home control system without wires.
- Sensing applications enable one to configure and run multiple systems from a single remote control.
- Sensing applications support the straightforward installation of wireless sensors to monitor a wide variety of conditions.
- Sensing applications facilitate the reception of automatic notification upon detection of unusual events.

Body-worn medical sensors (e.g., heartbeat sensors) are also emerging. These are battery-operated devices with network beacons occurring either every few seconds that could be worn by home-resident elderly or people with other medical conditions. These sensors have two ongoing processes: heartbeat time logging and transmission of heart rate and other information (instantaneous and average heart rate, body temperature, and battery voltage).

- **Medical Applications**

A number of hospitals and medical centers are exploring applications of WSN technology to a range of medical applications, including pre-hospital and in-hospital emergency care, disaster response, and stroke patient rehabilitation. WSNs have the potential to affect the delivery and study of resuscitative care by allowing vital signs to be collected and integrated automatically into the patient care record and used for real-time triage, correlation with hospital records, and long-term observation. WSNs permit home monitoring for chronic and elderly patients, facilitating long-term care and trend analysis; this in turn can sometimes reduce the length of hospital stays. WSNs also permit collection of long-term medical information that populates databases of clinical data; this enables longitudinal studies across populations and allows physicians to study the effects of medical intervention programs. These WSNs tend to be of the C2WSN category.

Vital sign data, such as pulse oximetry, are poorly integrated with pre-hospital and hospital-based patient care records. Harvard University and others have developed a small, wearable wireless pulse oximeter and two-lead electrocardiogram (EKG). These devices collect heart rate, oxygen saturation, and EKG data and relay it over a short-range (100-m) wireless network to any number of receiving devices, including PDAs, laptops, or ambulance-based terminals. The data can be displayed in real time and integrated into the developing pre-hospital patient care record. The sensor devices themselves can be programmed to process the vital sign data, for example, to raise an alert condition when vital signs fall outside normal parameters; any adverse change in patient status can then be signaled to a nearby EMT or paramedic.

In collaboration with the Motion Analysis Laboratory at the Spaulding Rehabilitation Hospital, Harvard University has also developed a tiny wearable device for monitoring the limb movements and muscle activity of stroke patients during rehabilitation exercise. These devices, consisting of three-axis accelerometer, gyro-scope, and electromyogram sensors, allow researchers to capture a rich data set of motion data for studying the effect of various rehabilitation exercises on this patient population.

In addition to the hardware platform, Harvard University developed a scalable software infrastructure called CodeBlue, for wireless medical devices. CodeBlue is designed to provide routing, naming, discovery, and security for wireless medical sensors, PDAs, PCs, and other devices that may be used to monitor and treat patients in a number of medical settings (see Figure 1.8). CodeBlue is designed to scale across a different network densities, ranging from sparse clinic and hospital deployments to very dense ad hoc deployments at a mass casualty site. Part of the CodeBlue system includes a system for tracking the location of individual patient devices indoors and outdoors using radio signal information.

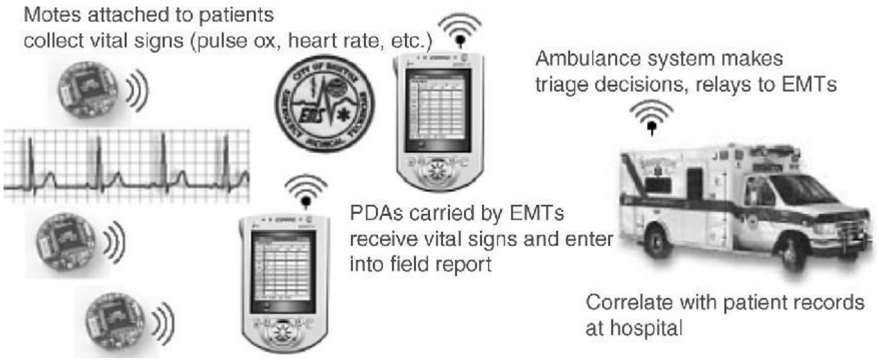

Motes attached to patients
collect vital signs (pulse ox, heart rate, etc.)
Ambulance system makes
triage decisions, relays to EMTs
PDAs carried by EMTs
receive vital signs and enter
into field report
Correlate with patient records
at hospital

Figure 1.8 Use of CodeBlue for emergency response: PDA displaying real-time vital signs of multiple patients. (Courtesy of Harvard University and Boston University School of Medicine.)

The ability to deploy WSNs that interconnect in an effective manner with unattended WNs is expected to have a significant bearing on the efficacy of military and civil applications such as, but not limited to, combat field surveillance, security, and disaster management. These WSNs process data assembled from multi-ple sensors in order to monitor events in an area of interest. For example, in a disaster management event, a large number of sensors can be dropped by a helicopter; net-working these sensors can assist rescue operations by locating survivors, identifying risky areas, and making the rescue crew more aware of the overall situation and improving overall safety. Some WSNs have camera-enabled sensors; one can have aboveground full-color visible-light cameras as well as belowground infrared cameras. The use of WSNs will limit the need for military personnel involvement in dangerous reconnaissance missions. Security applications include intrusion detection and criminal hunting. Some examples of WSN applications are:

- Military sensor networks to detect and gain as much information as possible about enemy movements, explosions, and other phenomena of interest
- Law enforcement and national security applications for inimical agent tracking or nefarious substance monitoring (e.g., see Figure 1.9)

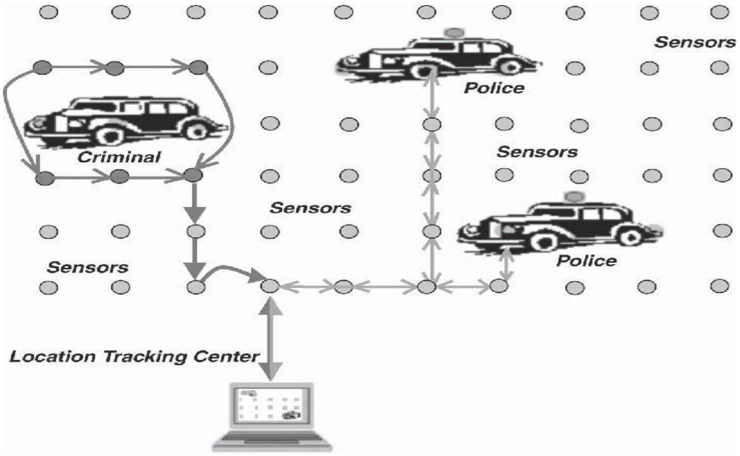


Figure 1.9 Law enforcement–national security application.

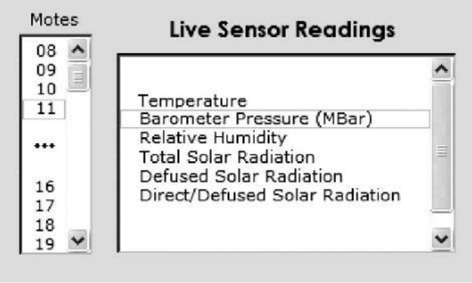


Figure 1.10 Typical real-time administrative access to distributed motes.

- Sensor networks to detect and characterize chemical, biological, radiological, nuclear, and explosive (CBRNE) attacks and material
- Sensor networks to detect and monitor environmental changes in plains, forests, oceans, and so on
- Wireless traffic sensor networks to monitor vehicle traffic on highways or in congested parts of a city
- Wireless surveillance sensor networks for providing security in shopping malls, parking garages, and other facilities
- Wireless parking lot sensor networks to determine which spots are occupied and which are free
- Borders monitoring with sensors and satellite uplinks

Figure 1.10 depicts the typical real-time administrative access to distributed WNs (motes) in an open-space sensor field. Real-time monitoring and sensor inter-rogation is typically supported. A number of illustrative examples are described in the subsections that follow. These examples just scratch the surface of the plethora of possible applications.

- **Sensor and Robots**

Two technologies appear poised for a degree of convergence: mobile robotics and wireless sensor networks. Some researchers expect that mobile robotics will use WSNs to achieve ubiquitous computing environments. For example, Intel envisions mobile robots acting as gateways into wireless sensor networks, such as into the Smart Dust networks of wireless motes. These robots embody sensing, actuation, and basic (miniaturized) robotics functions. The field of mobile robotics deals with mechanical aspects (the wheels, motors, grasping arms, or physical layout) as well as with the logic aspects (the microprocessors, the software, and the telemetry). Two questions of interest are :

Can a mobile robot act as a gateway into a wireless sensor network?
Can sensor networks take advantage of a robot’s mobility and intelligence?

To affect this convergence, inexpensive standards-based hardware, open-source operating systems, and off-the-shelf connectivity modules are required (e.g., Intel XScale microprocessors and Intel Centrino mobile technology).

One major issue with a mobile robot acting as a gateway is the communication between the robot and the sensor network. Some propose that a sensor network can be equipped with IEEE 802.11 capabilities to bridge the gap between robotics and wireless networks. For example, Intel recently demonstrated how a few motes equipped with 802.11 wireless capabilities can be added to a sensor network to act as wireless hubs. Other motes in the network then utilize each other as links to reach the 802.11-equipped hubs; the hubs forward the data packets to the main 802.11-capable gateway, which is usually a PC or laptop. Using some motes as hubs reduces the number of hops that any one data packet has to make to reach the main gateway, and also reduces power consumption across the sensor network. As an example, Intel recently installed small sensors in a vineyard in Oregon to monitor microclimates. The sensors measured temperature, humidity, and other factors to monitor the growing cycle of the grapes, then transmitted the data from sensor to sensor until the data reached a gateway. At the gateway, the data were interpreted and used to help prevent frostbite, mold, and other agricultural problems.

Intel, Carnegie Mellon University, University of Southern California, University of Pennsylvania, Northwestern, Georgia Tech, NASA, DARPA (the Defense Advanced Research Projects Agency), and NIST (National Institute of Standards and Technology) are just some of the institutions researching this topic. The Robotics Engineering Task Force (RETF; modeled after the Internet Engineering Task Force) has the goal of enabling government and university researchers to work collaboratively to establish standard software protocols and interfaces for robotics systems. The most pressing issue for the RETF is developing standards for commanding and controlling mobile robots.

Other examples of WSN applications include preventive maintenance for equipment in a semiconductor manufacturing fab, and sensor networks for theme parks. Both applications leverage the concept of heterogeneous WSNs, and both solve important business problems in their domains. At Intel’s semiconductor fabs, thousands of sensors track vibrations coming from various pieces of equipment to determine if the machines are about to fail. There is an established science that enables managers to determine the particular signature that a well-functioning machine should have. Typically, employees in the fab must gather the sensor data manually from each node - a costly and time-consuming process that is carried out periodically, on a schedule determined by the expected failure rate of the equipment. Going forward, networking the sensors could make the process more efficient and cost-effective. Intel reportedly plans to make use of the mote technology to build an application that acquires data automatically; pro-totypes have already been built (see Figure 1.11). Intel is also exploring the deployment of heterogeneous sensor networks in theme parks. Such networks could be used for multiple purposes. One potential use is monitoring the quality of water in tanks (see Figure 1.12); currently, such monitoring is done manually; a WSN can make the process more accurate and efficient. Another potential use of the network is to provide Internet access to park visitors. Visitors can use the wireless network to reserve a space at a particular park attraction or to learn more about an exhibit. The wireless network could improve park management as well.

Sensors could track attendance at park exhibits and rides, and manage-ment could use the network to access office applications from various stations throughout the park.

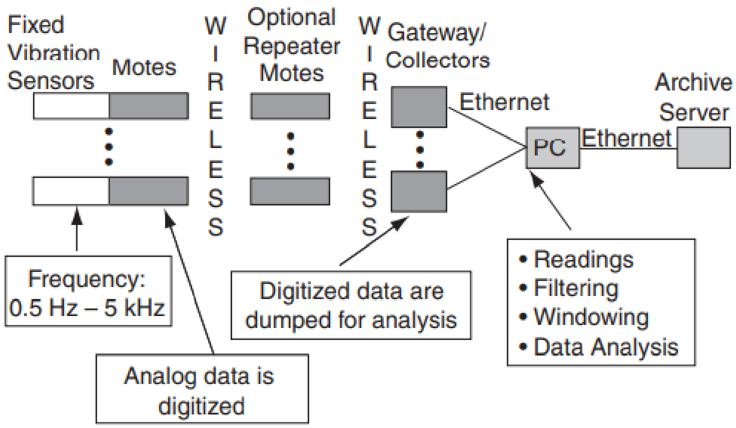


Figure 1.11 Intel fab environment with WSNs

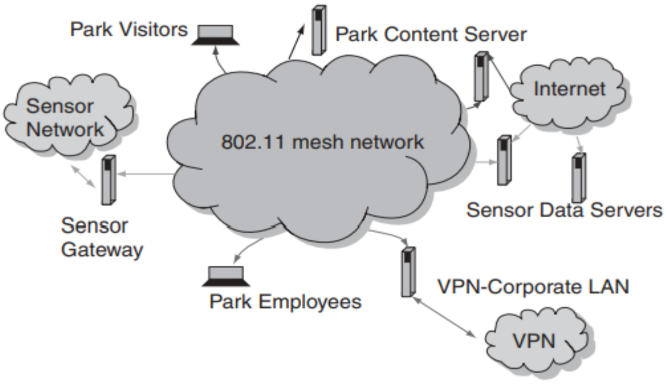


Figure 1.12 Themepark WSN example

- **Reconfigurable Sensor Networks**

Military applications require support for tactical and surveillance arrangements that employ reconfigurable sensor WNs that are capable of forming networks on the fly, assembling themselves without central control, and being deployed incrementally. Reconfigurable ‘‘smart’’ WNs are self-aware, self-configurable, and autonomous. Self-organizing WSNs utilize mechanisms that allow newly deployed WNs to establish connectivity (to build up a network topology) spontaneously. Also, these networks have mechanisms for managing WN mobility (if any), WN recon-figuration, and WN failure (if and when that happens).

- **Highway Monitoring**

Transportation (traffic flow) is a sector that is expected to benefit from increased monitoring and surveillance. A specific example follows. (Traffic in the United States is growing at three times the rate of population growth and causing an esti-mated $75 billion lost annually due to traffic congestion.) Traffic Pulse Technology is an example of a WSN developed by Traffic.com. The goal of this system (which uses stationary WNs; see Figure 2.13) is to collect data through a sensor network, process and store the data in a data center, and distribute those data through a variety of applications. Traffic Pulse is targeted for open-air environments; it provides real-time collection of data (e.g., to check temperature or monitor pollution levels). The system is installed along major highways; the digital sensor network gathers lane-by-lane data on travel speeds, lane occupancy, and vehicle counts. These basic data elements make it possible to calculate average speeds and travel times. The data are then transmitted to the data center for reformatting. The network monitors roadway conditions continuously on a 24/7 basis and provides updates to the data center in real time. The system collects key traffic information, including vehicle speeds, counts (volume), and roadway density, transmitting the data over a wireless network to a data center every 60 seconds.

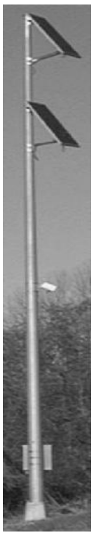


Figure 1.13 Typical highway traffic-sensing installation. (Courtesy of Traffic.com.)

In each major city, Traffic.com maintains a traffic pulse operations center that collects and reports on real-time event, construction, and incident data. This information supplements the data collected from the sensors. Each center produces the information through a wide range of methods: video, aircraft, mobile units, and monitoring of emergency and maintenance services frequencies. Applications include the following:

- Private traffic information providers in the United States: The company’s real-time and archived data offer valuable tools for a variety of commercial and governmental applications.

- Telematics: For mobile professionals and others, the company's traffic information complements in-vehicle navigation devices, informing drivers not only how to get from point A to point B but how long it will take to get there, or even direct them to an alternative route.

- **Military Applications**

A number of companies have developed WSNs that include customizable, sensor-laden, networked nodes and both mobile and Internet-hosted user interfaces. For example, Rockwell Scientific's wireless sensing network develop-ment system allows examination of issues relative to design, deployment, and use of microsensor networks. Wireless distributed microsensor networks consist of a collection of communicating nodes, where each node incorporates (1) one or more sensors for measuring the environment, (2) computing capability to process sensor data into ''high-value'' information and to accomplish local control, and a radio to communicate information to and from neighboring nodes and eventually to external users. The company[9] has developed new prototype devel-opment platforms for experimenting with microsensor networks under a number of government- and industry-sponsored programs (see Figure 1.14). The baseline prototype wireless sensing unit is based on an open, modular design using widely available commercial-off-the-shelf (COTS) technology. These nodes combine sensors (such as mechanical vibration, acoustic, and magnetic) with a commercial digital cordless telephone radio and an embedded commercial RISC microprocessor in a small package.

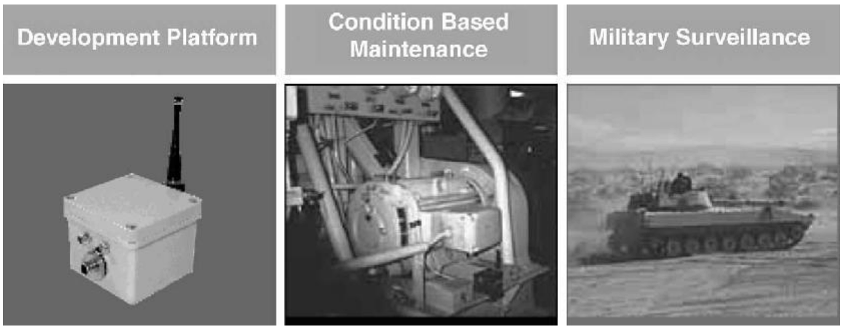


Figure 1.14 Military examples. (Courtesy of Rockwell Scientific.)

Condition-Based Monitoring Again as an illustrative example, Rockwell Scientific is developing WSNs specifically tailored to the requirements for monitoring complex machinery and processes. Their WSNs have been deployed on board U.S. Navy ships as part of a developmental program with the Office of Naval Research. Exploratory studies have also been done for use of WSNs on aircraft, rotorcraft, and spacecraft as part of an overall integrated vehicle health management system. Machinery maintenance has evolved from run to fail (no maintenance) to scheduled maintenance (e.g., change oil every three months) to condition-based maintenance (CBM). All three techniques are in current use. The economic trade-off is between the cost of the CBM equipment and the staffing resources expended to determine the machine's health and the cost of unexpected, as opposed to scheduled, repair and process downtime. With the emphasis of industry in the last couple of decades on just-in-time processes, unexpected machinery failure can be costly. The successful application of machinery monitoring programs can optimize the use of machinery and keep manufacturing costs in check by making the process more efficient. The costs associated with CBM can be allocated into equipment, installation, and labor costs in collecting and analyzing the machine health data. WSNs are positioned to minimize all three costs and, in particular, to eliminate the staffing costs, which often are the largest. With the continuing advances in data processing hardware and RF transceiver hardware (cell phone markets drive this), the technology is now becoming available to install compact monitoring systems on machinery that avoid the installation expense of data cabling through RF link technology; these systems provide a mechanism for data acquisition and analysis on the monitoring unit itself. The primary challenge faced by WSNs for machinery and process monitoring is related to the quality of the information produced by both the individual sensors and the distributed sensor network. Nodes located on individual components must not only be able to provide information on the present state of the component (e.g., a bearing or gearbox), but also provide an indicator of the remaining useful life of the component.

The approach taken at Rockwell Scientific has been to mount two parallel efforts. Existing diagnostic routines and expert systems are being ported to WSN hardware with modifications for autonomous data collection and analysis. The firm is also involved in developing advanced diagnostics algorithms for machinery vibration monitoring that provide advances over present systems. The main thrust in this area is to generalize diagnostic algorithms so that they do not depend on detailed knowledge of the machinery on which they are installed. Data-processing algorithms that determine critical machine parameters, such as the shaft speed or the number of rolling elements in a bearing, have been developed. The company is also developing the ability for distributed collections of WSN nodes located on machine components and/or throughout a process to provide information on the overall machine and/or process on which they are deployed. This is a primary advantage of a distributed sensing system in that it enables inferences from individual component data to be used to provide diagnostics for aspects of the system that are not being sensed directly. For example, monitoring bearing vibrations or motor currents can provide information not only on bearing health but also on the inception and severity of pump cavitation. Pump cavitation, in turn, can provide information on the state of valves located throughout a pumping process.

The dynamically reconfigurable nature of WSNs is being exploited by Rockwell Scientific in an application of WSNs to space vehicle status monitoring in collaboration with the Boeing Company. WSNs are deployed throughout space vehicles to perform a variety of missions during the different phases of the space flight. For example, during the launch phase, WSN nodes located on various critical components of the spacecraft can monitor vibration levels for out-of-compliance signals. During flight and re-entry, the WSN monitor structural disturbances caused by the significant temperature gradients encountered as different portions of the vehicle are alternately exposed and shadowed from the sun and atmosphere. This is accomplished via coherent collection and processing of vibration and strain data. Upon landing, critical components will once again be monitored for out-of-compliance signals. These data are used to determine those components needing postflight maintenance or replacement, enabling faster turnaround for the space vehicle, thereby lowering costs.

Military Surveillance For military users, an application focus of WSN technology has been area and theater monitoring. WSNs can replace single high-cost sensor assets with large arrays of distributed sensors for both security and surveillance applications. The WSN nodes are smaller and more capable than sensor assets presently in the inventory; the added feature of robust, self-organizing networking makes WSNs deployable by untrained troops in essentially any situation. Distributed sensing has the additional advantages of being able to provide redundant and hence highly reliable information on threats as well as the ability to localize threats by both coherent and incoherent processing among the distributed sensor nodes. WSNs can be used in traditional sensor network applications for large-area and perimeter monitoring and will ultimately enable every platoon, squad, and soldier to deploy WSNs to accomplish a number of mission and self-protection goals. Rockwell Scientific has been working with the U.S. Marine Corps and U.S. Army to test and refine WSN performance in desert, forest, and urban terrain.

For the urban terrain, WSNs are expected to improve troop safety as they clear and monitor intersections, buildings, and rooftops by providing continuous vigilance for unknown troop and vehicle activity. The primary challenge facing WSNs is accurate identification of the signal being sensed; one needs to develop state-of-the-art vibration, acoustic, and magnetic signal classification algorithms to accomplish this goal. Currently, WSNs run vibration detection algorithms based on energy thresholding; although this is a simple technique, it is subject to false alarms, leading to a desire for more sophisticated spectral signature algorithms. Low-power algorithms to classify a detected event as an impulsive event (e.g., either a footstep or gunshot) or vehicle (e.g., wheeled or tracked, light or heavy) have also been demonstrated.

The inclusion of multiple sensors on each node enables fusion of different sensed phenomenologies, leading to higher-quality information and decreased false alarm rates. Algorithms for fusing the seismic, acoustic, and magnetic sensors on a single node are being developed. Algorithms utilizing the advantages of a network of spatially separate nodes span a range of cooperative behaviors, each of which trades off detection quality versus energy consumption. Examples of cooperative fusion range from high-level decision corroboration (e.g., voting), to feature fusion, to full coherent beam formation. The examples discussed above are simply representative of many efforts under way at many companies involved in theater technology.

Borders Monitoring At press time Boeing Co. had secured a contract from the Department of Homeland Security to implement SBInet, the Secure Borders Initiative, along the northern and southern U.S. borders. The program was announced by DHS in 2005, and contracts were awarded in late 2006. The SBInet portion of the Secure Borders Initiative is the development of a technological infrastructure that facilitates the use of a variety of sensors and detection devices, and which enables that data to be forwarded to remote operations centers via Ku-band satellite uplinks.

- **Civil and Environmental Engineering Applications**

Sensors can be used for civil engineering applications. Research has been under way in recent years to develop sensor technology that is applicable for buildings, bridges, and other structures. The goal is to develop ‘‘smart structures’’ that are able to self-diagnose potential problems and self-prioritize requisite repairs. This technology is attractive for earthquake-active zones. Although routine mild tremors may not cause visible damage, they can give rise to hidden cracks that could eventually fail during a higher-magnitude quake. Furthermore, after a mild earthquake, a buil-ding's true structural condition may not be ostensively visible without some ‘‘below-the-skin’’ measurement. Smart Dust motes, tiny and inexpensive sensors developed by UC–Berkeley engineers, are promising in this regard (see Figure 2.15). The battery-powered matchbox-sized WNs operating on TinyOS are designed to sense a number of factors, ranging from light and temperature (for energy-saving applications) to dynamic response (for civil engineering analysis).

Figure 1.15 Motes. [Courtesy of Steve Glaser and David Pescovitz, Center for Information Technology Research in the Interest of Society (CITRIS) program, UC–Berkeley.]

Up to the present, wired seismic accelerometers (devices able to measure movement) have been used; however, these devices are expensive (several thousands of dollars each) and are difficult to install. This predicament limits the density of sensor deploy-ment, which in turn limits the planner's view of a building's structural integrity. As a result, a safety-impacting structural problem does not become visible until the entire building is affected. On the other hand, if sensors that cost a few hundreds of dollars and that can be installed relatively easily and quickly become available, one arrives at a situation where dense packs of sensors can be deployed to surround all critical beams and columns. This arrangement is able to provide detailed structural data. UC Berkeley's Richmond Field Station seismic laboratory is pursuing research in this area. Data from the Smart Dust motes is expected to increase the accuracy of finite element analyses, a method of computer modeling where mathematical equations represent a structure's behavior under given conditions.

- **Wildfire Instrumentation**

Collecting real-time data from wildfires is important for life safety considerations and allows predictive analysis of evolving fire behavior. One way to collect such data is to deploy sensors in the wildfire environment. FireBugs are small wireless sensors (motes) based on TinyOS that self-organize into networks for collecting real-time data in wildfire environments. The FireBug system combines state-of-the-art sensor hardware running TinyOS with standard off-the-shelf World Wide Web and database technology, allowing rapid deployment of sensors and behavior monitoring.

- **Habitat Monitoring**

As an illustrative example, in the recent past, the Intel Research Laboratory at Berkeley undertook a project with the College of the Atlantic in Bar Harbor and UC–Berkeley to deploy wireless sensor networks on Great Duck Island in Maine. These networks monitor the microclimates in and around nesting burrows used by Leach's storm petrel. The goal was to develop a habitat-monitoring kit that enables researchers worldwide to engage in nonintrusive and nondisruptive monitoring of sensitive wildlife and habitats. About three dozen motes were deployed on the island. Each mote has a microcontroller, a low-power radio, memory, and batteries. Sensor motes monitor the nesting habitat of Leach's storm petrel on the island and relay their readings into a satellite link that allows researchers to download real-time environmental data over the Internet. For habitat monitoring the planner needed sensors that can take readings for temperature, humidity, barometric pressure, and midrange infrared. Motes sample and relay their sensor readings periodically to computer base stations on the island.

1.7 Another Taxonomy of WSN Technology

The taxonomy is based on physical placement of the various sensors and the connectivity of these nodes to nodes in the wired infrastructure; the network configuration determines the amount of routing intelligence that needs to be supported in the sensor nodes. Specifically, key factors used in the classification process under discussion are the size of the system, the number of sensors used, the average (and/or maximum) distance (in hops) of the sensors to the wired infrastructure, and the distribution of the sensor nodes.

Three types of WSN system (technology) that have been described in are:

1. Non-propagating WSN systems
2. Deterministic routing WSN systems
   a. Aggregating
   b. Nonaggregating systems
3. Self-configurable and self-organizing WSN systems
   a. Aggregating
   b. Nonaggregating systems

In non-propagating WSN systems, WNs are not responsible to support dynamic routing of packets to end systems. This follows because the wired infrastructure is the basic connecting component in this case and WNs are generally in close proxi-mity (one hop) to the wired infrastructure. WNs collect and report their sensor mea-surements to nodes connected to the wired network, which, in turn, route the information to the end system. These systems are generally manually configurable and are highly deterministic in deployment distribution.

Environmental sensors deployed in buildings or within a physically restricted area belong to this category.

In deterministic routing WSN systems, the wired and wireless infrastructures both play an active role in routing packets. For packets to reach the wired infra-structure in these environments, the WNs have to route or forward packets through a number of wireless hops. However, the routes to the wired infrastructure are deter-ministic and can be configured manually. In home networking systems, the WNs are in prespecified positions and route information through predetermined routes. The number of nodes in such a system is usually relatively small.

Part-A Questions

1. Differentiate ad hoc networks and wireless sensor networks.
2. Define reconfigurable sensor network.
3. List any four applications of WSN.
4. State the important characteristics of WSN.
5. Identify the need for multi hop wireless communication in WSN?
6. Sketch the sensor node with its components.
7. Draw the architecture of WSN.
8. List the different types of nodes in WSN.
9. Identify the elements of WSN.
10. Compare sensor node and sink node.

Part-B Questions

1. List and sketch the components of a sensor node and also describe the Wireless Sensor Network architecture.
2. Elaborate the usage of wireless sensor network in military applications.
3. Determine the possibilities of using WSN in medical applications.
4. Organize the steps to monitor the habitat of our national animal.
5. Design an WSN architecture to control home appliances.

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF COMPUTING

DEPARTMENT OF INFORMATION TECHNOLOGY

UNIT - 2

SITA1501 – WIRELESS SENSOR NETWORKS AND ARCHITECTURE

UNIT - 2 ROUTING PROTOCOLS FOR AD HOC WIRELESS NETWORKS

Syllabus:

Designing issues, classification of routing protocols, table driven routing protocols, on demand routing protocol, Hybrid routing protocol, Hierarchical routing protocols. Multicast routing in Ad Hoc wireless networks: Operations and classification of multicast routing protocols, Tree based multicast routing protocol, Mesh based multicast routing protocol.

2. Introduction

Routing in MANETs To enable communication within a MANET, a routing protocol is required to establish routes between participating nodes. Because of limited transmission range, multiple network hops may be needed to enable data communication between two nodes in the network. Since MANET is an infrastructureless network, each mobile node operates not only as a host but also as a router, forwarding packets for other mobile nodes in the network. There are frequent unpredictable topological changes in these networks, which makes the task of finding and maintaining routes as difficult. Conventional routing protocols based on distance vector or link state algorithms can not be applied here, since the amount of routing related traffic would waste a large portion of the wireless bandwidth, and such discovered routes would soon become obsolete due to mobility of nodes In MANETs mobile nodes share the same frequency channel thereby limiting the network capacity. Thus one of the highly desirable properties of a routing protocol for MANETs is that it should be bandwidth efficient.

2.1 Designing issues

The major challenges that a routing protocol designed for ad hoc wireless networks faces are:

2.1.1 Mobility

- Network topology is highly dynamic due to movement of nodes. Hence, an ongoing session suffers frequent path breaks.
- Disruption occurs due to the movement of either intermediate nodes in the path or end nodes.
- Wired network routing protocols cannot be used in adhoc wireless networks because the nodes are here are not stationary and the convergence is very slow in wired networks.
- Mobility of nodes results in frequently changing network topologies.
- Routing protocols for adhoc wireless networks must be able to perform efficient and effective mobility management.

2.1.2 Bandwidth Constraint

- Abundant bandwidth is available in wired networks due to the advent of fiber optics and due to the exploitation of wavelength division multiplexing (WDM) technologies.
- In a wireless network, the radio band is limited , and hence the data rates it can offer are much less than what a wired network can offer .
- This requires that the routing protocols use the bandwidth optimally by keeping the overhead as low as possible.
- The limited bandwidth availability also imposes a constraint on routing protocols in maintaining the topological information.

2.1.3 Error-prone shared broadcast radio channel

- The broadcast nature of the radio channel poses a unique challenge in ad hoc wireless networks.
- The wireless links have time-varying characteristics in terms of link capacity and link-error probability.
- This requires that the adhoc wireless network routing protocol interact with the MAC layer to find alternate routes through better-quality links.
- Transmissions in ad hoc wireless network routing protocol interact with the MAC layer to find alternate routes through better-quality links.
- Therefore, it is required that ad hoc wireless network routing protocols find paths with less congestion.

2.1.4 Hidden and exposed terminal problems

- The hidden terminal problem refers to the collision of packets at a receiving node due to the simultaneous transmission of those nodes that are not within the direct transmission range of the receiver, but are within the transmission range of the receiver.
- Collision occurs when both transmits packets at the same time without knowing about the transmission of each other.

Ex: Consider figure 2.1. Here, if both node A and node C transmit to node B at the same time, their packets collide at node B. This is due to the fact that both node A and C are hidden from each other, as they are not within the direct transmission range of each other and hence do not know about the presence of each other.

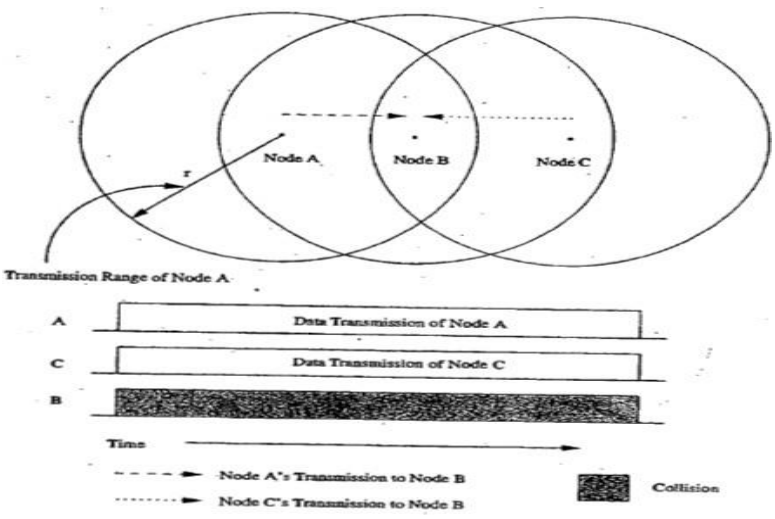

Node A
Node B
Node C
r
Transmission Range of Node A
A
C
B
Data Transmission of Node A
Data Transmission of Node C
Time
Node A's Transmission to Node B
Node C's Transmission to Node B
Collision

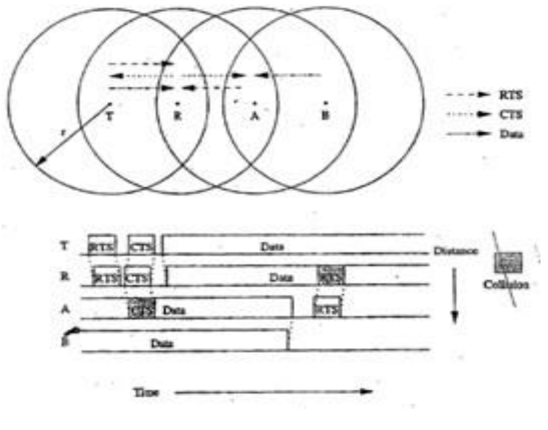
Fig.2.1 Hidden Terminal Problems

- Solution for this problem include medium access collision avoidance (MACA):

Transmitting node first explicitly notifies all potential hidden nodes about the forthcomingtransmission by means of a two-way handshake control protocol called RTS-CTS protocol exchange.

  - This may not solve the problem completely but it reduces the probability of collisions.

- Medium access collision avoidance for wireless (MACAW)
  - An improved version of MACA protocol.
  - Introduced to increase the efficiency.
  - Requires that a receiver acknowledges each successful reception of data packet.



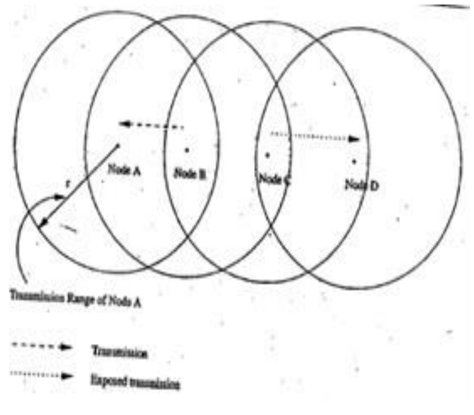


Fig.2.2 Hidden Terminal problem with RTS-CTS Fig.2.3 ExposedTerminal Problem

- Successful transmission is a way exchange mechanism, RTS-CTS-Data-ACK, as illustrated in figure Other solutions include floor acquisition multiple access (FAMA) and Dual busy tone multiple access(DBTMA).

- The exposed terminal problem refers to the inability of a node which is blocked due to transmission by a nearby transmitting node to transmit to another node.
- Ex.: consider the fig. 2.3. Here, if a transmission from node B to another node A is already in progress, node C cannot transmit to node D, as it concludes that its neighbor node B, is in transmitting mode and hence should not interfere with the on-going transmission. Thus, reusability of the radio spectrum is affected.

2.1.5 Resource Constraints

- Two essential and limited resources are battery life and processing power.
- Devices used in adhoc wireless networks require portability, and hence they also have weight and size constraints along with the restrictions on the power source.

2.2 Characteristics of an Ideal Routing Protocol for ad hoc wireless networks

A routing protocol for ad hoc wireless networks should have the following characteristics:

- It must be fully distributed routing involves high control overhead and hence is not scalable.
- It must be adaptive to frequent topology changes caused by the mobility of nodes.
- Route computation and maintenance must involve a minimum number of nodes. Each node in the network must have quick access to routes, that is, minimum connection setup time is desired.
- It must be localized, as global state maintenance involves a huge state propagation control overhead.
- It must be loop free and free from state routes.
- The number of packet collisions must be kept to a minimum by limiting the number of broadcasts made by each node. The transmissions should be reliable to reduce message loss and to prevent the occurrence of state routes.
- It must converge to optimal routes once the network topology becomes stable, The convergence must be quick.
- It must optimally use scarce resources such as bandwidth, computing power, memory and battery power,
- Every node in the network should try to store information regarding the stable local topology only. Changes in remote parts of the network must not cause updates in the topology information maintained by the node.
- It should be able to provide a certain level of Quality of Service (QoS) as demanded by the applications and should also offer support for time-sensitive traffic.

2.3 Classifications of Routing Protocols

A classification tree is shown below:

The routing protocol for adhoc wireless networks can be broadly classified into 4 categories based on

- Routing information update mechanism
- Use of temporal information for routing
- Routing topology
- Utilization of specific resources

2.3.1 Based on the routing information update mechanism

Ad hoc wireless network routing protocols can be classified into 3 major categories based on the routing information update mechanism. They are:

➢ Proactive or table-driven routing protocols:

- Every node maintains the network topology information in the form of routing tables by periodically exchanging routing information.
- Routing information is generally flooded in the whole network.
- Whenever a node requires a path to a destination, it runs an appropriate path-finding algorithm on the topology information it maintains.

➢ Reactive or on-demand routing protocols

- Do not maintain the network topology information.
- Obtain the necessary path when it is required, by using a connection establishment process.

.

➢ Hybrid routing protocols:

- Combine the best features of the above two categories.
- Nodes within a certain distance from the node concerned, or within a particular geographical region, are said to be within the routing zone of the given node.
- For routing within this zone, a table-driven approach is used.
- For nodes that are located beyond this zone, an on-demand approach is used.

2.3.2 Based on the use of temporal information for routing

The protocols that fall under this category can be further classified into two types:

➢ Routing protocols using past temporal information:

Use information about the past status of the links or the status of links at the time of routing tomake routing decisions.

➢ Routing protocols that use future temporal information:

Use information about the about the expected future status of the wireless links to makeapproximate routing decisions.

Apart from the lifetime of wireless links, the future status information also includes informationregarding the lifetime of the node, prediction of location, and prediction of link availability.

2.3.3 Based on the routing topology

Ad hoc wireless networks, due to their relatively smaller number of nodes, can make use of either a flattopology or a hierarchical topology for routing.

- Flat topology routing protocols:
  Make use of a flat addressing scheme similar to the one used in IEEE 802.3 LANs. It assumes the presence of a globally unique addressing mechanism for nodes in an ad hocwireless network.

- Hierarchical topology routing protocols:
  Make use of a logical hierarchy in the network and an associated addressing scheme. The hierarchy could be based on geographical information or it could be based on hop distance.

2.3.4 Based on the utilization of specific resources

- Power-aware routing:
  Aims at minimizing the consumption of a very important resource in the ad hoc wireless networkssuch as battery power. The routing decisions are based on minimizing the power consumption either logically or globallyin the network.
- Geographical information assisted routing:
  Improves the performance of routing and reduces the control overhead by effectively utilizing thegeographical information available.

2.4 Table-Driven Routing Protocols

- These protocols are the extensions of the wired routing protocols.
- They maintain the global topology information in the form of tables at every node.
- Tables are updated frequently in order to maintain consistent and accurate network state information.
- Ex.: Destination Sequenced Distance Vector routing protocol (DSDV), Wireless Routing Protocol (WRP), Source-Tree Adaptive Routing Protocol (STAR) and Cluster-head Gateway Switch Routing protocol (CGSR).

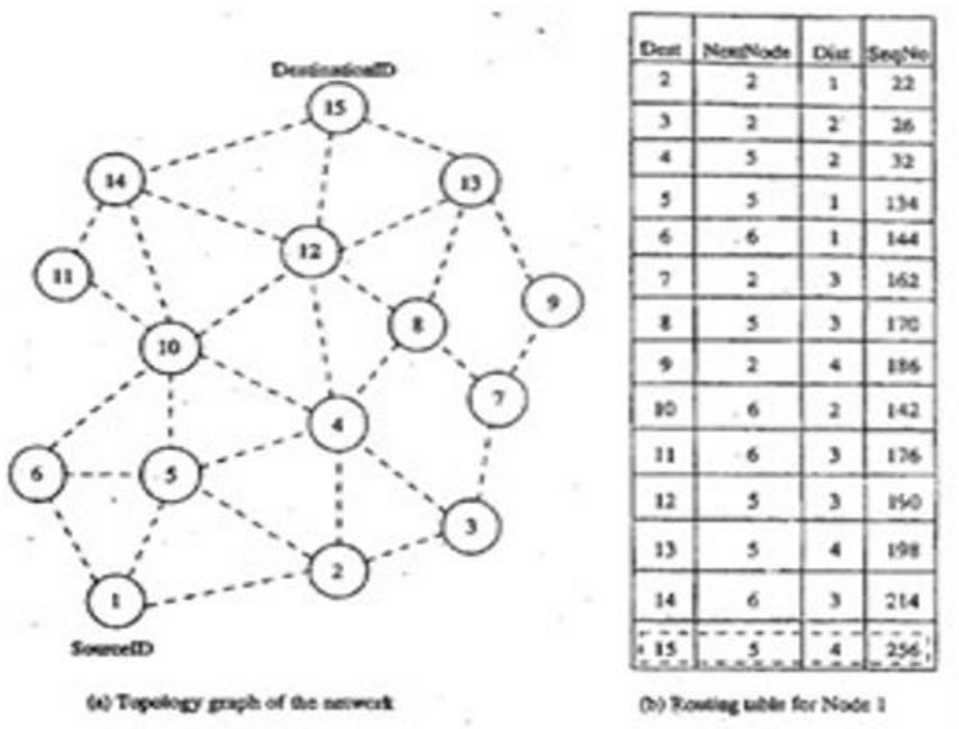
Proactive protocols attempt to evaluate continuously the routes within the network. It means proactive protocol continuously maintain the routing information, so that when a packet needs to be forwarded, the path is known already and can be immediately used. The family of distance vector protocols is an example of proactive scheme. The advantage of the proactive schemes is that whenever a route is needed, there is negligible delay in determining the route. Unfortunately, it is a big overhead to maintain routing tables in the

MANET environment. Therefore, this type of protocol has following common disadvantages:

  - Requires more amounts of data for maintaining routing information.
  - Low reaction on re-structuring network and failures of individual nodes.

2.4.1 Destination Sequenced Distance Vector routing protocol (DSDV)

- It is an enhanced version of the distributed Bellman-Ford algorithm where each node maintains a tablethat contains the shortest distance and the first node on the shortest path to every other node in the network.
- It incorporates table updates with increasing sequence number tags to prevent loops, to counter the count-to-infinity problem, and for faster convergence.
- As it is a table-driven routing protocol, routes to all destinations are readily available at every node at all times.
- The tables are exchanged between neighbors at regular intervals to keep an up-to-date view of the network topology.
- The table updates are of two types.
  - *Incremental updates:* Takes a single network data packet unit (NDPU). These are used whena node does not observe significant changes in the local topology.
  - *Full dumps:* Takes multiple NDPUs. It is done either when the local topology changessignificantly or when an incremental update requires more than a single NDPU.
- Table updates are initiated by a destination with a new sequence number which is always greater than the previous one.
- Consider the example as shown in figure 3.4(a). Here node 1 is the source node and node 15 is the destination. As all the nodes maintain global topology information, the route is already available as shownin figure 3.4 (b). Here the routing table node 1 indicates that the shortest route to the destination node isavailable through node 5 and the distance to it is 4 hops, as depicted in figure 3.4(b)
- The reconfiguration of a path used by an on-going data transfer session is handled by the protocol in the following way.
- The end node of the broken link initiates a table update message with the broken link's weight assigned to infinity and with a sequence number greater tha the stored sequence number for that destination.
- Each node upon receiving an update with weight infinity, quickly disseminates it to its neighbors in order to propagate the broken-link information to the whole.
- A node always assigns an odd number to the link break update to differentiate it from the even sequence number generated by the destination.
- Figure 3.5 shows the case when node 11 moves from its current position.

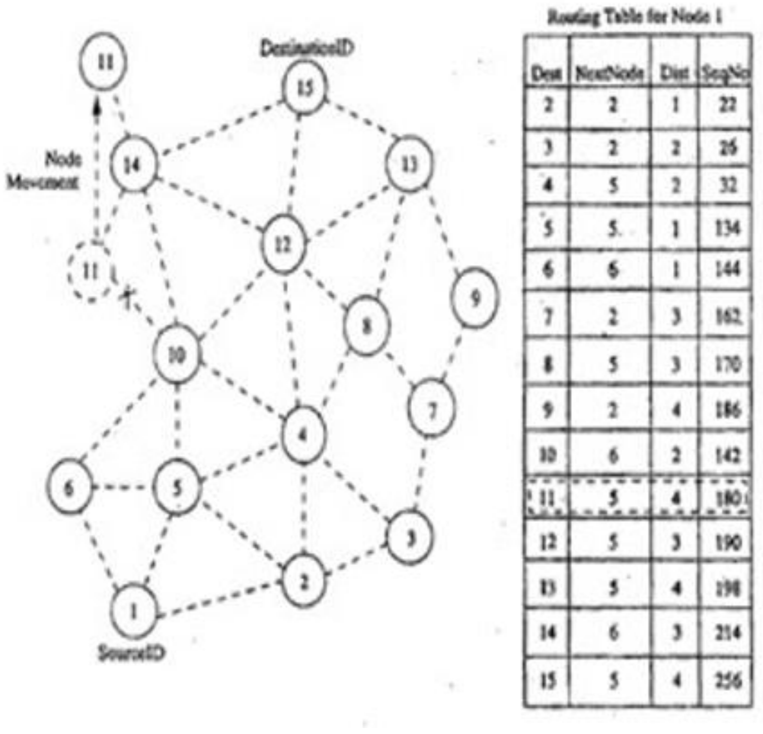


Dest	NextNode	Dist	SeqNo
2	2	1	22
3	2	2	26
4	5	2	32
5	5	1	134
6	6	1	144
7	2	3	162
8	5	3	170
9	2	4	186
10	6	2	142
11	6	3	176
12	5	3	190
13	5	4	198
14	6	3	214
15	5	4	256

(a) Topology graph of the network

(b) Routing table for Node 1

Figure: 3.4 Route establishment in DSDV



Routing Table for Node 1

Dest	NextNode	Dist	SeqNo
2	2	1	22
3	2	2	26
4	5	2	32
5	5.	1	134
6	6	1	144
7	2	3	162
8	5	3	170
9	2	4	186
10	6	2	142
11	5	4	180
12	5	3	190
13	5	4	198
14	6	3	214
15	5	4	256

Figure: 3.5 Route maintenance in DSDV

- Advantages

  - ✓ Less delay involved in the route setup process.
  - ✓ Mechanism of incremental update with sequence number tags makes the existing wired networkprotocols adaptable to ad hoc wireless networks.
  - ✓ The updates are propagated throughout the network in order to maintain an up-to-date view ofthe network topology at all nodes.

- Disadvantages

  - ✓ The updates due to broken links lead to a heavy control overhead during high mobility.
  - ✓ Even a small network with high mobility or a large network with low mobility can completelychoke the available bandwidth.
  - ✓ It suffers from excessive control overhead.
  - ✓ In order to obtain information about a particular destination node, a node has to wait for a tableupdate message initiated by the same destination node.
  - ✓ This delay could result in state routing information at nodes.

2.5 Reactive Protocols

Reactive protocols do not maintain routes but invoke a route determination procedure only on demand or we can say reactive protocols build the routes only on demand. Thus, when a route is required, some sort of global search procedure is initiated. The family of classical flooding algorithms belongs to the reactive protocol group. Examples of reactive ad-hoc network routing protocols include ad hoc on demand distance vector (AODV) and temporally ordered routing algorithm (TORA).

These protocols have the following advantages:

- o No large overhead for global routing table maintenance as in proactive protocols.
- o Reaction is quick for network restructure and node failure.
- o Even though reactive protocols have become the main stream for MANET routing, they still have the following disadvantages:
- o Latency time is high in route finding
- o Excessive flooding can lead to network clogging.

2.5.1 Dynamic Source Routing Protocol (DSR)

- Dynamic source routing is an on-demand routing protocol which is based on source routing. Designed to restrict the bandwidth consumed by control packets in adhoc wireless networks by eliminating the periodic table update messages
- It is very similar to AODV in that it forms a route on demand when a transmitting computer requests one. But, it uses source routing instead of relying on the routing table at each intermediate device. Many successive refinements have been made to dynamic source routing.
- This protocol works in two main phases:
  - Route discovery
  - Route maintenance
- When a node has a message to send, it contacts to the route cache to determine whether is it has a route to the destination. If an active route to the destination exists, it is used to send a message.
- Otherwise, a node initiates a route discovery by broadcasting a route request packet. The route request stores the destination address, the source address, and a unique identification number.
- Each device that receives the route request checks whether it has a route to the destination. If it does not, it adds its own address to the route record of the packet and then rebroadcasts the packet on its outgoing links.
- To minimize the no. of broadcasts, a mobile rebroadcast a packet only if it has not seen the packet before and its own address was not already in the route record.
- Advantages:

  - ✓ Uses a reactive approach which eliminates the need to periodically flood the network with table update messages
  - ✓ Route is established only when required
  - ✓ Reduce control overhead

- Disadvantages

  - ✓ Route maintenance mechanism does not locally repair a broken link
  - ✓ Stale route cache information could result in inconsistencies during route construction phase
  - ✓ Connection set up delay is higher
  - ✓ Performance degrades rapidly with increasing mobility
  - ✓ Routing overhead is more & directly proportional to path length

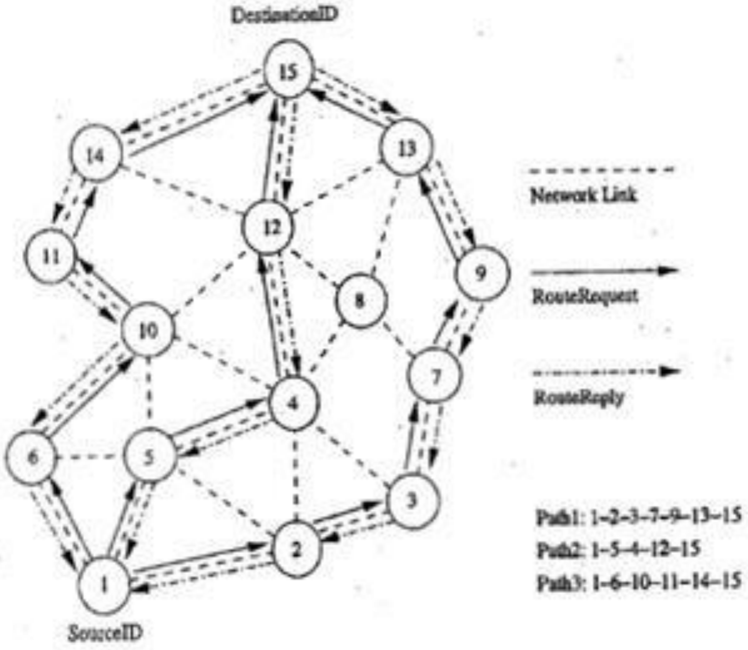


Fig. 3.6 Route establishment in DSR

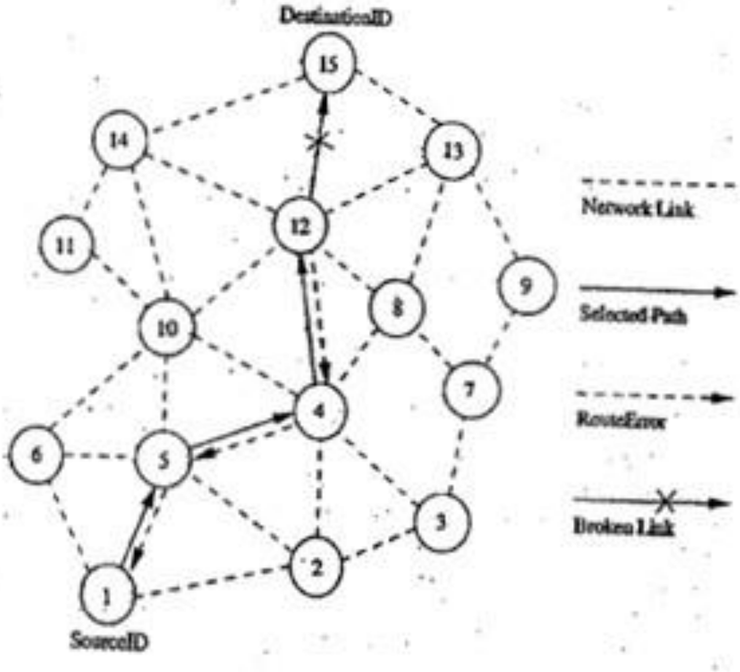


Figure: 3.7 Route maintenance in DSR

2.6. Hybrid Protocols

Hybrid protocols attempt to take advantage of best of reactive and proactive schemes. The basic idea behind such protocols is to initiate route discovery on demand but at a limited search cost. One of the popular hybrid protocols is zone routing protocol (ZRP).

2.6.1 Zone routing protocol (ZRP)

- Zone routing protocol is a hybrid of reactive and proactive protocols. It combines the advantage of both reactive and proactive schemes.
- ZRP was invented by Zygmunt Haas of Cornell University. Zone routing protocol finds loop free routes to the destination.
- ZRP divides the network into zones of variable size; size of the zone is determined radius of length ?, where the ? is the number of hops or nodes to the perimeter of the zone and not the physical distance.
- In other words we can say that, the neighborhood of the local node is called a routing zone. Specifically, a routing zone of the node is defined as the set of nodes whose minimum distance in hops from the node is no greater than the zone radius.
- A node maintains routes to all the destinations proactively in the routing zone. It also maintains its zone radius, and the overlap from the neighboring routing zones.
- To create a routing zone, the node must identify all its neighbors first which are one hop away and can be reached directly.
- The Process of neighbor discovery is governed by the NDP (Neighbor Discovery Protocol), a MAC level scheme. ZRP maintains the routing zones through a proactive component called the intra-zone routing protocol (IARP) and is implemented as a modified distance vector scheme. Thus IARP is responsible for maintaining routes within the routing zone.
- Another protocol called the inter-zone routing protocol (IERP) which is responsible for maintaining and discovering the routes to nodes beyond the routing zone.
- This type of process uses a query - response mechanism on-demand basis. IERP is more efficient than standard flooding schemes.
- When a source node send data to a destination which is not in the routing zone, the source initiates a route query packet.
- The latter identified by the tuple <source node ID, request number>. This request is then broadcasted to all the nodes in the source nodes periphery.
- When a node receives this query, it adds its own identification number (ID) to the query. Thus the sequence of recorded nodes presents a route from the current routing zone. Otherwise, if the destination is in the current routing zone of the node, a route reply is sent back to the source along the reverse from the accumulated record.
- A big advantage of this scheme is that a single route request can result in multiple replies of route. The source can determine the quality of these multiple routes based on such parameter as hop count or traffic and choose the best route to be used.

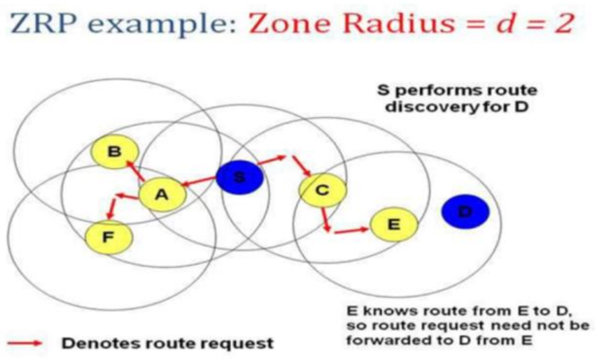

ZRP example: Zone Radius = d = 2
S performs route discovery for D
B
A
S
C
E
D
F
E knows route from E to D, so route request need not be forwarded to D from E
Denotes route request

Fig.: 3.8 Zone routing

2.7 Hierarchical Routing Protocols

With this type of protocol the choice of proactive and of reactive routing depends on the hierarchic level in which a node resides. The routing is initially established with some proactively prospected routes and then serves the demand from additionally activated nodes through reactive flooding on the lower levels. The choice for one or the other method requires proper attributation for respective levels. The main disadvantages of such algorithms are:

1. Advantage depends on depth of nesting and addressing scheme.
2. Reaction to traffic demand depends on meshing parameters.

Examples of hierarchical routing algorithms are:

- CBRP (Cluster Based Routing Protocol)
- Optimized Link State Routing protocol (OLSR)
- FSR (Fisheye State Routing protocol)
- Order One Network Protocol; Fast logarithm-of-2 maximum times to contact nodes. Supports large groups.
- ZHLS (Zone-based Hierarchical Link State Routing Protocol

2.7.1 Optimized Link State Routing Protocol (OLSR)

The Optimized Link State Routing Protocol (OLSR)[1] is an IP routing protocol optimized for mobile ad hoc networks, which can also be used on other wireless ad hoc networks. OLSR is a proactive link-state routing protocol, which uses hello and topology control (TC) messages to discover and then disseminate link state information throughout the mobile ad hoc network. Individual nodes use this topology information to compute next hop destinations for all nodes in the network using shortest hop forwarding paths.

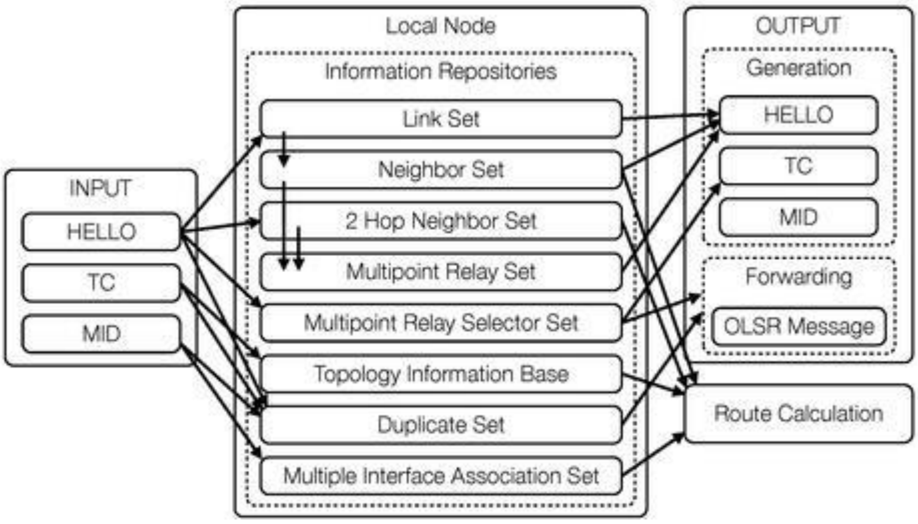


Fig.3.9. OLSR Data Flow

Link-state routing protocols such as Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS) elect a *designated router* on every link to perform flooding of topology information. In wireless ad hoc networks, there is different notion of a link, packets can and do go out the same interface; hence, a different approach is needed in order to optimize the flooding process. Using Hello messages, the OLSR protocol at each node discovers 2-hop neighbor information and performs a distributed election of a set of *multipoint relays* (MPRs). Nodes select MPRs such that there exists a path to each of its 2-hop neighbors via a node selected as an MPR. These MPR nodes then source and forward TC messages that contain the MPR selectors. This functioning of MPRs makes OLSR unique from other link state routing protocols in a few different ways: The forwarding path for TC messages is not shared among all nodes but varies depending on the source, only a subset of nodes source link state information, not all links of a node are advertised but only those that represent MPR selections.

Since link-state routing requires the topology database to be synchronized across the network, OSPF and IS-IS perform topology flooding using a reliable algorithm. Such an algorithm is very difficult to design for ad hoc wireless networks, so OLSR doesn't bother with reliability; it simply floods topology data often enough to make sure that the database does not remain unsynchronized for extended periods of time.

Multipoint relays (MPRs) relay messages between nodes. They also have the main role in routing and selecting the proper route from any source to any desired destination node. MPRs advertise link-state information for their MPR selectors (a node selected as a MPR) periodically in their control messages. MPRs are also used to form a route from a given node to any destination in route calculation. Each node periodically broadcasts a Hello message for the link sensing, neighbor detection and MPR selection processes

Benefits: Being a proactive protocol, routes to all destinations within the network are known and maintained before use. Having the routes available within the standard routing table can be useful for some systems and network applications as there is no route discovery delay associated with finding a new route. The routing overhead generated, while generally greater than that of a reactive protocol, does not increase with the number of routes being created. Default and network routes can be injected into the system by HNA messages allowing for connection to the internet or other networks within the OLSR MANET cloud. Network routes are something reactive protocols do not currently execute well. Timeout values and validity information is contained within the messages conveying information allowing for differing timer values to be used at differing nodes.

2.8 Multicasting Protocols:

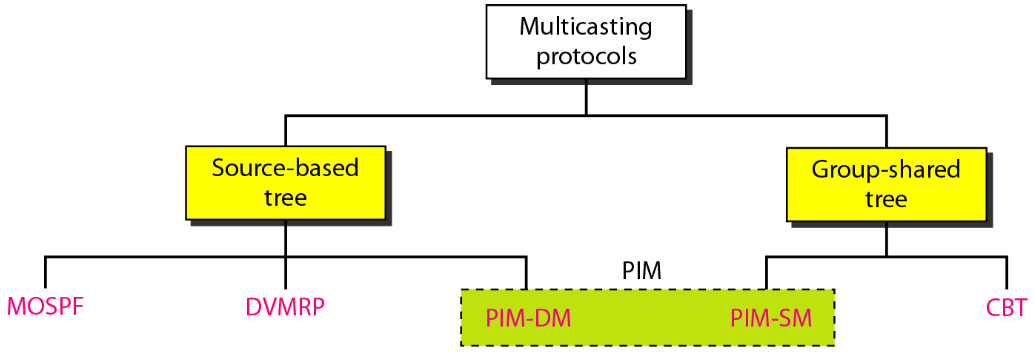


Fig.3.10 Classification

2.8.1 Multicast Link State Routing: MOSPF

- Multicast link state routing uses the source-based tree approach
- n (the number of group) topologies and n shortest path trees made
- Each router has a routing table that represents as many shortest path trees as there are groups
- MOSPF is an extension of the OSPF protocol that uses multicast link state routing to createsource based trees
- MOSPF requires a new link state update packet to associate the unicast address of a host withthe group address or addresses the host is sponsoring
- MOSPF is a data-driven protocol; the first time an MOSPF router see a datagram with a givensource and group address, the router constructs the Dijkstra shortest path tree

2.8.2 Multicast Distance Vector: DVMRP

- Multicast distance vector routing uses the source-based trees, but the router never actuallymakes a routing table
- Multicast routing does not allow a router to send its routing table to its neighbors. The idea isto create a table from scratch by using the information from the unicast distance vector tables
- Process based on four decision-making strategies. Each strategy is built on its predecessor
  - – Flooding
  - – Reverse Path Forwarding (RPF)

– Reverse Path Broadcasting (RPB)
– Reverse Path Multicasting (RPM)

- DVMRP: Strategies

  - Flooding broadcasts packets, but creates loops in the systems
  - Reverse path forwarding: RPF eliminates the loop in the flooding process
  - Reverse path broadcasting: RPB creates a shortest path broadcast tree from the source to each destination. It guarantees that each destination receives one and only one copy of the packet
  - Problem with RPF

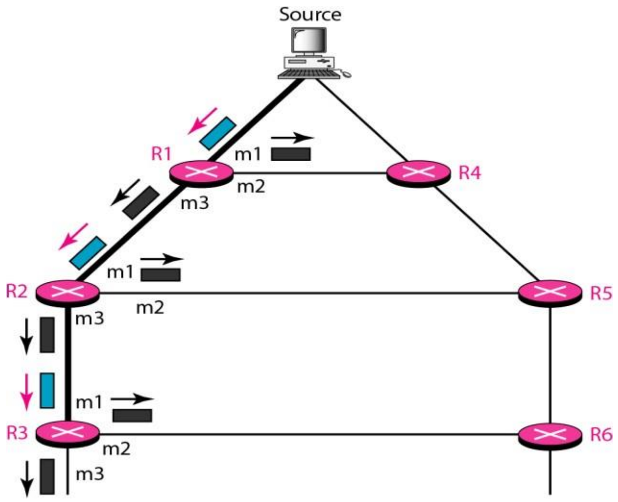


Fig.3.11 DVMRP

Part- A Questions

1. What is hidden terminal problem?
2. What are the responsibilities of routing protocol?
3. What are the major challenges in designing routing protocols?
4. Differentiate proactive and reactive protocols. Write examples for each.
5. List the characteristics of a routing protocol for ad hoc wireless networks.
6. List the major classification of routing protocol for ad hoc wireless network.
7. Based on routing information update mechanism how the routing protocols are classified?
8. How does energy aware routing work?
9. List the classification of routing protocols based on the routing information update mechanism.
10. List the advantages and disadvantages of DSDV routing protocols.
11. What is hybrid routing protocol?
12. List some examples of table driven routing protocols.

13. List the types of on-demand routing protocols
14. What do you mean by time to live (TTL)?
15. What are the advantages and disadvantages of dynamic source routing protocol?
16. Give the difference between Ad hoc on demand Distance vector routing protocol (AODV) anddynamic sequence routing protocol (DSR)

Part- B Questions

1. Explain on demand routing protocol in detail.
2. Explain the major challenges that a routing protocol designed for adhoc wireless networks.
3. List the characteristics of ideal routing protocol for ad hoc wireless network.
4. Discuss table driven protocols with examples.
5. Explain multicast routing algorithms in detail.

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

SCHOOL OF COMPUTING

DEPARTMENT OF INFORMATION TECHNOLOGY

UNIT - 3

SITA1501 – WIRELESS SENSOR NETWORKS AND ARCHITECTURE

UNIT - 3 SYSTEM ARCHITECTURE AND DESIGN ISSUES

Syllabus:

Design Constraints for Routing in Wireless Sensor Networks, Classification of Routing Protocols in Wireless Sensor Networks-Hierarchy Role of Nodes in the Network, Data Delivery Model, Optimization Techniques for Routing in Wireless Sensor Networks, Application of the Optimization Techniques: Routing Protocols

3.1 Design Constraints for Routing in Wireless Sensor Networks

Due to the reduced computing, radio and battery resources of sensors, routing protocols in wireless sensor networks are expected to fulfill the following requirements:

- *Autonomy:* The assumption of a dedicated unit that controls the radio and routing resources does not stand in wireless sensor networks as it could be an easy point of attack. Since there will not be any centralized entity to make the routing decision, the routing procedures are transferred to the network nodes.
- *Energy Efficiency*: Routing protocols should prolong network lifetime while maintaining a good grade of connectivity to allow the communication between nodes. It is important to note that the battery replacement in the sensors is infeasible since most of the sensors are randomly placed. Under some circumstances, the sensors are not even reachable. For instance, in wireless underground sensor networks, some devices are buried to make them able to sense the soil.
- *Scalability:* Wireless sensor networks are composed of hundreds of nodes so routing protocols should work with this number of nodes.
- Resilience: Sensors may unpredictably stop operating due to environmental reasons or to the battery consumption. Routing protocols should cope with this eventuality so when a current-in-use node fails, an alternative route could be discovered.
- *Device Heterogeneity:* Although most of the civil applications of wireless sensor network rely on homogenous nodes, the introduction of different kinds of sensors could report significant benefits. The use of nodes with different processors, transceivers, power units or sensing components may improve the characteristics of the network. Among other, the scalability of the network, the energy drainage or the bandwidth are potential candidates to benefit from the heterogeneity of nodes.
- *Mobility Adaptability:* The different applications of wireless sensor networks could demand nodes to cope with their own mobility, the mobility of the sink or the mobility of the event to sense. Routing protocols should render appropriate support for these movements.

3.2 Classification of Routing Protocols in Wireless Sensor Networks

Taking into account their procedures, routing protocols can be roughly classified according to the following criteria.

3.2.1 Hierarchy Role of Nodes in the Network

In the flat schemes, all sensor nodes participate with the same role in the routing procedures. On the other hand, the hierarchical routing protocols classify sensor nodes according to their functionalities. The network is then divided into groups or clusters. A leader or a cluster head is selected in the group to coordinate the activities within the cluster and to communicate with nodes outside the own cluster. The differentiation of nodes can be static or dynamic.

3.2.2 Data Delivery Model

Depending on the application, data gathering and interaction in wireless sensor networks could be accomplished on several ways. The data delivery model indicates the flow of information between the sensor nodes and the sink. The data delivery models are divided into the following classes: continuous, event-driven, query-driven or hybrid. In the continuous model, the nodes periodically transmit the information that their sensors are detecting at a pre-specified rate. In contrast, the query-driven approaches force nodes to wait to be demanded in order to inform about their sensed data. In the event-driven model, sensors emit their collected data when an event of interests occurs. Finally, the hybrid schemes combine the previous strategies so sensors periodically inform about the collected data but also response to queries. Additionally, they are also programmed to inform about events of interest.

3.3 Optimization Techniques for Routing in Wireless Sensor Networks

The particular characteristics of wireless sensor networks and their constraints have prompted the need for specific requirements to routing protocols. When compared to mobile ad hoc networks routing protocols, the algorithms in wireless sensor networks usually realize the following specifications:

3.3.1 Attribute-based:

In these algorithms, the sink sends queries to certain regions and waits for the response from the sensors located in this area. Following an attribute-value scheme, the queries inform about the required data. The selection of the attributes depends on the application. An important characteristic of these schemes is that the content of the data messages is analysed in each hop to make decisions about routing.

3.3.2 Energy Efficiency:

Multiple routes can communicate a node and the sink. The aim of energy-aware algorithms is to select those routes that are expected to maximize the network lifetime. To do so, the routes composed of nodes with higher energy resources are preferred.

3.3.3 Data Aggregation:

Data collected in sensors are derived from common phenomena so nodes in a close area usually share similar information. A way to reduce energy consumption is data aggregation. Aggregation consists of suppressing redundancy in different data messages. When the suppression is achieved by some signal processing techniques, this operation is called data fusion.

3.3.4 Addressing Scheme:

Wireless sensor networks are formed by a significant number of nodes so the manual assignation of unique identifiers is infeasible. The use of the MAC address or the GPS coordinates is not recommended as it introduces a significant payload. However, network-wide unique addresses are not needed to identify the destination node of a specific packet in wireless sensor networks. In fact, attribute-based addressing fits better with the specificities of wireless sensor networks. In this case, an attribute such as node location and sensor type is used to identify the final destination. Concerning these identifiers, two different approaches have been proposed. Firstly, the ID reuse scheme allows identifiers to be repeated in the network but keeping their uniqueness in close areas. In this way, a node knows that its identifier is unique in a k-hop neighborhood, being k a parameter to configure. On the other hand, the field-wide unique ID schemes guarantee that the identifiers are unique in the whole application. With this assumption, other protocols such as routing, MAC or network configurations can be simultaneously used.

3.3.5 Location-based:

When this technique is used, a node decides the transmission route according to the localization of the final destination and the positions of some other nodes in the network.

3.3.6 Multipath Communication:

With this technique, nodes use multiple paths from an origin to a destination in the network. As multipath communications are intended to increase the reliability and the performance of the network, these paths should not share any link. Multipath communications can be accomplished in two ways. Firstly, one path is established as the active communication routing while the other paths are stored for future need, i.e. when the current active path is broken. On the other hand, it is also possible to distribute the traffic among the multiple paths.

3.3.7 Quality of Service:

The network application business and its functionalities prompt the need for ensuring a QoS (Quality of Service) in the data exchange. In particular, effective sample rate, delay bounded and temporary precision are often required. Satisfying them is not possible for all the routing protocols as the demands may be opposite to the protocol principles. For instance, a routing protocol could be designed to extend the network lifetime while an application may demand an effective sample rate which forces periodic transmissions and, in turn, periodic energy consumptions. Figure 3.1 shows the relation of QoS and its dependence to the routing protocol goal and to the routing protocol strategy.

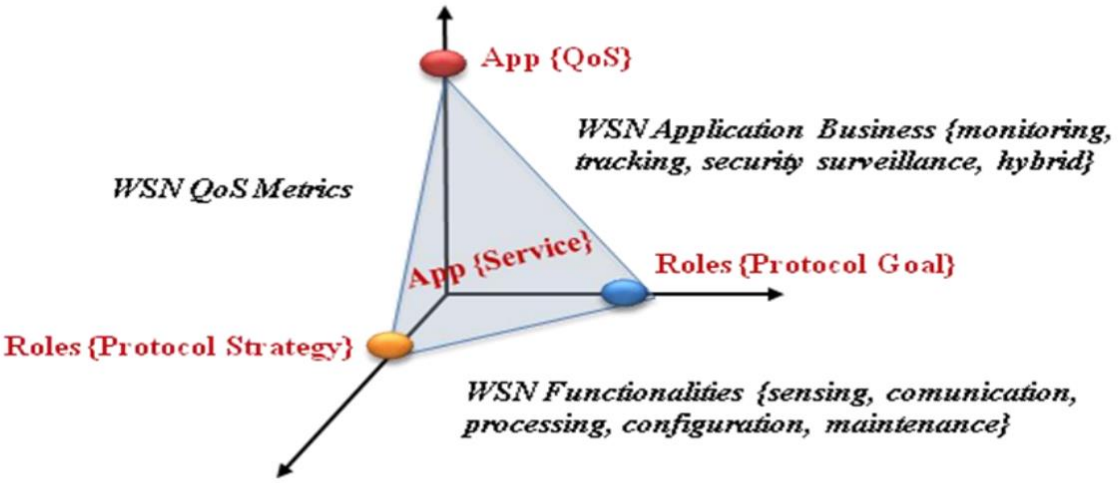
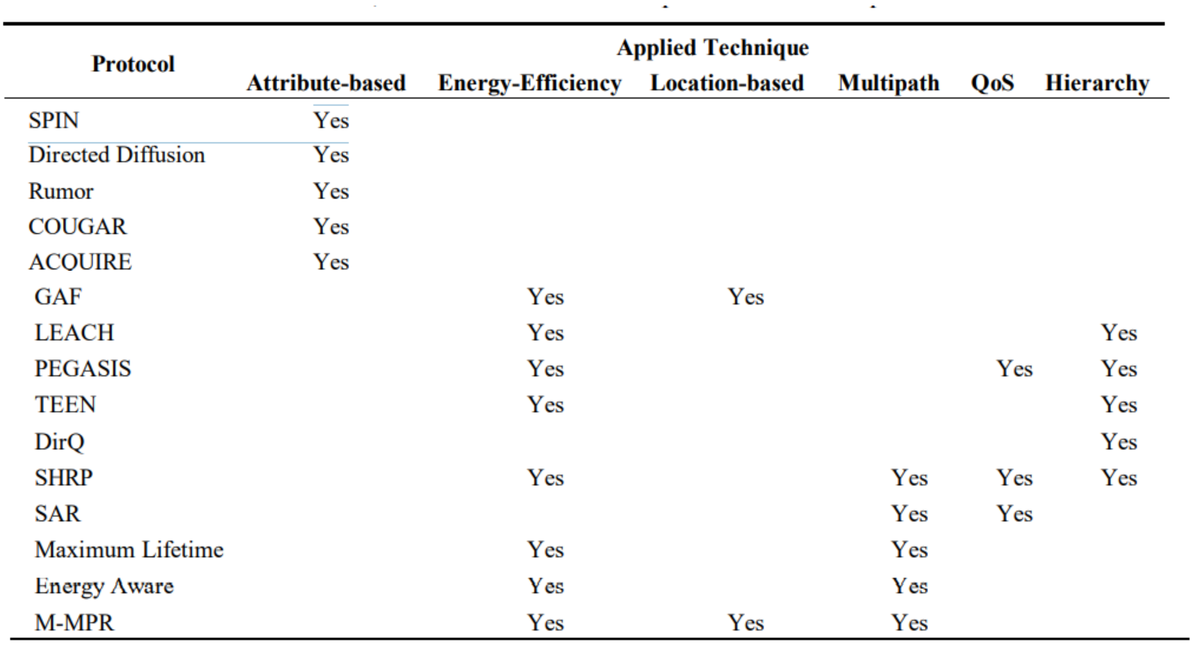

App {QoS}
WSN Application Business {monitoring, tracking, security surveillance, hybrid}
WSN QoS Metrics
App {Service}
Roles {Protocol Goal}
Roles {Protocol Strategy}
WSN Functionalities {sensing, comunication, processing, configuration, maintenance}

Figure 3.1 Relation of QoS and Routing Protocol Goal and Strategy

3.4 Application of the Optimization Techniques: Routing Protocols

By means of representative routing protocols, we present how the attribute-based, the geographic and the multipath techniques are usually applied into wireless sensor networks. Although the hierarchy is commonly considered a parameter for the classification of protocols, we will study it as an important technique used in routing protocols and therefore, we will also analyse some representative hierarchical routing protocols.

Table 3.1 Summary of the characteristics of the routing protocols



Protocol	Applied Technique					
	Attribute-based	**Energy-Efficiency**	**Location-based**	**Multipath**	**QoS**	**Hierarchy**
SPIN	Yes					
Directed Diffusion	Yes					
Rumor	Yes					
COUGAR	Yes					
ACQUIRE	Yes					
GAF		Yes	Yes			
LEACH		Yes				Yes
PEGASIS		Yes			Yes	Yes
TEEN		Yes				Yes
DirQ						Yes
SHRP		Yes		Yes	Yes	Yes
SAR				Yes	Yes	
Maximum Lifetime		Yes		Yes		
Energy Aware		Yes		Yes		
M-MPR		Yes	Yes	Yes		

3.4.1 Attribute-based or Data-centric Routing Protocols

In this category, the following protocols stand out:

3.4.1.1. SPIN (Sensor Protocols for Information via Negotiation)

SPIN (Sensor Protocols for Information via Negotiation), that efficiently disseminate information among sensors in an energy-constrained wireless sensor network. Nodes running a SPIN communication protocol name their data using high-level data descriptors, called meta-data. They use meta-data negotiations to eliminate the transmission of redundant data throughout the network. In addition, SPIN nodes can base their communication decisions both upon application-specific knowledge of the data and upon knowledge of the resources that are available to them. This allows the sensors to efficiently distribute data given a limited energy supply. Four specific SPIN protocols were simulated and analyzed: SPIN-PP and SPINEC, which are optimized for a point-to-point network, and SPIN-BC and SPIN-RL, which are optimized for a broadcast network.

In point-to-point networks, the sender announces that it has new data with an advertisement message to each neighbor. When the neighbor receives the message, the node checks the metadata to know if it already stores the data item. If the neighbor is interested in the information, it responds with a request message. Upon receiving it, the sender transmits the information in a data message. The neighbor that receives the data, inform about its availability to its own neighbors with an advertisement message. The three-handshake protocol is then repeated. The described process is known as SPIN-PP. The algorithm SPIN-EC introduces a technique in the nodes so when their current energy resources do not exceed a predetermined threshold that allows them to complete the three hand-shake protocols, they do not participate in the process. The SPIN-BC and SPIN-RL variants extend the algorithm to support broadcast transmissions. In this way, one advertisement message can reach all the neighbors. In this case, the neighbors do not respond immediately with a request message but they must wait a random time. To optimize the process, a node different from the advertising one cancels its own request message when it detects another similar message. Taking into account the broadcast transmission, the advertising node also responds with just one data message even when it has received multiple request messages.

Additionally, SPIN-RL incorporates some reliability functionalities. Specifically, nodes keep track of the advertisement messages that they receive and their corresponding originators. If they send a request message, but the announcing node does not respond in a given interval, the node asks again for the data with a request message. Comparing the SPIN protocols to other possible approaches, the SPIN protocols can deliver 60% more data for a given amount of energy than conventional approaches in a point-to-point network and 80% more data for a given amount of energy in a broadcast network. In addition, in terms of dissemination rate and energy usage, the SPIN protocols perform close to the theoretical optimum in both point-to-point and broadcast networks. One of the major advantages of these protocols is that nodes are only required to know its 1-hop neighborhood.

3.4.1.2 Directed Diffusion

As a data-centric protocol, applications in sensors label the data using attribute-value pairs. A node that demands the data generates a request where an interest is specified according to the attribute-value based scheme defined by the application. The sink usually injects an interest in the network for each application task. The nodes update an internal interest cache with the interest messages received. The nodes also keep a data cache where the recent data messages are stored. This structure helps on determining the data rate. On receiving this message, the nodes establish a reply link to the originator of the interest. This link is called gradient and it is characterized by the data rate, duration and expiration time. Additionally, the node activates its sensors to collect the intended data. The reception of an interest message makes the node establish multiple gradients (or first hop in a route) to the sink. In order to identify the optimum gradient, positive and negative reinforcements are used. There algorithm works with two types of gradients: exploratory and data gradients. Exploratory gradients are intended for route set-up and repair whereas data gradients are used for sending real data.

3.4.1.3 Rumor

In this algorithm, the queries generated by the sink are propagated among the nodes that have observed an event related to the queries. To do so, a node that observes an event inject a longlived packet called agent. The agents are propagated in the network so distant nodes have knowledge about which nodes have perceived certain events. To optimize the behavior of agents, when an agent reaches a node which has detected another event, the agent is still forwarded but aggregating the new discovered event. Additionally, the agents maintain a list of the recent visited nodes so loops are partially avoided. On reception of agents, nodes can acquire updated information about the events in the network. This knowledge is reflected in the nodes event caches. By using the event cache, a node can conveniently send a query message. However, some nodes may not be aware of the events originator. Under these circumstances, the query is sequentially propagated to one of the neighbors selected randomly. Once the query arrives at a node with an entry related to the demanded event in its event cache, the query is then forwarded through the learnt path. Following this procedure, the cost of flooding the network with the query is clearly suppressed.

3.4.1.4 COUGAR

Under this approach, the network is foreseen as a distributed database where some nodes containing the information are temporary unreachable. Since node stores historic values, the network behaves as a data warehouse. Additionally, it is worth noting that poor propagation conditions may lead to the storage of erroneous information in the nodes. Taking into account this circumstance, COUGAR provides a SQL-like interface extended to incorporate some clauses to model the probability distribution. The sink is responsible for generating a query plan which provides the hints to select a special node called the leader. The network leaders perform aggregation and transmit the results to the sink.

3.4.1.5. ACQUIRE

Active Query Forwarding in Sensor Networks algorithm also considers the wireless sensor network as a distributed database. In this scheme, a node injects an active query packet into the network. Neighboring nodes that detects that the packet contains obsolete information, emits an update message to the node. Then, the node randomly selects a neighbor to propagate the query which needs to resolve it. As the active query progress through network, it is progressively resolved into smaller and smaller components until it is completely solved. Then, the query is returned back to the querying node as a completed response.

3.4.2 Geographical Routing Protocols

These algorithms take advantage of the location information to make routing techniques more efficient. Specifically, neighbors exchange information about their location so when a node needs to forward a packet, it sends it to the neighbor which is assumed to be closest to the final destination. To operate, the source inserts the destination’s coordinates in the packets. The location information used in geographical algorithms can be derived from specific devices such as GPS or it can be modeled by virtual coordinates. Concerning geographical protocols, geocasting is the process by which a packet is delivered to the nodes placed in an area. This primitive is especially suitable in wireless sensor networks since the sink usually demands information from the nodes that are in a zone. The zone can be statically determined by the source node or it can be constructed dynamically by the relaying nodes in order to avoid some nodes that may cause a detour. On the other hand, in geographic-based rendezvous mechanisms, geographical locations are used as a rendezvous place for providers and seekers of information. Geographic-based rendezvous mechanisms can be used as an efficient means for service location and resource discovery, in addition to data dissemination and access in wireless sensor networks. The most popular forwarding techniques in geographical routing protocols are:

3.4.2.1 Greedy Algorithms

Under this approach, a node decides about the transmission path based on the position of its neighbors. To proceed, the source compares the localization of the destination with the coordinates of its neighbors. Then, it propagates the message to the neighbor which is closest to the final destination. The process is repeated until de packet reaches the intended destination. Several metrics related to the concept of closeness have been proposed for this context. Among them, the most popular metrics are the Euclidean distance and the projected line joining the relaying node and the destination. With this strategy, flooding processes are restricted to one-hop and the network is able to adapt proficiently to the topological changes. This simple forwarding rule is modified according to the reliability of links. In this proposal, the unreliable neighbors are not taken into account for the retransmissions. On the other hand, the geographic information is also used in SPEED (Statelss Protocol for End-to-End Delay) to estimate the delay of the transmitted packets. Similar to this algorithm, the greedy algorithm with the „most-forward-within-R‟ forwarding technique opts to select the most distant neighbor of the packet holder which is closer to the final destination as the next hop. In contrast, the „nearest-forward-process‟ chooses the nearest neighbor that is closer to the intended destination as the next relaying node. The main limitation of the greedy algorithms is that the transmission may fail when the current holder of the message has no neighbors closer to the destination than itself. This could occur even when there is a feasible path between the two extremes, for instance, when an obstacle is present. Aiming at overcoming this drawback, the “right hand‟ rule is suggested.

3.4.2.2 GAF (Geographic Adaptive Fidelity)

This protocol aims at optimizing the performance of wireless sensor networks by identifying equivalent nodes with respect to forwarding packets. Two nodes are considered to be equivalent when they maintain the same set of neighbor nodes and so they can belong to the same communication routes. Source and destination in the application are excluded from this characterization. To identify equivalent nodes, their positions are necessary. Additionally, a virtual grid is constructed. This grid is formed by cells whose size allows to state that all the nodes in one cell can directly communicate with the nodes belonging to adjacent cells and vice versa. In this way, the nodes in a cell are equivalent. Nodes identify equivalent nodes by the periodic exchange of discovery messages with the nodes in their cells. With the information contained in these messages, the nodes negotiate which one is going to support the communications. The other nodes will stay powered off. With this procedure, the routing fidelity is kept, that is, there is uninterrupted connectivity between communicating nodes. However, the elected node periodically rotates for fair energy consumption. To do so, the nodes wake up periodically.

3.4.3 Hierarchical Routing Protocols

The main objective of hierarchical routing is to reduce energy consumption by classifying nodes into clusters. In each cluster, a node is selected as the leader or the cluster head. The different schemes for hierarchical routings mainly differ in how the cluster head is selected and how the nodes behave in the inter and intra-cluster domain

3.4.3.1 LEACH (Low Energy Adaptive Clustering Hierarchy)

In LEACH the role of the cluster head is periodically transferred among the nodes in the network in order to distribute the energy consumption. The performance of LEACH is based on rounds. Then, a cluster head is elected in each round. For this election, the number of nodes that have not been cluster heads and the percentage of cluster heads are used. Once the cluster head is defined in the setup phase, it establishes a TDMA schedule for the transmissions in its cluster. This scheduling allows nodes to switch off their interfaces when they are not going to be employed. The cluster head is the router to the sink and it is also responsible for the data aggregation. As the cluster head controls the sensors located in a close area, the data aggregation performed by this leader permits to remove redundancy. A centralized version of this protocol is LEACH-C. This scheme is also based on time rounds which are divided into the set-up phase and the steady-phase. In the set-up phase, sensors inform the base station about their positions and about their energy level. With this information, the base station decides the structure of clusters and their corresponding cluster heads. Since the base station possess a complete knowledge of the status of the network, the cluster structure resulting from LEACH-C is considered an optimization of the results of LEACH.

3.4.3.2. PEGASIS (Power-Efficient Gathering in Sensor Information Systems)

It is considered an optimization of the LEACH algorithm. Rather than classifying nodes in clusters, the algorithm forms chains of the sensor nodes. Based on this structure, each node transmits to and receives from only one closest node of its neighbors. With this purpose, the nodes adjust the power of their transmissions. The node performs data aggregation and forwards it the node in the chain that communicates with the sink. In each round, one node in the chain is elected to communicate with the sink. The chain is constructed with a greedy algorithm.

3.4.3.3 TEEN (Threshold Sensitive Energy Efficient Sensor Network Protocol)

TEEN is other hierarchical protocol for reactive networks that responds immediately to changes in the relevant parameters. In this protocol a clusters head (CH) sends a hard threshold value and a soft one. The nodes sense their environment continuously. The first time a parameter from the attribute set reaches its hard threshold value, the node switches on its transmitter and sends its data. The nodes then transmits data in the current cluster period if the following conditions are true: the current value of the sensed attribute is greater than the hard threshold, and the current value of the sensed attribute differs from sensed value by an amount equal to or greater than the soft threshold. Both strategy looks to reduce energy spend transmitting messages. The main drawback of this scheme is that, if the thresholds are not reached, the nodes will never communicate; the user will not get any data from the network at all and will not come to know even if all the nodes die. Thus, this scheme is not well suited for applications where the user needs to get data on a regular basis

3.4.3.4 DirQ (Directed Query Dissemination)

DirQ [24] aims at optimizing the propagation of queries in a wireless sensor network. The main objective is that the queries are just propagated by the minimum number of nodes that ensure that the queries arrive at the nodes that are able to service the query. To do so, certain information is exchanged in the network. The periodicity of the update messages depend on the rate of variation of the physical parameters that the network is sensing. Then, each node autonomously maintains its own threshold (δ). If a sensor node has a value V of a desired parameter and the next measurement period gets the same or a similar value in the interval between (δ – V, V + δ) then it decides not to send anything to sink. However, if the sink does not receive any message from a specific node then it assumes that this node has a measured value that has not changed much from what has been reported recently. To allow a precise delivery of applications, all network nodes must be capable of storing information which can be considered a disadvantage depending on the amount of information stored in the topology and the number of nodes. DirQ is a protocol suitable for situations where the number of requests is high and times of transmission of requests are known.

3.4.4 Multipath Routing Protocols

In these protocols, a source knows multiple routes to a destination. The routes can be simultaneously used or one of them can be active while the others are maintained for future needs.

3.4.4. SAR (Sequential Assignment Routing)

SAR is one of the first protocols for wireless sensor networks that provide the notion of QoS routing criteria. It is based on the association of a priority level to each packet. Additionally, the links and the routes are related to a metric that characterizes their potential provision of quality of service. This metric is based on the delay and the energy cost. Then, the algorithm creates trees rooted at the one-hop neighbors of the sink. To do so, several parameters such as the packet priority, the energy resources and the QoS metrics are taken into account. The protocol must periodically recalculate the routes to be prepared in case of failure of one of the active nodes.

3.4.4.2 Maximum Lifetime Routing in Wireless Sensor Networks

This algorithm combines the energy consumption optimization with the use of multiple routes. In this algorithm an active route (also called the primary route) is monitored to control its residual energy. Meanwhile other routes can be discovered. If the residual energy of the active route does not exceed the energy of an alternative route, the corresponding secondary route is then used.

3.4.4.3 Energy Aware Routing in Wireless Sensor Networks

Once multiple paths are discovered, this algorithm associates a probability of use to each route. This probability is related to the residual energy of the nodes that form the route but it is also considers the cost of transmitting through that route.

3.4.4.4. M-MPR (Mesh Multipath Routing)

This protocol presents two operation mode. Firstly, in the disjoint MPR (D-MPR) with Selective Forwarding each packet is individually analyzed by the source and it is routed through different routes. Secondly, the D-MPR with data replication is based on the simultaneous emission of multiple copies of the same packet through different routes. Specifically, all the known routes that communicate the source and the destination propagate the packet. For the route discovery, information about the position of the nodes and about their residual energy is exchanged.

Part- A Questions

1. Differentiate unicasting and multicasting.
2. Determine the roles of nodes in WSN.
3. Identify the different kinds of algorithms which can be executed on wireless sensor networks
4. List the criteria by which the routing protocols for WSN are classified.
5. Find out the resource constraints of routing protocols for Wireless Sensor Network.
6. Organize the steps to identify cluster head.
7. Distinguish event-driven and query-driven data delivery models.
8. Compare continuous and hybrid data delivery models.
9. List the various types of data delivery models.
10. Identify the functions of cluster head.

Part- B Questions

1. Identify the design constraints of routing protocols for Wireless Sensor Network and explain in detail.
2. Classify the routing protocols based on hierarchy role of nodes and data delivery model. Explain any one routing protocol for WSN in detail.
3. Organize the requirements of optimization techniques for routing in wireless Sensor Networks.
4. Determine the applications of optimization techniques.

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF COMPUTING

DEPARTMENT OF INFORMATION TECHNOLOGY

UNIT - 4

SITA1501 – WIRELESS SENSOR NETWORKS AND ARCHITECTURE

UNIT 4 ROUTING PROTOCOLS FOR WIRELESS SENSOR NETWORKS

Syllabus:

Introduction, Data Dissemination and Gathering, Routing Challenges and Design Issues in Wireless Sensor Networks Network Scale and Time-Varying Characteristics, Resource Constraints, Sensor Applications Data Models, Routing Strategies in Wireless Sensor Networks: WSN Routing Techniques, Flooding and Its Variants, Sensor Protocols for Information via Negotiation, Low-Energy Adaptive Clustering Hierarchy, Power-Efficient Gathering in Sensor Information Systems, Directed Diffusion, Geographical Routing.

4.1 Introduction

WSNs are extremely versatile and can be deployed to support a wide variety of applications in many different situations, whether they are composed of stationary or mobile sensor nodes. The way these sensors are deployed depends on the nature of the application. In environmental monitoring and surveillance applications, for example, sensor nodes are typically deployed in an ad hoc fashion so as to cover the specific area to be monitored (e.g., C1WSNs). In health care–related applications, smart wearable wireless devices and biologically compatible sensors can be attached to or implanted strategically within the human body to monitor vital signs of the patient under surveillance. Once deployed, sensor nodes self-organize into an autonomous wireless ad hoc network, which requires very little or no maintenance. Sensor nodes then collaborate to carry out the tasks of the application for which they are deployed. Despite the disparity in the objectives of sensor applications, the main task of wireless sensor nodes is to sense and collect data from a target domain, process the data, and transmit the information back to specific sites where the underlying application resides. Achieving this task efficiently requires the development of an energy-efficient routing protocol to set up paths between sensor nodes and the data sink. The path selection must be such that the lifetime of the network is maximized. The characteristics of the environment within which sensor nodes typically operate, coupled with severe resource and energy limitation, make the routing problem very Challenging.

4.2 Data Dissemination and Gathering

The way that data and queries are forwarded between the base station and the location where the target phenomena are observed is an important aspect and a basic feature of WSNs. A simple approach to accomplishing this task is for each sensor node to exchange data directly with the base station. A single-hop-based approach, however, is costly, as nodes that are farther away from the base station may deplete their energy reserves quickly, thereby severely limiting the lifetime of the network. This is the case particularly where the wireless sensors are deployed to cover a large geographical region or where the wireless sensors are mobile and may move away from the base station. To address the shortcomings of the single-hop approach, data exchange between the sensors and the base stations is usually carried out using multihop packet transmission over short communication radius. Such an approach leads to significant energy savings and reduces considerably communication interference between sensor nodes competing to access the channel, particularly in highly dense WSNs. Data forwarding between the sensors where data are collected and the sinks where data are made available is illustrated in Figure 4.1.

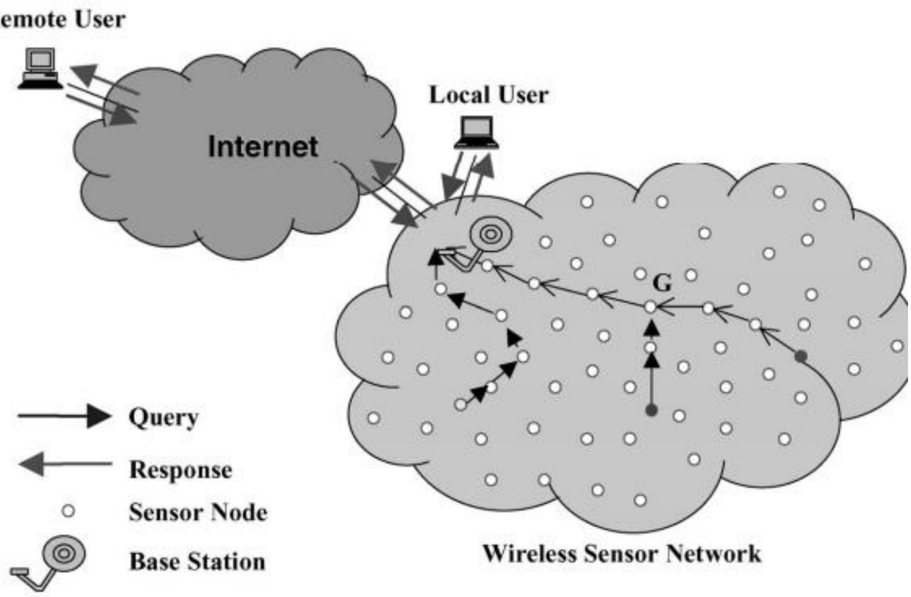


Figure 4.1 Multihop data and query forwarding

In response to queries issued by the sinks or when specific events occur within the area monitored, data collected by the sensors are transmitted to the base station using multihop paths. It is worth noting that depending on the nature of the application, sensor nodes can aggregate data correlated on their way to the base station. In a multihop WSN, intermediate nodes must participate in forwarding data packets between the source and the destination. Determining which set of intermediate nodes is to be selected to form a data-forwarding path between the source and the destination is the principal task of the routing algorithm. In general, routing in large-scale networks is inherently a difficult problem whose solution must address multiple challenging design requirements, including correctness, stability, and optimality with respect to various performance metrics. The intrinsic properties of WSNs, combined with severe energy and bandwidth constraints, bring about additional challenges that must be addressed to satisfy the traffic requirements of the application supported, while extending the lifetime of the network.

4.3 Routing Challenges and Design Issues in Wireless Sensor Networks

Although WSNs share many commonalities with wired and ad hoc networks, they also exhibit a number of unique characteristics which set them apart from existing networks. These unique characteristics bring to sharp focus new routing design requirements that go beyond those typically encountered in wired and wireless ad hoc networks. Meeting these design requirements presents a distinctive and unique set of challenges. These challenges can be attributed to multiple factors, including severe energy constraints, limited computing and communication capabilities, the dynamically changing environment within which sensors are deployed, and unique data traffic models and application-level quality of service requirements.

4.3.1 Network Scale and Time-Varying Characteristics

Sensor nodes operate with limited computing, storage, and communication capabilities under severe energy constraints. Due to large number of conceivable sensor-based applications, the densities of the WSNs may vary widely, ranging from very sparse to very dense. Furthermore, in many applications, the sensor nodes, in some cases numbering in the hundreds if not thousands, are deployed in an ad hoc and often unsupervised manner over wide coverage areas. In these networks, the behavior of sensor nodes is dynamic and highly adaptive, as the need to self-organize and conserve energy forces sensor nodes to adjust their behavior constantly in response to their current level of activity or the lack thereof. Furthermore, sensor nodes may be required to adjust their behavior in response to the erratic and unpredictable behavior of wireless connections caused by high noise levels and radio-frequency interference, to prevent severe performance degradation of the application supported.

4.3.2 Resource Constraints,

Sensor nodes are designed with minimal complexity for large-scale deployment at a reduced cost. Energy is a key concern in WSNs, which must achieve a long lifetime while operating on limited battery reserves. Multihop packet transmission over wireless networks is a major source of power consumption. Reducing energy consumption can be achieved by dynamically controlling the duty cycle of the wireless sensors. The energy management problem, however, becomes especially challenging in many mission-critical sensor applications. The requirements of these applications are such that a predetermined level of sensing and communication performance constraints must be maintained simultaneously. Therefore, a question arises as to how to design scalable routing algorithms that can operate efficiently for a wide range of performance constraints and design requirements. The development of these protocols is fundamental to the future of WSNs.

4.3.3 Sensor Applications Data Models

The data model describes the flow of information between the sensor nodes and the data sink. These models are highly dependent on the nature of the application in terms of how data are requested and used. Several data models have been proposed to address the data-gathering needs and interaction requirements of a variety of sensor applications. A class of sensor applications requires data collection models that are based on periodic sampling or are driven by the occurrence of specific events. In other applications, data can be captured and stored, possibly processed and aggregated by a sensor node, before they are forwarded to the data sink. Yet a third class of sensor applications requires bidirectional data models in which two-way interaction between sensors and data sinks is required. The need to support a variety of data models increases the complexity of the routing design problem. Optimizing the routing protocol for an application’s specific data requirements while supporting a variety of data models and delivering the highest performance in scalability, reliability, responsiveness, and power efficiency becomes a design and engineering problem of enormous magnitude.

4.4 Routing Strategies in Wireless Sensor Networks

The WSN routing problem presents a very difficult challenge that can be posed as a classic trade-off between responsiveness and efficiency. This trade-off must balance the need to accommodate the limited processing and communication capabilities of sensor nodes against the overhead required to adapt to these. In a WSN, overhead is measured primarily in terms of bandwidth utilization, power consumption, and the processing requirements on the mobile nodes. Finding a strategy to balance these competing needs efficiently forms the basis of the routing challenge. Furthermore, the intrinsic characteristics of wireless networks gives rise to the important question of whether or not existing routing protocols designed for ad hoc networks are sufficient to meet this challenge. Routing algorithms for ad hoc networks can be classified according to the manner in which information is acquired and maintained and the manner in which this information is used to compute paths based on the acquired information. Three different strategies can be identified: proactive, reactive, and hybrid. The proactive strategy, also referred to as table driven, relies on periodic dissemination of routing information to maintain consistent and accurate routing tables across all nodes of the network. The structure of the network can be either flat or hierarchical. Flat proactive routing strategies have the potential to compute optimal paths. The overhead required to compute these paths may be prohibitive in a dynamically changing environment. Hierarchical routing is better suited to meet the routing demands of large ad hoc networks. Reactive routing strategies establish routes to a limited set of destinations on demand. These strategies do not typically maintain global information across all nodes of the network. They must therefore, rely on a dynamic route search to establish paths between a source and a destination. This typically involves flooding a route discovery query, with the replies traveling back along the reverse path. The reactive routing strategies vary in the way they control the flooding process to reduce communication overhead and the way in which routes are computed and re-established when failure occurs. Hybrid strategies rely on the existence of network structure to achieve stability and scalability in large networks. In these strategies the network is organized into mutually adjacent clusters, which are maintained dynamically as nodes join and leave their assigned clusters. Clustering provides a structure that can be leveraged to limit the scope of the routing algorithm reaction to changes in the network environment. A hybrid routing strategy can be adopted whereby proactive routing is used within a cluster and reactive routing is used across clusters. The main challenge is to reduce the overhead required to maintain the clusters. In summary, traditional routing algorithms for ad hoc networks tend to exhibit their least desirable behavior under highly dynamic conditions. Routing protocol overhead typically increases dramatically with increased network size and dynamics. A large overhead can easily overwhelm network resources. Furthermore, traditional routing protocols operating in large networks require substantial internodal coordination, and in some cases global flooding, to maintain consistent and accurate information, which is necessary to achieve loop-free routing. The use of these techniques increases routing protocol overhead and convergence times. Consequently, although they are well adapted to operate in environments where the computation and communications capabilities of the network nodes are relatively high compared to sensor nodes, the efficiency of these techniques conflict with routing requirements in WSNs. New routing strategies are therefore required for sensor networks that are capable of effectively managing the trade-off between optimality and efficiency.

4.4.1 WSN Routing Techniques

The design of routing protocols for WSNs must consider the power and resource limitations of the network nodes, the time-varying quality of the wireless channel, and the possibility for packet loss and delay. To address these design requirements, several routing strategies for WSNs have been proposed. One class of routing protocols adopts a flat network architecture in which all nodes are considered peers. A flat network architecture has several advantages, including minimal overhead to maintain the infrastructure and the potential for the discovery of multiple routes between communicating nodes for fault tolerance. A second class of routing protocols imposes a structure on the network to achieve energy efficiency, stability, and scalability. In this class of protocols, network nodes are organized in clusters in which a node with higher residual energy, for example, assumes the role of a cluster head. The cluster head is responsible for coordinating activities within the cluster and forwarding information between clusters. Clustering has potential to reduce energy consumption and extend the lifetime of the network. A third class of routing protocols uses a data-centric approach to disseminate interest within the network. The approach uses attribute-based naming, whereby a source node queries an attribute for the phenomenon rather than an individual sensor node. The interest dissemination is achieved by assigning tasks to sensor nodes and expressing queries to relative to specific attributes. Different strategies can be used to communicate interests to the sensor nodes, including broadcasting, attribute-based multicasting, geo-casting, and anycasting. A fourth class of routing protocols uses location to address a sensor node. Location-based routing is useful in applications where the position of the node within the geographical coverage of the network is relevant to the query issued by the source node. Such a query may specify a specific area where a phenomenon of interest may occur or the vicinity to a specific point in the network environment.

4.4.2 Flooding and Its Variants

Flooding is a common technique frequently used for path discovery and information dissemination in wired and wireless ad hoc networks. The routing strategy is simple and does not rely on costly network topology maintenance and complex route discovery algorithms. Flooding uses a reactive approach whereby each node receiving a data or control packet sends the packet to all its neighbors. After transmission, a packet follows all possible paths. Unless the network is disconnected, the packet will eventually reach its destination. Furthermore, as the network topology changes, the packet transmitted follows the new routes. Figure 4.2 illustrates the concept of flooding in data communications network.

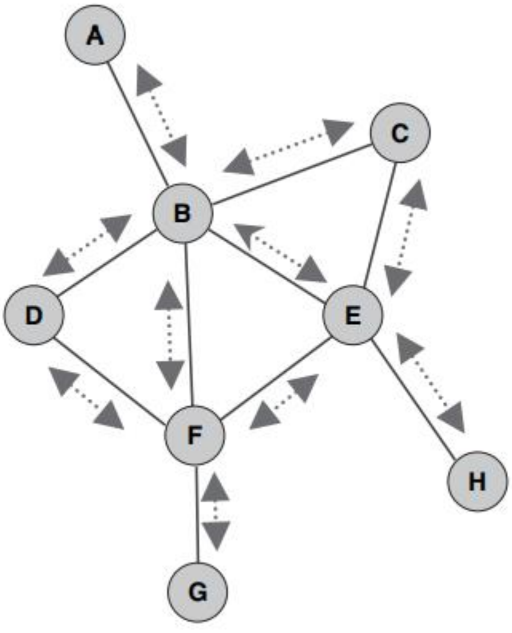


Figure 4.2 Flooding in data communications networks

As shown in the figure, flooding in its simplest form may cause packets to be replicated indefinitely by network nodes. To prevent a packet from circulating indefinitely in the network, a hop count field is usually included in the packet. Initially, the hop count is set to approximately the diameter of the network. As the packet travels across the network, the hop count is decremented by one for each hop that it traverses. When the hop count reaches zero, the packet is simply discarded. A similar effect can be achieved using a time-to-live field, which records the number of time units that a packet is allowed to live within the network. At the expiration of this time, the packet is no longer forwarded. Flooding can be further enhanced by identifying data packets uniquely, forcing each network node to drop all the packets that it has already forwarded. Such a strategy requires maintaining at least a recent history of the traffic, to keep track of which data packets have already been forwarded. Despite the simplicity of its forwarding rule and the relatively low-cost maintenance that it requires, flooding suffers several deficiencies when used in WSNs. The first drawback of flooding is its susceptibility to traffic implosion, as shown in Figure 4.3.

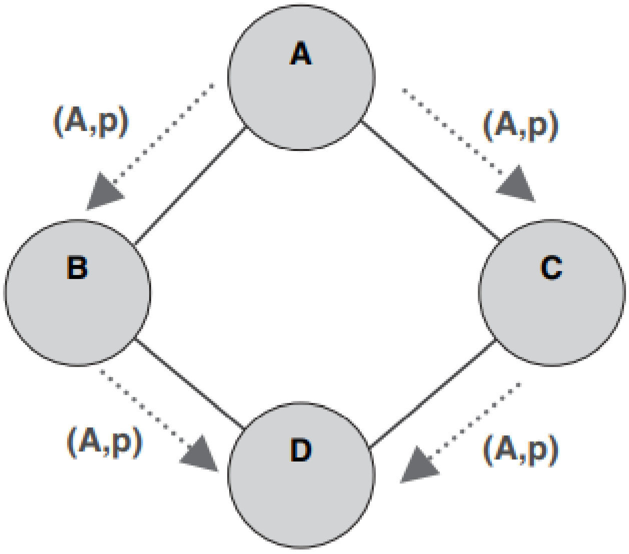


Figure 4.3 Flooding traffic implosion problem

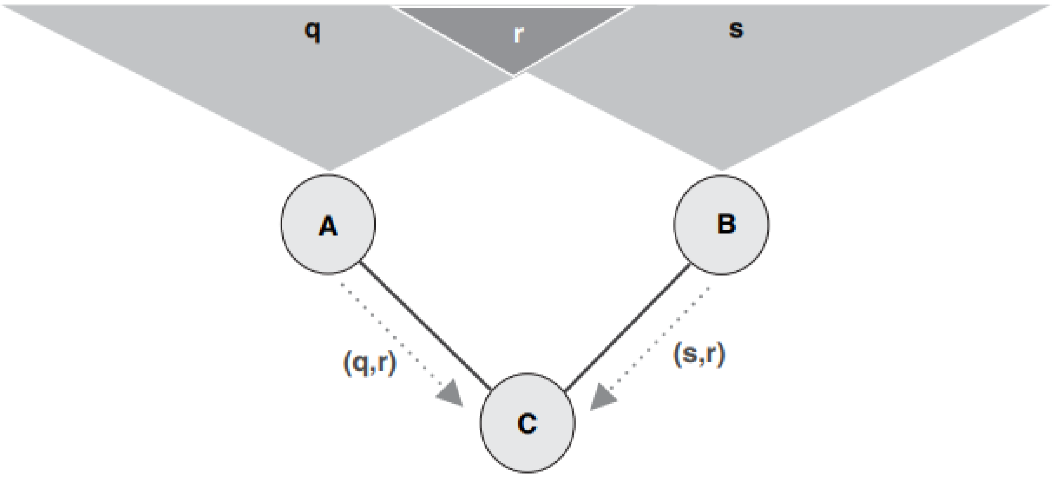


Figure 4.4 Flooding traffic overlapping problem

This undesirable effect is caused by duplicate control or data packets being sent repeatedly to the same node. The second drawback of flooding is the overlap problem to which it gives rise, as depicted in Figure 4.4. Overlapping occurs when two nodes covering the same region send packets containing similar information to the same node. The third and most severe drawback of flooding is resource blindness. The simple forwarding rule that flooding uses to route packets does not take into consideration the energy constraints of the sensor nodes. As such, the node's energy may deplete rapidly, reducing considerably the lifetime of the network. To address the shortcomings of flooding, a derivative approach, referred to as gossiping, has been proposed. Similar to flooding, gossiping uses a simple forwarding rule and does not require costly topology maintenance or complex route discovery algorithms. Contrary to flooding, where a data packet is broadcast to all neighbors, gossiping requires that each node sends the incoming packet to a randomly selected neighbor. Upon receiving the packet, the neighbor selected randomly chooses one of its own neighbors and forwards the packet to the neighbor chosen. This process continues iteratively until the packet reaches its intended destination or the maximum hop count is exceeded. Gossiping avoids the implosion problem by limiting the number of packets that each node sends to its neighbor to one copy. The latency that a packet suffers on its way to the destination may be excessive, particularly in a large network. This is caused primarily by the random nature of the protocol, which, in essence, explores one path at a time.

4.4.3 Sensor Protocols for Information via Negotiation

Sensor protocols for information via negotiation (SPIN) is a data-centric negotiation-based family of information dissemination protocols for WSNs. The main objective of these protocols is to efficiently disseminate observations gathered by individual sensor nodes to all the sensor nodes in the network. Simple protocols such as flooding and gossiping are commonly proposed to achieve information dissemination in WSNs. Flooding requires that each node sends a copy of the data packet to all its neighbors until the information reaches all nodes in the network. Gossiping, on the other hand, uses randomization to reduce the number of duplicate packets and requires only that a node receiving a data packet forward it to a randomly selected neighbor. The simplicity of flooding and gossiping is appealing, as both protocols use simple forwarding rules and do not require topology maintenance. The performance of these algorithms in terms of packet delay and resource utilization, however, quickly deteriorates with the size of the network and the traffic load. This performance drawback is typically caused by traffic implosion and geographical overlapping. Traffic implosion results in multiple copies of the same data being delivered to the same sensor node. Geographical overlapping, on the other hand, causes nodes covering the same geographical area to disseminate, unnecessarily, similar data information items to the network sensor nodes. Simple protocols such as flooding and gossiping do not alter their behavior to adapt communication and computation to the current state of their energy resource. This lack of resource awareness and adaptation may reduce the lifetime of the network considerably, as highly active nodes may rapidly deplete their energy resources. The main objective of SPIN and its related family members is to address the shortcomings of conventional information dissemination protocols and overcome their performance deficiencies. The basic tenets of this family of protocols are data negotiation and resource adaptation. Semantic-based data negotiation requires that nodes running SPIN ‘‘learn’’ about the content of the data before any data are transmitted between network nodes. SPIN exploits data naming, whereby nodes associate metadata with data they produce and use these descriptive data to perform negotiations before transmitting the actual data. A receiver that expresses interest in the data content can send a request to obtain the data advertised. This form of negotiation assures that data are sent only to interested nodes, thereby eliminating traffic implosion and reducing significantly the transmission of redundant data throughout the network. Furthermore, the use of meta data descriptors eliminates the possibility of overlap, as nodes can limit their requests to name only the data that they are interested in obtaining. Resource adaptation allows sensor nodes running SPIN to tailor their activities to the current state of their energy resources. Each node in the network can probe its associated resource manager to keep track of its resource consumption before transmitting or processing data. When the current level of energy becomes low, the node may reduce or completely eliminate certain activities, such as forwarding thirdparty metadata and data packets. The resource adaptation feature of SPIN allows nodes to extend their longevity and consequently, the lifetime of the network. To carry out negotiation and data transmission, nodes running SPIN use three types of messages. The first message type, ADV, is used to advertise new data among nodes. A network node that has data to share with the remaining nodes of the network can advertise its data by first transmitting an ADV message containing the metadata describing the data. The second message type, REQ, is used to request an advertised data of interest. Upon receiving an ADV containing metadata, a network node interested in receiving specific data sends a REQ message the metadata advertising node, which then delivers the data requested. The third message type, DATA, contains the actual data collected by a sensor, along with a metadata header. The data message is typically larger than the ADV and REQ messages. The latter messages only contain metadata that are often significantly smaller than the corresponding data message.

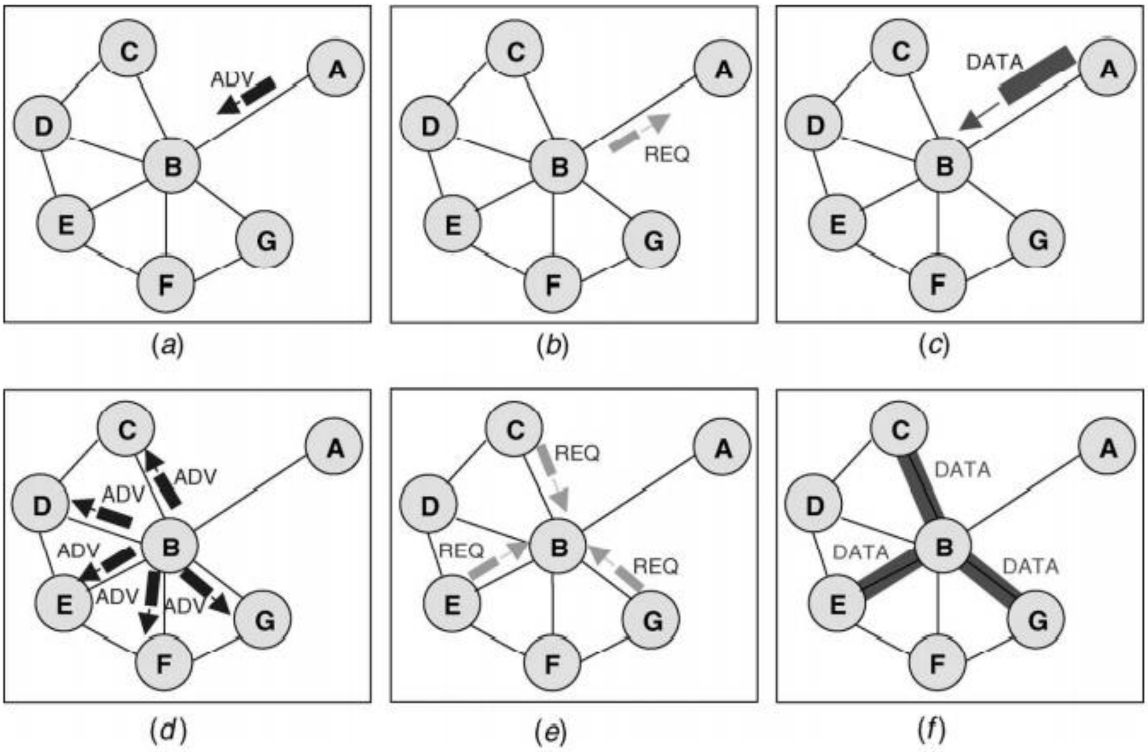


Figure 4.5 SPIN basic protocol operations

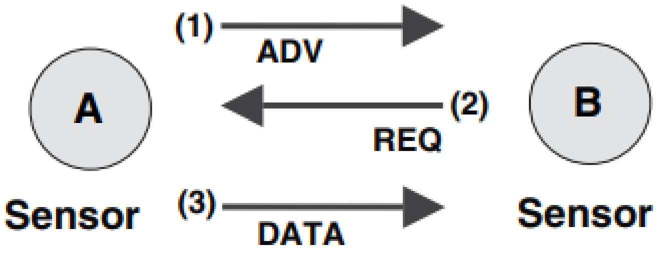


Figure 4.6 SPIN-PP three-way handshake protocol

Limiting the redundant transmission of data messages using semantic-based negotiation can result in significant reduction of energy consumption. The basic behavior of SPIN is illustrated in Figure 6.6, in which the data source, sensor node A, advertises its data to its immediate neighbor, sensor node B, by sending an ADV message containing the metadata describing its data. Node B expresses interest in the data advertised and sends a REQ message to obtain the data. Upon receiving the data, node B sends an ADV message to advertise the newly received data to its immediate neighbors. Only three of these neighbors, nodes C, E, and G, express interest in the data. These nodes issue a REQ message to node B, which eventually delivers the data to each of the requesting nodes. The simplest version of SPIN, referred to as SPIN-PP, is designed for a point-to-point communications network. The three-step handshake protocol used by SPIN-PP is depicted in Figure 6.7. In step 1, the node holding the data, node A, issues an advertisement packet (ADV). In step 2, node B expresses interest in receiving the data by issuing a data request (REQ). In step 3, node A responds to the request and sends a data packet to node B. This completes the three-step handshake procedure. SPIN-PP uses negotiation to overcome the implosion and overlap problems of the traditional flooding and gossiping protocols. A simulation-based performance study of SPIN-1 shows that the protocol reduces energy consumption by a factor of 3.5 compared to flooding. The protocol also achieves high data dissemination rates, nearing the theoretical optimum. An extension of this basic protocol, SPIN-EC, additionally incorporates a thresholdbased resource-awareness mechanism to complete data negotiation. When its energy level approaches the low threshold, a node running SPIN-EC reduces its participation in the protocol operations. In particular, a node engages in protocol operations only if it concludes that it can complete all the stages of the protocol operations without causing its energy level to decrease below the threshold.

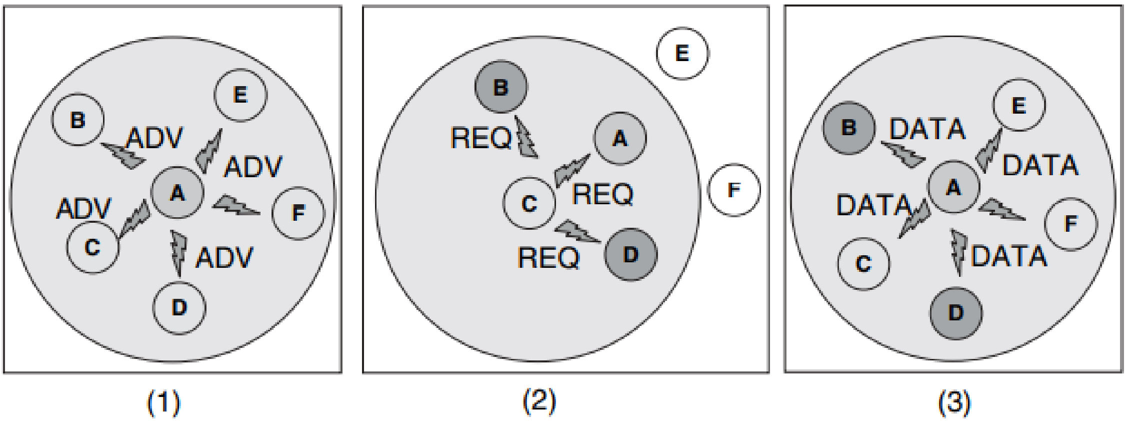


Figure 4.7 SPIN-BC protocol basic operations

Consequently, if a node receives an advertisement, it does not send out an REQ message if it determines that its energy resource is not high enough to transmit an REQ message and receive the corresponding DATA message. The simulation results of this protocol show that SPIN-EC disseminates 60% more data per unit energy than flooding. Furthermore, the data show that SPIN-EC comes very close to the ideal amount of data that can be disseminated per unit energy. Both SPIN-PP and SPIN-EC are designed for point-to-point communication. A third member of the SPIN family, SPIN-BC, is designed for broadcast networks. In these networks, nodes share a single channel for communications. In this class of networks, when a node sends out a data packet on the broadcast channel, the packet transmitted is received by all the other nodes within a certain range of the sending node. The SPIN-BC protocol takes advantage of the broadcasting capability of the channel and requires that a node which has received an ADV message does not respond immediately with an REQ message. Instead, the node waits for a certain amount of time, during which it monitors the communications channel. If the node hears an REQ message issued by another node which is interested in receiving the data, it cancels its own request, thereby eliminating any redundant requests for the same message. Furthermore, upon receiving an REQ message, the advertising node sends the data message only once, even when it receives multiple requests for the same message. The basic operations of the SPIN-BC protocol are depicted in Figure 6.8. In this configuration, the node holding the data, node A, sends a ADV packet to advertise the data to its neighbors. All nodes hear the advertisement, but node C is first to issue a REQ packet to request the data from node A. Nodes B and D hear the broadcast request and refrain from issuing their own REQ packets. Nodes E and F either have no interest in the data advertised or intentionally delay their requests. Upon hearing node C's request, node A replies by sending the data packet. All nodes within the transmission range of A receive the data packet, including nodes E and F. In broadcast environments, SPIN-BC has the potential to reduce energy consumption by eliminating redundant exchange of data requests and replies. The last protocol of the SPIN family, SPIN-RL, extends the capabilities of SPIN-BC to enhance its reliability and overcome message transmission errors caused by a lossy channel. Enhanced reliability is achieved by periodic broadcasting of ADV and REQ messages. Each node in SPIN-BC keeps track of the advertisements it hears and the nodes where these advertisements originate. If a node requesting specific data of interest does not receive the data requested within a certain period of time, it sends the request again. Furthermore, improved reliability can be provided by readvertising metadata periodically. Finally, SPIN-RL nodes limit the frequency with which they resend the data messages. After sending out a data message, a node waits for a certain time period before it responds to other requests for the same data message. The SPIN protocol family addresses the major drawbacks of flooding and gossiping. Simulation results show that SPIN is more energy efficient than flooding or gossiping. Furthermore, the results also show that the rate at which SPIN disseminates data is greater than or equal to the rate of either of these protocols. SPIN achieves these gains by localizing topology changes and eliminating dissemination of redundant information through semantic negotiation. It is worth noting, however, that localized negotiation may not be sufficient to cover the entire network and ensure that all interested nodes receive the data advertisement and eventually, the data of interest. Such a situation may occur if intermediate nodes may not express interest in the data and drop the corresponding ADV message upon receiving it. This shortcoming may prevent the use of SPIN for specific applications such as monitoring for intrusion detection and critical infrastructure protection.

4.4.4 Low-Energy Adaptive Clustering Hierarchy

Low-energy adaptive clustering hierarchy (LEACH) is a routing algorithm designed to collect and deliver data to the data sink, typically a base station. The main objectives of LEACH are:

- Extension of the network lifetime
- Reduced energy consumption by each network sensor node
- Use of data aggregation to reduce the number of communication messages

To achieve these objectives, LEACH adopts a hierarchical approach to organize the network into a set of clusters. Each cluster is managed by a selected cluster head. The cluster head assumes the responsibility to carry out multiple tasks. The first task consists of periodic collection of data from the members of the cluster. Upon gathering the data, the cluster head aggregates it in an effort to remove redundancy among correlated values. The second main task of a cluster head is to transmit the aggregated data directly to the base station. The transmission of the aggregated data is achieved over a single hop. The network model used by LEACH is depicted in Figure 4.8. The third main task of the cluster head is to create a TDMA-based schedule whereby each node of the cluster is assigned a time slot that it can use for transmission. The cluster head advertises the schedule to its cluster members through broadcasting. To reduce the likelihood of collisions among sensors within and outside the cluster, LEACH nodes use a code-division multiple access–based scheme for communication.

The basic operations of LEACH are organized in two distinct phases. These phases are illustrated in Figure 4.9. The first phase, the setup phase, consists of two steps, cluster-head selection and cluster formation. The second phase, the steady-state phase, focuses on data collection, aggregation, and delivery to the base station. The duration of the setup is assumed to be relatively shorter than the steady-state phase to minimize the protocol overhead.

At the beginning of the setup phase, a round of cluster-head selection starts. The cluster-head selection process ensures that this role rotates among sensor nodes, thereby distributing energy consumption evenly across all network nodes. To determine if it is its turn to become a cluster head, a node, n, generates a random number, v, between 0 and 1 and compares it to the cluster-head selection threshold, T(n). The node becomes a cluster head if its generated value, v, is less than T(n). The cluster-head selection threshold is designed to ensure with high probability that a predetermined fraction of nodes, P, is elected cluster heads at each round. Further, the threshold ensures that nodes which served in the last 1/P rounds are not selected in the current round.

To meet these requirements, the threshold T(n) of a competing node n can be expressed as follows:

$$T(n) = \begin{cases} 0 & \text{if } n \notin G \\ \dfrac{P}{1 - P(r \bmod (1/P))} & \forall n \in G \end{cases}$$

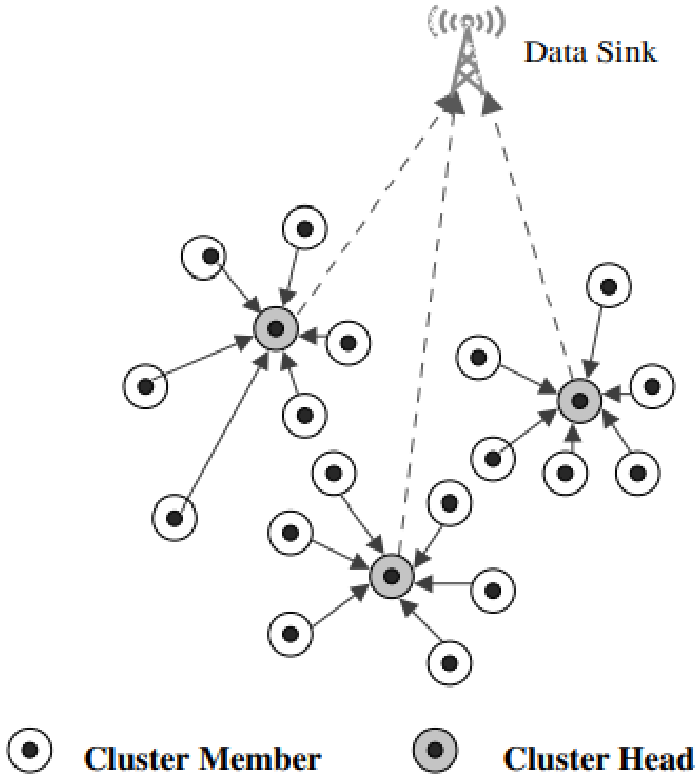


Figure 4.8 LEACH network model

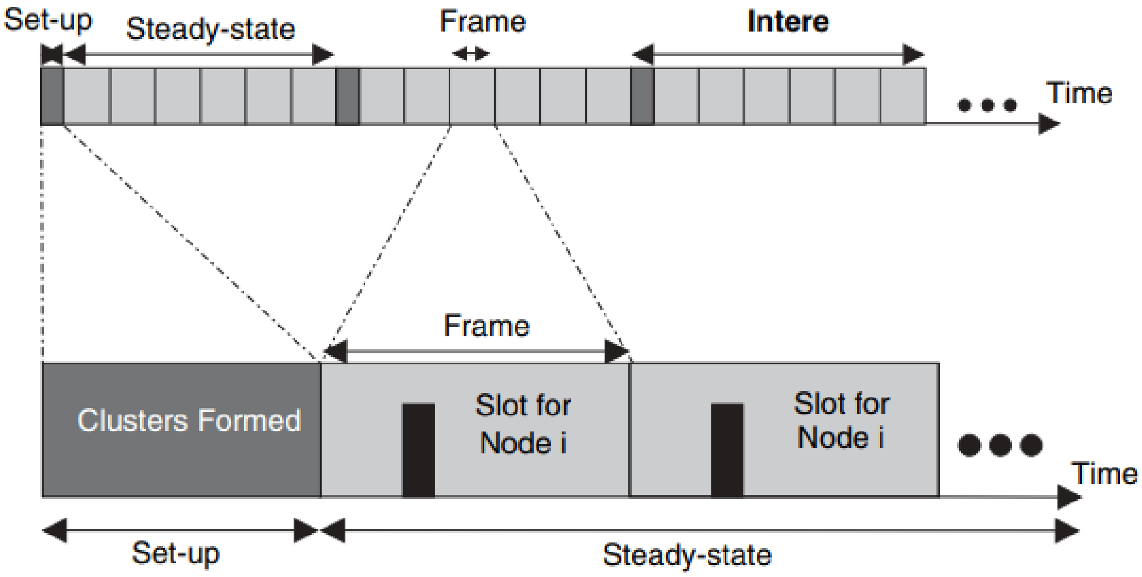


Figure 4.9 LEACH phases

The variable G represents the set of nodes that have not been selected to become cluster heads in the last 1=P rounds, and r denotes the current round.

The predefined parameter, P, represents the cluster-head probability. It is clear that if a node has served as a cluster head in the last 1=P rounds, it will not be elected in this round. At the completion of the cluster-head selection process, every node that was selected to become a cluster head advertises its new role to the rest of the network. Upon receiving the cluster-head advertisements, each remaining node selects a cluster to join. The selection criteria may be based on the received signal strength, among other factors. The nodes then inform their selected cluster head of their desire to become a member of the cluster. Upon cluster formation, each cluster head creates and distributes the TDMA schedule, which specifies the time slots allocated for each member of the cluster. Each cluster head also selects a CDMA code, which is then distributed to all members of its cluster. The code is selected carefully so as to reduce intercluster interference.

The completion of the setup phase signals the beginning of the steady-state phase. During this phase, nodes collect information and use their allocated slots to transmit to the cluster head the data collected. This data collection is performed periodically. Simulation results show that LEACH achieves significant energy savings. These savings depend primarily on the data aggregation ratio achieved by the cluster heads. Despite these benefits, however, LEACH suffers several shortcomings. The assumption that all nodes can reach the base station in one hop may not be realistic, as capabilities and energy reserves of the nodes may vary over time from one node to another. Furthermore, the length of the steady-state period is critical to achieving the energy reduction necessary to offset the overhead caused by the cluster selection process. A short steady-state period increases the protocol’s overhead, whereas a long period may lead to cluster head energy depletion. Several algorithms have been proposed to address these shortcomings.

4.4.5 Power-Efficient Gathering in Sensor Information Systems

Power-efficient gathering in sensor information systems (PEGASIS) and its extension, hierarchical PEGASIS, are a family of routing and information-gathering protocols for WSNs. The main objectives of PEGASIS are twofold. First, the protocol aims at extending the lifetime of a network by achieving a high level of energy efficiency and uniform energy consumption across all network nodes. Second, the protocol strives to reduce the delay that data incur on their way to the sink. The network model considered by PEGASIS assumes a homogeneous set of nodes deployed across a geographical area. Nodes are assumed to have global knowledge about other sensors’ positions. Furthermore, they have the ability to control their power to cover arbitrary ranges. The nodes may also be equipped with CDMA-capable radio transceivers. The nodes’ responsibility is to gather and deliver data to a sink, typically a wireless base station. The goal is to develop a routing structure and an aggregation scheme to reduce energy consumption and deliver the aggregated data to the base station with minimal delay while balancing energy consumption among the sensor nodes.

Contrary to other protocols, which rely on a tree structure or a cluster-based hierarchical organization of the network for data gathering and dissemination, PEGASIS uses a chain structure. Based on this structure, nodes communicate with their closest neighbors. The construction of the chain starts with the farthest node from the sink. Network nodes are added to the chain progressively, starting from the closest neighbor to the end node. Nodes that are currently outside the chain are added to the chain in a greedy fashion, the closest neighbor to the top node in the current chain first, until all nodes are included. To determine the closest neighbor, a node uses the signal strength to measure the distance to all its neighboring nodes. Using this information, the node adjusts the signal strength so that only the closest node can be heard. A node within the chain is selected to be the chain leader. Its responsibility is to transmit the aggregated data to the base station.

The chain leader role shifts in positioning the chain after each round. Rounds can be managed by the data sink, and the transition from one round to the next can be tripped by a high-powered beacon issued by the data sink. Rotation of the leadership role among nodes of the chain ensures on average a balanced consumption of energy among all the network nodes. It is worth noting, however, that nodes assuming the role of chain leadership may be arbitrarily far away from the data sink. Such a node may be required to transmit with high power in order to reach the base station Data aggregation in PEGASIS is achieved along the chain. In its simplest form, the aggregation process can be performed sequentially as follows. First, the chain leader issues a token to the last node in the right end of the chain. Upon receiving the token, the end node transmits its data to its downstream neighbor in the chain toward the leader.

The neighboring node aggregates the data and transmits them to its downstream neighbor. This process continues until the aggregated data reach the leader. Upon receiving the data from the right side of the chain, the leader issues a token to the left end of the chain, and the same aggregation process is carried out until the data reach the leader. Upon receiving the data from both sides of the chain, the leader aggregates the data and transmits them to the data sink. Although simple, the sequential aggregation scheme may result in long delays before the aggregated data are delivered to the base station. Such a sequential scheme, however, may be necessary if arbitrarily close simultaneous transmission cannot be carried out without signal interference.

A potential approach to reduce the delay required to deliver aggregated data to the sink is to use parallel data aggregation along the chain. A high degree of parallelism can be achieved if the sensor nodes are equipped with CDMA-capable transceivers. The added ability to carry out arbitrarily close transmissions without interference can be used to ‘‘overlay’’ a hierarchical structure onto the chain and use the embedded structure to perform data aggregation. At each round, nodes at a given level of the hierarchy transmit to a close neighbor in the upper level of the hierarchy.

This process continues until the aggregated data reach the leader at the top level of the hierarchy. The latter transmits the final data aggregate to the base station. To illustrate the chain-based approach, consider the example depicted in Figure 4.10. In this example it is assumed that all nodes have global knowledge of the network and employ a greedy algorithm to construct the chain.

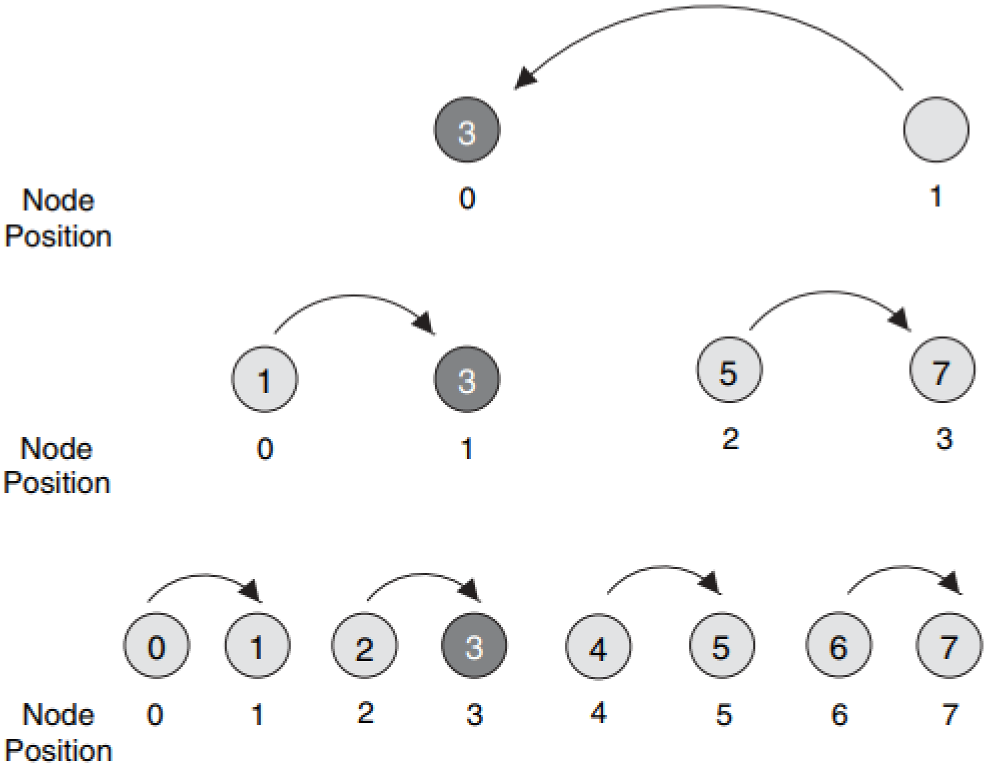
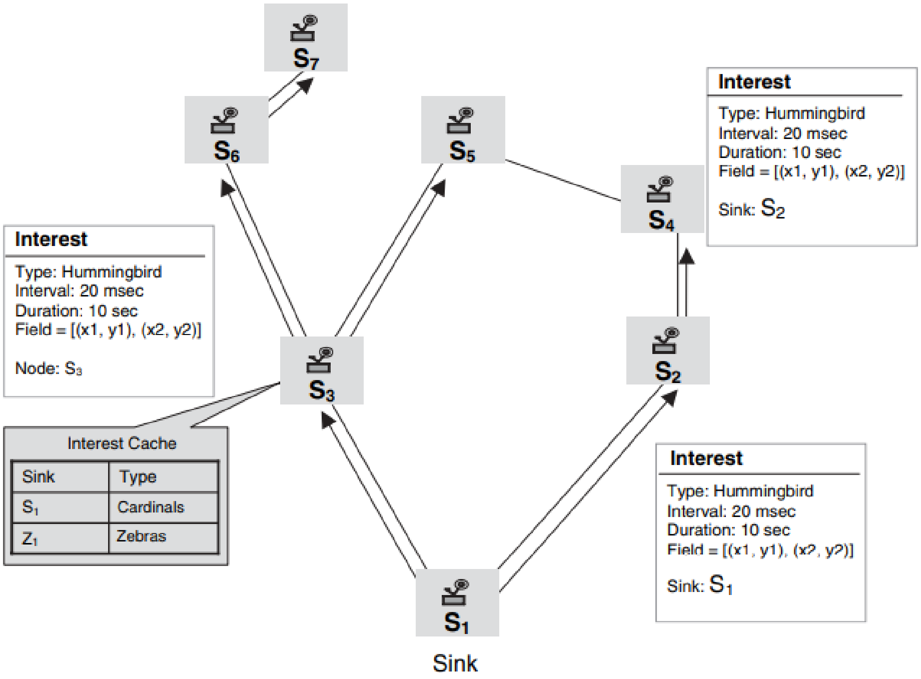


Figure 4.10 Chain-based data gathering and aggregation scheme

Furthermore, it is assumed that nodes take turns in transmitting to the base station such that node i mod N, where N represents the total number of nodes, is responsible for transmitting the aggregate data to the base station in round i. Based on this assignment, node 3, in position 3 in the chain, is the leader in round 3. All nodes in an even position must send their data to their neighbor to the right. At the next level, node 3 remains in an odd position. Consequently, all nodes in an even position aggregate their data and transmit them to their right neighbors. At the third level, node 3 is no longer in an odd position. Node 7, the only node beside node 3 to rise to this level, aggregates its data and sends them to node 3. Node 3, in turn, aggregates the data received with its own data and sends them to the base station. The chain-based binary approach leads to significant energy reduction, as nodes operate in a highly parallel manner. Furthermore, since the hierarchical, treelike structure is balanced, the scheme guarantees that after log2N steps, the aggregated data arrive at the leader. The chain-based binary aggregation scheme has been used in PEGASIS as an alternative to achieving a high degree of parallelism. With CDMA-capable sensor nodes, it has been shown that the scheme performs best with respect to the energy-delay product needed per round of data gathering, a metric that balances the energy and delay cost. The sequential scheme and the CDMA-based fully parallel scheme constitute two endpoints of the design spectrum. A third scheme, which does not require the node transceivers to be equipped with CDMA capabilities, strikes a balance between the two extreme schemes and achieves some level of parallelism. The basic idea of the scheme is to restrict simultaneous transmission to nodes that are spatially separated. Based on this restriction, hierarchical PEGASIS creates a three-level hierarchy in which the total number of network nodes is divided into three groups. Data are aggregated simultaneously within each group and exchanged between groups. The data aggregated eventually reach the leader, which delivers them to the data sink. It is worth noting that simultaneous transmission must be carefully scheduled to avoid interference. Furthermore, the three-level hierarchy must be restructured properly to allow leadership rotation among group nodes. The simulation results of the hierarchical extension of PEGASIS show considerable improvement over schemes such as LEACH. Further, the hierarchical scheme has been shown to outperform the original PEGASIS scheme by a factor of 60.

4.4.6 Directed Diffusion

Directed diffusion is a data-centric routing protocol for information gathering and dissemination in WSNs. The main objective of the protocol is to achieve substantial energy savings in order to extend the lifetime of the network. To achieve this objective, directed diffusion keeps interactions between nodes, in terms of message exchanges, localized within a limited network vicinity. Using localized interaction, direct diffusion can still realize robust multipath delivery and adapt to a minimal subset of network paths. This unique feature of the protocol, combined with the ability of the nodes to aggregate response to queries, results into significant energy savings. The main elements of direct diffusion include interests, data messages, gradients, and reinforcements. Directed diffusion uses a publish-and-subscribe information model in which an inquirer expresses an interest using attribute–value pairs. An interest can be viewed as a query or an interrogation that specifies what the inquirer wants. Table 6.1 shows an example that illustrates how an interest in hummingbirds can be expressed using a set of attribute–value pairs. Sensor nodes, which can service the interest, reply with the corresponding data. For each active sensing task, the data sink periodically broadcasts an interest message to each neighbor. The message propagates throughout the sensor network as an interest for named data. The main purpose of this exploratory interest message is to determine if there exist sensor nodes that can service the sought-after interest. All sensor nodes maintain an interest cache. Each entry of the interest cache corresponds to a different interest. The cache entry contains several fields, including a timestamp field, multiple gradient fields for each neighbor, and a duration field. The timestamp field contains the timestamp of the last matching interest received. Each gradient field specifies both the data rate and the direction in which data are to be sent. The value of the data rate is derived from the interval attribute of the interest. The duration field indicates the approximate lifetime of the interest. The value of the duration is derived from the timestamp of the attribute. Figure 4.11 illustrates interest propagation in a WSN. A gradient can be thought of as a reply link pointing toward the neighboring node from which the interest is received. The diffusion of interests across the entire network, coupled with the establishment of gradients at the network nodes, allows the discovery and establishment of paths between the data sinks that are interested in the named data and the nodes that can serve the data. A sensor node that detects an event searches its interest cache for an entry matching the interest. If a match is identified, the node first computes the highest event rate requested among all its outgoing gradients. It then sets its sensing subsystem to sample the events at this highest rate. The node then sends out an event description to each neighbor for which it has a gradient. A neighboring node that receives a data searches for a matching interest entry in its cache. If no match is found, the node drops the data message with no further action. If such a match exists, and the data message received does not have a matching data cache entry, the node adds the message to the data cache and sends the data message to the neighboring nodes.



Interest

Type: Hummingbird
Interval: 20 msec
Duration: 10 sec
Field = [(x1, y1), (x2, y2)]

Node: S_3

Interest Cache

Sink	Type
S_1	Cardinals
Z_1	Zebras

Interest

Type: Hummingbird
Interval: 20 msec
Duration: 10 sec
Field = [(x1, y1), (x2, y2)]

Sink: S_2

Interest

Type: Hummingbird
Interval: 20 msec
Duration: 10 sec
Field = [(x1, y1), (x2, y2)]

Sink: S_1

Figure 4.11 Interest propagation

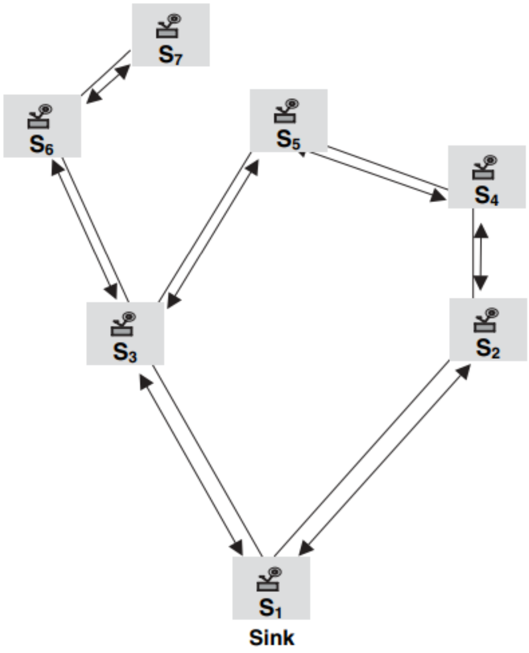


Figure 4.12 Initial gradient setup

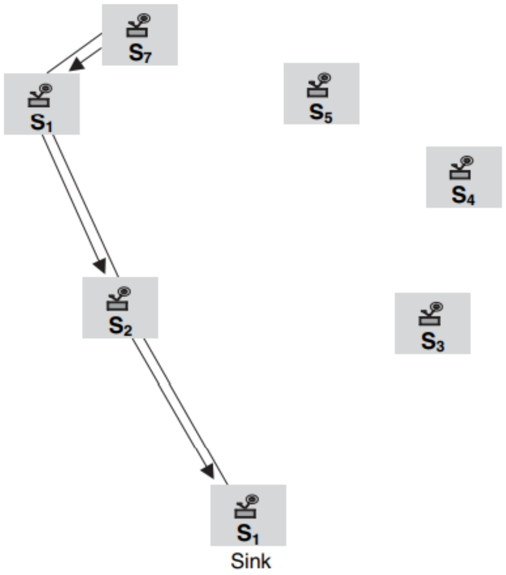


Figure 4.13 Data delivery along a reinforced path.

Upon receiving an interest, a node checks its interest cache to determine if an entry exists in its cache for this interest. If such an entry does not exist, the receiving node creates a new cache entry. The node then uses the information contained in the interest to instantiate the parameters of the newly created interest field. Furthermore, the entry is set to contain a single gradient field, with the event rate specified, pointing toward the neighboring node from which the interest is received. If a match exists between the interest received and a cache entry, the node updates the timestamp and duration fields of the matching entry. If the entry contains no gradient for the sender of the interest, the node adds a gradient with the value specified in the interest message. If the matching interest entry contains a gradient for the interest sender, the node simply updates the timestamp and duration fields. A gradient is removed from its interest entry when it expires. Figure 4.12 shows the initial gradient setup. During the gradient setup phase, a sink establishes multiple paths. The sink can use these paths to higher-quality events by increasing its data rate. This is achieved through a path reinforcement process. The sink may choose to reinforce one or several particular neighbors. To achieve this, the sink resends the original interest message, at a higher data rate, across the paths selected, thereby reinforcing the source nodes on the paths to send data more frequently. The path performing most often can then be retained while negatively reinforcing the remaining paths. Negative reinforcement can be achieved by timing out all high-data-rate gradients in the network, except for those that are explicitly reinforced. Figure 4.13 shows data delivery along a reinforced path.

Link failures caused by environmental factors affecting the communications channel, as well as node failures or performance degradation caused by node energy dissipation or complete depletion, can be repaired in directed diffusion. These failures are typically detected by reduced rate or data loss. When a path between a sensing node and the data sink fails, an alternative path, which is sending at lower rates, can be identified and reinforced. Lossy links can also be negatively reinforced by either sending interests with the exploratory data rate or simply by letting the neighbor's cache expire over time. Directed diffusion has the potential for significant energy savings. Its localized interactions allow it to achieve relatively high performance over unoptimized paths. Furthermore, the resulting diffusion mechanisms are stable under a range of network dynamics. Its data-centric approach obliterates the need for node addressing. The directed diffusion paradigm, however, is tightly coupled into a semantically driven query-on-demand data model. This may limit its use to applications that fit such a data model, where the interest-matching process can be achieved efficiently and unambiguously.

4.4.7 Geographical Routing

The main objective of geographical routing is to use location information to formulate an efficient route search toward the destination. Geographical routing is very suitable to sensor networks, where data aggregation is a useful technique to minimize the number of transmissions toward the base station by eliminating redundancy among packets from different sources. The need for data aggregation to reduce energy consumption shifts the computation and communications model in sensor networks from a traditional address-centric paradigm, where the interaction is between two addressable endpoints of communications, to a data-centric paradigm, where the content of the data is more important than the identity of the node that gathers the data. In this new paradigm, an application may issue a query to inquire about a phenomenon within a specific physical area or near the vicinity of a landmark. For example, scientists analyzing traffic flow patterns may be interested in determining the average number, size, and speed of vehicles that travel on a specific section of a highway. The identity of the sensors that collect and disseminate information about traffic flow on a specific section of the highway is not as important as the data content. Furthermore, multiple nodes that happen to be located in the targeted section of the highway may participate in collecting and aggregating the data in order to answer the query. Traditional routing approaches, which are typically designed to discover a path between two addressable endpoints, are not well suited to handling geographically specific multidimensional queries. Geographical routing, on the other hand, leverages location information to reach a destination, with each node’s location used as its address. In addition to its compatibility with data-centric applications, geographical routing requires low computation and communication overhead. In traditional routing approaches such as the one used in distributed shortest-path routing protocols for wired networks, knowledge of the entire network topology, or a summary thereof, may be required for a router to compute the shortest path to each destination. Furthermore, to maintain correct paths to all destinations, routers are called upon to update the state describing the current topology in a periodic fashion and when link failure occurs. The need to update the topology state constantly may lead to substantial overhead, proportional to the product of the number of routers and the rate of topological changes in the network. Geographical routing, on the other hand, does not require maintaining a ‘‘heavy’’ state at the routers to keep track of the current state of the topology. It requires only the propagation of single-hop topology information, such as the position of the ‘‘best’’ neighbor to make correct forwarding decisions. The self-describing nature of geographical routing, combined with its localized approach to decision, obliterates the need for maintaining internal data structures such as routing tables. Consequently, the control overhead is reduced substantially, thereby enhancing its scalability in large networks. These attributes make geographical routing a feasible solution for routing in resource-constrained sensor networks.

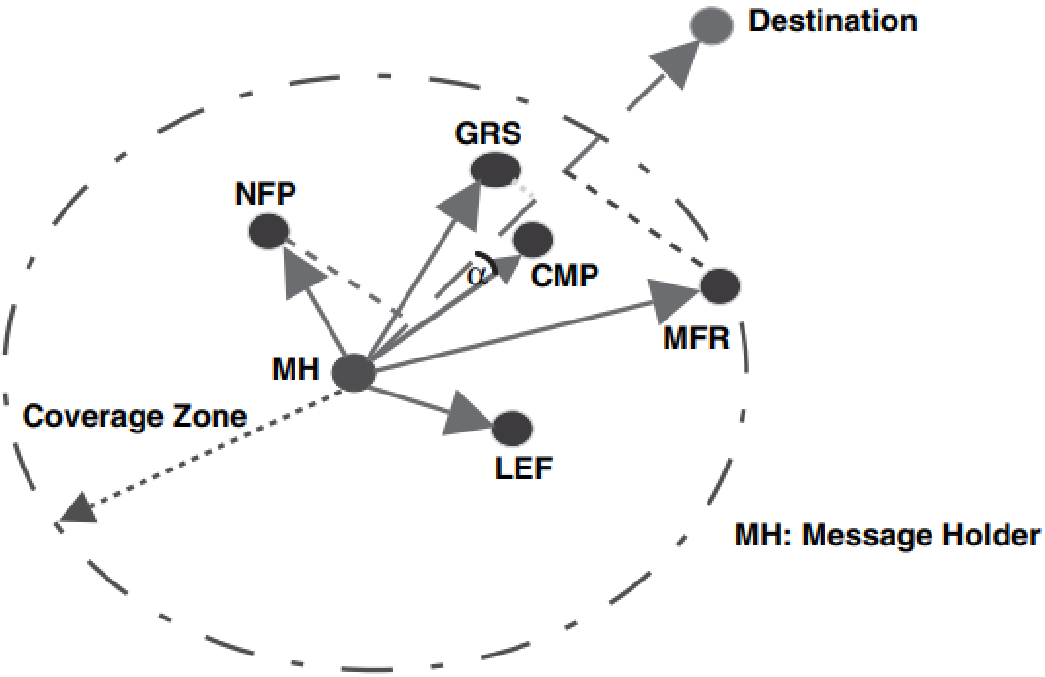


Figure 4.14 Geographical routing forwarding strategies

Part-A Questions

1. Identify the resource constraints for routing.
2. Find the drawbacks of flooding.
3. List the objectives of LEACH protocol.
4. Distinguish data dissemination and data aggregation.
5. Compare flooding and gossiping.
6. Determine the routing protocols which uses location to address a node.
7. Identify the routing protocols which uses attribute -based naming.
8. Find the fields of interest cache.
9. Compare geocasting and multicasting.
10. Differentiate timestamp and duration.

Part-B Questions

1. Explain the routing challenges and design issues in Wireless Sensor Network.
2. Directed diffusion keeps interactions between nodes, in terms of message exchanges, localized within a limited network vicinity. Explain.
3. SPIN is to efficiently disseminate observations gathered by individual sensor nodes to all the sensor nodes in the network. Explain
4. Establish the different phases of LEACH with a neat sketch and explain.
5. Describe about Power Efficient Gathering in Sensor Information Systems.

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited “A” Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

SCHOOL OF COMPUTING

DEPARTMENT OF INFORMATION TECHNOLOGY

UNIT -5

SITA1501 – WIRELESS SENSOR NETWORKS AND ARCHITECTURE

UNIT 5 TRANSPORT LAYER SECURITY PROTOCOLS FOR AD HOC WIRELESS NETWORKS

Syllabus:

Designing issues, classification of transport layer solutions, feedback-based TCP, TCP bus, Ad Hoc TCP, Security in Ad hoc wireless networks, Issues and challenges in security provisioning, Key management, Secure routing in Ad hoc wireless networks. Quality of Service: Issues and challenges in providing QoS in Ad Hoc wireless networks, classification of QoS solutions.

5.1 Designing issues:

- *Induced traffic:*

  Unlike wired networks, ad hoc wireless networks utilize multi-hop radio relaying. A link-level transmission affects the neighbor nodes of both the sender and receiver of the link. In a path having multiple links, transmission at a particular link affects one upstream link and one downstream link. This traffic at any given link (or path) due to the traffic through neighboring links (or paths) is referred to as induced traffic. This is due to the broadcast nature of the channel and the location-dependent contention on the channel. This induced traffic affects the throughput achieved by the transport layer protocol

- *Induced throughput unfairness:*

  This refers to the throughput unfairness at the transport layer due to the throughput/delay unfairness existing at the lower layers such as the network and MAC layers. For example, an ad hoc wireless network that uses IEEE 802.11 DCF as the MAC protocol may experience throughput unfairness at the transport layer as well. A transport layer protocol should consider these in order to provide a fair share of throughput across contending flows.

- *Separation of congestion control, reliability, and flow control:*

  A transport layer protocol can provide better performance if end-to-end reliability, flow control, and congestion control are handled separately. Reliability and flow control are end-to-end activities, whereas congestion can at times be a local activity. The transport layer flow can experience congestion with just one intermediate link under congestion. Hence, in networks such as ad hoc wireless networks, the performance of the transport layer may be improved if these are separately handled. While separating these, the most important objective to be considered is the minimization of the additional control overhead generated by them.

- *Power and bandwidth constraints:*

Nodes in ad hoc wireless networks face resource constraints including the two most important resources: (i) power source and (ii) bandwidth. The performance of a transport layer protocol is significantly affected by these constraints.

- *Misinterpretation of congestion:*

  Traditional mechanisms of detecting congestion in networks, such as packet loss and retransmission timeout, are not suitable for detecting the network congestion in ad hoc wireless networks. This is because the high error rates of wireless channel, location-dependent contention, hidden terminal problem, packet collisions in the network, path breaks due to the mobility of nodes, and node failure due to a drained battery can also lead to packet loss in ad hoc wireless networks. Hence, interpretation of network congestion as used in traditional networks is not appropriate in ad hoc wireless networks.

- *Completely decoupled transport layer:*

  Another challenge faced by a transport layer protocol is the interaction with the lower layers. Wired network transport layer protocols are almost completely decoupled from the lower layers. In ad hoc wireless networks, the cross-layer interaction between the transport layer and lower layers such as the network layer and the MAC layer is important for the transport layer to adapt to the changing network environment.

- *Dynamic topology:*

  Some of the deployment scenarios of ad hoc wireless networks experience rapidly changing network topology due to the mobility of nodes. This can lead to frequent path breaks, partitioning and remerging of networks, and high delay in reestablishment of paths. Hence, the performance of a transport layer protocol is significantly affected by the rapid changes in the network topology.

5.2 Classification of transport layer solutions

Figure 5.1 shows a classification tree for some of the transport layer protocols dis cussed in this chapter. The top-level classification divides the protocols as extensions of TCP for ad hoc wireless networks and other transport layer protocols which are not based on TCP. The solutions for TCP over ad hoc wireless networks can further be classified into split approaches and end-to-end approaches.

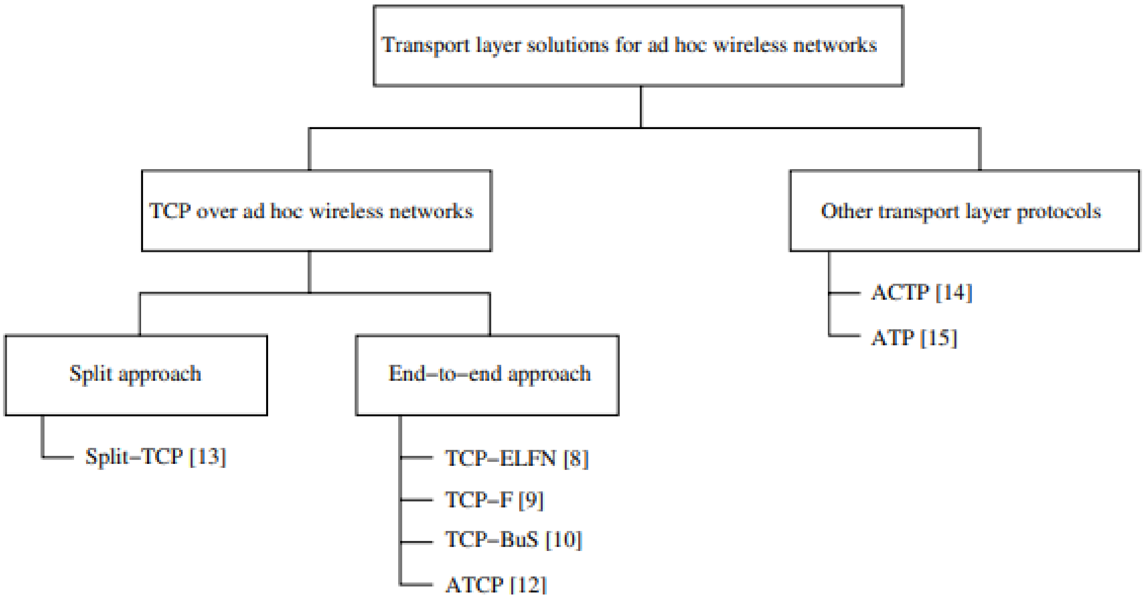
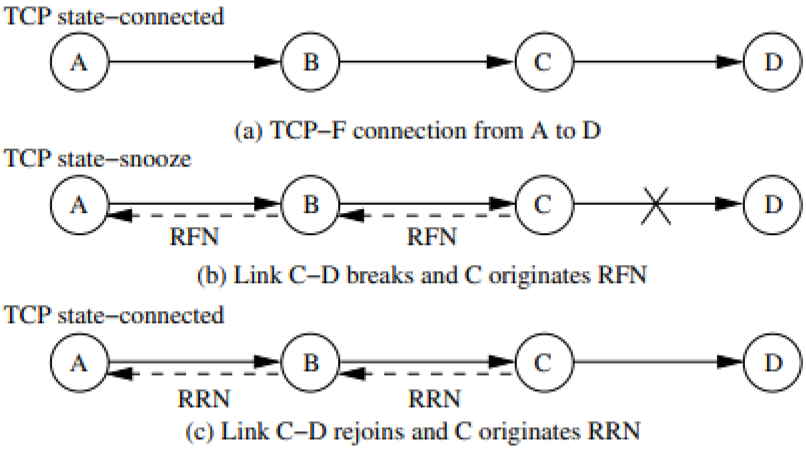

Transport layer solutions for ad hoc wireless networks
TCP over ad hoc wireless networks
Other transport layer protocols
ACTP [14]
ATP [15]
Split approach
End–to–end approach
Split–TCP [13]
TCP–ELFN [8]
TCP–F [9]
TCP–BuS [10]
ATCP [12]

Fig. 5.1 Classification of transport layer solutions

5.3 Feedback-based TCP

Feedback-based TCP [also referred to as TCP feedback (TCP-F) proposes modifications to the traditional TCP for improving performance in ad hoc wireless networks. It uses a feedback-based approach. TCP-F requires the support of a reliable link layer and a routing protocol that can provide feedback to the TCP sender about the path breaks. The routing protocol is expected to repair the broken path within a reasonable time period. TCP-F aims to minimize the throughput degradation resulting from the frequent path breaks that occur in ad hoc wireless networks. During a TCP session, there could be several path breaks resulting in considerable packet loss and path reestablishment delay. Upon detection of packet loss, the sender in a TCP session invokes the congestion control algorithm leading to the exponential back-off of retransmission timers and a decrease in congestion window size.

In TCP-F, an intermediate node, upon detection of a path break, originates a route failure notification (RFN) packet. This RFN packet is routed toward the sender of the TCP session. The TCP sender's information is expected to be obtained from the TCP packets being forwarded by the node. The intermediate node that originates the RFN packet is called the failure point (FP). The FP maintains information about all the RFNs it has originated so far. Every intermediate node that forwards the RFN packet understands the route failure, updates its routing table accordingly, and avoids forwarding any more packets on that route. If any of the intermediate nodes that receive RFN has an alternate route to the same destination, then it discards the RFN packet and uses the alternate path for forwarding further data packets, thus reducing the control overhead involved in the route reconfiguration process. Otherwise, it forwards the RFN toward the source node. When a TCP sender receives an RFN packet, it goes into a state called snooze. In the snooze state, a sender stops sending any more packets to the destination, cancels all the timers, freezes its congestion window, freezes the retransmission timer, and sets up a route failure timer. This route failure timer is dependent on the routing protocol, network size, and the network dynamics and is to be taken as the worst-case route reconfiguration time. When the route failure timer expires, the TCP sender changes from the snooze state to the connected state. Figure 5.2 shows the operation of the TCP-F protocol. In the figure, a TCP session is set up between node A and node D over the path A-B-C-D [refer to Figure 5.2 (a)]. When the intermediate link between node C and node D fails, node C originates an RFN packet and forwards it on the reverse path to the source node [see Figure 5.2 (b)]. The sender's TCP state is changed to the snooze state upon receipt of an RFN packet. If the link CD rejoins, or if any of the intermediate nodes obtains a path to destination node D, a route reestablishment notification (RRN) packet is sent to node A and the TCP state is updated back to the connected state [Figure 5.2 (c)]. As soon as a node receives an RRN packet, it transmits all the packets in its buffer, assuming that the network is back to its original state. This can also take care of all the packets that were not acknowledged or lost during transit due to the path break. In fact, such a step avoids going through the slow-start process that would otherwise have occurred immediately after a period of congestion. The route failure timer set after receiving the RFN packet ensures that the sender does not remain in the snooze state indefinitely. Once the route failure timer expires, the sender goes back to the connected state in which it reactivates the frozen timers and starts sending the buffered and unacknowledged packets. This can also take care of the loss of the RRN packet due to any possible subsequent congestion. TCP-F permits the TCP congestion control algorithm to be in effect when the sender is not in the snooze state, thus making it sensitive to congestion in the network.



(a) TCP−F connection from A to D

(b) Link C−D breaks and C originates RFN

(c) Link C−D rejoins and C originates RRN

Figure 5.2 Operation of TCP-F

- *Advantages and Disadvantages*

TCP-F provides a simple feedback-based solution to minimize the problems arising out of frequent path breaks in ad hoc wireless networks. At the same time, it also permits the TCP congestion control mechanism to respond to congestion in the network. TCP-F depends on the intermediate nodes' ability to detect route failures and the routing protocols' capability to re-establish a broken path within a reasonably short duration. Also, the FP should be able to obtain the correct path (the path which the packet traversed) to the TCP-F sender for sending the RFN packet. This is simple with a routing protocol that uses source routing [i.e., dynamic source routing (DSR)]. If a route to the sender is not available at the FP, then additional control packets may need to be generated for routing the RFN packet. TCP-F has an additional state compared to the traditional TCP state machine, and hence its implementation requires modifications to the existing TCP libraries. Another disadvantage of TCP-F is that the congestion window used after a new route is obtained may not reflect the achievable transmission rate acceptable to the network and the TCP-F receiver.

5.4 TCP-Bus

TCP with buffering capability and sequence information (TCP-BuS) is similar to the TCP-F and TCP-ELFN in its use of feedback information from an intermediate node on detection of a path break. But TCP-BuS is more dependent on the routing protocol compared to TCP-F and TCP-ELFN. TCP-BuS was proposed, with associativity-based routing (ABR) protocol as the routing scheme. Hence, it makes use of some of the special messages such as localized query (LQ) and REPLY, defined as part of ABR for finding a partial path. These messages are modified to carry TCP connection and segment information. Upon detection of a path break, an upstream intermediate node [called pivot node (PN)] originates an explicit route disconnection notification (ERDN) message. This ERDN packet is propagated to the TCP-BuS sender and, upon reception of it, the TCP-BuS sender stops transmission and freezes all timers and windows as in TCP-F. The packets in transit at the intermediate nodes from the TCP-BuS sender to the PN are buffered until a new partial path from the PN to the TCP-BuS receiver is obtained by the PN. In order to avoid unnecessary retransmissions, the timers for the buffered packets at the TCP-BuS sender and at the intermediate nodes up to PN use timeout values proportional to the round-trip time (RTT). The intermediate nodes between the TCP-BuS sender and the PN can request the TCP-BuS sender to selectively retransmit any of the lost packets. Upon detection of a path break, the downstream node originates a route notification (RN) packet to the TCP-BuS receiver, which is forwarded by all the downstream nodes in the path. An intermediate node that receives an RN packet discards all packets belonging to that flow. The ERDN packet is propagated to the TCP-BuS sender in a reliable way by using an implicit acknowledgment and retransmission mechanism. The PN includes the sequence number of the TCP segment belonging to the flow that is currently at the head of its queue in the ERDN packet. The PN also attempts to find a new partial route to the TCP-BuS receiver, and the availability of such a partial path to destination is intimated to the TCP-BuS sender through an explicit route successful notification (ERSN) packet. TCP-BuS utilizes the route reconfiguration mechanism of ABR to obtain the partial route to the destination. Due to this, other routing protocols may require changes to support TCP-BuS. The LQ and REPLY messages are modified to carry TCP segment information, including the last successfully re ceived segment at the destination. The LQ packet carries the sequence number of the segment at the head of the queue buffered at the PN and the REPLY carries the sequence number of the last successful segment the TCP-BuS receiver received. This enables the TCP-BuS receiver to understand the packets lost in transition and those buffered at the intermediate nodes. This is used to avoid fast retransmission requests usually generated by the TCP-BuS receiver when it notices an out-of-order packet delivery. Upon a successful LQ-REPLY process to obtain a new route to the TCP-BuS receiver, PN informs the TCP-BuS sender of the new partial path using the ERSN packet. When the TCP-BuS sender receives an ERSN packet, it resumes the data transmission. Since there is a chance for ERSN packet loss due to congestion in the network, it needs to be sent reliably. The TCP-BuS sender also periodically originates probe packets to check the availability of a path to the destination. Figure 5.3 shows an illustration of the propagation of ERDN and RN messages when a link between nodes 4 and 12 fails. When a TCP-BuS sender receives the ERSN message, it understands, from the sequence number of the last successfully received packet at the destination and the sequence number of the packet at the head of the queue at PN, the packets lost in transition. The TCP-BuS receiver understands that the lost packets will be delayed further and hence uses a selective acknowledgment strategy instead of fast retransmission. These lost packets are retransmitted by the TCP-BuS sender. During the retransmission of these lost packets, the network congestion between the TCP-BuS sender and PN is handled in a way similar to that in traditional TCP.

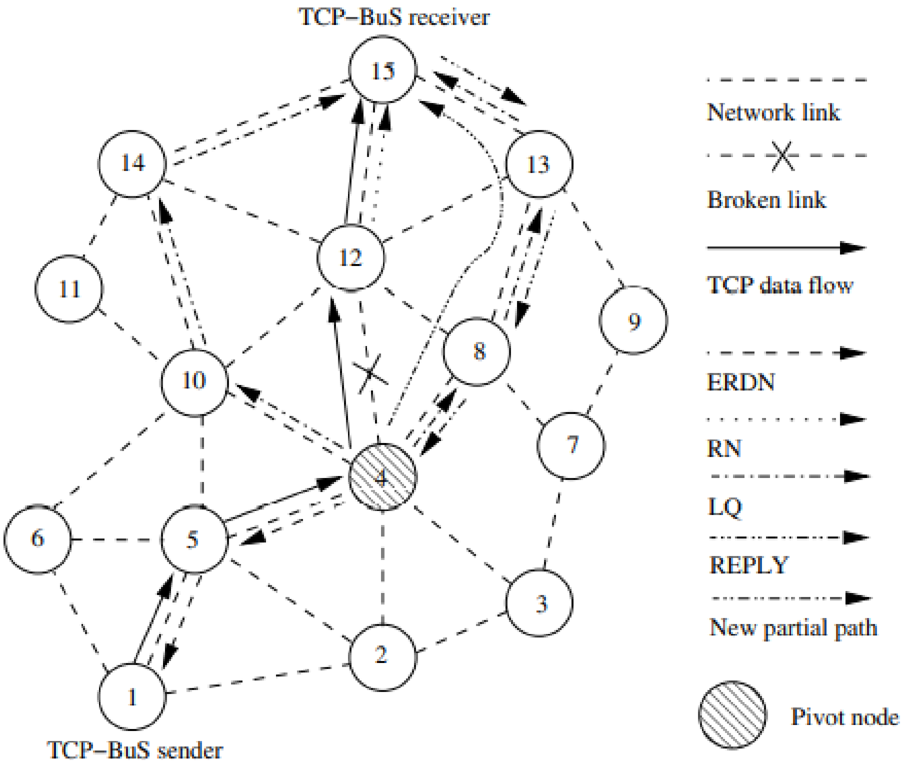


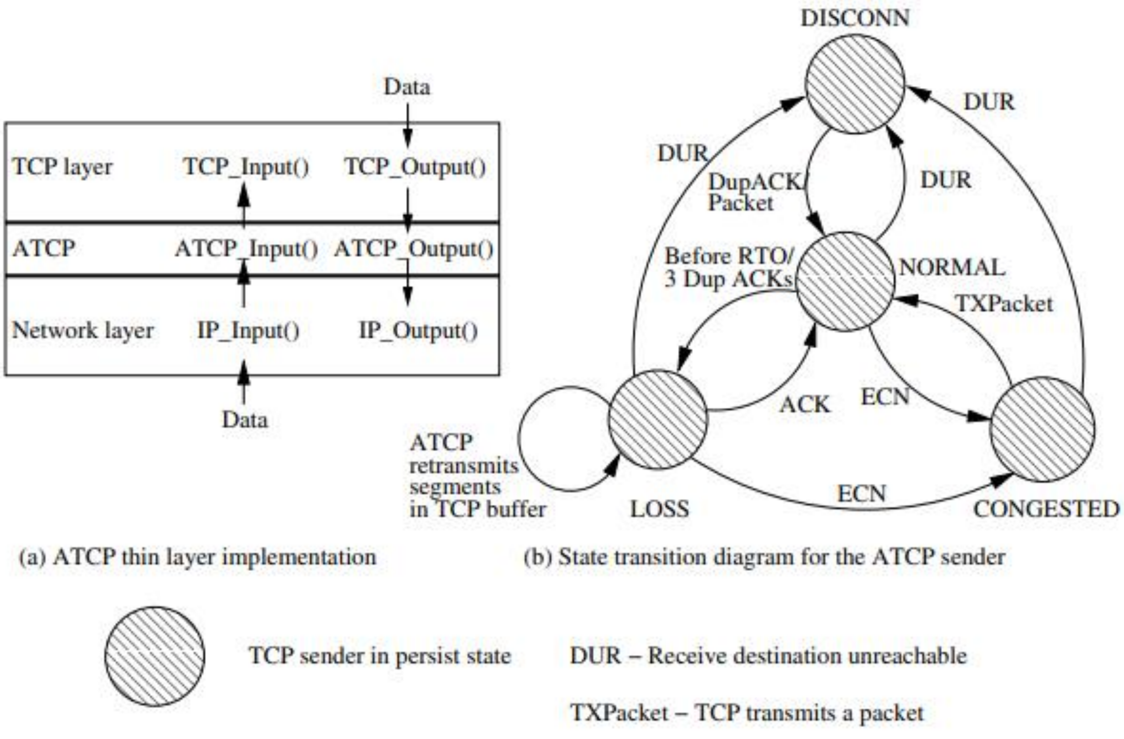
Figure 5.3 Operation of TCP-Bus

- *Advantages and Disadvantages*

The advantages of TCP-BuS include performance improvement and avoidance of fast retransmission due to the use of buffering, sequence numbering, and selective acknowledgment. TCP-BuS also takes advantage of the underlying routing proto cols, especially the on-demand routing protocols such as ABR. The disadvantages of TCP-BuS include the increased dependency on the routing protocol and the buffering at the intermediate nodes. The failure of intermediate nodes that buffer the packets may lead to loss of packets and performance degradation. The dependency of TCP-BuS on the routing protocol may degrade its performance with other routing protocols that do not have similar control messages as in ABR.

5.5 Ad Hoc TCP

Similar to TCP-F and TCP-ELFN, ad hoc TCP (ATCP) also uses a network layer feedback mechanism to make the TCP sender aware of the status of the network path over which the TCP packets are propagated. Based on the feedback information received from the intermediate nodes, the TCP sender changes its state to the persist state, congestion control state, or the retransmit state. When an intermediate node finds that the network is partitioned, then the TCP sender state is changed to the persist state where it avoids unnecessary retransmissions. When ATCP puts TCP in the persist state, it sets TCP’s congestion window size to one in order to ensure that TCP does not continue using the old congestion window value. This forces TCP to probe the correct value of the congestion window to be used for the new route. If an intermediate node loses a packet due to error, then the ATCP at the TCP sender immediately retransmits it without invoking the congestion control algorithm. In order to be compatible with widely deployed TCP based networks, ATCP provides this feature without modifying the traditional TCP. ATCP is implemented as a thin layer residing between the IP and TCP protocols. The ATCP layer essentially makes use of the explicit congestion notification (ECN) for maintenance of the states. Figure 5.4 (a) shows the thin layer implementation of ATCP between the traditional TCP layer and the IP layer. This does not require changes in the existing TCP protocol. This layer is active only at the TCP sender. The major function of the ATCP layer is to monitor the packets sent and received by the TCP sender, the state of the TCP sender, and the state of the network. Figure 5.4 (b) shows the state transition diagram for the ATCP at the TCP sender. The four states in the ATCP are (i) NORMAL, (ii) CONGESTED, (iii) LOSS, and (iv) DISCONN. When a TCP connection is established, the ATCP sender state is in NORMAL. In this state, ATCP does not interfere with the operation of TCP and it remains invisible. When packets are lost or arrive out-of-order at the destination, it generates duplicate ACKs. In traditional TCP, upon reception of duplicate ACKs, the TCP sender retransmits the segment under consideration and shrinks the contention window.



(a) ATCP thin layer implementation

(b) State transition diagram for the ATCP sender

Figure 5.4. An illustration of ATCP thin layer and ATCP state diagram

But the ATCP sender counts the number of duplicate ACKs received and if it reaches three, instead of forwarding the duplicate ACKs to TCP, it puts TCP in the persist state and ATCP in the LOSS state. Hence, the TCP sender avoids invoking congestion control. In the LOSS state, ATCP retransmits the unacknowledged segments from the TCP buffer. When a new ACK comes from the TCP receiver, it is forwarded to TCP and the TCP sender is removed from the persist state and then the ATCP sender changes to the NORMAL state. When the ATCP sender is in the LOSS state, the receipt of an ECN message or an ICMP source quench message changes it to the CONGESTED state. Along with this state transition, the ATCP sender removes the TCP sender from the persist state. When the network gets congested, the ECN4 flag is set in the data and the ACK packets. When the ATCP sender receives this ECN message in the normal state, it changes to the CONGESTED state and just remains invisible, permitting TCP to invoke normal congestion control mechanisms. When a route failure or a transient network partition occurs in the network, ATCP expects the network layer to detect these and inform the ATCP sender through an ICMP destination unreachable (DUR) message. Upon reception of the DUR message, ATCP puts the TCP sender into the persist state and enters into the DISCONN state. It remains in the DISCONN state until it is connected and receives any data or duplicate ACKs. On the occurrence of any of these events, ATCP changes to the NORMAL state. The connected status of the path can be detected by the acknowledgments for the periodic probe packets generated by the TCP sender. The receipt of an ICMP DUR message in the LOSS state or the CONGESTED state causes a transition to the DISCONN state. When ATCP puts TCP into the persist state, it sets the congestion window to one segment in order to make TCP probe for the new congestion window when the new route is available. In summary, ATCP tries to perform the activities listed in Table 5.1.

Table 5.1. The actions taken by ATCP

Event	Action
Packet loss due to high BER	Retransmits the lost packets without reducing congestion window
Route recomputation de lay	Makes the TCP sender go to persist state and stop transmission until new route has been found
Transient partitions	Makes the TCP sender go to persist state and stop transmission until new route has been found
Out-of-order packet de livery due to multipath routing	Maintains TCP sender unaware of this and retransmits the packets from TCP buffer
Change in route	Recomputes the congestion window

- *Advantages and Disadvantages*

  Two major advantages of ATCP are (i) it maintains the end-to-end semantics of TCP and (ii) it is compatible with traditional TCP. These advantages permit ATCP to work seamlessly with the Internet. In addition, ATCP provides a feasible and efficient solution to improve throughput of TCP in ad hoc wireless networks. The disadvantages of ATCP include (i) the dependency on the network layer protocol to detect the route changes and partitions, which not all routing protocols may implement and (ii) the addition of a thin ATCP layer to the TCP/IP protocol stack that requires changes in the interface functions currently being used.

5.6 Security in Ad hoc wireless networks

Due to the unique characteristics of ad hoc wireless networks, such networks are highly vulnerable to security attacks compared to wired networks or infrastructure-based wireless networks.

5.6.1 Network Security Requirements

A security protocol for ad hoc wireless networks should satisfy the following requirements. The requirements listed below should in fact be met by security protocols for other types of networks also.

• *Confidentiality:* The data sent by the sender (source node) must be comprehensible only to the intended receiver (destination node). Though an intruder might get hold of the data being sent, he/she must not be able to derive any useful information out of the data. One of the popular techniques used for ensuring confidentiality is data encryption.

• *Integrity:* The data sent by the source node should reach the destination node as it was sent: unaltered. In other words, it should not be possible for any malicious node in the network to tamper with the data during transmission.

• *Availability:* The network should remain operational all the time. It must be robust enough to tolerate link failures and also be capable of surviving various attacks mounted on it. It should be able to provide the guaranteed services whenever an authorized user requires them.

• *Non-repudiation:* Non-repudiation is a mechanism to guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message. Digital signatures, which function as unique identifiers for each user, much like a written signature, are used commonly for this purpose

5.6.2 Issues and challenges in security provisioning

Designing a fool-proof security protocol for ad hoc wireless is a very challenging task. This is mainly because of certain unique characteristics of ad hoc wireless networks, namely, shared broadcast radio channel, insecure operating environment, lack of central authority, lack of association among nodes, limited availability of resources, and physical vulnerability. A detailed discussion on how each of the mentioned characteristics causes difficulty in providing security in ad hoc wireless networks is given below.

• *Shared broadcast radio channel:* Unlike in wired networks where a separate dedicated transmission line can be provided between a pair of end users, the radio channel used for communication in ad hoc wireless networks is broad cast in nature and is shared by all nodes in the network. Data transmitted by a node is received by all nodes within its direct transmission range. So a malicious node could easily obtain data being transmitted in the network. This problem can be minimized to a certain extent by using directional antennas.

• *Insecure operational environment:* The operating environments where ad hoc wireless networks are used may not always be secure. One important application of such networks is in battlefields. In such applications, nodes may move in and out of hostile and insecure enemy territory, where they would be highly vulnerable to security attacks.

• *Lack of central authority:* In wired networks and infrastructure-based wireless networks, it would be possible to monitor the traffic on the network through certain important central points (such as routers, base stations, and access points) and implement security mechanisms at such points. Since ad hoc wireless networks do not have any such central points, these mechanisms cannot be applied in ad hoc wireless networks.

• *Lack of association:* Since these networks are dynamic in nature, a node can join or leave the network at any point of the time. If no proper authentication mechanism is used for associating nodes with a network, an intruder would be able to join into the network quite easily and carry out his/her attacks.

• *Limited resource availability:* Resources such as bandwidth, battery power, and computational power (to a certain extent) are scarce in ad hoc wireless networks. Hence, it is difficult to implement complex cryptography-based security mechanisms in such networks.

- *Physical vulnerability:* Nodes in these networks are usually compact and hand-held in nature. They could get damaged easily and are also vulnerable to theft.

5.7 Key management

Ad hoc wireless networks pose certain specific challenges in key management due to the lack of infrastructure in such networks. Three types of infrastructure have been identified which are absent in ad hoc wireless networks. The first is the network infrastructure, such as dedicated routers and stable links, which ensure communication with all nodes. The second missing infrastructure is services such as name resolution, directory, and TTPs. The third missing infrastructure in ad hoc wireless networks is the administrative support of certifying authorities.

Password-Based Group Systems Several solutions for group keying in ad hoc wireless networks have been suggested. The example scenario for implementation is a meeting room, where different mobile devices want to start a secure session. Here, the parties involved in the session are to be identified based on their location, that is, all devices in the room can be part of the session. Hence, relative location is used as the criterion for access control. If a TTP which knows the location of the participants exists, then it can implement location-based access control. A prior shared secret can be obtained by a physically more secure medium such as a wired network. This secret can be obtained by plugging onto a wired network first, before switching to the wireless mode. A password-based system has been explored where, in the simplest case, a long string is given as the password for users for one session. However, human beings tend to favor natural language phrases as passwords, over randomly generated strings. Such passwords, if used as keys directly during a session, are very weak and open to attack because of high redundancy, and the possibility of reuse over different sessions. Hence, protocols have been proposed to derive a strong key (not vulnerable to attacks) from the weak passwords given by the participants. This password-based system could be two-party, with a separate exchange between any two participants, or it could be for the whole group, with a leader being elected to preside over the session. Leader election is a special case of establishing an order among all participants. The protocol used is as follows. Each participant generates a random number, and sends it to all others. When every node has received the random number of every other node, a common pre-decided function is applied on all the numbers to calculate a reference value. The nodes are ordered based on the difference between their random number and the reference value. Threshold Cryptography Public key infrastructure (PKI) enables the easy distribution of keys and is a scalable method. Each node has a public/private key pair, and a certifying authority (CA) can bind the keys to the particular node. But the CA has to be present at all times, which may not be feasible in ad hoc wireless networks. It is also not advisable to simply replicate the CA at different nodes. A scheme based on threshold cryptography has been proposed by which n servers exist in the ad hoc wireless network, out of which any $(t+1)$ servers can jointly perform any arbitration or authorization successfully, but t servers cannot perform the same. Hence, up to t compromised servers can be tolerated. This is called an $(n, t + 1)$ configuration, where $n \geq 3t + 1$. To sign a certificate, each server generates a partial signature using its private key and submits it to a combiner. The combiner can be any one of the servers. In order to ensure that the key is combined correctly, $t + 1$ combiners can be used to account for at most t malicious servers. Using $t + 1$ partial signatures (obtained from itself and t other servers), the combiner computes a signature and verifies its validity using a public key. If the verification fails, it means that at least one of the t + 1 keys is not valid, so another subset of t + 1 partial signatures is tried. If the combiner itself is malicious, it cannot get a valid key, because the partial signature of itself is always invalid.

The scheme can be applied to asynchronous networks, with no bound on message delivery or processing times. This is one of the strengths of the scheme, as the requirement of synchronization makes the system vulnerable to DoS attacks. An adversary can delay a node long enough to violate the synchrony assumption, thereby disrupting the system. Sharing a secret in a secure manner alone does not completely fortify a system. Mobile adversaries can move from one server to another, attack them, and get hold of their private keys. Over a period of time, an adversary can have more than t private keys. To counter this, share refreshing has been proposed, by which servers create a new independent set of shares (the partial signatures which are used by the servers) periodically. Hence, to break the system, an adversary has to attack and capture more than t servers within the period between two successive refreshes; otherwise, the earlier share information will no longer be valid. This improves protection against mobile adversaries.

Self-Organized Public Key Management for Mobile Ad Hoc Networks have proposed a completely self-organized public key system for ad hoc wireless networks. This makes use of absolutely no infrastructure – TTP, CA, or server – even during initial configuration. The users in the ad hoc wireless network issue certificates to each other based on personal acquaintance. A certificate is a binding between a node and its public key. These certificates are also stored and distributed by the users themselves. Certificates are issued only for a specified period of time and contain their time of expiry along with them. Before it expires, the certificate is updated by the user who had issued the certificate. Initially, each user has a local repository consisting of the certificates issued by him and the certificates issued by other users to him. Hence, each certificate is initially stored twice, by the issuer and by the person for whom it is issued. Periodically, certificates from neighbors are requested and the repository is updated by adding any new certificates. If any of the certificates are conflicting (e.g., the same public key to different users, or the same user having different public keys), it is possible that a malicious node has issued a false certificate. A node then labels such certificates as conflicting and tries to resolve the conflict. Various methods exist to compare the confidence in one certificate over another. For instance, another set of certificates obtained from another neighbor can be used to take a majority decision. This can be used to evaluate the trust in other users and detect malicious nodes. If the certificates issued by some node are found to be wrong, then that node may be assumed to be malicious. A certificate graph is defined as a graph whose vertices are public keys of some nodes and whose edges are public-key certificates issued by users. When a user X wants to obtain the public key of another user Y, he/she finds a chain of valid public key certificates leading to Y. The chain is such that the first hop uses an edge from X, that is, a certificate issued by X, the last hop leads into Y (this is a certificate issued to Y), and all intermediate nodes are trusted through the previous certificate in the path. The protocol assumes that trust is transitive, which may not always be valid.

5.8 Secure routing in Ad hoc wireless networks

Unlike the traditional wired Internet, where dedicated routers controlled by the Internet service providers (ISPs) exist, in ad hoc wireless networks, nodes act both as regular terminals (source or destination) and also as routers for other nodes. In the absence of dedicated routers, providing security becomes a challenging task in these networks. Various other factors which make the task of ensuring secure communication in ad hoc wireless networks difficult include the mobility of nodes, a promiscuous mode of operation, limited processing power, and limited availability of resources such as battery power, bandwidth, and memory. In this, we show how some of the well-known traditional routing protocols for ad hoc networks fail to provide security.

5.8.1 Requirements of a Secure Routing Protocol for Ad Hoc Wireless Networks:

The fundamental requisites of a secure routing protocol for ad hoc wireless networks are listed as follows:

• *Detection of malicious nodes:* A secure routing protocol should be able to detect the presence of malicious nodes in the network and should avoid the participation of such nodes in the routing process. Even if such malicious nodes participate in the route discovery process, the routing protocol should choose paths that do not include such nodes.

• *Guarantee of correct route discovery:* If a route between the source and the destination nodes exists, the routing protocol should be able to find the route, and should also ensure the correctness of the selected route.

• *Confidentiality of network topology:* An information disclosure attack may lead to the discovery of the network topology by the malicious nodes. Once the network topology is known, the attacker may try to study the traffic pattern in the network. If some of the nodes are found to be more active compared to others, the attacker may try to mount (e.g., DoS) attacks on such bottleneck nodes. This may ultimately affect the on-going routing process. Hence, the confidentiality of the network topology is an important requirement to be met by the secure routing protocols.

• *Stability against attacks:* The routing protocol must be self-stable in the sense that it must be able to revert to its normal operating state within a finite amount of time after a passive or an active attack. The routing protocol should take care that these attacks do not permanently disrupt the routing process. The protocol must also ensure Byzantine robustness, that is, the protocol should work properly even if some of the nodes, which were earlier participating in the routing process, turn out to become malicious at a later point of time or are intentionally damaged.

5.8.2 Security-Aware Ad Hoc Routing Protocol:

The security-aware ad hoc routing (SAR) protocol uses security as one of the key metrics in path finding. A framework for enforcing and measuring the attributes of the security metric has been provided. This framework also enables the use of different levels of security for different applications that use SAR for routing. In ad hoc wireless networks, communication between end nodes through possibly multiple intermediate nodes is based on the fact that the two end nodes trust the intermediate nodes. SAR defines level of trust as a metric for routing and as one of the attributes for security to be taken into consideration while routing. The routing protocol based on the level of trust is explained using Figure 5.5. As shown in Figure 5.5, two paths exist between the two officers O1 and O2 who want to communicate with each other. One of these paths is a shorter path which runs through private nodes whose trust levels are very low. Hence, the protocol chooses a longer but secure path which passes through other secure (officer) nodes.

In the AODV protocol, the source node broadcasts a Route Request packet to its neighbors. An intermediate node, on receiving a Route Request packet, forwards it further if it does not have a route to the destination. Otherwise, it initiates a Route Reply packet back to the source node using the reverse path traversed by the Route Request packet. In SAR, a certain level of security is incorporated into the packet-forwarding mechanism. Here, each packet is associated with a security level which is determined by a number calculation method. Each intermediate node is also associated with a certain level of security. On receiving a packet, the intermediate node compares its level of security with that defined for the packet. If the node's security level is less than that of the packet, the Route Request is simply discarded. If it is greater, the node is considered to be a secure node and is permitted to forward the packet in addition to being able to view the packet. If the security levels of the intermediate node and the received packet are found to be equal, then the intermediate node will not be able to view the packet (which can be ensured using a proper authentication mechanism); it just forwards the packet further. Nodes of equal levels of trust distribute a common key among themselves and with those nodes having higher levels of trust. Hence, a hierarchical level of security could be maintained. This ensures that an encrypted packet can be decrypted (using the common key) only by nodes of the same or higher levels of security compared to the level of security of the packet. Different levels of trust can be defined using a number calculated based on the level of security required. It can be calculated using many methods. Since timeliness, in-order delivery of packets, authenticity, authorization, integrity, confidentiality, and non-repudiation are some of the desired characteristics of a routing protocol, a suitable number can be defined for the trust level for nodes and packets based on the number of such characteristics taken into account. The SAR mechanism can be easily incorporated into the traditional routing protocols for ad hoc wireless networks. It could be incorporated into both on-demand and table-driven routing protocols. The SAR protocol allows the application to choose the level of security it requires. But the protocol requires different keys for different levels of security. This tends to increase the number of keys required when the number of security levels used increases.

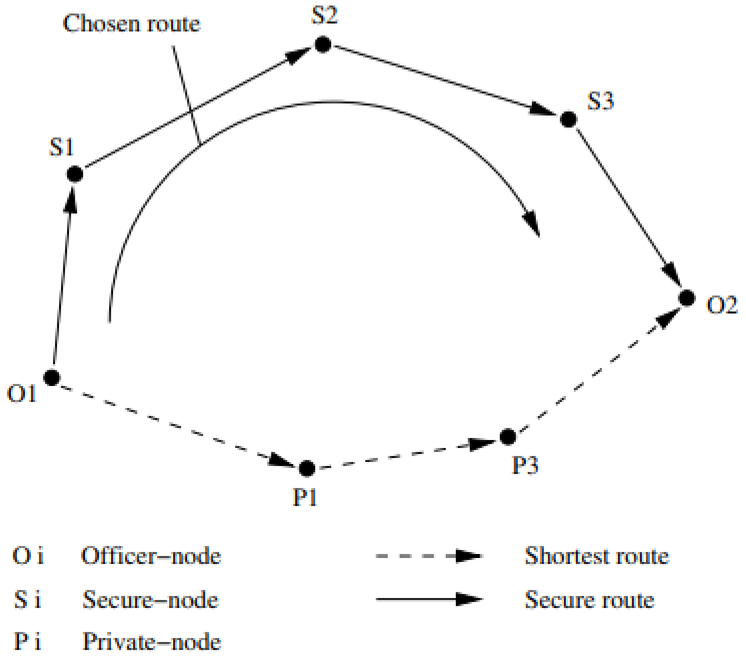


Figure 5.5 Illustration of the level of trust metric

5.9 Quality of Service: Issues and challenges in providing QoS in Ad Hoc wireless networks

Providing QoS support in ad hoc wireless networks is an active research area. Ad hoc wireless networks have certain unique characteristics that pose several difficulties in provisioning QoS. Some of the characteristics are dynamically varying network topology, lack of precise state information, lack of a central controller, error-prone shared radio channel, limited resource availability, hidden terminal problem, and insecure medium. A detailed discussion on how each of the above-mentioned characteristics affects QoS provisioning in ad hoc wireless networks is given below. Dynamically varying network topology: Since the nodes in an ad hoc wireless network do not have any restriction on mobility, the network topology changes dynamically. Hence, the admitted QoS sessions may suffer due to frequent path breaks, thereby requiring such sessions to be re-established over new paths. The delay incurred in re-establishing a QoS session may cause some of the packets belonging to that session to miss their delay targets/deadlines, which is not acceptable for applications that have stringent QoS requirements.

- ***Imprecise state information:*** In most cases, the nodes in an ad hoc wireless network maintain both the link-specific state information and flow-specific state information. The link specific state information includes bandwidth, delay, delay jitter, loss rate, error rate, stability, cost, and distance values for each link. The flow-specific information includes session ID, source address, destination address, and QoS requirements of the flow (such as maximum bandwidth requirement, minimum bandwidth requirement, maximum delay, and maximum delay jitter). The state information is inherently imprecise due to dynamic changes in network topology and channel characteristics. Hence, routing decisions may not be accurate, resulting in some of the real-time packets missing their deadlines.

- ***Lack of central coordination:*** Unlike wireless LANs and cellular networks, ad hoc wireless networks do not have central controllers to coordinate the activity of nodes. This further complicates QoS provisioning in ad hoc wireless networks.
- ***Error-prone shared radio channel:*** The radio channel is a broadcast medium by nature. During propagation through the wireless medium, the radio waves suffer from several impairments such as attenuation, multipath propagation, and interference (from other wireless devices operating in the vicinity).
- ***Hidden terminal problem:*** The hidden terminal problem is inherent in ad hoc wireless networks. This problem occurs when packets originating from two or more sender nodes, which are not within the direct transmission range of each other, collide at a common receiver node. It necessitates the retransmission of the packets, which may not be acceptable for flows that have stringent QoS requirements. The RTS/CTS control packet exchange mechanism, proposed in and adopted later in the IEEE 802.11 standard, reduces the hidden terminal problem only to a certain extent.
- ***Limited resource availability:*** Resources such as bandwidth, battery life, storage space, and processing capability are limited in ad hoc wireless networks. Out of these, bandwidth and battery life are critical resources, the availability of which significantly affects the performance of the QoS provisioning mechanism. Hence, efficient resource management mechanisms are required for optimal utilization of these scarce resources.
- ***Insecure medium:*** Due to the broadcast nature of the wireless medium, communication through a wireless channel is highly insecure. Therefore, security is an important issue in ad hoc wireless networks, especially for military and tactical applications. Ad hoc wireless networks are susceptible to attacks such as eavesdropping, spoofing, denial of service, message distortion, and impersonation. Without sophisticated security mechanisms, it is very difficult to provide secure communication guarantees. Some of the design choices for providing QoS support are described below.
- ***Hard state versus soft state resource reservation:*** QoS resource reservation is one of the very important components of any QoS framework (a QoS framework is a complete system that provides required/promised services to each user or application). It is responsible for reserving resources at all intermediate nodes along the path from the source to the destination, as requested by the QoS session. QoS resource reservation mechanisms can be broadly classified into two categories: hard state and soft state reservation mechanisms. In hard state resource reservation schemes, resources are reserved at all intermediate nodes along the path from the source to the destination throughout the duration of the QoS session. If such a path is broken due to network dynamics, these reserved resources have to be explicitly released by a de-allocation mechanism. Such a mechanism not only introduces additional control overhead, but may also fail to release resources completely in case a node previously belonging to the session becomes unreachable. Due to these problems, soft state resource reservation mechanisms, which maintain reservations only for small time intervals, are used. These reservations get refreshed if packets belonging to the same flow are received before the

timeout period. The soft state reservation timeout period can be equal to packet inter-arrival time or a multiple of the packet inter-arrival time. If no data packets are received for the specified time interval, the resources are deallocated in a decentralized manner without incurring any additional control overhead. Thus no explicit teardown is required for a flow. The hard state schemes reserve resources explicitly and hence, at high network loads, the call blocking ratio will be high, whereas soft state schemes provide high call acceptance at a gracefully degraded fashion.

- ***Stateful versus stateless approach:*** In the stateful approach, each node maintains either global state information or only local state information, while in the case of a stateless approach, no such information is maintained at the nodes. State information includes both the topology information and the flow specific information. If global state information is available, the source node can use a centralized routing algorithm to route packets to the destination. The performance of the routing protocol depends on the accuracy of the global state information maintained at the nodes. Significant control overhead is incurred in gathering and maintaining global state information. On the other hand, if mobile nodes maintain only local state information (which is more accurate), distributed routing algorithms can be used. Even though control overhead incurred in maintaining local state information is low, care must be taken to obtain loop-free routes. In the case of the neither stateless approach, neither flow specific nor link specific state information is maintained at the nodes. Though the stateless approach solves the scalability problem permanently and reduces the burden (storage and computation) on nodes, providing QoS guarantees becomes extremely difficult.
- ***Hard QoS versus soft QoS approach:***

  The QoS provisioning approaches can be broadly classified into two categories: hard QoS and soft QoS approaches. If QoS requirements of a connection are guaranteed to be met for the whole duration of the session, the QoS approach is termed a hard QoS approach. If the QoS requirements are not guaranteed for the entire session, the QoS approach is termed a soft QoS approach. Keeping network dynamics of ad hoc wireless networks in mind, it is very difficult to provide hard QoS guarantees to user applications. Thus, QoS guarantees can be given only within certain statistical bounds. Almost all QoS approaches available in the literature provide only soft QoS guarantees.

5.10 Classification of QoS solutions.

The QoS solutions can be classified in two ways. One classification is based on the QoS approach employed, while the other one classifies QoS solutions based on the layer at which they operate in the network protocol stack.

Classifications of QoS Approaches As shown in Figure 5.6, several criteria are used for classifying QoS approaches. The QoS approaches can be classified based on the interaction between the routing protocol and the QoS provisioning mechanism, based on the interaction between the network and the MAC layers, or based on the routing information update mechanism. Based on the interaction between the routing protocol and the QoS provisioning mechanism, QoS approaches can be classified into two categories: coupled and decoupled QoS approaches. In the case of the coupled QoS approach, the routing protocol and the QoS provisioning mechanism closely interact with each other for delivering QoS guarantees. If the routing protocol changes, it may fail to ensure QoS guarantees. But in the case of the decoupled approach, the QoS provisioning mechanism does not depend on any specific routing protocol to ensure QoS guarantees.

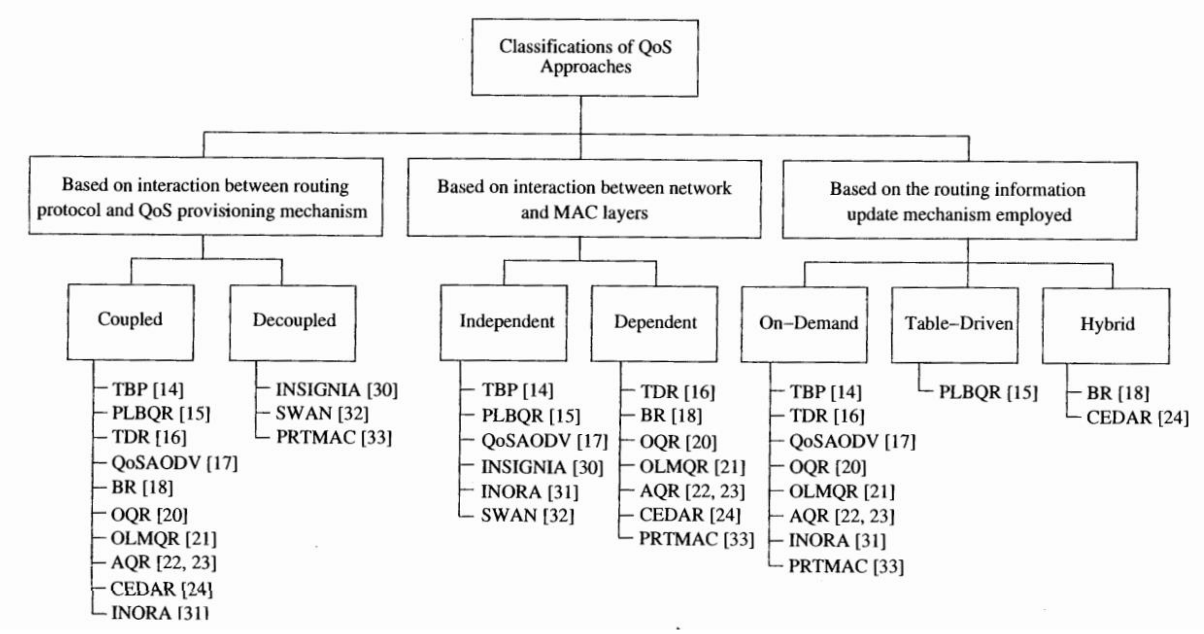


Figure 5.6. Classifications of QoS approaches

Similarly, based on the interaction between the routing protocol and the MAC protocol, QoS approaches can be classified into two categories: independent and dependent QoS approaches. In the independent QoS approach, the network layer is not dependent on the MAC layer for QoS provisioning. The dependent QoS approach requires the MAC layer to assist the routing protocol for QoS provisioning. Finally, based on the routing information update mechanism employed, QoS approaches can be classified into three categories, namely, table-driven, on demand, and hybrid QoS approaches. In the table-driven approach, each node in the network maintains a routing table which aids in forwarding packets. In the on-demand approach, no such tables are maintained at the nodes, and hence the source node has to discover the route on the fly. The hybrid approach incorporates features of both the table-driven and the on-demand approaches.

Part-A Questions

1. Compare TCP-F and TCP-Bus
2. List the major security threats that exist in ad hoc wireless networks.
3. Determine the major objectives of the transport layer protocol.
4. List the issues and challenges in security provisioning of transport layer.

5. Organize the steps involved in secure routing.
6. Identify the reasons for the requirement of secure routing protocols.
7. What is feedback-based TCP?
8. List some of the network layer attacks.
9. Define Ad hoc TCP.
10. Classify the transport layer solutions.

Part-B Questions

1. Identify the issues in designing a transport layer protocol for ad hoc wireless networks.
2. Determine reason for TCP's poor performance in ad hoc wireless network? Explain.
3. List various network layer attacks and describe about any one attack in detail.
4. Find the challenges in providing QoS in ad hoc wireless networks.
5. Classify QoS solutions and explain in detail.

TEXT / REFERENCE BOOKS :

1. Ibrahiem M.M. El Emary, Ramakrishnan.S, "Wireless Sensor Networks From Theory to Applications", CRC Press, 2013.

2. Fei Hu, Xiaojun Cao, "Wireless Sensor Networks Principles and Practice", CRC Press, 2010.

3. MounirFrikha, " Ad hoc Networks Routing, Qos and Optimization", Wiley, 2011.

4. Raheem, Beyah, Janise McNair, Cherita Corbett, Security in Ad hoc and Sensor Networks", World Scientific, 2010.