Simulation &Analysis of a Power Aware Routing Protocol in Wireless Sensor Networks

Thesis submitted in partial fulfillment of the requirements for the award the of degree of

Master of Engineering

in

Computer Science and Engineering

Thapar University, Patiala

Submitted By:

Subhash Junas

(80732022)

Under the supervision of

Dr. A. K. Verma

(Assistant Professor CSED)

May 2009

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

CERTIFICATE

I hereby certify that the matter which is being presented in the thesis entitled, "*Simulation & Analysis of a Power Aware Routing Protocol in Wireless Sensor Networks*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Computer Science and Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authenticated record of my own work carried out under the supervision of *Dr. Anil Kumar Verma* and refers other researcher's works which are duly listed in the reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Subhash Junas

ME-CSE 80732022

This to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Dr. A K Verma,

Assistant Professor,

Computer science and Engineering Department

Thapar University, Patiala

Countersigned by:

Professor and Head

Computer Science and Engineering Department

Thapar University, Patiala

Dr. R.K Sharma

Dean, Academic Affairs

Thapar University, Patiala

ACKNOWLEDGEMENT

"The need to be right all the time is the biggest bar to new ideas. It is better to have ideas for some of them to be wrong than to be always right by having no ideas at all"

-Edward de Bono

I am highly thankful to my guide Dr. Anil Kumar Verma, Assistant Professor, Computer Science & Engineering Department, Thapar University, Patiala, for his advice, motivation, guidance, moral support, efforts, and the attitude with which he solved all my queries in making this thesis possible. It has been a great honor to work under him.

I am also thankful to Dr. Rajesh Bhatia, Head of Department and Professor Seema Bawa, Computer Science and Engineering Department, Thapar University, Patiala for providing us with adequate infrastructure in carrying the research work.

I am also thankful to all he faculty of the computer science and engineering for their support all thorough my ME course. I am also thankful to my friends Vaibhav bhadade, Jasmeet Singh, Mukesh Kumar, Mayank Kuamr, Vineet Khera, Sandeep Khode, and Sushil Yadav for their kind and noble support.

I am forever indebted to my parents for their constant support and encouragement throughout the past years. This thesis is devoted to them.

Subhash Junas

Simulation &Analysis of a Power Aware Routing Protocol in Wireless Sensor Networks

Abstract

Wireless sensor network has revolutionized the networking field due to their capabilities to provide data **anytime, anywhere and anyhow**. This ability of wireless revolution has also opened the other arenas, such as MANET, VANET and wireless sensor networks.

The wireless sensor network (WSN) has proved to be useful in the different filed of the applications, where the traditional network have failed. It is their characteristics like limited power consumption, ability to withstand harsh environmental conditions, ability to cope with the node failures, Mobility of nodes, Dynamic network topology, heterogeneity of nodes, large scale of deployment, handling unattended operations which provide them the ability to solve the complex problems such as temperature control, humidity control, vehicular movement, lightening conditions, pressure, soil make up, noise levels, the presence or absence of certain kinds of objects, mechanical stress levels on attached objects, the current characteristics such as speed, direction, and size of an object.

As wireless sensor network has changed the definition of the networking field by providing data anywhere, anytime and anyhow but there are some factors that influences the working of WSN and one of them in the power consumption and battery life of the sensor nodes. In this work a power aware protocol has been proposed. The major factor that is affecting the routing is considered as the battery life of the sensor nodes working within the Wireless Sensor Network (WSN). Further the battery life of the sensor nodes being the major concern in WSN, so this work proposed a routing approach that can efficiently increase the battery life of the sensor nodes.

Keywords: WSNs, TinyOS, Simulators, battery life, power consumption.

Table of Content

LIST OF FIGURES

LIST OF ABBREVIATIONS

WSN: Wireless Sensor Network
MANET: Mobile Adhoc Network
VANET: Vehicular Adhoc Network
TOS: TinyOS
ADCs: Analog-to-Digital
RF: Radio Frequency
GPS: Global Positioning System
NBC: Nuclear Biological and Chemical
LAN: Local Area Network
TCP: Transmission Control Protocol
IEEE: Institute of Electrical and Electronics Engineers
TOSSIM: TinyOS Sensor Simulator
CPU: Central Processing Unit
MAC: Media Access Control
OPNET: Open Network
DoS: Denial-of-Service
TTL: Time-to-Live
ARQ: Automatic Repeat reQuest
SIR: Signal-To-Interface Ratio

CHAPTER 1
INTRODUCTION

1.1 Wireless Sensor Networks

A Wireless Sensor Network (WSN) is a network comprised of numerous small independent sensor nodes or motes. They merge a broad range of information technology; hardware, software, networking, and programming methodologies [1]. A wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations [1].

The development of wireless sensor networks was originally motivated by military applications such as battlefield surveillance. However, wireless sensor networks are now used in many civilian application areas, including environment and habitat monitoring, healthcare applications, home automation, and traffic control.

As we know that WSN consist of the large numbers of sensors but in addition to the sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller, and an energy source, usually a battery. The size of a single sensor node can vary from shoebox-sized nodes down to devices the size of grain of dust, although functioning 'motes' of genuine microscopic dimensions have yet to be created. Each mote in a wireless sensor network is a self-contained unit comprised of a power supply (generally batteries), a communication device (radio transceivers), a sensor or sensors, analog-to-digital converters (ADCs), a microprocessor, and data storage [1, 2].

The motes self organize themselves, into wireless networks (**Figure 1.1**) and data from the motes is relayed to neighboring motes until it reaches the desired destination for processing [3]. Each mote has very limited resources in terms of processing speed, storage capacity and communication bandwidth. In addition, their lifetime is determined by their ability to conserve power [4]. These limitations are a significant factor and must be addressed when designing and implementing a wireless sensor network for a specific application.

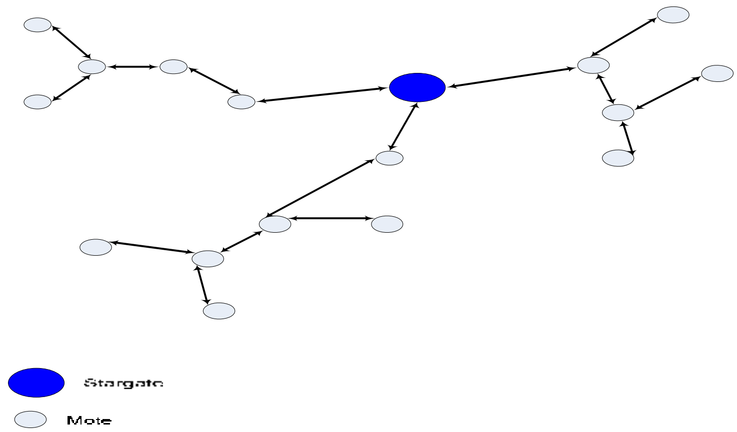


Figure 1.1: Example of a Flat Network [3]

The cost of sensor nodes is similarly variable, ranging from hundreds of dollars to a few cents, depending on the size of the sensor network and the complexity required of individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and bandwidth. A sensor network normally constitutes a wireless ad-hoc network, meaning that each sensor supports a multi-hop routing algorithm (several nodes may forward data packets to the base station).

Wireless Sensor Networks can be applied to a range of applications monitoring of space which includes environmental and habitat monitoring, indoor climate control, surveillance etc. monitoring things for example structural monitoring, condition-based equipment maintenance etc. and monitoring the interactions of things with each other and the surrounding space e.g., emergency response, disaster management, healthcare etc. The majority of these applications may be split into two classifications: **data collection and event detection.**

1.2 Data Collection versus Event Detection

As stated above, in general wireless sensor networks can be categorized into one of two types, data collection or event detection networks. In many applications where data collection is the goal, the sensors may be required to collect data for short periods at set times of the day. In this case, most of the time the sensor node will be asleep thus conserving power. However, when a wireless sensor network is to be employed for event detection, such as detecting the ignition of a fire, it would be anticipated that the sensor nodes must remain awake thus consuming their precious limited power [5].

1.3 Hardware and Tools for Developing a Wireless Sensor Network

A variety of hardware and tools are available for deploying and testing wireless sensor networks. Brief descriptions are provided hereafter.

1.3.1 Sensor Mote Hardware Architecture

Currently one of the most popular research platforms is the Mica2 sensor mote shown in Figure 1.2. It uses the TinyOS (TOS) Distributed Software Operating System, has a 325, 433 or 868/916 MHz multi-channel radio transceiver and an expansion connector that can be used for light, temperature, RH, barometric pressure, acceleration/seismic, acoustic and magnetic sensor boards [6].

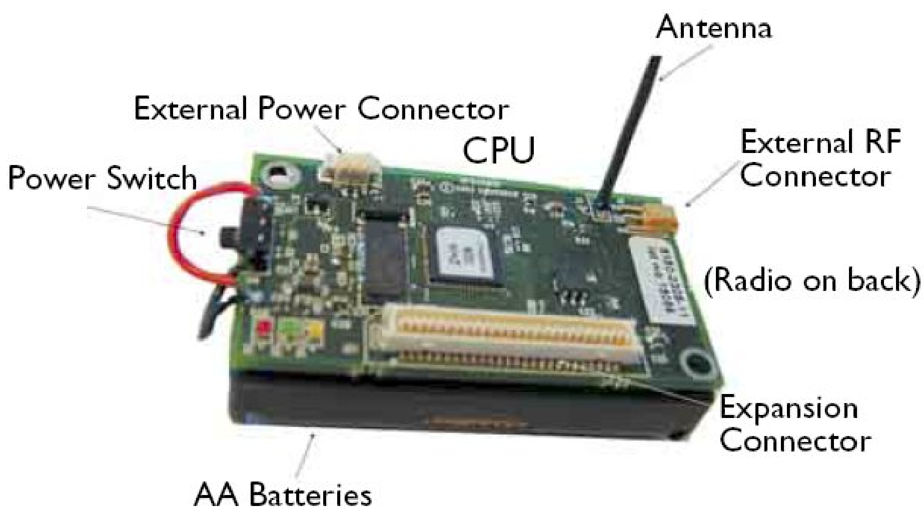


Figure 1.2: Mica2 Sensor Mote [6]

1.4 Working Description

A typical wireless sensor network consists of a base station and several nodes distributed or positioned in the environment of interest. The base stations are one or more distinguished components of the WSN with much more computational, energy and communication resources. They act as a gateway between sensor nodes and the end user.

Each node is expected to detect events of interest and estimate parameters that characterize these events. The resulting information at a node needs to be transmitted to the base station either directly or in "multi-hop" fashion involving automatic routing through several other nodes in the network. Implementation of such a network requires hardware components and corresponding software modules to program these components in a cooperative manner.

A commercial hardware platform that we have been investigating consists of processor cum radio boards commonly referred to as "motes". Each mote, a battery-powered device, consists of a sensor unit, a power unit, a two-way ISM band radio transceiver unit (includes an RF antenna), an ADC unit, a processor that runs TinyOS-based code, and longer memory capable of storing up to 100,000 measurements.

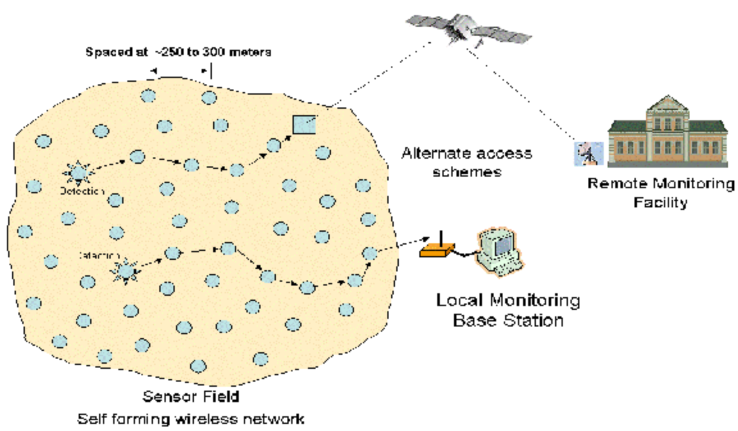
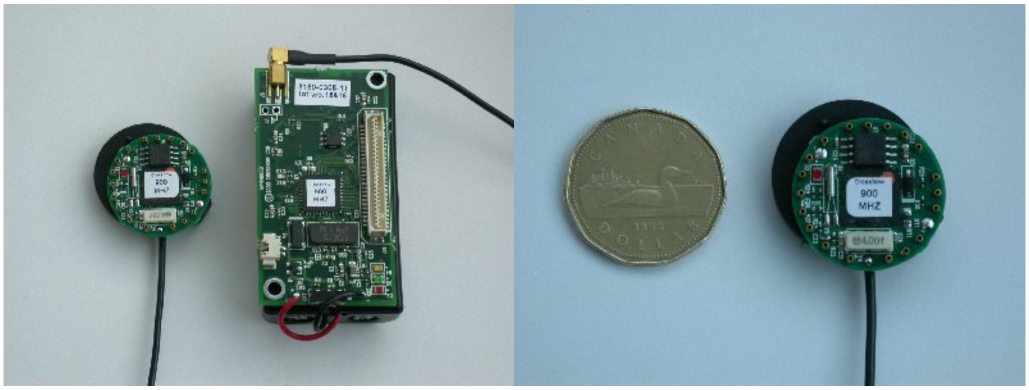


Figure 1.3 wireless sensor networks

A base station consists of a mote attached to a mote-interface-board that is interfaced to a PC via the parallel port. Two types of motes in this commercial system, namely, mica2dot and mica2 are shown in Figure 1. Included in the product are sensor boards that directly mate with a mote. These boards consist of sensor modules e.g. accelerometer, magnetometer, microphone, photo sensor, thermistor, and in addition, allow integration of user's own sensors.

Reception and transmission is available at different frequencies. Professional mote development kits available in the market allow frequencies of 315MHz, 433 MHz and 868/916 MHz with typical coverage of 1000ft, 1000ft and 500ft respectively. Each mote is powered by an external battery.



(a) (b)

Figure 1.4 (a) Mica2dot and mica2 motes supplied by Crossbow Technologies Inc. (b) As can be seen, mica2dot is comparable to a loonies in terms of size.

Power management in wireless sensor networks is an important aspect that will drive research in the next decade, and low-power sensor and RF transceivers are crucial for key applications in this area. The state-of-the-art in power management involves defining “wake” and “sleep” cycles. A mote draws mA current levels in data-gathering or wake mode and only draws μA current levels in sleep mode. Various other practical aspects including feasibility studies are being pursued.

In terms of software, the operating system for programming this particular sensor network is called TINYOS [7]. TinyOS is an event-driven operating system designed for sensor network nodes that have very limited resources (e.g. 8K bytes of program memory, 512 bytes of RAM). The programming language used to control and run the hardware components to perform user-defined tasks is called nesC. nesC is an extension to C designed to embody the structuring concepts and execution model of TinyOS.

1.5 Classification and application of WSNs

As the technology gains popularity, research is becoming important in both theoretical and application domains [8] [9]. The applications for WSNs are many and varied, but typically involve some kind of monitoring, tracking, and controlling. Specific applications for WSNs include habitat monitoring, object tracking, nuclear reactor control, fire detection, and traffic monitoring. In a typical application, a WSN is scattered in a region where it is meant to collect data through its sensor nodes. **We identify two classes of application examples below.**

A. Stationary Network:

We define a stationary network as a network of sensor nodes, in which, each sensor node's position is fixed relative to the base station and other nodes in the network. A demonstrated application in this direction is humidity monitoring in a vineyard [10]. Data acquired by a mote is transmitted to the base station, which then processes the information and triggers necessary actions such as localized watering. There are two possible cases to transmit data. When a node is in direct wireless contact i.e. in the range of the base station, direct communication is possible. When a node is not in the range, it transmits data in an ad-hoc environment also referred to as multi-hop. Implementation of an efficient multi-hop system requires optimal routing [11] to facilitate shortest route, reduced power consumption and improved transmission.

B. Network in Motion:

An example in this category is a herd of animals on an extensive farm, where each animal is equipped with a sensor node. The animals are in constant motion relative to each other as well as the base station. Such a complicated mobility management requires an even more sophisticated implementation of routing algorithms. In order to maximally benefit from wireless sensor networks of this type, we foresee additional hardware requirements in the form of GPS devices and other forms of mote location.

1.5.1 Area monitoring

Area monitoring is a common application of WSNs. In area monitoring, the WSN is deployed over a region where some phenomenon is to be monitored. As an example, a large quantity of sensor nodes could be deployed over a battlefield to detect enemy intrusion instead of using landmines.

When the sensors detect the event being monitored (heat, pressure, sound, light, electro-magnetic field, vibration, etc), the event needs to be reported to one of the base stations, which can take appropriate action (e.g., send a message on the internet or to a satellite).

1.5.2 A real Life Application

As part of the on-going work, a comprehensive "design framework" of an Intelligent Wireless patient-monitoring system has been developed. This framework includes

Real-time sensing of patient's vital parameters using the motes, and wireless transmission of such critical information over radio frequencies to the base-station. Subsequent data processing on a PC will allow detection of certain medical emergencies, and automatic alerting of medical staff. The overall framework is shown in Figure 1.5.

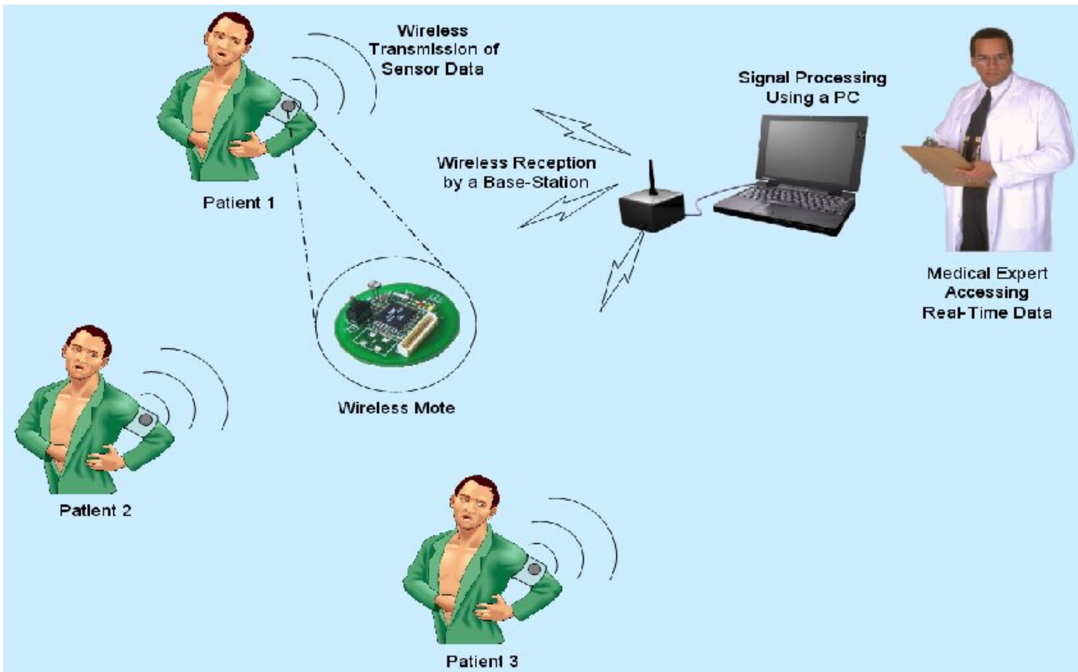


Figure 1.5 WSN application

In phase I of the project, a simplified prototype system utilizing a basic wireless network kit and sensor components have been implemented for the purpose of experimentation.

In phase II of the project, biomedical oriented low-power high-precision sensors will be designed and possibilities of developing a digital (Bluetooth) version of the wireless Senor system will be studied for advancement of the prototype. Prototype system will be tested in a hospital environment and will be refined towards meeting ethical, medical and technological standards in phase III. Potential deliverables will include:

- An Wireless Sensor self-organizing wireless system capable of close Patient-Monitoring even when patients are in motion.
- A Smart-Bed with increased automation in terms of emergency detection.

These capabilities could lead to efficient use of our valuable healthcare resources.

1.5.3 Military applications

Wireless sensor networks can be an integral part of military command, control, communications, computing, intelligence, surveillance, reconnaissance and targeting systems. The rapid deployment, self-organization and fault tolerance characteristics of sensor networks make them a very promising sensing technique for military.

Some of the military applications of sensor networks are monitoring friendly forces, equipment and ammunition; battlefield surveillance; reconnaissance of opposing forces and terrain; targeting; battle damage assessment; and nuclear, biological and chemical (NBC) attack detection and reconnaissance.

1.5.4 Environmental applications

Some environmental applications of sensor networks include tracking the movements of birds, small animals, and insects; monitoring environmental conditions that affect crops and livestock; irrigation; macro instruments for large-scale Earth monitoring and planetary exploration; chemical/biological detection; precision agriculture; biological, Earth, and environmental monitoring in marine, soil, and atmospheric contexts; forest fire detection; meteorological or geophysical research; flood detection; bio-complexity mapping of the environment; and pollution study.

Forest fire detection: Since sensor nodes may be strategically, randomly, and densely deployed in a forest, sensor nodes can relay the exact origin of the fire to the end users before the fire is spread uncontrollable.

1.5.5 Home applications: Home automation

As technology advances, smart sensor nodes and actuators can be buried in appliances, such as vacuum cleaners, micro-wave ovens, refrigerators, and VCRs. These sensor nodes inside the domestic devices can interact with each other and with the external network via the Internet or Satellite. They allow end users to manage home devices locally and remotely more easily.

1.5.6 Other commercial applications

Some of the commercial applications are monitoring material fatigue; building virtual keyboards; managing inventory; monitoring product quality; constructing smart office spaces; environmental control in office buildings; robot control and guidance in automatic manufacturing environments; interactive toys; interactive museums; factory process control and automation; monitoring disaster area; smart structures with sensor nodes embedded inside; machine diagnosis; transportation; factory instrumentation; local control of actuators; detecting and monitoring car thefts; vehicle tracking and detection; and instrumentation of semiconductor processing chambers, rotating machinery, wind tunnels, and anechoic chambers.

Detecting and monitoring car thefts: Sensor nodes are being deployed to detect and identify threats within a geographic region and report these threats to remote end users by the Internet for analysis.

1.5.7 Managing inventory control

Each item in a warehouse may have a sensor node attached. The end users can find out the exact location of the item and tally the number of items in the same category. If the end users want to insert new inventories, all the users need to do is to attach the appropriate sensor nodes to the inventories. The end users can track and locate where the inventories are at all times.

1.5.8 Vehicle tracking and detection

There are two approaches as described in to track and detect the vehicle:

First, the line of bearing of the vehicle is determined locally within the clusters and then it is forwarded to the base station,

Second, the raw data collected by the sensor nodes are forwarded to the base station to determine the location of the vehicle.

1.6 Factors influencing sensor network design

A sensor network design is influenced by many factors, which include fault tolerance; scalability; production costs; operating environment; sensor network topology; hardware constraints; transmission media; and power consumption.

These factors are important because they serve as a guideline to design a protocol or an algorithm for sensor networks. In addition, these influencing factors can be used to compare different schemes.

1.6.1 Fault tolerance

Some sensor nodes may fail or be blocked due to lack of power, have physical damage or environmental interference. The failure of sensor nodes should not affect the overall task of the sensor network. This is the reliability or fault tolerance issue. Fault tolerance is the ability to sustain sensor network functionalities without any interruption due to sensor node failures.

Sensor nodes being deployed in a house to keep track of humidity and temperature levels, the fault tolerance requirement may be low since this kind of sensor networks is not easily damaged or interfered by environmental noise.

On the other hand, sensor nodes being deployed in a battlefield for surveillance and detection, then the fault tolerance has to be high because the sensed data are critical and sensor nodes can be destroyed by hostile actions. As a result, the fault tolerance level depends on the application of the sensor networks, and the schemes must be developed with this in mind.

1.6.2 Scalability

The number of sensor nodes deployed in studying a phenomenon may be in the order of hundreds or thousands. Depending on the application, the number may reach an extreme value of millions. The density can range from few sensor nodes to few hundred sensor nodes in a region, which can be less than 10 m in diameter.

In general, the density can be as high as 20 sensor nodes/m. A home may contain around two dozens of home appliances containing sensor nodes, but this number will grow if sensor nodes are embedded into furniture and other miscellaneous items. For habitat monitoring application, the number of sensor nodes ranges from 25 to 100 per region.

1.6.3 Environment

Sensor nodes are densely deployed either very close or directly inside the phenomenon to be observed. Therefore, they usually work unattended in remote geographic areas. They may be working:

- in busy intersections,

- in the interior of a large machinery,
- at the bottom of an ocean,
- inside a twister,
- on the surface of an ocean during a tornado,
- in a biologically or chemically contaminated field,
- in a battlefield beyond the enemy lines,
- in a home or a large building,
- in a large warehouse,

This list gives us an idea about under which conditions sensor nodes are expected to work. They work under high pressure in the bottom of an ocean, in harsh environments such as a battlefield, under extreme heat and cold such as in the nozzle of an aircraft engine or in arctic regions, and in an extremely noisy environment such as under intentional jamming.

1.6.4 Power consumption

The wireless sensor node, being a micro-electronic device, can only be equipped with a limited power source. In some application scenarios, replenishment of power resources might be impossible.

Sensor node lifetime, therefore, shows a strong dependence on battery lifetime. In a multihop Wireless Sensor network, each node plays the dual role of data originator and data router. The disfunctioning of few nodes can cause significant topological changes and might require re-routing of packets and re-organization of the network. Hence, power conservation and power management take on additional importance. It is for these reasons that researchers are currently focusing on the design of power-aware protocols and algorithms for sensor networks.

In other mobile and Wireless Sensor networks, power consumption has been an important design factor because power efficiency is an important performance metric, directly influencing the network lifetime. Application specific protocols can be designed by appropriately trading off other performance metrics such as delay and throughput with power efficiency.

The main task of a sensor node in a sensor field is to detect events, perform quick local data processing, and then transmit the data. Power consumption can hence be divided into three domains: sensing, communication, and data processing. Sensing power varies with the nature of applications. Sporadic sensing might consume lesser power than constant event monitoring.

1.7 TinyOS (Operating System for WSN)

The event-driven nature of sensor networks means that testing an individual mote is insufficient. Programs must be tested at scale and in complex and rich conditions to capture a wide range of interactions. Deploying hundreds of motes is a daunting task, the focus of work shifts from research to maintenance, which is time-consuming due to the failure rate of individual motes.

A simulator can deal with these difficulties, by providing controlled, reproducible environments, by enabling access to tools such as debuggers, and by postponing deployment until code is well tested and algorithms are understood. TinyOS is an operating system specifically designed for sensor networks. It has a component-based programming model, provided by the nesC [12] language, a dialect of C.

TinyOS is not an OS in the traditional sense. A TinyOS program is a graph of components, each of which is an independent computational entity. Each TinyOS component has a frame, a structure of private variables that can only be referenced by that component.

Components have three computational abstractions: *commands*, *events*, and *tasks*. Commands and events are mechanisms for inter-component communication, while tasks are used to express intra-component concurrency. A *command* is typically a request to a component to perform some service, such as initiating a sensor reading, while an *even t*signals the completion of that service. Events may also be signaled asynchronously, for example, due to hardware interrupts or message arrival. From a traditional OS perspective, commands are analogous to down calls and events to up calls. Commands and events cannot block: rather, a request for service is *split phase* in that the request for service (the command) and the completion signal (the corresponding event) are decoupled. The command returns immediately and the event signals completion at a later time.
Rather than performing a computation immediately, commands and event handlers may post a *task*, a function executed by the TinyOS scheduler at a later time. This

allows commands and events to be responsive, returning immediately while deferring extensive computation to tasks. While tasks may perform significant computation, their basic execution model is run-to-completion, rather than to run indefinitely. This allows tasks to be much lighter-weight than threads. Tasks represent internal concurrency within a component and may only access that component's frame.

The TinyOS task scheduler uses a non-preemptive, FIFO scheduling policy. TinyOS abstracts all hardware resources as components. For example, calling the getData() command on a sensor component will cause it to later signal a dataReady() event when the hardware interrupt fires. While many components are entirely software-based, the combination of split-phase operations and tasks makes this distinction transparent to the programmer. In both cases an event signals that the encryption operation is complete. ADC, ClockC, UART, SlavePin and SpiByteFifo are example hardware abstraction components. TinyOS commands and events are very short, due to limited code space and a finite state machine style of decomposition. The rich event processing model means an event or command call path can traverse several components.

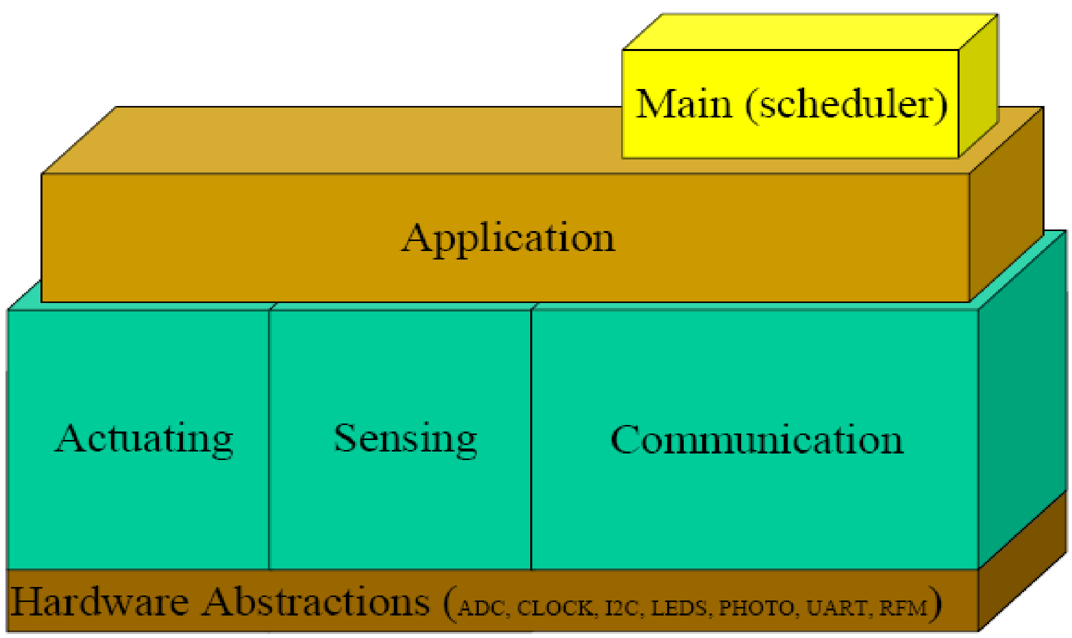


Figure 1.6 TinyOS Structure (Consist of scheduler and graph of components)

The TinyOS component model allows us to easily change the target platform from mote hardware to simulation by only replacing a small number of low-level components. The event-driven execution model can be exploited for efficient event-driven simulation, and the whole program compilation process can be re-targeted for the simulator's storage model and native instruction set.

The static component memory model of TinyOS simplifies state management for these large collections. Setting the right level of simulation abstraction can accurately capture the behavior and interactions of TinyOS applications.

1.8 TOSSIM: A Simulator for TinyOS Sensor Networks

1.8.1 The Necessity of Network Simulation

The emergence of wireless sensor networks brought many open issues to network designers. Traditionally, the three main techniques for analyzing the performance of wired and wireless networks are *analytical methods*, *computer simulation*, *and physical measurement*.

However, because of many constraints imposed on sensor networks, such as energy limitation, decentralized collaboration and fault tolerance, algorithms for sensor networks tend to be quite complex and usually defy analytical methods that have been proved to be fairly effective for traditional networks. Furthermore, few sensor networks have come into existence, for there are still many unsolved research problems, so measurement is virtually impossible. It appears that simulation is the only feasible approach to the quantitative analysis of sensor networks.

The event-driven nature of sensor networks means that testing an individual mote is insufficient. Programs must be tested at scale and in complex and rich conditions to capture a wide range of interactions. Deploying hundreds of motes is a daunting task, the focus of work shifts from research to maintenance, which is time-consuming due to the failure rate of individual motes.

A simulator can deal with these difficulties, by providing controlled, reproducible environments, by enabling access to tools such as debuggers, and by postponing deployment until code is well tested and algorithms are understood

1.8.2 TOSSIM

TOSSIM is a discrete event simulator for TinyOS sensor networks. Instead of compiling a TinyOS application for a mote, users can compile it into the TOSSIM [13] framework, which runs on a PC. This allows users to debug, test, and analyze algorithms in a controlled and repeatable environment. As TOSSIM runs on a PC, users can examine their TinyOS code using debuggers and other development tools.

TOSSIM's primary goal is to provide a high fidelity simulation of TinyOS applications. For this reason, it focuses on simulating TinyOS and its execution, rather than simulating the real world. While TOSSIM can be used to understand the causes of behavior observed in the real world, it does not capture all of them, and should not be used for absolute evaluations.

CHAPTER 2

LITERATURE SURVEY

Most wireless sensor devices today operate on batteries. Hence, power consumption becomes an important issue. To maximize the lifetime of Wireless Sensor Network devices, the power consumption rate of each node must be evenly distributed, and the overall transmission power for each connection request must be minimized.

Since existing wired routing protocols cannot be used directly in Wireless Sensor Network environment, therefore different routing schemes should be proposed for Wireless Sensor networks. The performance of a Wireless Sensor wireless network is dependent on the routing algorithm or protocol employed.

2.1 Characteristics of Wireless Sensor Network.

A Wireless Sensor Network is an autonomous system consisting of sensor nodes that do not rely on the presence of any fixed network infrastructure. Depending on the nodes' geographical positions, their transceiver coverage patterns, transmission power levels, and co-channel interference levels, a network can be formed and unformed on the fly. This Wireless Sensor network topology changes as sensor nodes migrate, "disappear" (failure or depletion of battery capacity), or adjust their transmission and reception characteristics. The main characteristics of Wireless Sensor networks are:

- **Dynamic topology**: Nodes are free to move about arbitrarily. In addition, radio propagation conditions change rapidly over time. Thus, the network topology may change randomly and rapidly over unpredictable times.
- **Bandwidth constraints and variable link capacity**: Wireless links have significantly lower capacity than wired links. Due to the effects of multiple accesses, multipath fading, noise, and signal interference, the capacity of a wireless link can be degraded over time and the effective throughput may be less than the radio's maximum transmission capacity.
- **Energy constrained nodes**: Sensor nodes rely on batteries for proper operation. Since a Wireless Sensor network consists of several nodes, depletion of batteries in these nodes will have a great influence on overall

network performance. Therefore, one of the most important protocol design factors is related to device energy conservation.

- **Multi-hop communications**: Due to signal propagation characteristics of wireless transceivers, Wireless Sensor networks require the support of multihop communications; that is, sensor nodes that cannot reach the destination node directly will need to relay their messages through other nodes.
- **Limited security**: Mobile wireless networks are generally more vulnerable to security threats than wired networks. The increased possibility of eavesdropping, spoofing, and denial-of-service (DoS) attacks should be carefully considered when a Wireless Sensor wireless network system is designed. To support mobile computing in Wireless Sensor wireless networks, a sensor node must be able to communicate with other sensor nodes, which may not lie within its radio transmission range. To support Wireless Sensor Network communications under the influence of the above-mentioned factors, an Wireless Sensor routing protocol will need to perform four functions, namely:
- Determining/detecting changing network topology
- Maintaining network topology/connectivity
- Scheduling of packet transmission and channel assignment

2.2 Determination of Network Topology

A Wireless Sensor routing protocol must determine and monitor the dynamics of network topology over time. Since multihop communications are needed in Wireless Sensor networks, the routing protocol must ensure that links in the route have strong connectivity. There must exist at least one path from any node to any other node in a no partitioned network.

A node in the Wireless Sensor network must be aware of its surrounding environment and the neighboring nodes with which it can directly communicate. It must take into account the relative difficulties of forming links with those nodes and the benefits such connections will bring to the network on the whole (e.g., bandwidth usage, transmission delay, throughput, power consumption). Basically, there are two approaches in providing Wireless Sensor network connectivity:

- *Flat-routed* network architecture
- *Hierarchical* network architectures [14]

In the flat-routed network architecture, all the nodes are equal and packet routing is done based on peer-to-peer connections. However, in hierarchical networks, at least one node in each lower layer is designated to serve as a gateway or coordinator to higher layers.

2.3 Maintaining Network Connectivity under Changing Radio Conditions

Since the location of each sensor node can change over time, the network topology also changes frequently as a result. Topological changes can occur due to:

- The breakdown of a sensor node in a hostile environment
- The failure of a connected link due to signal interference and changes in signal propagation

Conditions Therefore, a Wireless Sensor routing protocol must be able to dynamically update the status of its links and reconfigure itself in order to maintain strong connectivity to support communications among the nodes

2.4 Transmission Scheduling and Channel Assignment

Since a new radio transmission by a sensor node may affect an existing communication link through signal interference, an efficient packet scheduling and channel assignment algorithm is needed to ensure that the new transmissions will not conflict with an existing one. In addition, it will ensure efficient use of the limited available bandwidth. In a single-channel radio system, transmissions over the wireless media should be adequately scheduled in order to maintain sufficient spatial distances among the nodes.

2.5 Packet Routing

Compared to wired networks with static nodes, Wireless Sensor networks require a highly adaptive routing scheme to cope with the high rate of topology changes. This implies that the routing protocol should propagate topology changes and compute updated routes to the destination. In general, existing routing protocols can be classified as either:

- Table-driven
- On-demand
- Hybrid [14, 15]

Table-driven protocols attempt to continuously update the routes within the network so that when a packet needs to be forwarded, the route is already known and can immediately be used. The family of distance-vector or link-state algorithms is examples of table-driven schemes. On the other hand, on-demand schemes invoke a route discovery procedure only on a need basis. Thus, when a route is needed, some sort of global or localized search procedure is employed.

2.6 Desired Properties of Wireless Sensor Routing Protocols

The desirable properties of Wireless Sensor routing protocols are discussed below.

2.6.1 Distributed Implementation

Since Wireless Sensor wireless networks are autonomous and self-organizing systems, routing protocols must be distributed in nature without relying on centralized authorities.

2.6.2 Efficient Utilization of Bandwidth

If a routing protocol incurs excessive control traffic, the available network bandwidth will be consumed by control traffic. This can impact communication performance. Since the bandwidth of a wireless network is limited, reduction of control overhead is an important design factor.

2.6.3 Efficient Utilization of Battery Capacity

The lifetime of sensor nodes will deeply impact the performance of the Wireless Sensor Network. To prolong the lifetime of each node, Wireless Sensor routing protocols should consider power consumption. For example, routing protocols should be able to accommodate sleep periods without causing any adverse consequences; that is, hosts can stop transmitting and/or receiving for arbitrary periods of time when it is ideal. Moreover, transmission power can be used as a routing metric.

2.6.4 Optimization of Metrics

End-to-end throughput and delay are widely used performance metrics in wired and wireless networks. However, since the network topology is dynamically changing, the bandwidth and battery power are important factors in Wireless Sensor networks but one has to consider other metrics as well. Such metrics can influence the design of routing protocols, and there exist trade-offs in using different metrics. For example, although on-demand routing algorithms can reduce control overhead (i.e., optimizing the bandwidth), it requires some route acquisition time (i.e., the time required to discover and establish a route when desired), thus increasing end-to-end delay. Therefore, in routing protocol design one should optimize some reasonable metrics in addition to others. The following is a list of metrics worthy of consideration:

- Maximum end-to-end throughput
- Minimum end-to-end delay
- Shortest path/minimum hop
- Minimum total power (battery capacity)
- Load balancing (least congested path)
- Minimum overhead (bandwidth)
- Adaptability to the changing topology
- Association stability [16] (i.e., longevity of routes)
- Route relaying load [16]

2.6.5 Fast Route Convergence

Since the network topology dynamically changes, routing protocols should provide a new and stable route as soon as possible after a topology change.

2.6.6 Freedom from Loops

The paths derived from the routing tables of all nodes should not have loops. Some routing algorithms can cause temporary loops (i.e., a small fraction of packets spinning around the network for an arbitrary period of time). Looping of packets can result in considerable overhead in terms of bandwidth and power consumption. Using time-to-live (TTL) values can help mitigate this problem, but a more structured and well-thought-out approach is desirable because it can lead to better communication performance.

2.6.7 Power-Efficient Wireless Sensor Network and Efficient Utilization of Battery Power

Since most sensor nodes of a Wireless Sensor network today operate using batteries, it is important to minimize the power consumption of the entire network (implying maximizing the lifetime of ad hoc networks). The power required by each sensor node can be classified into two categories:

- Communication-related power
- Non-communication-related power

The former can be further divided into two parts, namely:

- Processing power
- Transceiver power

Each sensor node spends some processing power to execute network algorithms and run applications. Transceiver power refers to the power used by the radio transceiver to communicate with the other sensor nodes. In mobile power consumption, each protocol layer is closely coupled. For example, if a routing protocol requires frequent updates of routing information, it is difficult to implement sleep mode at the data link layer. We therefore briefly summarize the power conservation schemes for each layer below.

***Physical Layer* —**

Currently, research has been done for hardware and circuit-level power optimization and management within a wireless device (e.g., low-power displays, low-power CPUs, power-efficient computing algorithms). Within a single wireless device, improved power efficiency may be achieved for conventional components (e.g., CPU and disk) by turning them off or slowing them down when not needed [17]. At the physical layer, transmission power can be adjusted. The use of excessive transmission power can increase the interference to other hosts and will cause an increase in transmission power by other hosts. Therefore, physical layer functions should include transmitting data at the minimum power level to maintain links, and adapt to changes in transmission environment (due to host mobility or co-channel interference) [18]. Summarizing, power control can maintain a link at the minimum power level, but can also prolong an existing link against interference by increasing the transmission power.

Data Link Layer —

At the data link layer, energy conservation can be achieved by using effective retransmission request schemes and sleep mode operation. The data link layer provides error-free communication between two nodes. It detects transmission errors and then retransmits the correct packet using automatic repeat request (ARQ). In Wireless Sensor networks, due to the presence of mobility and co-channel interference, transmission errors can occur frequently, which lead to frequent retransmission requests. Since retransmissions increase power consumption and cause higher interference to other users, a new efficient retransmission request scheme is needed for Wireless Sensor networks. One possible scheme is the following: when a transmitter does not receive acknowledgments after packet transmissions, the retransmission request scheme senses that the channel is bad and ceases retransmitting to not waste power when the chances of successful reception are dim [17]. This scheme, therefore, reduces unnecessarily power wastage at the expense of transmission delay. Another solution is for the retransmission request scheme to increase the retransmission power. This reduces the possibility of transmission errors but increases the signal to- interference ratio (SIR) of the network. Therefore, to appropriately determine when and at what power level a sensor node should attempt retransmission is an important issue for minimizing power incurred at the data link layer [18]. A node transmitting packets to its downstream nodes will be overheard by all neighboring nodes. Hence, all neighboring nodes will consume power even though the packet transmission was not directed to them [19]. Therefore, to reduce power consumption a node's transceiver should be powered off (snooze or not listening) when not in use.

Network Layer —

In wired networks, the emphasis has traditionally been on maximizing end-to-end throughput and minimizing delay. In general, paths are computed based on minimizing hop count or delay. To maximize the lifetime of sensor nodes, routing algorithms must select the best path from the viewpoint of power constraints as part of route stability. When power efficiency is considered, Wireless Sensor networks will require a routing algorithm that can evenly distribute packet-relaying loads to each node to prevent nodes from being overused or abused [20]. By maximizing the lifetime of all nodes, the time before the network is partitioned is prolonged.

2.7 Power-Efficient Wireless Sensor Routing Protocol

At the network layer, routing algorithms must select the best path to minimize the total power needed to route packets on the network and maximize the lifetime of all nodes. Following are the routing approaches that have been proposed to achieve the goal of power efficiency.

2.7.1 Minimum Total Transmission Power Routing

(MTPR) — In wireless communications, radio propagation can be modeled with a 1/*dn* transmit power roll off (usually, $n = 2$ for short distance and $n = 4$ for longer distance). For successful transmissions, the signal-to-noise ratio (SNR) received at a host *nj* should be greater than a specified predetection threshold Ψj. This threshold Ψj is closely related to the bit error rate (BER) of the received signal. For successful transmissions from a host *ni* to *nj*, the SNR at host *nj* should satisfy the following equation:

$$SNR_j = \frac{P_i G_{i,j}}{\sum_{k \neq i} P_K G_{k,j} + \eta_j} \Psi_j(BER),$$

Equation (1)

Where *Pi* is the transmission power of host *ni*, *Gi,j* is the path gain between hosts *ni* and *nj* and ηj is the thermal noise at host *nj*. Therefore, the minimum transmission power is dependent on interference noise, distance between hosts, and desired BER. To obtain the route with the minimum total power, the transmission power *P*(*ni*, *nj*) between hosts *nj* and *nj* can be used as a metric [19]. The total transmission power for route *l*, *Pl*, can be derived from

$$P_l = \sum_{i=0}^{D-1} P(n_i, n_{n+1}) \quad \text{for all node } n_i \in \text{route},$$

Equation (2)

where *n*0 and *nD* are the source and destination nodes, respectively. Therefore, the desired route k can be obtained from

$$P_k = \min_{l \in A} P_l,$$

where *A* is the set containing all possible routes. The above function can be solved by a standard shortest path algorithm such as Dijkstra or Bellman-Ford. In [20], Dijkstra's shortest path algorithm was modified to obtain the minimum total power route. However, since transmission power depends on distance1 (proportional to *dn*), this algorithm will select routes with more hops than other routing algorithms. In general, the more nodes involved in routing packets, the greater the end-to-end delay. In addition, a route consisting of more nodes is more likely to be unstable, because the probability that intermediate nodes will move away is higher. Hence, from the standpoint of minimum hops, the route obtained from the above algorithm is not attractive. To overcome this problem, transceiver power (the power used when receiving data) as well as transmission power were considered as a cost metric, and the distributed Bellman-Ford algorithm was used [21]. At node *nj*, it computes

$$C_{i,j} = P_{\text{transmit}}(n_i, n_j) + P_{\text{transceiver}}(n_j) + Cost(n_j),$$

Equation (3)

where *ni* is a neighboring node of *nj*, *P*transceiver(*nj*) is the transceiver power at node *nj*, and *Cost*(*nj*) is the total power cost from the source node to node *nj*. This value is sent to node *ni*. Subsequently, at node *ni* it computes its power cost by using the following equation:

$$Cost(n_i) = \min_{j \in NH(i)} C_{i,j}, \text{ where}$$
$$NH(i) = \{j; n_j \text{ is a neighbor node of } n_i\}.$$

Equation (4)

The path with minimum cost from the source node to node *ni* is selected. This procedure is repeated until the destination node is reached. In this algorithm, *P*transceiver (*nj*) helps the algorithm find routes with fewer hops than the MTPR algorithm because generally the transceiver power is identical for hosts using the same transceiver. However, it has a critical disadvantage. Although this metric can reduce the total power consumption of the overall network, it does not reflect directly on the lifetime of each host. If the minimum total transmission power routes are via a specific host, the battery of this host will be exhausted quickly, and this host will die of battery exhaustion soon.

2.7.2 Minimum Battery Cost Routing (MBCR) —

This routing scheme proposed that the remaining battery capacity of each host is a more accurate metric to describe the lifetime of each host [21]. Let *ct i* be the battery capacity of a host *ni* at time *t* ranging between 0 and 100. We define *fi* as a battery cost function of a host *ni*. Now, suppose a node's willingness to forward packets is a function of its remaining battery capacity. The less capacity it has, the more reluctant it is.

As proposed, one possible choice for *fi* is

$$f_i(c_i^t) = \frac{1}{c_i^t}.$$

Equation (1)

As the battery capacity decreases, the value of cost function for node *ni* will increase. The battery cost *Rj* for route *i*, consisting of *D* nodes, is

$$R_j = \sum_{i=0}^{D_j - 1} f_i(c_i^t).$$

Equation (2)

Therefore, to find a route with the maximum remaining battery capacity, we should select a route *i* that has the minimum battery cost.

$$R_i = \min\{R_j \mid j \in A\},$$

Equation (3)

where *A* is the set containing all possible routes. Since battery capacity is directly incorporated into the routing protocol, this metric prevents hosts from being overused, thereby increasing their lifetime and the time until the network is partitioned [21]. If all nodes have similar battery capacity, this metric will select a shorter-hop route. However, because only the summation of values of battery cost functions is considered, a route containing nodes with little remaining battery capacity may still be selected. Initially, it seems that the lifetime of all nodes will be elongated. However, on closer examination, since there is no guarantee that minimum total transmission power paths will be selected under all circumstances, it can consume more power to transmit user traffic from a source to a destination, which actually reduces the lifetime of all nodes.

2.8 Research Challenge

Wireless sensor networks (WSN) are currently receiving significant attention due to their unlimited potential. However, it is still very early in the lifetime of such systems and many research challenges exist. There are many research challenges for wireless sensor networks. Research challenges that must be met before WSN become pervasive are discussed as follows.

2.8.1 Real-world Protocols:

Network protocols that account for the key realities in wireless communication are required. New research is needed to:

- Measure and assess how the theoretical properties of wireless communication are exhibited in today's and tomorrow's sensing and communication devices,
- Establish better models of communication realities to feed back into improved simulation tools,
- Invent new network protocols that account for the communication realities of real world environments,
- Test the individual solutions on real platforms in real world settings, and

2.8.2 Real-Time:

WSN deal with real world environments. In many cases, sensor data must be delivered within time constraints so that appropriate observations can be made or actions taken. Very few results exist to date regarding meeting real-time requirements in WSN. Most protocols either ignore real-time or simply attempt to process as fast as possible and hope that this speed is sufficient to meet deadlines [22].

Dealing with real-time usually identifies the need for differentiated services, e.g., routing solutions need to support different classes of traffic; guarantees for the important traffic and less support for unimportant traffic.

It is important not only to develop real-time protocols for WSN, but associated analysis techniques must also be developed.

2.8.3 Power Management (in the large):

Low-cost deployment is one acclaimed advantage of sensor networks. Limited processor bandwidth and small memory are two arguable constraints in sensor networks. However, the energy constraint is unlikely to be solved soon due to slow progress in developing battery capacity. Moreover, the untended nature of sensor nodes and hazardous sensing environments preclude battery replacement as a feasible solution. On the other hand, the surveillance nature of many sensor network applications requires a long lifetime; therefore, it is a very important research issue to provide a form of energy-efficient surveillance service for a geographic area. Much of the current research focuses on how to provide full or partial sensing coverage in the context of energy conservation.

2.8.4 Security and Privacy:

WSN are limited in their energy, computation, and communication capabilities. In contrast to traditional networks, sensor nodes are often deployed in accessible areas, presenting a risk of physical attacks. Sensor networks interact closely with their physical environment and with people, posing additional security problems. Because of these reasons current security mechanisms are inadequate for WSN. These new constraints pose new research challenges on key establishment, secrecy and authentication, privacy, robustness to denial-of-service attacks [23], secure routing, and node capture.

To achieve a secure system, security must be integrated into every component, since components designed without security can become a point of attack. Consequently, security and privacy pervade every aspect of system design. One challenge is how to secure wireless communication links against eavesdropping and tampering. Overall, security is a difficult challenge for any system. The severe constraints and demanding environments of WSN make computer security for these systems even more challenging.

CHAPTER 3
PROBLEM STATEMENT

Power management is the major concern in the Wireless Sensor Networks. So, power consumption and battery life of the sensor nodes becomes an important issue and the same has been considered in the presented work. The issues for the wireless sensor networks that needs to be considered are, the power consumption rate of each node must be reduced and maximizing the lifetime of the sensor network. These two objectives can be achieved when routing within the wireless sensor network is handled properly along with the battery life of the sensor nodes which is partly satisfied in previous work. So this research work has proposed a routing approach based on battery life of the sensor nodes working within the Wireless Sensor Network.

The battery life constraint is unlikely to be solved soon due to slow progress in developing battery capacity. Moreover, the untended nature of sensor nodes and hazardous sensing environments preclude battery replacement as a feasible solution. On the other hand, the surveillance nature of many sensor network applications requires a long battery life; therefore, it is a very important research issue to provide a kind of solution to the battery life that can meet the needs of the upcoming applications.

3.1 Motivation

Efficient routing approach plays a significant role in saving the battery consumption of the network and to maximize the network lifetime. A requirement of a routing approach is that it must be designed to find optimal paths from the source to the sink and at the same time consumes the energy of the nodes along those paths within acceptable levels.

This will enhance the lifetime of the sensor network. In sensor networks, when the data flow is routed along the minimum energy path it results in reduced total energy consumption and hence an increased network lifetime. In taking the shortest path, if the energy metric is also taken into account, then considerable saving in energy can be expected at each node.

3.2 Objectives: To design and develop a routing protocol for the WSN having the following two characteristics:

- Minimizes the battery consumption in WSN
- Maximizing the lifetime of the sensor network

3.3 Assumptions for Simulation:

- The nodes in the network are stationary.
- All nodes have similar capabilities, (processing/communication) and equal significance.
- Each of the nodes is battery operated and the battery is not recharged.
- Nodes are left unattended after deployment.
- Distance between the nodes is same.

3.4 Routing Approach Based on Battery Life: BATLIFE

Let the sensor nodes of density 'n' be deployed uniformly in a sensor field. A sensor network can be modeled by means of a weighted graph G (N, B), with sensor node set N = {N0, N1….Nn}, the source node being denoted by Label BaseStation and the destination node by Label sink node respectively. B denotes the batteries of the nodes.

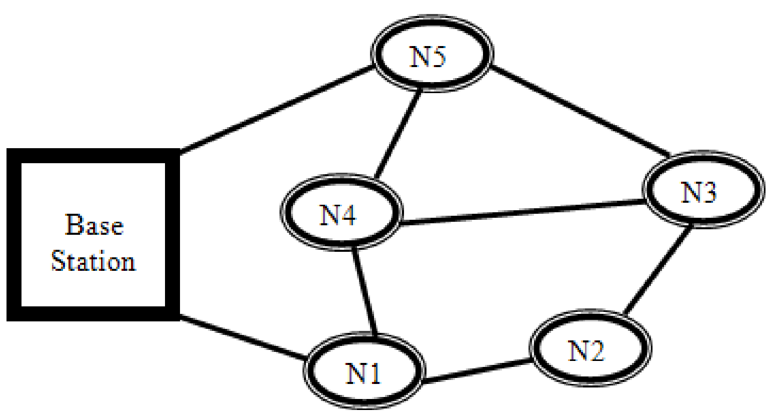


Figure 3.1 Sensor Network of 5 nodes, where N represents Nodes

The solution proposed a routing approach based on the battery life that can be used to find out the best possible route from the base station to the sink node. So it works on the following two phases:

- ***Propagation phase**:* Process starts to find out the neighboring nodes to the base station with the minimum battery life.
- ***Communication Phase**:* Data is sent from source to destination, using the information from the earlier phase. This is when paths are chosen according to the battery life.

3.4.1 Propagation phase: BaseStation initiates the process of finding the neighboring node with the minimum battery life.

- Every intermediate node forwards the request only to the neighbors with the minimum battery life. If Ns is the source and Ni and Nj are the neighboring nodes at a given instant then the next node where is to be send depends on the following equation

$$B\ (Ni) < B\ (Nj), \text{ from the Ns, B (Battery)}$$

- This is continued until destination node is reached.

3.4.2 Communication Phase:

The data communication phase is realized by modifying the breadth first search algorithm to find out the neighboring nodes and selecting the node with the minimum battery. This algorithm helps to find out the available battery life of the neighboring nodes. Node with the minimum battery life available has been selected from the network for the routing purpose and so on till the route reached to the sink node.

During the routing process if the battery of the any node within the route will be exhausted then the route will be changed on the basis of the node with the minimum battery left. For the simulation purpose I have assumed five nodes (node1, node2, node3, node4, node5) serving as the sensor nodes and a **Base Station** serving as the source for the network. Out the five nodes **node3** is serving as the **Sink Node**. It also consists of the routing options available from the Base Station to the Sink Node. As the routing process progresses the route that is being used gets highlighted. A node with the green color indicates the battery available and a Node with the red color indicates that it has been exhausted completely. The simulation and analysis has been shown in the chapter 4.

CHAPTER 4

SIMULATION AND ANALYSIS

The name of the simulator is "BATLIFE" and it stands for "battery life". The simulator BATLIFE provides a routing approach based on the battery life of the sensor nodes. . For the simulation purpose I have assumed five nodes (node1, node2, node3, node4, node5) serving as the sensor nodes and a **Base Station** serving as the source for the network. Out the five nodes **node3** is serving as the **Sink Node**.

It also consists of the routing options available from the Base Station to the Sink Node. As the routing process progresses the route that is being used gets highlighted. A node with the green color indicates the battery available and a Node with the red color indicates that it has been exhausted completely. First figure comprises of the start button that needs to be clicked in order to start the BATLIFE simulator.

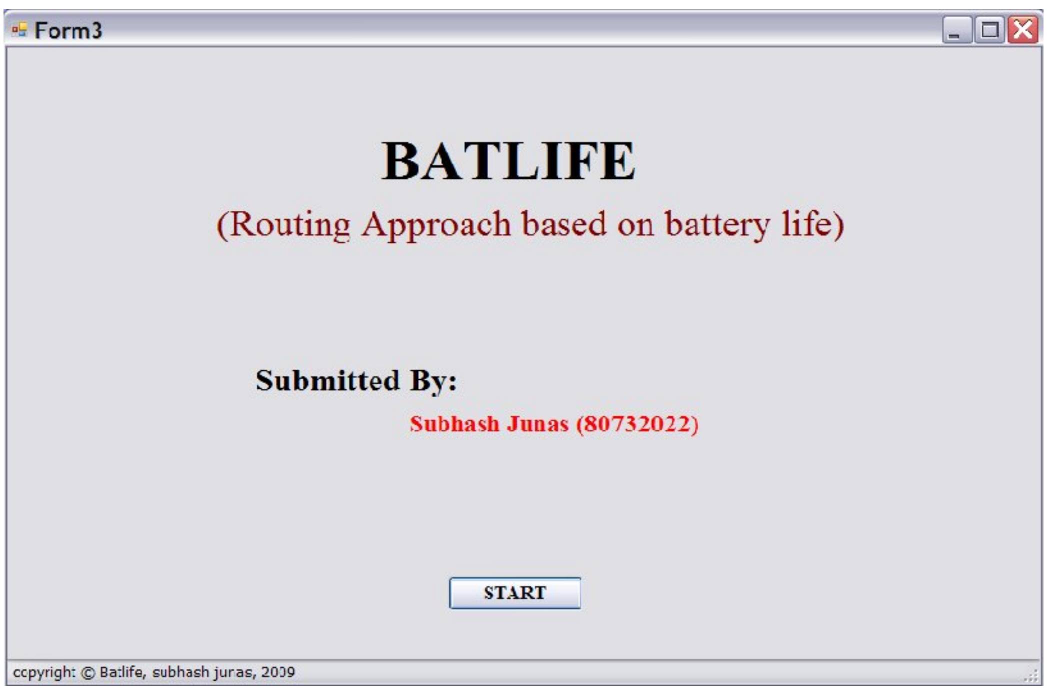


Figure 4.1 BATLIFE step1

After clicking the start button the second figure that appears would ask for the number of processes that to be entered. These much processes would get operated on that network.

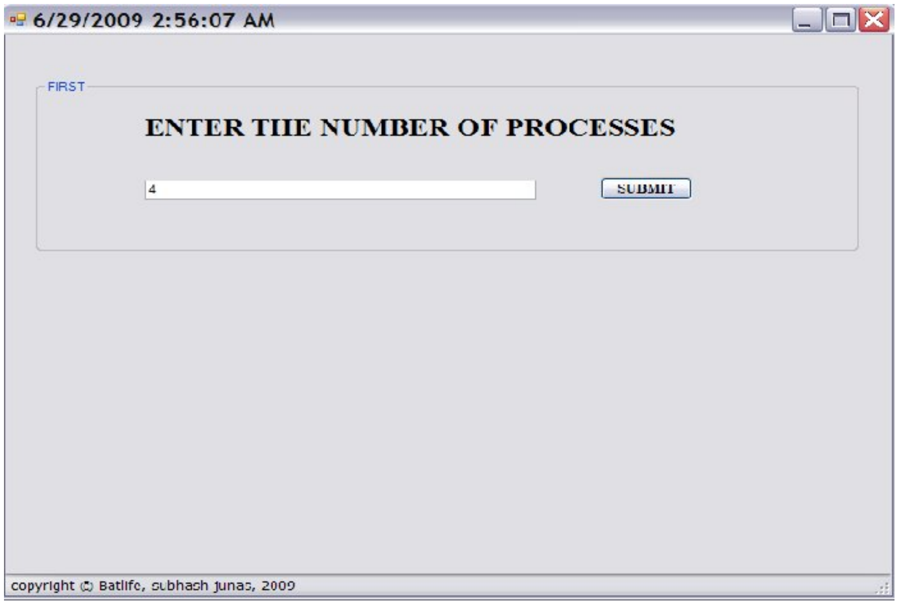


Figure 4.2 BATLIFE step2

On Submitting the number of processes the third figure would show the number of processes entered and would ask to enter the burst rate and the number of nodes. As the BATLIFE works for the five nodes so the number 5 needs to be entered.

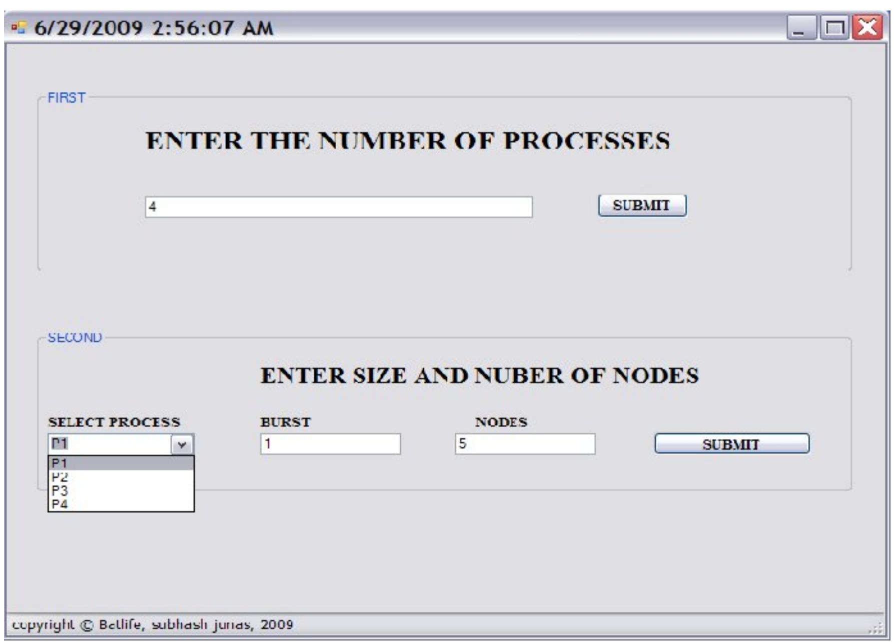


Figure 4.3 BATLIFE step3

The fourth figure shows the number of nodes and burst rate entered. Here it would ask to enter the battery status of the nodes. The battery entered will indicate the battery available to the nodes at the start of the simulation.

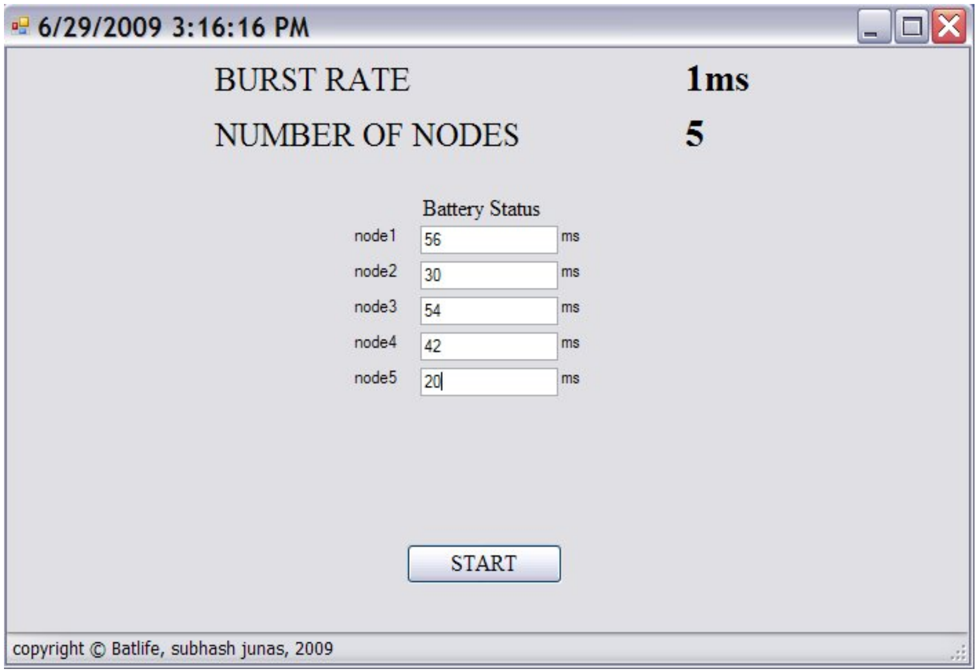


Figure 4.4 BATLIFE step4

On submitting the battery life of the nodes it would appear on the network of the figure five and along with that it will show the number of the options available for the routing purpose. Here label in pink indicate that the node3 is the Sink Node or the destination node and label in Blue indicate that the Base station is the source of the network. Label in Dark green will indicate the route being used for the routing.

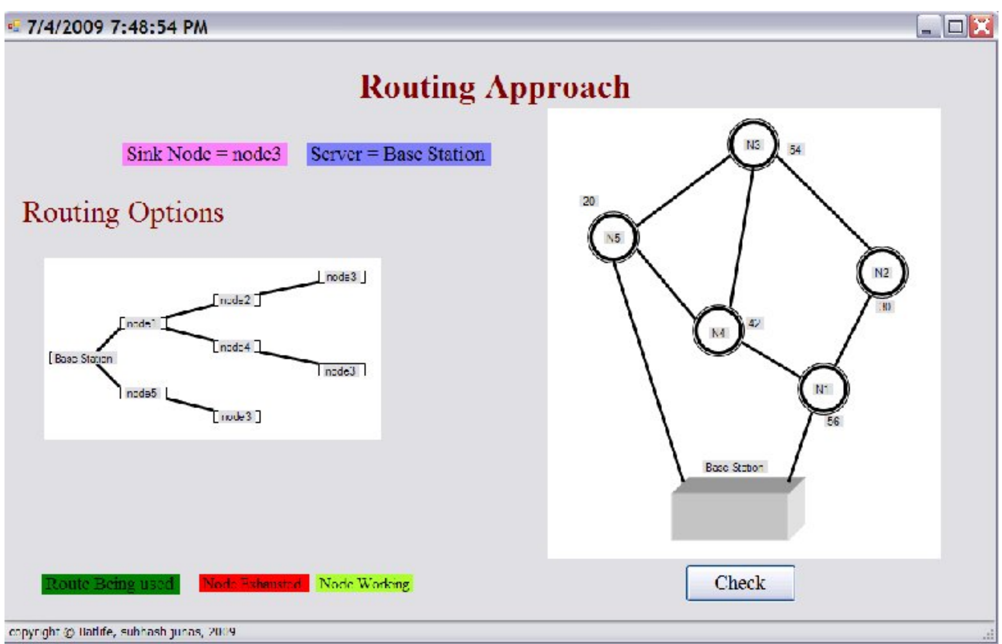


Figure 4.5 BATLIFE step5

On clicking the check button the routing process starts. Out of the available nodes from the base station, node1 and node5 have been selected and compared. As the battery of the node5 is lower than the node1 there fore it has been selected for the start of the routing form the base station which is shown in the figure 6. The route (base station→node5→node3) being used has been shown in the routing options (Dark Green Color).

The decreasing numbers on the nodes indicate the batteries being consumed and the batteries being consumed by the nodes reduced in the exponential way on the basis of the burst applied.

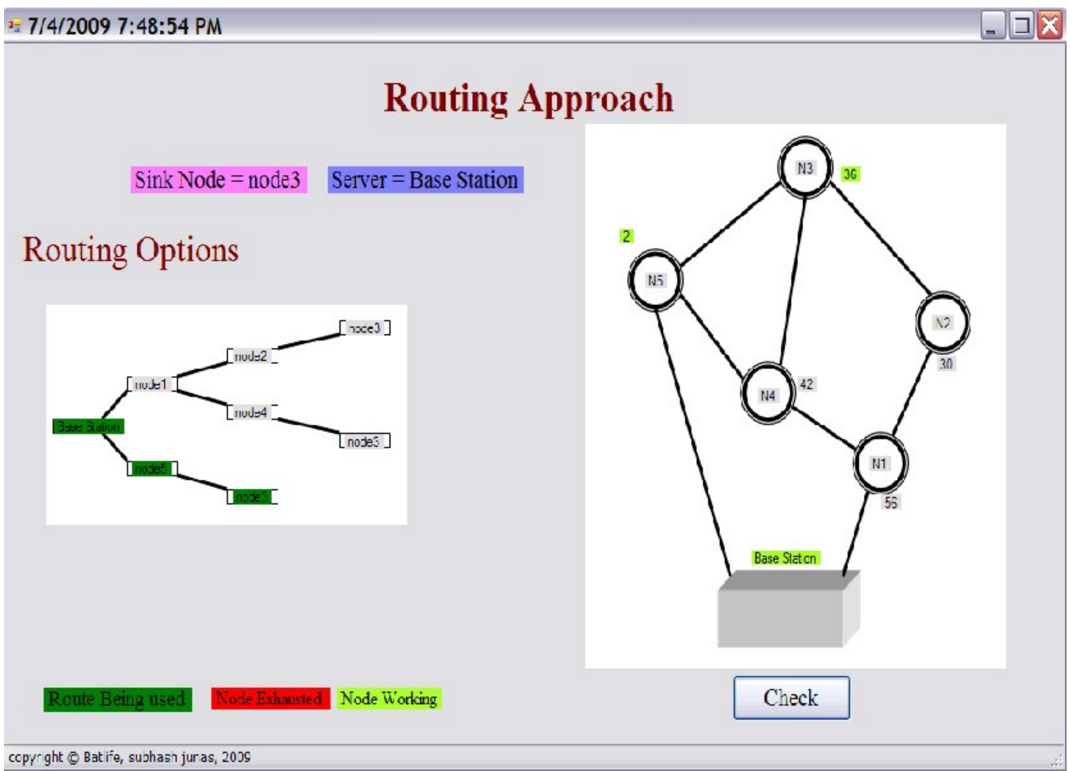


Figure 4.6 BATLIFE step6

The node5 turns into red when it is totally exhausted and along with that the route BaseStation→node5→node3 is disconnected.

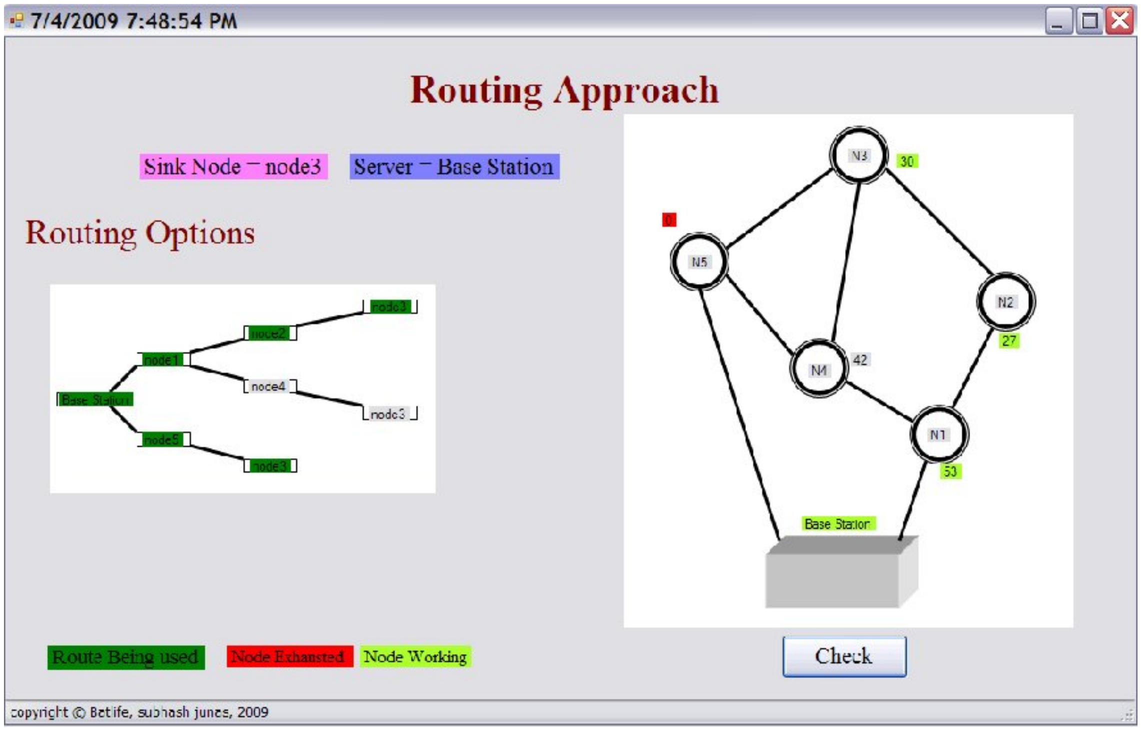


Figure 4.7 BATLIFE step7

The time node5 is exhausted a new route from the base station to the node3 started **(BaseStation→node1→node2→node3)**

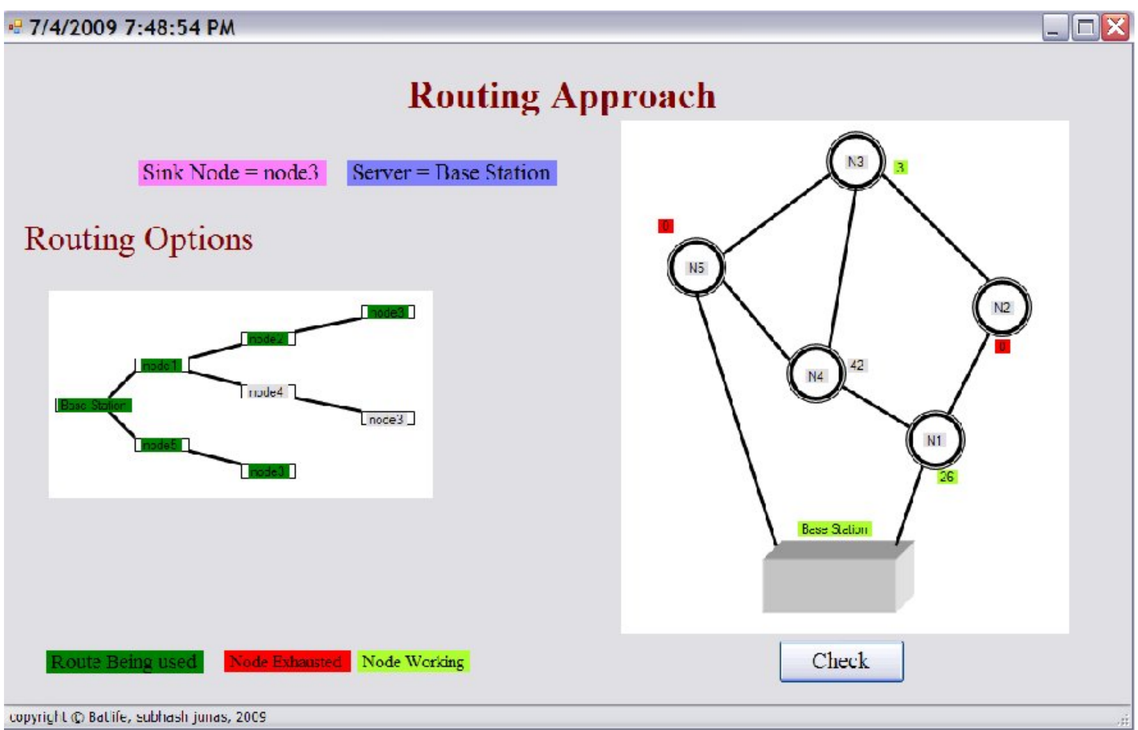


Figure 4.8 BATLIFE step8

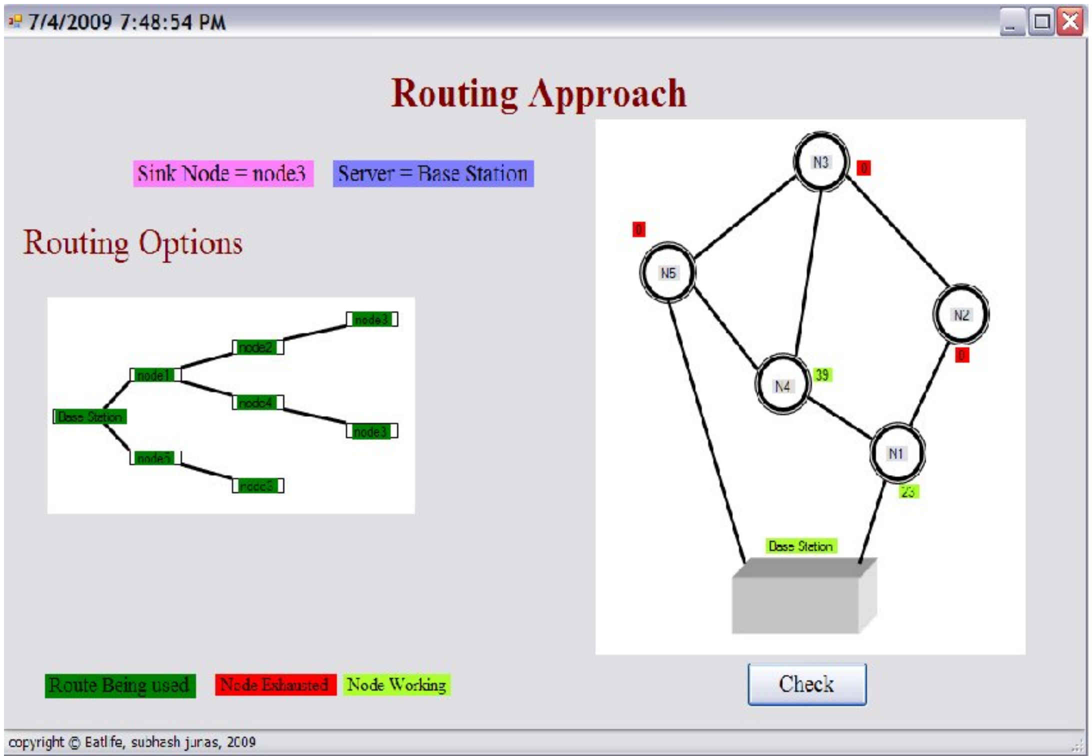


Figure 4.9 BATLIFE step9

After decreasing exponentially when node2 gets totally exhausted it becomes red and a new route BaseStation→node1→node4→node3 to the node3 started from the base station.

The benefit of this routing approach is that it conserves the battery life of the sensor nodes. When the base station is using a particular route then only the nodes within that route will remain active and rest of the nodes will be in the sleeping mode. As this approach selects the nodes with the minimum battery life and in that way the nodes with the higher battery life can be saved for the applications where more battery is needed.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusions

Wireless sensor network has the capabilities to provide data **anytime, anywhere and anyhow,** along with that the wireless has also opened the doors to the other areas, such as MANET, VANET and wireless sensor networks. Although there are many issues regarding wireless sensor networks but now these days the major concern has been given to the routing protocols and efficient utilization of the batteries of the sensor nodes.

Since many routing protocols has been proposed by the researchers but not much work has been reported in the field of Power Aware Routing. In the presented work, a simulator has been designed to cater to the different requirements of WSN helping the numbers of network nodes and the topology fixed. The result confirm with the logical understanding of the route-discovery mechanism in a WSN using the power aware routing. As a result more number of nodes in the WSN can live for a greater amount of time.

5.2 Future Work

The simulation has been carried out in an ‘isolated mode’ and it would be interesting to simulate the protocol on other factor for e.g. Route-discovery time, route-maintenance time etc. further the proposed work would be of much help if it can be integrated with TOSSIM.

REFERENCES

[1] Overview of Sensor Networks, D. Culler, D. Estrin and M. Srivastava, Computer, Vol. 37, Issue No. 8, August 2004, pp. 41 – 49, http://csdl2.computer.org/comp/mags/co/2004/08/r8041.pdf

[2] Sensor Network Operation, http://www.intel.com/research/exploratory/sensornetwork_operation.htm

[3] Intel Motes and Wireless Sensor Networks, http://www.intel.com/research/downloads/SNOverviewCD.pdf

[4] Sensor Nets/ RFID, http://www.intel.com/research/exploratory/instrument_world.htm

[5] On Random Event Detection with Wireless Sensor Networks, P. K. Dutta, Masters Thesis, Ohio State University, 2004, http://www.cs.berkeley.edu/~prabal/pubs/masters/dutta04masters.pdf

[6] Crossbow Technology Inc. Data Sheet, http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf

[7] University of California at Berkeley's website on Tiny Operating System. http://today.cs.berkeley.edu/tos/

[8] Proc. First ACM Conference on Embedded Networked Sensor Systems, Los Angeles, CA, November 2003.

[9] Proc. First ACM International Workshop on Sensor Networks and Applications, Atlanta, GA, September 2002.

[10] http://www.intel.com/labs/features/rs01031.htm, website of Intel Corporation, 2200 Mission College Blvd.Santa Clara, CA 95052.

[11] D. Ganesan, R. Govindan, S. Shenker and D. Estrin, "Highly resilient, energy efficient multipath routing in wireless sensor networks," *Mobile Computing and Communications Review*, vol. 1, no. 2, pp. 10-24, 2002

[12] D. Gay, P. Levis, R. von Behren, M. Welsh, E. Brewer, and D. Culler. The nesC Language: A Holistic Approach to Networked Embedded Systems. In *Proceedings of Programming Language Design and Implementation (PLDI)*, June 2003.

[13] P.Levis, N. Lee, M. Welsh, and D. Culler. TOSSIM: Accurate and Scalable Simulation of EntireTinyOS Applications To Appear in Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003).

[14] E. Royer and C.-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *IEEE Pers. Commun.*, vol. 6, no. 2, Apr. 1999, pp. 46–55.

[15] S.-J. Lee, M. Gerla, and C.-K. Toh, "A Simulation Study of Table-Driven and On-Demand Routing Protocols for Mobile Ad Hoc Networks," *IEEE Network*, July 1999.

[16] C.-K. Toh, "Associativity Based Routing for Ad Hoc Mobile Networks," *Wireless Pers. Commun. J.*, Special Issue on Mobile Networking and Computing Systems, vol. 4, no. 2, Mar. 1997.

[17] P. Latteieri, C. Schurgers, and M. B. Srivastava, "Adaptive Link Layer Strategies for Energy Efficient Wireless Networking," *ACM WINET*.

[18] A. Alwan *et al.*, "Adaptive Mobile Multimedia Networks," *IEEE Pers. Commun.*, Apr. 1996.

[19] S. Singh and C. S. Raghavendra, "PAMAS-Power Aware Multi-Access protocol with Signaling for Ad Hoc Networks," *ACM Commun. Rev.*, July 1998.

[20] K. Scott and N. Bambos, "Routing and Channel Assignment for Low Power Transmission in PCS," *Proc. ICUPC '96*, vol. 2, pp. 498-502, Oct. 1996.

[21] S. Singh, M. Woo, and C. S. Raghavendra, "Power- Aware Routing in Mobile Ad Hoc Networks,"*Proc. MobiCom '98*, Dallas, TX, Oct. 1998.

[22] C. Lu, B. Blum, T. Abdelzaher, J. Stankovic, and T. He, RAP: A Real-Time Communication Architecture for Large-Scale Wireless Sensor Networks, RTAS, June 2002.

[23] A. Wood and J. Stankovic, Denial of Service in Sensor Networks, IEEE Computer, Vol. 35, No. 10, October 2002, pp. 54-62.

PAPER PUBLISHED /COMMUNICATED

[1] Subhash Junas, Dr. Anil Kumar Verma, “Routing In Wireless Sensor Network: A Review”, in International Conference on Wireless Networks & Embedded System (WECON-2008) October 18-19, 2009 (Status: Accepted)