MASTER OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING

AN ENERGY EFFICIENT CROSS-LAYER APPROACH FOR WIRELESS SENSOR NETWORKS

By

Md. Imran Hossain Jony

Department of Electrical and Electronic Engineering

Islamic University of Technology (IUT)

Organization of Islamic Cooperation (OIC)

Board Bazar, Gazipur-1704, Bangladesh

October-2015

DECLARATION OF CANDIDATE

I hereby declare that the work reported in the M.Sc. thesis entitled "**AN ENERGY EFFICIENT CROSS-LAYER APPROACH FOR WIRELESS SENSOR NETWORKS**" submitted at **Islamic University of Technology (IUT),** Board Bazar, Gazipur, Bangladesh, is an authentic record of my work carried out under the supervision of **Prof. Dr. Mohammad Rakibul Islam**. I have not submitted this work elsewhere for any other degree or diploma. I am fully responsible for the contents of my M.Sc. thesis.

Md. Imran Hossain Jony
Student Number: 122604
Academic Year: 2012-2013

CERTIFICATE OF APPROVAL

The thesis titled "**An Energy Efficient Cross-Layer Approach for Wireless Sensor Networks**" submitted by **Md. Imran Hossain Jony** bearing Student No. 122604 of Academic Year 2012-2013 has been found as satisfactory and accepted as partial fulfilment of the requirement for the degree of Master of Science in Electrical and Electronic Engineering on October 29, 2015.

BOARD OF EXAMINERS

1. **Dr. Mohammad Rakibul Islam** — Chairman (Supervisor)
   Professor
   Department of Electrical & Electronic Engineering
   IUT, Gazipur-1704, Bangladesh.

2. **Dr. Md. Shahid Ullah** — Member (Ex-Officio)
   Professor and Head
   Department of Electrical & Electronic Engineering
   IUT, Gazipur-1704, Bangladesh.

3. **Dr. Md. Ashraful Hoque** — Member
   Professor
   Department of Electrical & Electronic Engineering
   IUT, Gazipur-1704, Bangladesh.

4. **Dr. Syed Iftekhar Ali** — Member
   Associate Professor
   Department of Electrical & Electronic Engineering
   IUT, Gazipur-1704, Bangladesh.

5. **Dr. Mohammed Imamul Hassan Bhuiyan** — Member (External)
   Professor
   Department of Electrical & Electronic Engineering
   BUET, Dhaka-1000, Bangladesh.

TO MY PARENTS & TEACHERS

TABLE OF CONTENTS

CHAPTER 5 INTERFACING WITH NETWORK SIMULATOR-2 **40**

LIST OF ACRONYMS AND ABBREVIATIONS

Abbreviation	Definition
ACK	Acknowledgement
AODV	Ad-hoc on-Demand Distance Vector
ARP	Address Resolution Protocol
ARQ	Automatic Repeat reQuest
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
CENS	Centre for Embedded Network Sensing
CSMA	Carrier Sense Multiple Access
CTS	Clear To Send
DCF	Distributed Coordination Function
DSDV	Destination-Sequenced Distance-Vector Routing
DSR	Dynamic Source Routing
EOA	Energy Optimization Algorithm
ESRT	Event-to-Sink Reliable Transport
ETSI	European Telecommunications Standards Institute
FEC	Forward Error Correction
FDM	Frequency Division Multiplexing
GSM	Global System for Mobile Communications
IEEE	Institution of Electrical and Electronics Engineers
IMEP	Internet MANET Encapsulation Protocol
IP	Internet Protocol
ISM	Industrial, Scientific and Medical
IEEE	Institution of Electrical and Electronics Engineers
ISI	Inter Symbol Interference
ISM	Industrial Scientific and Medical
LEACH	Low-Energy Adaptive Clustering Hierarchy
MAC	Medium Access Control

MIMO	Multiple Input Multiple Output
MANET	Mobile Ad-hoc Network
MISO	Multiple Input Single Output
MTU	Maximum Transmission Unit
NACK	No Acknowledgement
NS	Network Simulator
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiplexing Access
PRR	Packet Reception Rate
PHY	Physical
PTS	Partial Transmit Sequence
PUMA	Protocol for Unified Multicasting through Announcements
QoS	Quality of Service
RREP	Route Reply Packet
RREQ	Route Recovery
RRER	Route Error
RTS	Request To Send
SCADA	Supervisory Control and Data Acquisition
SIMO	Single Input Multiple Output
SISO	Single Input Single Output
SMAC	Sensor MAC
SMP	Sensor Management Protocol
SQDDP	Sensor Query and Data Dissemination Protocol
SQTL	Sensor Query and Tasking Language
STBC	Space-Time Bock Code-encoded
SNR	Signal to Noise Ratio
TCL	Tool Command Language
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TORA	Temporarily Ordered Routing Algorithm

UDP	User Datagram Protocol
WSN	Wireless Sensor Network
WLAN	Wireless Local Area Network
XLP	Cross-Layer Protocol

LIST OF SYMBOLS

Symbol	Definition
PL	Path Loss
$\mathbf{E_{rx}}$	Receive Energy
$\mathbf{E_{tx}}$	Transmit Energy
$\mathbf{E_{sleep}}$	Sleep Energy
$\mathbf{T_{transition}}$	Transition Time

LIST OF FIGURES

LIST OF TABLE

ACKNOWLEDGEMENTS

First of all, I would like to express my deep and sincere gratitude to my supervisor Dr. Mohammad Rakibul Islam, Professor in Electrical and Electronic Engineering Department of Islamic University of Technology (IUT), Dhaka, Bangladesh for his kind guidance, support and encouragement. His regular encouragement and supervision on this thesis really made me to complete this work within this very short time. His valuable advice and constructive criticism guided me to learn and enjoy the art of research. His ability to quickly understand the depth of the problem and suggestions has always inspired me. In every sense, this thesis could never have come into existence without his help. Learning from him has been a very fruitful and enjoyable experience. I shall always remain grateful to him.

I would like to express my deepest cordial thanks to my parents for their support and love.

I am also grateful to the Head of the department Prof. Dr. Md. Shahid Ullah for his inspirations throughout my career and encouragement to complete the work. My heartfelt acknowledgement goes to Prof. Dr. Md. Ashraful Hoque, Dr. Syed Iftekhar Ali and Prof. Dr. Mohammed Imamul Hassan Bhuiyan for being members of my committee.

Last but not the least I am thankful to Allah, who gave me enough patience, strength and courage to complete this work. I am highly grateful to the Almighty.

ABSTRACT

Energy minimization has become a burning issue for Wireless Sensor Networks (WSNs) which are mainly event based systems and rely on the collective effort of several micro-sensor nodes continuously observing a physical phenomenon. Energy efficient approaches or tools are the key to prolong the lifetime of the sensor nodes. This research presents a cross-layer approach between the medium access control (MAC) and the network layer to achieve energy efficiency. To get a bigger insight, number of sensor nodes is increased and performance of the whole network is evaluated. Ad-hoc On-Demand Distance Vector (AODV) is used in this work as routing protocol in the network layer along with IEEE-802.11 protocol in the MAC layer. Simulation results (Figure 4.8 and Figure 4.9) show that the cross-layer approach obtains significant energy savings compared with Sensor MAC (S-MAC) protocol and traditional approach. It also shows that the performance of IEEE-802.11 MAC can be made more efficient by implementing the proposed cross-layer approach.

CHAPTER 1

INTRODUCTION

Energy efficient communication is desirable in many recent wireless sensor applications. Traditional layered approaches can achieve only limited energy efficiency due to the shorter battery lifetime which is a great obstacle to efficient communication. As sensor nodes are battery operated, therefore, increasing the battery lifetime is the prime concern for energy efficient communication. Cross layer approach is a potential candidate to fulfil the requirements to reduce the energy consumption of sensor nodes and increase the battery lifetime.

1.1 Background of WSNs

Wireless sensor networks (WSNs) are getting popular day-by-day for applicability, reliability and flexibility to implement in any environment. These are mainly event-based systems that exploit the collective effort of densely deployed micro-sensor nodes which continuously observe certain physical phenomenon. The main objective of any WSN application is to detect event features from the collective information provided by sensor nodes. In a typical sensor network, information collected by multiple local sensors need to be transmitted to a remote central processor. If the remote processor is far away from the source node then the information will first be transmitted to a relay node and then multi hop-based routing will be used to forward the data to its final destination. The main challenge for achieving this objective is mainly posed by the severe energy and processing constraints of low-end wireless sensor nodes as those are battery operated. Therefore, conserving the energy of those sensor nodes has become the most popular research area for the developers to enhance the lifetime of the nodes.

The majority of wireless sensor network deployments will fall into one of the three class templates which are: environmental data collection, security monitoring, and sensor node tracking [1]. One of the major fields of application of sensor nodes is military including battlefield surveillance and monitoring, detection of attack by weapons of mass destruction, such as chemical, biological, or nuclear and guidance systems of intelligent missiles. Sensors can also be used in environmental applications such as fire and flood detection, habitat exploration of animals. Sensors can be extremely useful in diagnosis of diseases and monitoring the patients [2], [3], [4]. Small sensor devices can be used in the patient's body to monitor their physiological data such as blood pressure or heart rate. These data collected from the sensors can be used from a SCADA (Supervisory Control and Data Acquisition) system to alert the concerned doctor on detection of an anomaly. Such systems provide patients a greater freedom of movement instead of their being confined to a hospital. Therefore, the patents need not to be confined themselves to the boundary of the hospitals, rather they can move anywhere with freedom and relief.

Sensors are also getting popular in commercial applications at home and in industries. Smart sensor nodes are used inside the air conditioners to sense the temperature of a room and then reacted according to the signal sent by remote controllers handled by the user. Sensors can be placed inside several home appliances like ovens, refrigerators, and vacuum cleaners to interact with each other and be remote-controlled. A "smart environment" can be created inside the home which adapts itself according to the user's needs or tastes. For instance, almost everything in the room such as, the lighting, music, and ambiance can be automatically set or controlled according to the user's preferences.

Similar type of control can also be useful in office buildings too. The airflow and temperature of different parts of the building can be automatically controlled using sensor nodes. Sensor networks can be deployed to monitor temperature in a large region [5], [6], [7]. For example, the Centre for Embedded Network Sensing (CENS) [5] is focused on environmental and habitat monitoring. The sensed data is used to study the vegetation response to climatic trends. The inventory control system of warehouses could be improved by installing sensors on the products to track their movement.

Bio-sensors can be deployed along the national borders to detect the smuggling of bio weapons by terrorists. Networks of video, acoustic, and other sensors can be used to track suspected targets. Some examples can be found in [8], [9], [10].

Image sensors and other types of sensors have been used at road way intersections to monitor traffic conditions [11], [12]. In the future, the information collected by the image sensors will be automatically processed by a network centre, which will perform traffic control functions related to signalling and responding to accidents. A more advanced concept proposes to attach sensors on the body of vehicles [13]. When two vehicles pass each other, they can exchange some critical information such as traffic jam locations.

Therefore it can be summarized that the applications of wireless sensor networks are endless, limited only by the human imagination. Considering all of these huge applications and usefulness of wireless sensor nodes, it has become a serious issue to conserve the battery power of the sensors to prolong their lifetime.

Despite the widespread acceptance of WSN, it has some drawbacks:

- Sensor nodes are battery operated; therefore, lifetime of those nodes reduces along with the consecutive use for communication.
- Implementing a new protocol is very complicated and costly as the layers are compactly integrated.

1.2 Motivation

Increasing the energy efficiency of wireless sensor networks is one of the greatest challenges faced by wireless communication engineers. The available energy of the battery is scarce and costly, whereas, there is a huge demand for data communication created by increasing number of sensor nodes used in several applications. Due to the limited energy and difficulty to recharge a large number of sensors, energy efficiency and maximizing network lifetime have been the most important design goals for WSNs. However, channel fading, interference, and radio irregularity pose big challenges on the design of energy efficient communication and routing protocols in the multi-hop WSNs.

As the MIMO technology has the potential to dramatically increase the channel capacity and reduce transmission energy consumption in fading channels [14], cooperative MIMO schemes have been proposed for WSNs to improve communication performance [15]-[18]. In those schemes, multiple individual single-antenna nodes cooperate on information transmission and/or reception for energy-efficient communications.

Cui et al. [15] analysed a cooperative MIMO scheme with Alamouti code for single-hop transmissions in WSNs. They considered radio applications in sensor networks, where the nodes were operated on batteries so that energy consumption might be minimized, while satisfying given throughput and delay requirements. In this context, they analysed the best modulation and transmission strategy to minimize the total energy consumption required to send a given number of bits. The total energy consumption includes both the transmission energy and the circuit energy consumption. They first considered multi-input-multi-output (MIMO) systems based on Alamouti diversity schemes, which has good spectral efficiency and also more circuitry that consumes energy. They then extended their energy-efficiency analysis of MIMO systems to individual single-antenna nodes that cooperate to form multiple-antenna transmitters or receivers. By transmitting and/or receiving information jointly, they showed that tremendous energy saving is possible for transmission distances larger than a given threshold, even when they took into account the local energy cost necessary for joint information transmission and reception. They also showed that over some distance ranges, cooperative MIMO transmission and reception can simultaneously achieve both energy savings and delay reduction.

Li [16] proposed a delay and channel estimation scheme without transmission synchronization for decoding for such cooperative MIMO schemes. He proposed a new transmission scheme which uses two transmitting sensors and space-time block codes to provide transmission diversity in distributed wireless sensor networks with neither antenna-arrays nor transmission synchronisation is proposed. Full diversity and full rate were achieved which enhanced power/bandwidth efficiency and reliability. Simulations demonstrated its superior performance in saving transmission energy.

Li et al. [17] proposed a space-time block code-encoded (STBC) encoded cooperative transmission scheme for WSNs without perfect synchronization. The efficiency of STBC cooperative transmission was studied within low-energy adaptive clustering hierarchy (LEACH), which was a typical networking or communication protocol for wireless sensor networks. Cooperation protocol with low overhead or operating cost was proposed by the authors and synchronization requirements among cooperating sensors were discussed. Energy efficiency was analysed as a trade-off between the reduced transmission energy consumption and the increased electronic and overhead energy consumption. Simulations showed that with proper design, cooperative transmission could enhance energy efficiency and prolong sensor network lifetime.

Jayaweera [18] considered the training overhead of such schemes. Energy efficiency of MIMO techniques in wireless sensor networks was analysed by him. Assuming a cooperative sensor network, the energy consumption of MIMO-based wireless sensor networks was compared with conventional SISO sensor networks. The dependence of energy efficiency on coherence time of the fading process and communications distance was considered in his research. His results showed the applicability of MIMO techniques in sensor networks with judicious system design.

For multi-hop communication a novel cluster based scheme was also presented [19]. A cluster-based cooperative multiple-input-multiple-output (MIMO) scheme was proposed to reduce the adverse impacts caused by radio irregularity and fading in multi-hop wireless sensor networks. This scheme extended the LEACH protocol to enable the multi-hop transmissions among clusters by incorporating a cooperative MIMO scheme into hop-by-hop transmissions. Through the adaptive selection of cooperative nodes and the coordination between multi-hop routing and cooperative MIMO transmissions, the scheme could gain effective performance improvement in terms of energy efficiency and reliability. Based on the energy consumption model developed in the paper, the optimal parameters to minimize the overall energy consumption were found, such as the number of clusters and the number of cooperative nodes. Simulation results exhibited that the proposed scheme could effectively save energy and prolong the network lifetime.

These all were associated to reduce energy consumption by selecting the nodes with better specifications. Any of those methods did not try to work out with the inside layers of sensor nodes. Then the concept of cross layer with different protocols comes up to mitigate the demand for consuming less energy for the sensor nodes.

1.3 Problem Definition

Wireless broadband access networks allow people and devices to have high speed connections to the backbone network from any place at any time. Low power wireless network design is a key issue for energy constrained wireless communication. Wireless sensor network is an area where the energy efficient design is mandatory.

There are lots of cross layer implementations (discussed in chapter-3) done by several researchers where a specific protocol is used to cross two different layers to reduce the energy consumption of sensor nodes. Those are very complicated and costly to implement. Our focus is on developing a cross-layer approach which is not only energy efficient but also easy to implement in practice. This research proposes to pass any specific information from a layer to another by strengthening the interfacing between those two layers.

Interfacing between MAC layer and network layer is the main issue to develop our cross layer approach. AODV routing protocol creates a routing table inside the network layer. This routing table consists of several parameters like source and destination addresses, hop count etc. Combining this with the MAC layer is the main issue for implementation of our cross layer approach.

IEEE 802.11 MAC and AODV routing protocol inside network layer can be a good candidate for implementing such a concept as because AODV utilizes routing table which can make MAC more efficient if the table can be accessed from the MAC layer itself. This is the main challenge for this cross layer approach to be successfully implemented.

1.4 Goal and Scope of the Thesis

The goals met by the thesis are listed as follows:

- Investigation of cross layer approach for WSN and development of an energy efficient communication scheme using cross layer approach.
- Improving the MAC layer functionalities of older version (IEEE 802.11).
- Performance evaluation and comparison of our cross layer approach with the traditional layered approach using software simulation.

1.5 Thesis Layout

This dissertation is organized into six chapters.

- **Chapter 1** represents the background of the present work, motivation and objectives, contributions of this thesis and related work with this thesis.

- **Chapter 2** presents the technical background and introduction to cross layer. It deals with the fundamental description of cross layer approach and explains the importance of it to reduce energy consumption of sensor nodes.

- **Chapter 3** consists of several parts consisting of the definition of cross-layer approach, historical background of cross-layer approach and the algorithm of our cross-layer approach.

- **Chapter 4** presents with the system model and concept of our cross-layer approach is discussed. Analysis and comparison are presented here with the simulation results.

- **Chapter 5** presents a comprehensive study on ns2 for creating a wireless sensor network scenario for our proposed approach.

- **Chapter 6** summarizes the conclusion of each of the contributing chapters and list possible directions for future work.

CHAPTER 2

ARCHITECTURE OF WIRELESS SENSOR NETWORKS

The demand for wireless sensor nodes and sensor network is increasing day-by-day. New applications are emerging, not just in the wired systems, but also in the wireless mobile systems. In order to achieve something significant, it is important to study the network architecture for WSNs.

2.1 Network Architecture

A sensor network typically consists of a large number of sensor nodes densely deployed in a region of interest, and one or more data sinks or base stations that are located close to or inside the sensing region, as shown in Figure 2.1. The sink(s) sends queries or commands to the sensor nodes in the sensing region while the sensor nodes collaborate to accomplish the sensing task and send the sensed data to the sink(s). Meanwhile, the sink(s) also serves as a gateway to outside networks, for example, the Internet. It collects data from the sensor nodes, performs simple processing on the collected data, and then sends relevant information (or the processed data) via the Internet to the users who requested it or use the information.

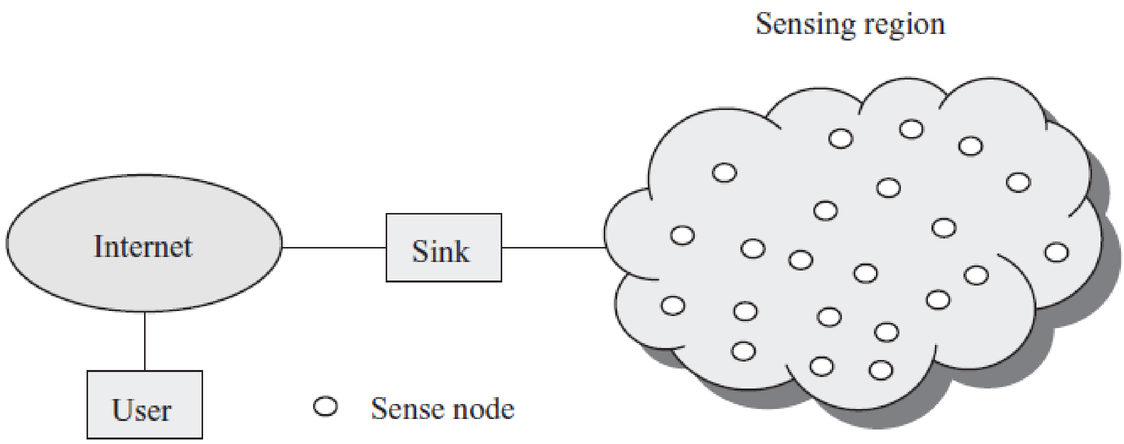

Figure 2.1: Network Architecture

To send data to the sink, each sensor node can use single-hop long-distance transmission, which leads to the single-hop network architecture, as shown in Figure 2.2. However, long-distance transmission is costly in terms of energy consumption. In sensor networks, the energy consumed for communication is much higher than that for sensing and computation.

For example, the energy consumed for transferring one bit of data to a receiver at 100 m away is equal to that needed to execute 3,000 instructions [20]. The ratio of energy consumption for communicating 1 bit over the wireless medium to that for processing the same bit could be in the range of 1,000-10,000 [21, 22]. Furthermore, the energy consumed for transmission dominates the total energy consumed for communication and the required transmission power grows exponentially with the increase of transmission distance. Therefore, it is desired to reduce the amount of traffic and transmission distance in order to increase energy savings and prolong network lifetime.

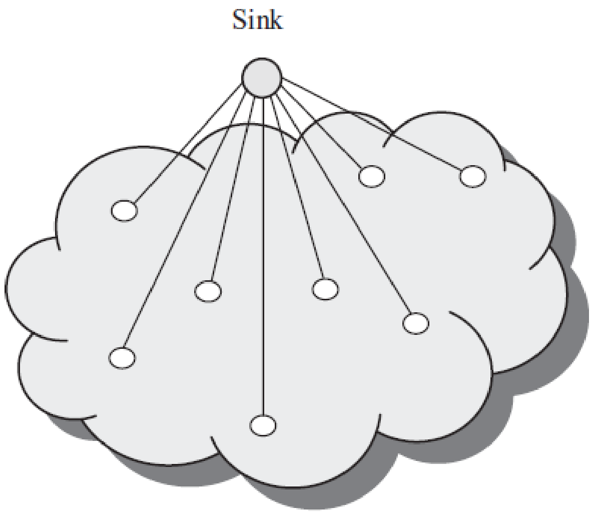

Figure 2.2: Single-hop Network Architecture

For this purpose, multi-hop short-distance communication is highly preferred. In most sensor networks, sensor nodes are densely deployed and neighbor nodes are close to each other, which makes it feasible to use short-distance communication. In multi-hop communication, a sensor node transmits its sensed data toward the sink via one or more intermediate nodes, which can reduce the energy consumption for communication. The architecture of a multi-hop network can be organized into two types: flat and hierarchical [23], which are described in the next two sections.

2.1.1 Flat Architecture

In a flat network, each node plays the same role in performing a sensing task and all sensor nodes are peers. Due to the large number of sensor nodes, it is not feasible to assign a global identifier to each node in a sensor network. For this reason, data gathering is usually accomplished by using data-centric routing, where the data sink transmits a query to all nodes in the sensing region via flooding and only the sensor nodes that have the data matching the query will respond to the sink. Each sensor node communicates with the sink via a multi-hop path and uses its peer nodes as relays. Figure 2.3 illustrates the typical architecture of a flat network.

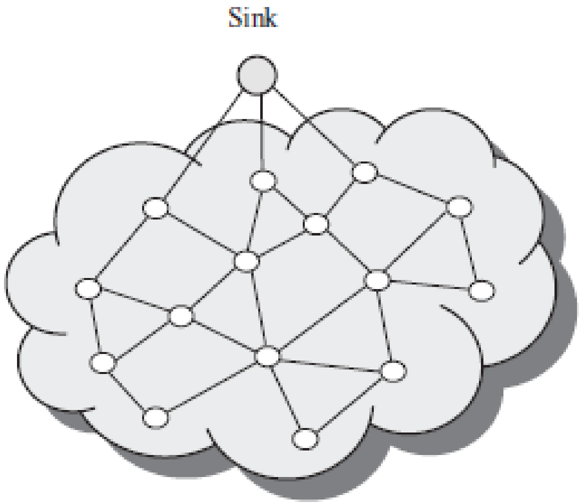

Figure 2.3: Flat Network Architecture

2.1.2 Hierarchical Architecture

In a hierarchical network, sensor nodes are organized into clusters, where the cluster members send their data to the cluster heads while the cluster heads serve as relays for transmitting the data to the sink. A node with lower energy can be used to perform the sensing task and send the sensed data to its cluster head at short distance, while a node with higher energy can be selected as a cluster head to process the data from its cluster members and transmit the processed data to the sink. This process can not only reduce the energy consumption for communication, but also balance traffic load and improve scalability when the network size grows. Since all sensor nodes have the same transmission capability, clustering must be periodically performed in order to balance the traffic load among all sensor nodes. Moreover, data aggregation can be performed at cluster heads to reduce the amount of data transmitted to the sink and improve the energy efficiency of the network [24]. The major problem with clustering is how to select the cluster heads and how to organize the clusters [25]. In this context, there are many clustering strategies. According to the distance between the cluster members and their cluster heads, a sensor network can be organized into a single-hop clustering architecture or a multi-hop clustering architecture, as shown in Figs. 2.4 and 2.5, respectively [26].

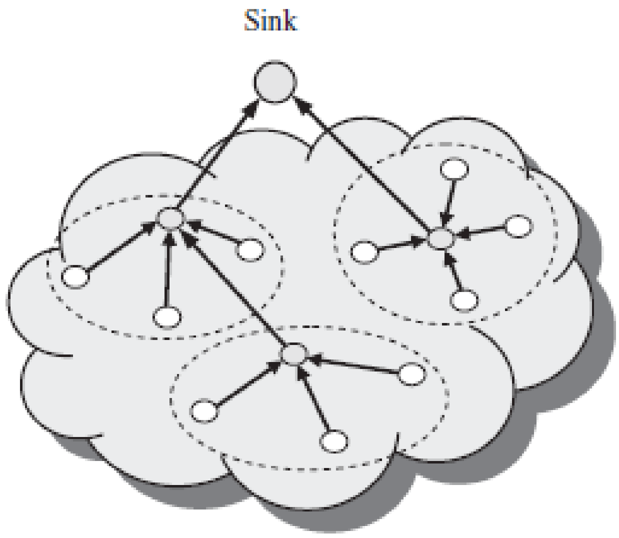

Figure 2.4: Single-hop Clustering Architecture

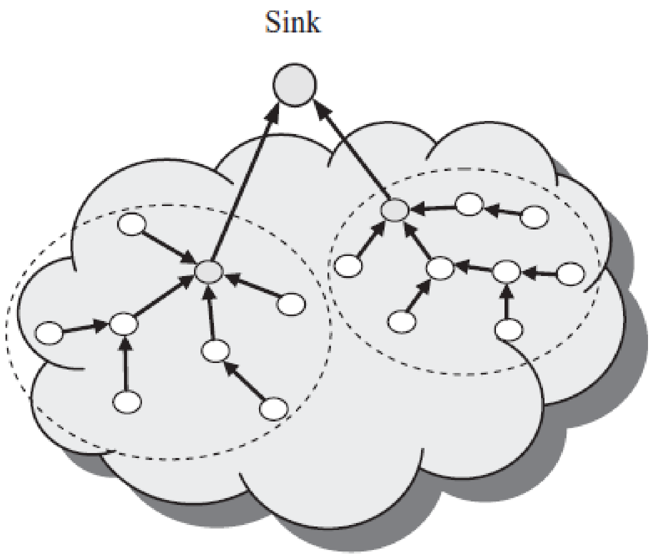

Figure 2.5: Multi-hop Clustering Architecture

According to the number of tiers in the clustering hierarchy, a sensor network can be organized into a single-tier clustering architecture or a multitier clustering architecture. Figure 2.6 illustrates an example of the multitier clustering architecture. To address the clustering problem, a variety of clustering algorithms have been proposed in the literature [26]-[31].

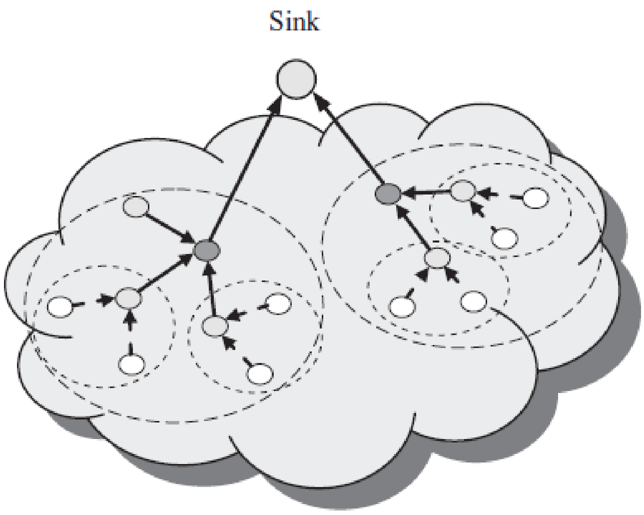

Figure 2.6: Multi-tier Clustering Architecture

2.2 Classification of WSNs

WSNs are application specific. A sensor network is usually deployed for a specific application and thus has some different characteristics. According to different criteria, WSNs can be classified into different categories [32].

2.2.1 Static and Mobile Network

According to the mobility of sensor nodes, a sensor network can be static or mobile. In a static sensor network, all sensor nodes are static without movement, which is the case for many applications. However, some sensor applications require mobile nodes to accomplish a sensing task. A wireless biosensor network using autonomously controlled animals is a typical example of mobile sensor networks [33]. Compared with static sensor networks, which is simpler to control and easier to implement, the design of mobile sensor networks must consider the mobility effect, which increases the complexity of implementation.

2.2.2 Deterministic and Nondeterministic Network

According to the deployment of sensor nodes, a sensor network can be deterministic or nondeterministic. In a deterministic sensor network, the positions of sensor nodes are pre-planned and are fixed once deployed. This type of network can only be used in some limited situations, where the pre-planned deployment is possible. In most situations, however, it is difficult to deploy sensor nodes in a pre-planned manner because of the harsh or hostile environments. Instead, sensor nodes are randomly deployed without preplanning and engineering. Obviously, nondeterministic networks are more scalable and flexible, but require higher control complexity.

2.2.3 Static-Sink and Mobile-Sink Network

A data sink in a sensor network can be static or mobile. In a static- sink network, the sink(s) is static with a fixed position located close to or inside a sensing region. All sensor nodes send their sensed data to the sink(s). Obviously, a static sink makes the network simpler to control, but it would cause the hotspot effect [23]. The amount of traffic that sensor nodes are required to forward increases dramatically as the distance to the data sink becomes smaller. As a result, sensor nodes closest to the data sink tend to die early, thus resulting in network partition and even disrupting normal network operation. In a mobile-sink network, the sink(s) moves around in the sensing region to collect data from sensor nodes, which can balance the traffic load of sensor nodes and alleviate the hotspot effect in the network.

2.2.4 Single-Sink and Multi-sink Network

A sensor network can have a single sink or multiple sinks. In a single-sink network, there is only one sink located close to or inside the sensing region. All sensor nodes send their sensed data to this sink. In a multi-sink network, there may be several sinks located in different positions close to or inside the sensing region. Sensor nodes can send their data to the closest sink, which can effectively balance the traffic load of sensor nodes and alleviate the hotspot effect in the network.

2.2.5 Single-hop and Multi-hop Network

According to the number of hops between a sensor node and the data sink, a sensor network can be classified into single-hop or multi-hop. In a single-hop network, all sensor nodes transmit their sensed data directly to the sink, which makes network control simpler to implement. However, this requires long-range wireless communication, which is costly in terms of both energy consumption and hardware implementation. The furthest nodes from the data sink will die much more quickly than those close to the sink. Also, the overall traffic load in the network may increase rapidly with the increase of the network size, which would cause more collisions, and thus increase energy consumption and delivery latency.

In a multi-hop network, sensor nodes transmit their sensed data to the sink using short-range wireless communication via one or more intermediate nodes. Each intermediate node must perform routing and forward the data along a multi-hop path. Moreover, data aggregation can be performed at an intermediate node to eliminate data redundancy, which can reduce the total amount of traffic in the network and thus improve the energy efficiency of the network. In general, a single-hop network has simpler network architecture and thus is easier to control. It is suitable for applications in small sensing areas with sparsely deployed sensor nodes. Multi-hop networks have a wider range of applications at the cost of higher control complexity.

2.2.6 Self-Reconfigurable and Non-Self-Configurable Network

According to the configurability of sensor nodes, a sensor network can be self-configurable or non-self-configurable. In a non-self-configurable network, sensor nodes have no ability to organize themselves into a network. Instead, they have to rely on a central controller to control each sensor node and collect information from them. Therefore, this type of networks is only suitable for small-scale networks. In most sensor networks, however, sensor nodes are able to autonomously organize and maintain their connectivity by themselves and collaboratively accomplish a sensing task. A network with such self-configurability is suitable for large-scale networks to perform complicated sensing tasks.

2.2.7 Homogeneous and Heterogeneous Network

According to whether sensor nodes have the same capabilities, a sensor network can be homogeneous or heterogeneous [34]. In a homogeneous network, all sensor nodes have the same capabilities in terms of energy, computation, and storage. In contrast, a heterogeneous network has some sophisticated sensor nodes that are equipped with more processing and communicating capabilities than normal sensor nodes. In this case, the network can assign more processing and communication tasks to those sophisticated nodes in order to improve its energy efficiency and thus prolong the lifetime.

2.3 Protocol Stack

The protocol stack for WSNs consists of five protocol layers: the physical layer, data link layer, network layer, transport layer and application layer [32]. The application layer contains a variety of application-layer protocols to generate various sensor network applications. The transport layer is responsible for reliable data delivery required by the application layer. The network layer is responsible for routing the data from the transport layer. The data link layer is primarily responsible for data stream multiplexing, data frame transmission and reception, medium access, and error control. The physical layer is responsible for signal transmission and reception over a physical communication medium, including frequency generation, signal modulation, transmission and reception, data encryption, and so on.

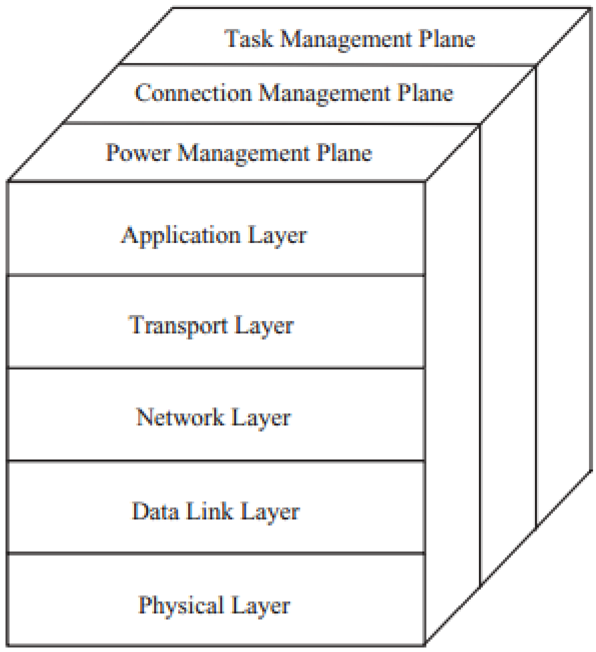

Figure 2.7: Protocol Stack

On the other hand, the protocol stack can be divided into a group of management planes across each layer [35], including power, connection, and task management planes. The power management plane is responsible for managing the power level of a sensor node for sensing, processing, and transmission and reception, which can be implemented by employing efficient power management mechanisms at different protocol layers. For example, at the MAC layer, a sensor node can turn off its transceiver when there is no data to transmit and receive. At the network layer, a sensor node may select a neighbour node with the most residual energy as its next hop to the sink.

The connection management plane is responsible for the configuration and reconfiguration of sensor nodes to establish and maintain the connectivity of a network in the case of node deployment and topology change due to node addition, node failure, node movement, and so on. The task management plane is responsible for task distribution among sensor nodes in a sensing region in order to improve energy efficiency and prolong network lifetime. Since sensor nodes are usually densely deployed in a sensing region and are redundant for performing a sensing task, not all sensor nodes in the sensing region are required to perform the same sensing task. Therefore, a task management mechanism can be used to perform task distribution among multiple sensors.

2.3.1 Application Layer

The application layer includes a variety of application layer protocols that perform various sensor network applications, such as query dissemination, node localization, time synchronization, and network security. For example, the sensor management protocol (SMP) [36] is an application layer management protocol that provides software operations to perform a variety of tasks, for example, exchanging location related data, synchronizing sensor nodes, moving sensor nodes, scheduling sensor nodes, and querying the status of sensor nodes. The sensor query and data dissemination protocol (SQDDP) provides user applications with interfaces to issue queries, respond to queries, and collect responses [36]. The sensor query and tasking language (SQTL) provides a sensor programming language used to implement middleware in WSNs [37]. Although many sensor network applications have been proposed, their corresponding application layer protocols still need to be developed.

2.3.2 Transport Layer

In general, the transport layer is responsible for reliable end-to-end data delivery between sensor nodes and the sink(s). Due to the energy, computation, and storage constraints of sensor nodes, traditional transport protocols cannot be applied directly to sensor networks without modification. For example, the conventional end-to-end retransmission based error control and the window-based congestion control mechanisms used in the transmission control protocol (TCP) cannot be used for sensor networks directly because they are not efficient in resource utilization. On the other hand, sensor networks are application specific. A sensor network is usually deployed for a specific sensing application, for example, habitat monitoring, inventory control, and battlefield surveillance. Different applications may have different reliability requirements, which have a big impact on the design of transport layer protocols. In addition, data delivery in sensor networks primarily occurs in two directions: upstream and downstream. In the upstream, the sensor nodes transmit their sensed data to the sink(s), while in the downstream the data originated from the sink(s), for example, queries, commands, and programming binaries, are sent from the sink(s) to the source sensor nodes. The data flows in the two directions may have different reliability requirements. For example, the data flows in the upstream direction are loss-tolerant because the sensed data are usually correlated or redundant to a certain extent. In the downstream, however, the data flows are queries, commands, and programming binaries sent to the sensor nodes, which usually require 100% reliable delivery. Therefore, the unique characteristics of sensor networks and the specific requirements of different applications present many new challenges in the design of transport layer protocols for WSNs.

2.3.3 Network Layer

The network layer is responsible for routing the data sensed by source sensor nodes to the data sink(s). In a sensor network, sensor nodes are deployed in a sensing region to observe a phenomenon of interest. The observed phenomenon or data need to be transmitted to the data sink. In general, a source node can transmit the sensed data to the sink either directly via single-hop long-range wireless communication or via multi-hop short-range wireless communication. However, long-range wireless communication is costly in terms of both energy consumption and implementation complexity for sensor nodes. In contrast, multi-hop short-range communication can not only significantly reduce the energy consumption of sensor nodes, but also effectively reduce the signal propagation and channel fading effects inherent in long-range wireless communication, and is therefore preferred. Since sensor nodes are densely deployed and neighbour nodes are close to each other, it is possible to use multi-hop short-range communication in sensor networks. In this case, to send the sensed data to the sink, a source node must employ a routing protocol to select an energy efficient multi-hop path from the node itself to the sink. However, routing protocols for traditional wireless networks are not suitable for sensor networks because they do not consider energy efficiency as the primary concern. Also, data from the sensing region toward the sink exhibit a unique many-to-one traffic pattern in sensor networks [23].

The combination of multi-hop (i.e., hop-by-hop) and many-to-one communications results in a significant increase in transit traffic intensity and thus packet congestion, collision, loss, delay, and energy consumption as data move closer toward the sink. The sensor nodes closer to the sink, typically within a small number of hops, will lose a larger number of packets and consume much more energy than the nodes further away from the sink, thus largely reducing the operational lifetime of the entire network. Therefore, it is important to take into account the energy constraint of sensor nodes as well as the unique traffic pattern in the design of the network layer and routing protocols. In this context, a large amount of research has been conducted and a variety of routing protocols have been proposed to address various application scenarios of sensor networks.

2.3.4 Data Link Layer

The data link layer is responsible for data stream multiplexing, data frame creation and detection, medium access, and error control in order to provide reliable point-to-point and point-to-multipoint transmissions. One of the most important functions of the data link layer is medium access control (MAC). The primary objective of MAC is to fairly and efficiently share the shared communication resources or medium among multiple sensor nodes in order to achieve good network performance in terms of energy consumption, network throughput, and delivery latency. However, MAC protocols for traditional wireless networks cannot be applied directly to sensor networks without modification because they do not take into account the unique characteristics of sensor networks, in particular, the energy constraint. For example, the primary concern in a cellular system is to provide quality of service (QoS) to users.

Energy efficiency is only of secondary importance because there is no power limit with the base stations and the mobile users can replenish the batteries in their handsets. In MANETs, mobile nodes are equipped with portable devices powered by battery, which is also replaceable. In contrast, the primary concern in sensor networks is energy conservation for prolonging network lifetime, which makes traditional MAC protocols unsuitable for sensor networks. Therefore, a large amount of research work has been conducted on MAC and a variety of MAC protocols have been proposed to address different application scenarios. Another important function of the data link layer is error control in data transmission. In many applications, a sensor network is deployed in a harsh environment where wireless communication is error prone. In this case, error control becomes indispensable and critical for achieving link reliability or reliable data transmission. In general, there are two main error control mechanisms:

Forward Error Correction (FEC) and Automatic Repeat reQuest (ARQ): ARQ achieves reliable data transmission by retransmitting lost data packets or frames. Obviously, this incurs significant retransmission overheads and additional energy consumption, and therefore is not suitable for sensor networks. FEC achieves link reliability by using error control codes in data transmission, which introduces additional encoding and decoding complexities that require additional processing resources in sensor nodes. However, FEC can significantly reduce the channel bit error rate (BER) for any given transmission power. Given the energy constraint of sensor nodes, FEC is still the most efficient solution to error control in sensor networks. To design a FEC mechanism, the choice of the error control code is very important because a well-chosen error control code can obtain a good coding gain and several orders of magnitude reduction in BER.

Meanwhile, the additional processing power consumed for encoding and decoding must also be considered. Therefore, a trade-off should be optimized between the additional processing power and the corresponding coding gain in order to have a powerful, energy efficient, and low complexity FEC mechanism.

2.3.5 Physical Layer

The physical layer is responsible for converting bit streams from the data link layer to signals that are suitable for transmission over the communication medium. For this purpose, it must deal with various related issues, for example, transmission medium and frequency selection, carrier frequency generation, signal modulation and detection, and data encryption. In addition, it must also deal with the design of the underlying hardware, and various electrical and mechanical interfaces.

Medium and frequency selection is an important problem for communication between sensor nodes. One option is to use radio and the industrial, scientific and medical (ISM) bands that are licence-free in most countries. The main advantages of using the ISM bands include free use, large spectrum, and global availability [38]. However, the ISM bands already have been used for some communication systems, such as cordless phone systems and wireless local area networks (WLANs). On the other hand, sensor networks require a tiny, low cost, and ultralow power transceiver. For these reasons, the 433 MHz ISM band and the 917 MHz ISM band have been recommended for use in Europe and North America, respectively.

Many projects have used radio frequency (RF) circuits in the hardware design for sensor nodes, such as the mAMPS project [39], where the sensor node uses a 2.4 GHz transceiver, and that in [40], where the sensor node uses a single channel RF transceiver operating at 916 MHz In addition to radio, optical or, infrared medium can be a possible option. For example, the Smart Dust project [41] used the optical medium for transmission. However, both require that a sender and its receiver be within the sight distance to communicate with each other, which limits their use to a certain extent [42].

This chapter deals with the basics and network architecture of WSNs with the classification of it as well. Functionalities of all the layers of WSNs are also discussed along with the protocol stack.

CHAPTER 3

OVERVIEW OF CROSS-LAYER APPROACH

Wireless sensor networks (WSNs) require energy efficient protocols to improve the network lifetime. In order to minimize the energy consumption, extensive research has been conducted in the literature on designing energy efficient protocols at each layer aside. Regarding the MAC layer [43], the most common way to conserve energy consists in putting the transceiver and the processor of a sensor node into a low power sleep state when it is unused. As such, the wasted energy due to collisions, overhearing and idle listening is reduced. On the other hand, works in [44], [45] addressed the problem at the network layer by proposing new routing solutions that take into account the sleep state of some nodes.

3.1 Historical Background

Several methods are proposed to save the energy of the sensor nodes. Cooperative communication is one of the efficient methods introduced earlier [46]. Multi-hop networks use some form of cooperation by enabling intermediate nodes to forward the message from source to destination. The main advantages of cooperative communications are:

- Higher spatial diversity: resistance to both small scale and shadow fading.
- Higher throughput/lower delay: higher achievable data rates, fewer retransmissions, and lower transmission delay.
- Reduced interference/lower transmitted power: better frequency reuse in a cellular/WLAN deployment.
- Adaptability to network conditions: opportunistic use and redistribution of network energy and bandwidth.

However, in order to realize a fully cooperative network, research at the physical layer should be coupled with higher layers of the protocol stack, in particular, the MAC sub-layer and the network layer.

Spatial-temporal correlation is another one. It has been shown in [47] that exploiting the spatial and temporal correlation improve energy efficiency of communication in WSN. Due to high density in the network topology, sensor observations are highly correlated in the space domain. Furthermore, the nature of the physical phenomenon constitutes the temporal correlation between each consecutive observation of a sensor node. These spatial and temporal correlations along with the collaborative nature of the WSN bring significant potential advantages for the development of efficient communication protocols well-suited for the WSN paradigm.

In [48], the authors analyzed that the event-to-sink reliable transport (ESRT) protocol which ensures reliable event detection with minimum energy expenditure. It included a congestion control component that served the dual purpose of achieving reliability and conserving energy. Importantly, the algorithms of ESRT mainly run on the sink, with minimal functionality required at resource constrained sensor nodes. ESRT protocol operation was determined by the current network state based on the reliability achieved and congestion condition in the network. This self-configuring nature of ESRT made it robust to random, dynamic topology in WSN. Furthermore, ESRT could also accommodate multiple concurrent event occurrences in a wireless sensor field. Analytical performance evaluation and simulation results showed that ESRT converges to the desired reliability with minimum energy expenditure, starting from any initial network state.

Directed diffusion is another approach to reduce energy consumption of sensor nodes [49]. Directed diffusion is data-centric in that all communication is for named data. All nodes in a directed diffusion based networks are application aware. This enables diffusion to achieve energy savings by selecting empirically good paths and by caching and processing data in-network (e.g., data aggregation). In [50], the authors showed that reception based forwarding strategies were more efficient than purely distance-based strategies. They also showed that relative blacklisting schemes reduce disconnections and achieve higher delivery rates than absolute blacklisting schemes and that Automatic Repeat Request (ARQ) schemes become more important in larger networks. Based on an analytical link loss model, they study the distance-hop trade-off via mathematical analysis and extensive simulations of a wide array of blacklisting/link-selection strategies; they also validate some strategies using a set of real experiments on motes. Their analysis, simulations and experiments all showed that the product of the packet reception rate (PRR) and the distance traversed towards destination was the optimal forwarding metric for the ARQ case, and was a good metric even without ARQ. Nodes using this metric often take advantage of neighbors in the transitional region (highvariance links). Their results also showed that reception based forwarding strategies were more efficient than purely distance-based strategies; relative blacklisting schemes reduced disconnections and achieved higher delivery rates than absolute blacklisting schemes; and that ARQ schemes became more important in larger networks.

Several works on WSN reveals that cross-layer integration and design techniques also result in significant improvement in terms of energy conservation [51], [52]. Their approach [51] combined medium access organization with routing. Losningen [52] Cross-Layer Approach was merged with data link layer and network layer to provide retransmission, monitoring and tracking facilities for the sensor nodes. It would act as an energy-efficient protocol in ubiquitous network and wireless communication by maximizing network's lifetime. Network Simulator-2 (NS2) was used to verify the proposed protocol for maximization of network's lifetime by analysing the one among the performance metrics, latency. The simulation results showed that the network's lifetime was increased by conserving the energy of the sensor nodes in Wireless Sensor Network. The proposed approach could be used for battle field surveillance where the uninterrupted data would be highly needed, habitat/wildlife monitoring/tracking and environmental monitoring etc.

The energy consumption for medium access control (MAC) and physical layer is analyzed in [53] and [54]. In [53], the authors proposed a detailed energy survey of the physical, data link, and network layer by analytical techniques. They also showed the impact of regular sleep periods on node energy consumption and present a comparison analysis of single-hop vs. multi-hop communications in the energy realm. A detailed energy expenditure analysis of not only the physical layer but also the link and network layer provides a basis for developing new energy efficient wireless sensor networks. Regular, coordinated sleeping extends the lifetime of sensor nodes, but systems can only benefit from sleeping in terms of transmitted packets if the data arrival rate to the system is low.

Energy efficiency is the driving motivation for it can be considered the most important factor for wireless sensor networks because of the power constraints set by battery operation. Radio solutions in the lower ISM bands are attractive because of their relatively easy implementation and low power consumption. However, the data rates of these commercial radios are also relatively low, limiting transmittable frame sizes to a few tens of octets along with strict duty cycle requirements. From the analysis they extracted in [53], key parameters of selected MAC protocols and showed that some traditional mechanisms, such as binary exponential back-off, have some inherent problems. They also argued that single-hop communications has up to 40% lower energy consumption than multi-hop forwarding within the feasible transmission distances of an ISM radio.

In [54], the authors suggested to combine new communication protocols with hardware solutions able to further reduce the nodes' power consumption. In this work, a cross-layer solution, based on the combined use of a duty-cycling protocol and a new kind of active wake-up circuit, was presented and validated by using a test bed approach. The resulting solution significantly reduced idle listening periods by awakening the node only when a communication was detected. Specifically, an MAC scheduler managed the awakenings of a commercial power detector connected to the sensor node, and, if an actual communication was detected, it enabled the radio transceiver. The effectiveness of the proposed cross-layer protocol had been thoroughly evaluated by means of tests carried out in an outdoor environment.

In [55], the authors proposed an Energy Optimization Approach based on Cross-Layer for Wireless Sensor Networks named as EOA, which considered the joint optimal design of the physical, medium access control (MAC), and routing layer. The focus of EOA was on the computation of optimal transmission power, routing, and duty-cycle schedule that optimize the WSNs energy-efficiency. They first proposed a feedback algorithm that computes the proper transmission power level between nodes. Then, routing protocol could make use of the transmission power as a metric by choosing route with optimal power consumption to forward packets. Finally, the cross-layer routing information was exploited to form a duty-cycle schedule in MAC layer. EOA is validated on a CROSSBOW's MicaZ mote platform, and evaluated using the TOSSIM simulator, the simulation results showed that EOA was an energy-efficient approach and able to achieve significant performance improvement as well.

A cross-layer approach was established between MAC layer and network layer where the grid-quorum system was used on MAC layer and on the network layer the authors proposed to find query paths based on the power cost incurred by grid quorums used by nodes along a path [56].

A unified cross-layer protocol (XLP) is presented in [57-58] which achieves congestion control, routing and medium access control in a cross-layer fashion. XLP integrates functionalities of physical to transport layer into a cross-layer protocol using the concept of initiative determination which proves to be more energy efficient than the previous protocols. Initiative determination enables receiver-based contention, initiative-based forwarding, local congestion control and distributed duty cycle operation. The operation of the XLP is devised based on the new notion of initiative determination, which constitutes the core of the XLP and implicitly incorporates the intrinsic communication functionalities required for successful communication in WSN. Based on the initiative concept, XLP performs received based contention, local congestion control, and distributed duty cycle operation in order to realize efficient and reliable communication in WSN. Analytical performance evaluation and simulation experiment results show that XLP significantly outperforms the traditional layered protocol architectures in terms of both network performance and implementation complexity.

All of those methods basically focus on the cross-layer protocols rather than using any existing protocol to implement the cross-layer approach. The focus of us is kept on this issue and the method proposed here uses the technique of using Ad-hoc On-Demand Distance Vector (AODV) protocol inside the MAC layer which reduces the complexity of implementing a new protocol combining several layers. This also makes IEEE 802.11 MAC more efficient in terms of energy consumption.

3.2 Algorithm

In this research, an approach to execute cross-layer between MAC layer and network layer is established which achieves energy efficiency more than a traditional layered approach. The interfacing between these two layers has been strengthened to pass the routing table from network to MAC layer. At first the identity of the routing table of network layer is characterized by a symbol. Then the symbol is called upon from the MAC layer. Therefore, the MAC layer decides whether a signal needs to be sent to the upper layers or not (for receiving node) and to the specific sensor node or not (for transmitting node). Combination of these two processes reduces wastage of energy throughout the network by ensuring more synchronized and efficiently scheduled data transmission and reception.

Among several MAC layer protocols, IEEE 802.11 MAC seemed to be inefficient comparing several other MAC layer protocols. It has been shown that IEEE 802.11 MAC still can be efficient if our cross-layer approach is imposed on it. Our results also outperform S-MAC in several extents. S-MAC utilizes coordinated adaptive sleeping where the concept of duty cycle is initialized to keep sensor nodes active for a certain period of time to save energy [59].

3.2.1 MAC Layer Design Choices

Link layer Automatic Repeat Request (ARQ) refers to the hop-to-hop recovery of frames that arrive with errors. The primary design choice investigated at the MAC layer was whether or not to employ link layer recovery via ARQ for packets. The MAC layer used in our evaluations was 802.11 [60].

The primary reliability mechanisms provided by 802.11 are RTS/CTS, ACK, and randomized slot selection. RTS/CTS is the media access control packet exchange that guarantees that single transmitter will gain exclusive access to a shared transmission space. The ACK packet is sent by the receiver upon receipt of a data packet to inform the transmitter when successful transmission has occurred. This is a basic “stop-and-wait” ARQ mechanism where the transmitter times out and retransmits when an ACK does not arrive within a window of expectation. The 802.11 MAC does not employ RTS/CTS or ACK for multicast and broadcast transmissions due to ACK and CTS “implosion”. However, it attempts to reduce the probability of broadcast collision by randomly selecting a transmission slot once an idle media is sensed. Clients of this MAC layer can choose to employ ARQ or not by selecting unicast or broadcast addresses. Three different modes when considering MAC layer ARQ are utilized:

3.2.1.1 No ARQ

All transmissions are sent with a randomized send time and a broadcast MAC address. Unicasting is accomplished by address screening at the routing layer. Such transmissions do not employ MAC layer reliability mechanisms such as RTS/CTS and ACK. In this mode, reliability is completely deferred to the transport or application layer. There are several possible benefits to this scheme. Firstly, there is a significant amount of overhead over time connected with the exchange of RTS/CTS and ACK packets that is avoided. Secondly, routing protocols like diffusion attempt to select high quality (lower error rate) paths for data transmission. The reliability mechanisms in 802.11 can make poor paths mistakenly look reliable to higher layers.

3.2.1.2 ARQ Always

All transmissions are sent via a stop-and-wait ARQ protocol with a single node address. This transmission method utilizes RTS/CTS and ACK with retries to bolster perceived reliability. When a node wishes to communicate with multiple neighbors, each neighbor must be sent a unicast packet. The number of ARQ retransmissions attempted before giving up is configurable. This method also has certain benefits for sensor networks. Packets that travel on the links identified in route discovery will be delivered with a high degree of reliability, despite the transient interference typical in a wireless domain.

3.2.1.3 Selective ARQ

It is a combination of No ARQ and ARQ. In this scheme packets sent to single neighbors employ a stop-and-wait ARQ mechanism. Packets sent to multiple neighbors have no ARQ. This method attempts to combine the benefits of both ARQ and No ARQ. Data and control packets traveling on established paths are unicast, using ARQ to bolster reliability. Packets used in route-discovery are broadcast to all neighbors without ARQ. Poor paths are statistically not selected for reinforcement, and the route discovery procedure does not pay the overhead for reliability.

3.2.2 Transport Layer Design Choices

The transfer of data that is larger than the network Maximum Transmission Unit (MTU) is a particularly difficult task in wireless communication and, more specifically in directed diffusion. Although protocols such as 802.11 have fragmentation and reassembly facilities, there are limits on the size of an entity can be broken up, and guaranteed delivery is not provided [60, 61]. A single missing fragment from a large binary object (such as executable code) may render the data entity useless; therefore, transport layer facilities are required.

Traditional transport layers, like TCP, assume that the primary cause of packet loss is congestion. As such, their focus is on congestion control. In wireless sensor networks the primary problem is packet loss due to interference or low power.

The design decisions for the transport layer are primarily concerned with the balance of hop-by-hop vs. end-to-end functionality. Repair requests could be initiated by sinks (receiver end-points), or by in-network nodes on an established path. Obviously the type of MAC that any transport layer runs over will have a profound effect on how well the transport layer performs. The two transport layer schemes are [62]:

3.2.2.1 End-to-End Selective Request NACK

The need for repair and the generation of repair requests takes place only at the sinks. Repair requests for specific missing fragments travel on a reverse reinforced path from sink to source, where the missing data is retransmitted.

3.2.2.2 Hop-by-Hop Selective Request NACK and Repair from Cache

In this paradigm, each caching node on the reinforced path, from source to sink, caches the fragments which make up a large data entity. When such nodes sense a missing fragment, a repair request is sent to the next hop on the reverse reinforced path toward the source. If the requested fragment is in the local cache, a response is sent. If not, the NACK is forwarded to the next hop toward the source.

This chapter deals with the concept of cross-layer along with the relevant works done with cross-layer implementations. It also presents the algorithm of our proposed approach and design choices of MAC and transport layer.

CHAPTER 4

PROPOSED CROSS-LAYER APPROACH

4.1 System Model

The system considered in our approach is a clustered wireless sensor network shown in Figure 4.1. Communication link connecting two wireless sensor nodes can in general be Multiple Input Multiple Output (MIMO), Multiple Input Single Output (MISO), Single Input Multiple Output (SIMO) or Single Input Single Output (SISO). At first it is needed to look into the MAC layer along with the network layer. The protocols used to determine who goes next on a multi-access channel belong to a sub-layer of the data link layer which is called the MAC sub-layer. Therefore it determines the ability of a node to efficiently share the wireless medium with the other nodes in the network. One of the main objectives of the MAC layer is to keep the energy consumption low by turning off the radio module as often as possible.

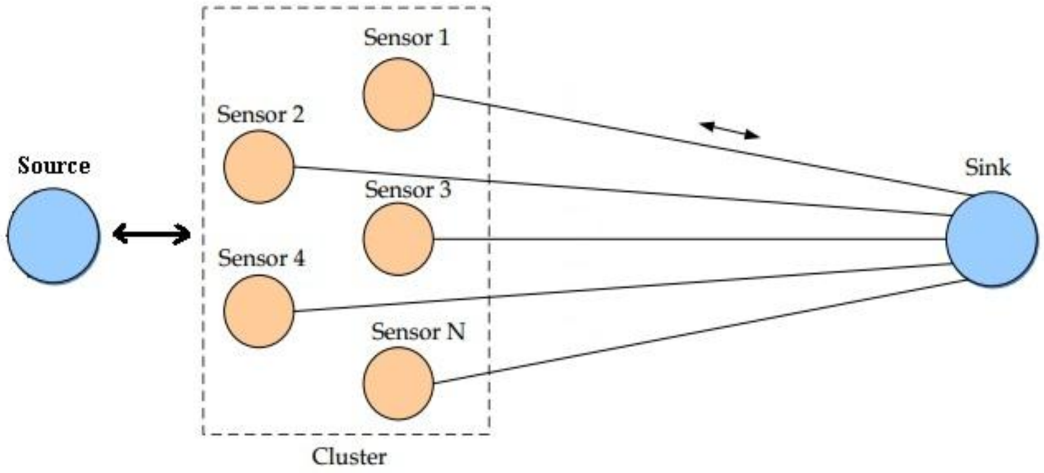

Figure 4.1: System Model

In order to design energy aware MAC protocols, the main causes of energy consumption need to be taken into consideration and which are idle listening, overheads, overhearing and collisions presented in [63]. Therefore, these factors need to be minimized in order to achieve the energy efficiency. But there exists a tradeoff for the optimal design. For example, the protocol which aims to reduce idle monitoring and collisions always requires extra synchronizations and overheads, whereas, an increase in energy waste occurs due to collisions while reducing these factors.

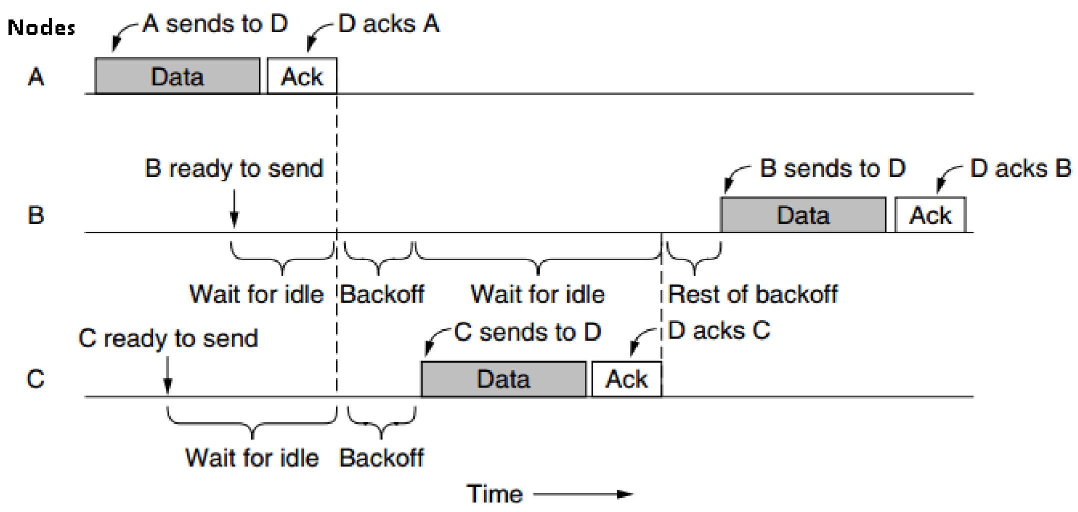

Figure 4.2: Sending a frame with MAC

That is why the proposal presented here focuses on the interfacing between the layers rather than implementing a protocol. This is a very simple method to implement which reduces the complexity and cost of the system along with the energy.

Several protocols are there to functionalize this sub-layer and among all of those, IEEE 802.11 MAC protocol tries to avoid collisions with a protocol called CSMA/CA (CSMA with Collision Avoidance) which can be found in [64]. Figure 4.2 illustrates the scenario clearly where it is shown that node A is the first to send a frame. While A is sending, nodes B and C become ready to send. They see that the channel is busy and wait for it to become idle. Shortly after A receives an acknowledgement, the channel goes idle.

However, rather than sending a frame right away and colliding, B and C both perform a back-off. C picks up a short back-off, and thus sends first. B halts its countdown while it senses that C is using the channel and resumes after C has received an acknowledgement. B soon completes its back-off and sends its frame.

This scheduling of MAC layer would be much more efficient if the information about the nodes' address and routing channel are provided to that very layer. We would like to use AODV routing protocol in the network layer as it stores a route table which consists of destination address, next hop address, destination sequence number and life time of any route. Our aim is to provide MAC layer with this route table which makes scheduling more efficient. When a node wishes to send a packet to some destination then it checks its routing table to determine if it has a current route to the destination. If it finds a root then it forwards the packet to next hop node or initiates a route discovery process if it is otherwise. Route discovery process begins with the creation of a Route Request (RREQ) packet which is created by source node. For each destination, a node maintains a list of precursor nodes, to route through them. Precursor nodes help in route maintenance. Life-time is also updated every time a route is used and if a route is never used within its life time then it expires. Therefore, energy will not be wasted. Say for example, B node has never been able to send any data to D. If the routing table is not there in the MAC layer, then this layer will schedule the data to be sent to D again. But again D will not be able to receive the data which will cause unnecessary loss of transmission energy.

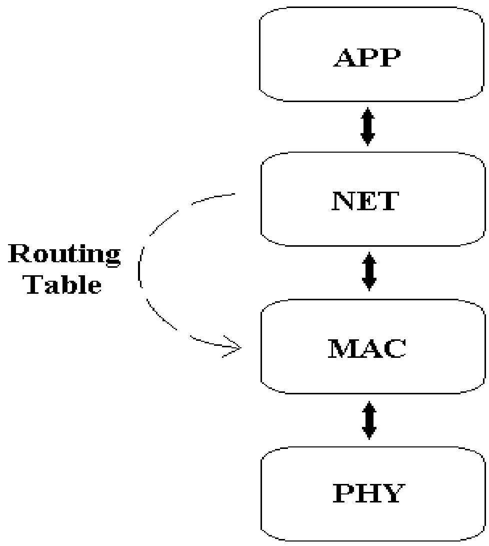

Figure 4.3: Overview of our cross-layer approach

Even if D receives the data from another node as WSN has that property, it will not send the data to its upper layer for further processing to communicate with B which will reduce the receiving energy as well. Rather D will turn into sleep mode if it never receives any data from any sensor node. Therefore our cross-layer approach will reduce energy consumption by controlling the flow of data throughout the network.

4.2 Route Discovery

When an intermediate node receives a RREQ, the node sets up a reverse route entry for the source node in its route table which consists of source IP address, source sequence number,

number of hops to source node, IP address of node from which RREQ was received. Any node can send a RREP (Route Reply Packet) to the source using the reverse route which also contains life time field. In order to respond to RREQ a node should have unexpired entry for the destination and sequence number of destination at least as great as in RREQ (for loop prevention) in its route table.

If both conditions are met and the IP address of the destination matches with that in RREQ the node responds to RREQ by sending a RREP back using unicasting. If conditions are not satisfied, then node increments the hop count in RREQ and broadcasts to its neighbors. Ultimately the RREQ will make to the destination. Figure 4.4 to Figure 4.7 will explain the procedure in detail.

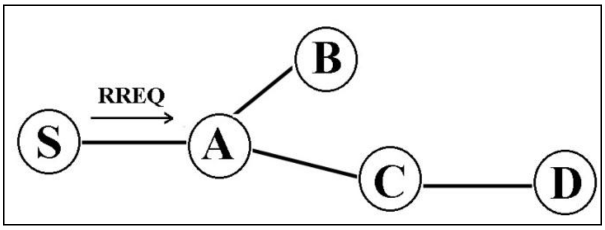

Figure 4.4: Node S sends RREQ to Node A

From Figure 4.4 it can be seen that S node needs a route to reach destination node D. Therefore S creates a RREQ which enters D’s IP address and sequence number, S’s IP address and sequence number, hop count (= 0). Node S broadcasts RREQ to neighbors and node A receives it which makes a reverse route entry for S where, destination = S, next hop = S, hop count = 1. A has no routes to D, so it rebroadcasts RREQ to its neighbors which is shown in Figure 4.5.

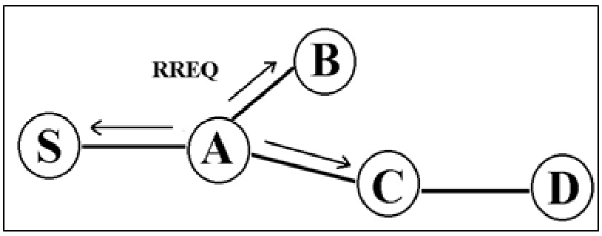

Figure 4.5: Node A sends RREQ to neighbors

Node C receives RREQ and makes a reverse route entry for S where, destination = S, next hop = A, hop count = 2. As it has a route to D and the sequence for route to D is greater than D's sequence in RREQ, so C creates RREP. An intermediate node which knows a route with a smaller sequence number cannot send RREP. A new RREQ by node S for a destination is assigned a higher destination sequence number. C's RREP enters D's IP address and sequence number, S's IP address, hop count to D (= 1).

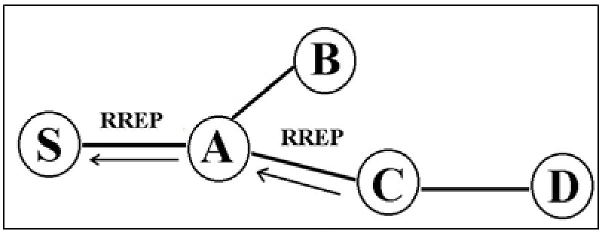

Figure 4.6: Node A sends RREP to Node S

Node A receives RREP and makes a forward route entry to D where, destination = D, next hop = C and hop count = 2. Then node A unicasts RREP to node S. Any node may receive multiple RREP for a given destination from more than one neighbor.

The node only forwards the first RREP it receives or may forward another RREP if that has greater destination sequence number or a smaller hop count. Rest of those are discarded which reduces the number of RREP propagating towards the source. Source can begin data transmission upon receiving the first RREP.

Now, after receiving RREP node S makes a forward route entry to D where, destination = D, next hop = A, hop count = 3. Then it sends data packet to D where the route is S-A-C-D. Broadcast transmission also follows the same procedure to send data packet to source to sink.

4.3 Route Failure

If the link between node C and node D breaks, then C creates Route Error (RERR) message rather creating a RREP and also invalidates the route to D in the routing table. RERR contains list of all destinations which are unreachable. Figure 4.7 illustrates the route failure scenario.

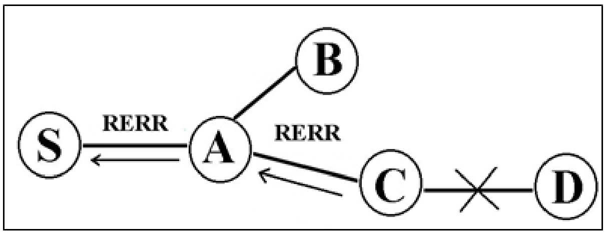

Figure 4.7: Link failure between node C and D

4.4 Simulation parameters

To analyze the performance of our cross-layer approach, we have used network simulator-2 (ns-2) software. This software can generate a real life networking scenario with ease. This is a discrete event simulator targeted at networking research which provides substantial support for simulation of Transmission Control Protocol (TCP), routing, and multicast protocols over wired and wireless (local and satellite) networks.

A clustered wireless network topology has been created with an event area of 800 x 800 m^2 where the source and sink nodes are also defined to several coordinates. The agents used in the transport layer are UDP (User Datagram Protocol) and Null agents as they do not use any acknowledgement while data transmission and thereby consume less energy than TCP agents.

TCP ensures reliable data transmission with the acknowledgements they use after receiving a data or packet along with more energy consumption. As our research is mainly concerned about the energy issues, therefore, TCP agents are not used here. All the nodes then communicate with the sink node to establish the network.

Then we have investigated several performance metrics varying the number of senor nodes. After that process we have applied our cross-layer approach to every sensor nodes and the same performance metrics are again being analyzed to make some comparative study with the other approaches. Number of nodes is varied in between 2 to 10 for every time the simulation is done. Several simulation parameters are presented in Table 4.1. These parameters are taken just as like as the parameters used for implementing S-MAC and traditional approach in order to achieve an equal base for comparative study.

Table 4.1: Simulation Parameters

Parameters	Value
Initial Energy	10 J
PL	55 dB
E_{rx}	13.5 mW
E_{tx}	24.75 mW
E_{sleep}	1.5 mW
$T_{transition}$	16 ms
Frame Length	5s
Buffer Length	30
Energy Threshold	100 µJ
CBR Packet Size	512 B

The following performance metrics have been investigated in the evaluations:

- Energy Efficiency: This is one of the most important metrics in WSNs. In our analysis, this metric is definitely the most important one as we are focusing on reduction of energy consumption of the sensor nodes. We consider the total energy consumption of all the sensor nodes here. So, lower value of that indicates more energy efficiency.

- Goodput: It is defined as the ratio between the total number of unique packets received at the sink and the total number of packets sent by all the source nodes. This metric will ensure the communication reliability of the network. Higher value of it indicates more reliable communication.

- Throughput: It is defined as the bits per second received at the sink. We have considered all the packets here, not only the unique packets.

4.5 Performance Evaluation

Simulation results have been presented with the graphs shown in Figure 4.8, Figure 4.9, Figure 4.10, Figure 4.11 and Figure 4.12.

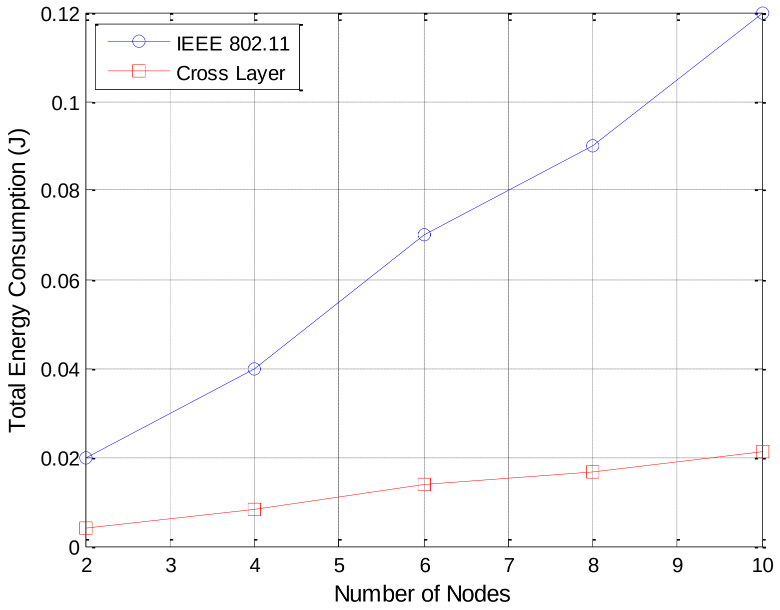

Figure 4.8: Energy Comparison between IEEE 802.11 MAC and Cross-layer Approach

At first, total energy consumption (in joules) throughout the network is observed varying the number of nodes. Figure 4.8 shows the comparison between our cross-layer approach with the IEEE 802.11 MAC protocol and Figure 4.9 with the S-MAC of different duty cycles.

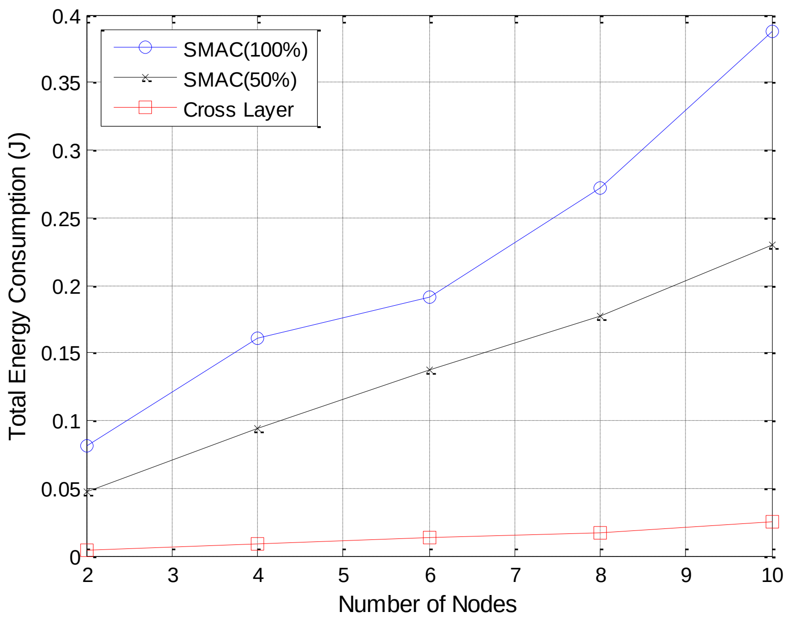

Figure 4.9: Energy Comparison between S-MAC and Cross-layer Approach

From both of the figures it is clearly visible that, our cross-layer approach consumes less energy than traditional 802.11 MAC and S-MAC. S-MAC with 50% duty cycle means the node is active for half of the period and for the other half it remains in sleep mode. Therefore, it will definitely consume less energy than the S-MAC with 100% duty cycle which is clearly visible in Figure 4.9. But our cross-layer approach outperforms both of those considering the fact of energy consumption throughout the network. The comparison gets more visible for increasing the number of sensor nodes. Therefore, our cross-layer approach works significantly for the larger sized sensor networking scenario.

Then the goodput analysis is presented in Figure 4.10 and Figure 4.11. Goodput is the ratio of data received to data transmitted or sent by a node.

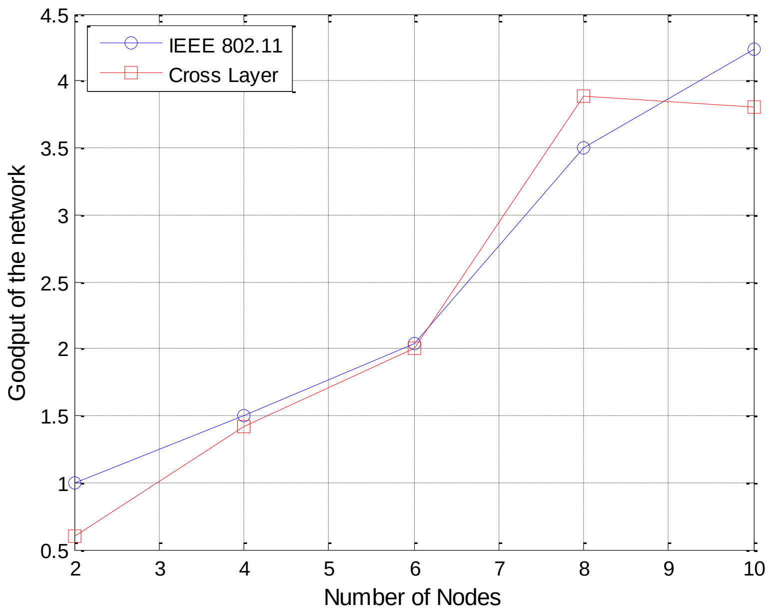

Figure 4.10: Goodput Comparison between IEEE 802.11 MAC and Cross-layer Approach

It is clearly observed from Figure 4.10 that, our approach not only consumes less energy but it also maintains a good number of data receptions to transmissions ratio. Though for several nodes goodput reduces to IEEE 802.11 as because our cross-layer approach controls the data flow to save energy. Although then it performs well considering the amount of energy it consumes along with maintaining a good throughput which is visible in Figure 4.11. Therefore all these figures need to analyze together to draw a conclusion about any of the approaches.

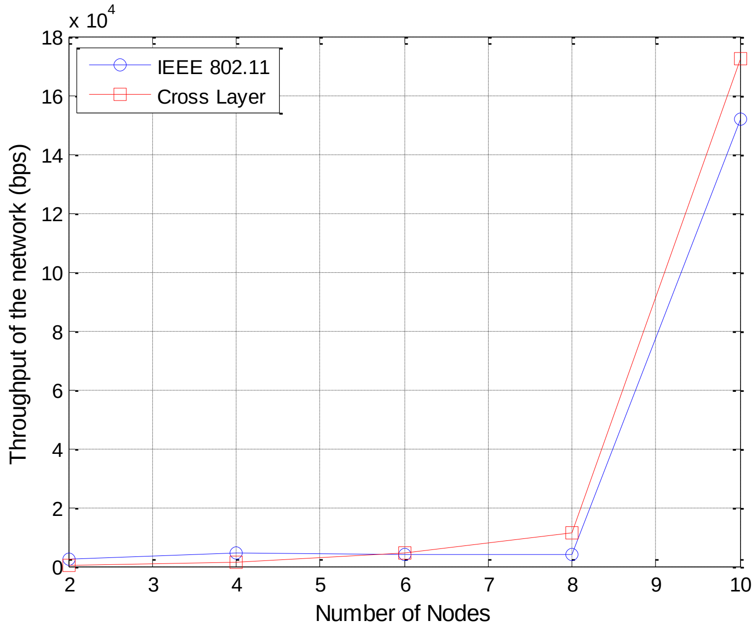

Figure 4.11: Throughput Comparison between IEEE 802.11 MAC and Cross-layer Approach

So from Figure 4.10, it can be concluded that the proposed approach can have less goodputs in several points of the graph than IEEE 802.11 MAC, but it receives more data throughout the simulation time which is shown in Figure 4.11. Therefore, this little bit of trade-off could easily be dealt with considering the overall performance.

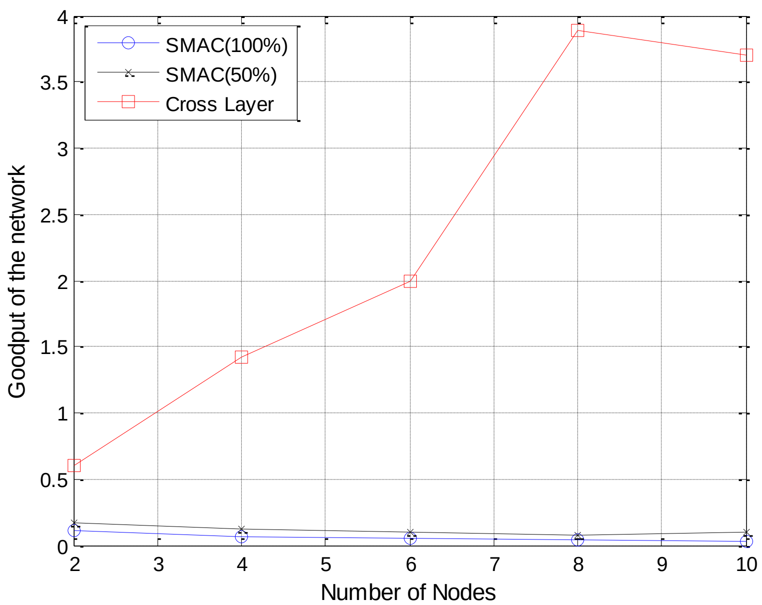

Figure 4.12: Goodput Comparison between S-MAC and Cross-layer Approach

The goodput comparison between our cross-layer approach and S-MAC with different duty cycles has also been taken into consideration which can be found in Figure 4.12. S-MAC with duty cycles of 100% and 50% are considered only for our analysis. 50% of duty cycle denotes that the node is active for 50% of the time and remains off for the other 50% of the time. Figure 4.12 suggests that S-MAC maintains a poor number of data receptions and transmissions ratio for its duty cycle operation and it is totally outperformed by our cross-layer approach.

In addition to the performance regarding simulation issues, the implementation issues are also very important. Our cross-layer approach does not necessarily require any external hardware or complex circuitries, rather a simple technique inside the software makes the interfacing between the two layers stronger and thus the efficiency is achieved. As because of single buffer the upper layers have to wait for the lower layers to process the data. This may lead to significant energy consumption if the packet needs not to be processed further. This problem is also solved by implementing cross-layer approach.

CHAPTER 5

INTERFACING WITH NETWORK SIMULATOR-2

5.1 Basic Wireless Model

The wireless model essentially consists of the MobileNode at the core, with additional supporting features that allows simulations of multi-hop ad-hoc networks, wireless LANs etc. The MobileNode object is a split object. A MobileNode thus is the basic Node object with added functionalities of a wireless and mobile node like ability to move within a given topology, ability to receive and transmit signals to and from a wireless channel etc. A major difference between them, though, is that a MobileNode is not connected by means of Links to other nodes or mobile nodes. In this section we shall describe the internals of MobileNode, its routing mechanisms, the routing protocols DSDV, AODV, TORA, PUMA and DSR, creation of network stack allowing channel access in MobileNode, brief description of each stack component, trace support and movement/traffic scenario generation for wireless simulations.

5.2 Network Components in a Node

The network stack for a sensor node consists of a link layer (LL), an ARP module connected in interface priority queue (IFq), a mac layer (MAC), a network interface (netIF), all connected to the channel. These network components are related and plumbed together in OTcl.

5.2.1 Link Layer

The LL used by sensor node is discussed in this section. The only difference being the link layer for sensor node has an ARP module connected to it which resolves all IP to hardware (MAC) address conversions. Normally for all outgoing (into the channel) packets, the packets are handed down to the LL by the Routing Agent. The LL hands down packets to the interface queue. For all incoming packets (out of the channel), the mac layer hands up packets to the LL which is then handed off at the node_entry_ point. The class LL is implemented in ~ns/ll.{cc,h} and ~ns/tcl/lan/ns-ll.tcl.

5.2.2 ARP

The Address Resolution Protocol (implemented in BSD style) module receives queries from Link layer. If ARP has the hardware address for destination, it writes it into the mac header of the packet. Otherwise it broadcasts an ARP query, and caches the packet temporarily. For each unknown destination hardware address, there is a buffer for a single packet. In case, additional packets to the same destination are sent to ARP, the earlier buffered packet is dropped.

Once the hardware address of a packet's next hop is known, the packet is inserted into the interface queue. The class ARPTable is implemented in ~ns/arp.{cc,h} and ~ns/tcl/lib/ns-mobilenode.tcl.

5.2.3 Interface Queue

The class PriQueue is implemented as a priority queue which gives priority to routing rotocol packets, inserting them at the head of the queue. It supports running a filter over all packets in the queue and removes those with a specified destination address. See ~ns/priqueue.{cc,h} for interface queue implementation.

5.2.4 MAC Layer

Historically, ns-2 (prior to release ns-2.33) has used the implementation of IEEE 802.11 distributed coordination function (DCF) from CMU. Starting with ns-2.33, several 802.11 implementations are available.

5.2.5 Tap Agents

Agents that subclass themselves as class Tap defined in mac.h can register themselves with the mac object using method installTap(). If the particular Mac protocol permits it, the tap will promiscuously be given all packets received by the mac layer, before address filtering is done.

Further can be found in ~ns/mac.{cc,h} for class Tap implementation.

5.2.6 Network Interfaces

The Network Interphase layer serves as a hardware interface which is used by mobile node to access the channel. The wireless shared media interface is implemented as class Phy/WirelessPhy. This interface subject to collisions and the radio propagation model receives packets transmitted by other node interfaces to the channel. The interface stamps each transmitted packet with the meta-data related to the transmitting interface like the transmission power, wavelength etc. This meta-data in packet header is used by the propagation model in receiving network interface to determine if the packet has minimum power to be received and/or captured and/or detected (carrier sense) by the receiving node. The model approximates the DSSS radio interface (Lucent WaveLan direct-sequence spread-spectrum). Details can be found ~ns/phy.{cc.h} and ~ns/wireless-phy.{cc,h} for network interface implementations.

5.2.7 Radio Propagation Model

It uses Friss-space attenuation ($1/r^2$) at near distances and an approximation to Two ray Ground ($1/r^4$) at far distances. The approximation assumes specular reflection off a flat ground plane. Details are there in ~ns/tworayground.{cc,h} for implementation.

5.2.8 Antenna

An omni-directional antenna having unity gain is used by mobilenodes. The file ~ns/antenna.{cc,h} can be used for implementation details.

5.3 Wireless Topology Creation

As our consideration deals with the static nodes, therefore MobileNode class is converted inside the tcl scripts with no node movements. When MobileNode is the basic nsNode object with added functionalities like movement, ability to transmit and receive on a channel that allows it to be used to create mobile, wireless simulation environments. The class MobileNode is derived from the base class Node. MobileNode is a split object. The mobility features including node movement, periodic position updates, maintaining topology boundary etc are implemented in C++ while plumbing of network components within MobileNode itself (like classifiers, dmux , LL, Mac, Channel etc) have been implemented in Otcl. The functions and procedures described in this subsection can be found in ~ns/mobilenode.{cc,h}, ~ns/tcl/lib/ns mobilenode.tcl, ~ns/tcl/mobility/dsdv.tcl, ~ns/tcl/mobility/dsr.tcl, ~ns/tcl/mobility/tora.tcl. Example scripts can be found in ~ns/tcl/ex/wireless-test.tcl and ~ns/tcl/ex/wireless.tcl.

While the first example uses a small topology of 3 nodes, the second example runs over a topology of 50 nodes. These scripts can be run simply by typing $ns tcl/ex/wireless.tcl (or /wireless-test.tcl). The five ad-hoc routing protocols that are currently supported are Destination Sequence Distance Vector (DSDV), Dynamic Source Routing (DSR), Temporally ordered Routing Algorithm (TORA), Ad-hoc On-demand Distance Vector (AODV) and Protocol for Unified Multicasting Through Announcements (PUMA). The primitive to create a mobile node is described below. The old APIs for creating a mobile node depended on which routing protocol was used, like set mnode [$opt(rp)-create-mobile-node $id] where $opt(rp) defines "DSDV", "AODV", "TORA", "DSR" or "PUMA" and ID is the index for the mobile node. But the old API's use is being deprecated and the new API is described in Appendix A.

The above API configures for a mobile node with all the given values of ad-hoc routing protocol, network stack, channel, topography, propagation model, with wired routing turned on or off (required for wired-cum-wireless scenarios) and tracing turned on or off at different levels (router, mac, agent). In case hierarchical addressing is being used, the hier address of the node needs to be passed as well. The code in Appendix A describes the procedure to create wireless nodes.

The above procedure creates a mobilenode (split) object, creates an ad-hoc routing agent as specified, creates the network stack consisting of a link layer, interface queue, mac layer, and a network interface with an antenna, uses the defined propagation model, interconnects these components and connects the stack to the channel.

5.4 Different MAC Protocols

There are several MAC protocols available in ns2 which are listed below:

5.4.1 802.11 MAC protocol

Historically, ns-2 (prior to release ns-2.33) has used the implementation of IEEE 802.11 distributed coordination function (DCF) from CMU. Starting with ns-2.33, several 802.11 implementations are available.

5.4.2 Preamble based TDMA protocol

This work is still at a preliminary stage, some practical issues, such as: contention in the preamble phase and time slot reuses in a multi-hop environment are not considered. Unlike contention based MAC protocol (802.11, for example), a TDMA MAC protocol allocates different time slots for nodes to send and receive packets. The superset of these time slots is called a TDMA frame.

Node Currently, ns supports a single hop, preamble-based TDMA MAC protocol. With this protocol, a TDMA frame contains preamble besides the data transmission slots. Within the preamble, every node has a dedicated sub-slot and uses it to broadcast the destination node id of outgoing packet. Other nodes listen in the preamble and record the time slots to receive packets. Like other common TDMA protocols (GSM, for example), each node has a data transmission slot to send packets. To avoid unnecessary power consumption, each node turns its radio on and off explicitly by invoking node API set_node_sleep (). The radio only needs to be on when: in the preamble phase (takes one slot time) and there is a packet to send and receive. The preamble is implemented as a central data structure tdma_preamble, which is accessible to all the nodes. At the beginning of a frame, each node writes the destination node id into its sub-slot in preamble if it has a packet to send. Following preamble phase, each node sends packet in its data transmission slot and checks the preamble to determine if there is a packet to receive in other slots. The following parameters are user configurable: the wireless link bandwidth, the slot length packet_slot_len, and the number of nodes max_node_num.

5.5 Different Routing Protocols

The five different ad-hoc routing protocols currently implemented for mobile networking in ns2 are DSDV, DSR, AODV, TORA and PUMA.

5.5.1 DSDV

In this routing protocol routing messages are exchanged between neighbouring mobile nodes (i. e, mobile nodes that are within range of one another). Routing updates may be triggered or routine. Updates are triggered in case routing information from one of the neighbours forces a change in the routing table. A packet for which the route to its destination is not known is cached while routing queries are sent out. The packets are cached until route-replies are received from the destination. There is a maximum buffer size for caching the packets waiting for routing information beyond which packets are dropped.

All packets destined for the mobile node are routed directly by the address demux to its port demux. The port demux hands the packets to the respective destination agents. A port number of 255 is used to attach routing agent in mobile nodes. The mobile nodes also use a default-target in their classifier (or address demux).

In the event a target is not found for the destination in the classifier (which happens when the destination of the packet is not the mobile node itself), the packets are handed to the default-target which is the routing agent. The routing agent assigns the next hop for the packet and sends it down to the link layer. The routing protocol is mainly implemented in C++.

5.5.2 DSR

This section briefly describes the functionality of the dynamic source routing protocol. As mentioned earlier the SRNode is different from the MobileNode. The SRNode’s entry_ points to the DSR routing agent, thus forcing all packets received by the node to be handed down to the routing agent. This model is required for future implementation of piggy-backed routing information on data packets which otherwise would not flow through the routing agent. The DSR agent checks every data packet for source-route information. It forwards the packet as per the routing information. In case it does not find routing information in the packet, it provides the source route, if route is known, or caches the packet and sends out route queries if route to destination is not known. Routing queries, always triggered by a data packet with no route to its destination, are initially broadcast to all neighbours. Route replies are send back either by intermediate nodes or the destination node, to the source, if it can find routing info for the destination in the route-query. It hands over all packets destined to itself to the port dmux. In SRNode the port number 255 points to a null agent since the packet has already been processed by the routing agent.

5.5.3 TORA

Tora is a distributed routing protocol based on "link reversal" algorithm. At every node a separate copy of TORA is run for every destination. When a node needs a route to a given destination it broadcasts a QUERY message containing the address of the destination for which it requires a route. This packet travels through the network until it reaches the destination or an intermediate node that has a route to the destination node. This recipient node then broadcasts an UPDATE packet listing its height with the destination. As this node propagates through the network each node updates its height to a value greater than the height of the neighbour from which it receives the UPDATE. This results in a series of directed links from the node that originated the QUERY to the destination node. If a node discovers a particular destination to be unreachable it sets a local maximum value of height for that destination. In case the node cannot find any neighbour having finite height with this destination it attempts to find a new route. In case of network partition, the node broadcasts a CLEAR message that resets all routing states and removes invalid routes from the network. TORA operates on top of IMEP (Internet MANET Encapsulation Protocol) that provides reliable delivery of route-messages and informs the routing protocol of any changes of the links to its neighbours. IMEP tries to aggregate IMEP and TORA messages into a single packet (called block) in order to reduce overhead. For link-status sensing and maintaining a list of neighbour nodes, IMEP sends out periodic BEACON messages which are answered by each node that hears it by a HELLO reply message.

5.5.4 AODV

AODV is a combination of both DSR and DSDV protocols. It has the basic route-discovery and route-maintenance of DSR and uses the hop-by-hop routing, sequence numbers and beacons of DSDV. The node that wants to know a route to a given destination generates a ROUTE REQUEST. The route request is forwarded by intermediate nodes that also create a reverse route for itself from the destination. When the request reaches a node with route to destination it generates a ROUTE REPLY containing the number of hops requires reaching destination. All nodes that participate in forwarding this reply to the source node create a forward route to destination. This state created from each node from source to destination is a hop-by-hop state and not the entire route as is done in source routing.

AODV is the main concern for us as our thesis work deals only with this protocol.

5.5.5 PUMA

The Protocol for Unified Multicasting through Announcements (PUMA) is a distributed, receiver initiated, mesh based multicast routing protocol. By default, the first receiver in a multicast group acts as the core (i.e., rendezvous point) for that particular group. PUMA uses a simple and very efficient control message, a multicast announcement, to maintain the mesh. Besides that, multiple meshes can be compiled into a single announcement bucket. PUMA does not require any unicast protocol, and all transmissions are broadcasts. Even though broadcast transmissions are unreliable, the mesh itself introduces some redundancy, and because the mesh includes only group members and the nodes interconnecting them, broadcasts remain scoped within the mesh. As a multicast announcement propagates throughout the mesh, nodes learn the shortest path to the core.

In this way data packets can be quickly routed to the core. On its way toward the core, two things can happen to a data packet: (a) the packet goes all the way until it reaches the core, or (b) a mesh member is hit before reaching the core. Once a data packet reaches the mesh, the packet propagates only inside the mesh. The core is not a single point of failure, because when the core fails a group member quickly takes the core role.

5.6 IEEE 802.11 MAC

Prior to release ns-2.33, there was only one main-tree 802.11 model, although other researchers were maintaining third-party patches on the web. Starting with ns-2.33, there are multiple choices in the main distribution. The first extension described below (infrastructure mode) extends the legacy model to include infrastructure mode. However, the last two items (802.11Ext and dei802mr) are complete replacements for the legacy model. Therefore, researchers now have a choice of 802.11 models, and should carefully read the documentation and code of each one to understand the best fit for the job.

5.6.1 802.11 DCF from CMU

This model has been the only model available in the main ns source tree prior to release ns-2.33. It uses a RTS/CTS/DATA/ACK pattern for all unicast packets and simply sends out DATA for all broadcast packets. The implementation uses both physical and virtual carrier sense. The class Mac802_11 is implemented in ~ns/mac-802_11.{cc,h}.

5.6.2 802.11 infrastructure extensions

Ilango Purushothaman from the University of Washington has implemented infrastructure extensions to the above 802.11 model, and fixed some bugs along the way. The extensions include passive and active scanning, authentication, association, inter-AP communications, and mobility support (handoff).

It needs to be noted that this model still supports single-channel scenarios only.

5.6.3 802.11Ext

A team from Mercedes-Benz Research and Development North America and from University of Karlsruhe have collaborated to develop a completely new 802.11 Mac and Phy model, called Mac802_11Ext and WirelessPhyExt, respectively. The new model contains the following features:

- Structured design of MAC functionality modules: transmission, reception, transmission coordination, reception coordination, back-off manager, and channel state monitor
- Cumulative SINR computation
- MAC frame capture capabilities
- Multiple modulation scheme support
- Packet drop tracing at the PHY layer
- Nakagami fading model

5.7 Cross-Layer Development

Development of cross-layer in ns2 requires some modification in system files. Steps are mentioned below:

(a) Identifying the routing table function in aodv.cc file.
(b) Defining the routing table function using a symbol in mac header file.
(c) This symbol is then called from tcl script while defining the sensor nodes.

Remaining energy of the nodes can be found in the trace files after the simulation is run. Using these values, energy efficiency of the nodes can be calculated.

5.8 Energy Model Implementation

Energy Model, as implemented in ns, is a node attribute. The energy model represents level of energy in a mobile host. The energy model in a node has an initial value which is the level of energy the node has at the beginning of the simulation. This is known as initialEnergy_. It also has a given energy usage for every packet it transmits and receives. These are called txPower_ and rxPower_. The files where the energy model is defined are ns/energymodel [.cc and .h]. Other functions/methods described in this chapter may be found in ns/wireless-phy.cc, ns/cmu-trace.cc, ns/tcl/lib[ns-lib.tcl, nsnode.tcl, ns-mobilenode.tcl].

Since the energy model is a node attribute, it can be defined by the codes included in Appendix A.

CHAPTER 6

CONCLUSIONS

6.1 Summary of Work

This research tries to represent that traditional layered protocol can still be efficient if cross layer approach is performed in it. It has made IEEE 802.11 MAC more energy efficient just crossing it with the network layer and result shows that it outperforms S-MAC as well which is more energy efficient than IEEE 802.11 MAC. It also reduces the complexity and expenditure of implementing a new protocol inside sensor network as no external functionalities have been included.

The ultimate goal of our cross layer approach is to utilize the resource of one layer to another layer. AODV routing protocol uses route table to every nodes, which leads to more energy consumption along with the implementation complexity when works with the MAC layer of IEEE 802.11 module. But it can be seen that combining these two layers with a little effort using our cross layer approach can have a significant impact for several performance metrics of the network.

In traditional layered structures, like in TinyOS [65], the upper layers have to wait for the lower layers to process the data as because a single buffer is used to process data or packet for all the layers. This may lead to significant energy consumption if the packet needs not to be processed further. This scheduling can be done more efficiently if the access of the routing table is given to the MAC layer from the network layer.

6.2 Future Scopes

The search for a good cross-layer approach still faces many challenges. Future research could be carried out in the following directions to further improve the cross-layer approach for WSNs.

- Criterion for parameters selection to cross-layer implementation in WSNs.

  This thesis derived the goodput, throughput and energy efficiency of cross-layer approach for WSNs varying the number of nodes. There are several parameters like, end-to-end delay estimation and overhead calculations can be taken into consideration while varying the distance.

- Implementation of the cross-layer approach over Zig-Bee Platform.

  This thesis derived an evaluation of cross-layer approach only on IEEE 802.11 platform. There is a good opportunity to develop and evaluate this approach on the latest platform of Zig-Bee which is also known as IEEE 802.15.4.

REFERENCES

[1] Jason Lester Hill, “System Architecture for Wireless Sensor Networks”, PhD Thesis, University of California at Berkeley, Spring 2003.

[2] H. Baldus, K. Klabunde, and G. Muesch, “Reliable setup of medical body-sensor networks”, In Proc. EWSN, Janurary 2004.

[3] P. Bauer, M. Sichitiu, R. Istepanian, and K. Premaratne, “The mobile patient: wireless distributed sensor networks for patient monitoring and care”, In Proc. IEEE EMBS International Conference on Information Technology Applications in Biomedicine, November 2000.

[4] N. F. Timmons and W. G. Scanlon, “Analysis of the performance of IEEE 802.15.4 for medical sensor body area networking”, In Proc. First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, October 2004.

[5] B. Charny, “ZDNet News: Wireless Research Senses the Future”, http://research.cens.ucla.edu, December 2002.

[6] R. Holman, J. Stanley, and T. Ozkan-Haller, “Applying video sensor networks to near shore environment monitoring,” IEEE Pervasive Computing, vol. 2, no. 4, pp. 14-21, October-December 2003.

[7] K. Martinez, J. K. Hart, and R. Ong., “Environmental sensor networks”, Computing Journal, vol. 37, no. 8, pp. 50-56, August 2004.

[8] R. Hills, “Sensing for danger”, Sci. Technol. Rep., July 2001.

[9] J. Nemeroff, L. Garcia, D. Hampel, and S. DiPierro, “Networked sensor communications”, In Proc. MILCOM, 2002.

[10] E. M. Petriu, T. E. Whalen, R. Abielmona, and A. Stewart, “Robotic sensor agents: a new generation of intelligent agents for complex environment monitoring”, IEEE Instrumentation and Measurement Magazine, vol. 7, no. 3, pp. 46-51, September 2004.

[11] T. T. Hsieh, “Using sensor networks for highway and traffic applications”, IEEE Potentials, vol. 23, no. 2, pp. 13-16, April-May 2004.

[12] C. E. Smith, C. A. Richards, S. A. Brandt, and N. P. Papanikolopoulos, “Visual tracking for intelligent vehicle-highway systems”, IEEE Transactions on Vehicular Technology, vol. 45, no. 4, pp. 744-759, November 1996.

[13] D. Estrin, R. Govinda, J. Heidemann, and S. Kumar, “Next century challenges: Scalable coordination in sensor networks”, In Proc. MobiCOM, 1999.

[14] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, “Space-time block codes from orthogonal designs”, IEEE Transactions on Information Theory, vol. 45, no. 5, pp. 1456-1467, 1999.

[15] S. Cui, A. J. Goldsmith, and A. Bahai, “Energy-efficiency of MIMO and cooperative MIMO techniques in sensor networks”, IEEE Journal on Selected Areas in Communications, vol. 22, no. 6, pp. 1089-1098, 2004.

[16] X. Li, “Energy efficient wireless sensor networks with transmission diversity”, IEE Electronics Letters, vol. 39, no. 24, pp. 1753-1755, 2003.

[17] X. Li, M. Chen, and W. Liu, “Application of STBC-encoded cooperative transmissions in wireless sensor networks”, IEEE Signal Processing Letters, vol. 12, no. 2, pp. 134-137, 2005.

[18] S. K. Jayaweera, “Energy analysis of MIMO techniques in wireless sensor networks”, in Proceedings of 38th Annual Conference on Information Sciences and Systems (CISS ’04), Princeton, NJ, USA, March 2004.

[19] Yong Yuan, Min Chen, and Taekyoung Kwon, “A Novel Cluster-Based Cooperative MIMO Scheme for Multi-Hop Wireless Sensor Networks”, EURASIP Journal on Wireless Communications and Networking, vol. 2006, Article ID. 72493, pp. 1-9, 2006.

[20] G. Pottie, and W. Kaiser, “Wireless Integrated Sensor Networks (WINS)”, Communications of the ACM, vol. 43, no. 5, pp. 51-58, May 2000.

[21] J. Hill, R. Szewcyk, A. Woo, D. Culler, S. Hollar, and K. Pister, “ System architecture directions for networked sensors ”, in Proceedings of 9th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLoS IX), Cambridge, MA, pp. 93-104, Nov. 2000.

[22] W. M. Merrill, K. Sohrabi, L. Girod, J. Elson, F. Newberg, and W. Kaiser,“ Open standard development platforms for distributed sensor networks ”, in Proceedings of SPIE-Unattended Ground Sensor Technologies and Applications IV, vol. 4743, Orlando, FL, pp. 327-337, Apr. 2002.

[23] J. N. Al-Karaki, and A. E. Kamal, “Routing techniques in wireless sensor networks: A survey”, IEEE Wireless Communications, vol. 11, no. 6, pp. 6-28, December 2004.

[24] R. Rajagopalan, and P. Varshney, “Data-aggregation techniques in sensor networks: A survey”, IEEE Communications and Surveys and Tutorials, vol. 8, no. 4, pp. 48-63, 2006.

[25] A. A. Abbasi, and M. Younis, “A survey on clustering algorithms for wireless sensor networks”, Computer Communications, vol. 30, nos. 14-15, pp. 2826-2841, October 2007.

[26] G. Gupta, and M. Younis, “Load-balanced clustering of wireless sensor networks”, in Proceedings of 2003 IEEE International Conference on Communications, Anchorage, AK, pp. 1848-1852, May 2003.

[27] S. Bandyopadhyay, and E. J. Coyle, “An energy efficient hierarchical clustering algorithm for wireless sensor networks”, in Proceedings of IEEE INFOCOM ’03, vol. 3, San Francisco, pp. 1713-1723, Mar.-Apr. 2003.

[28] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless micro-sensor networks”, IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660-670, Oct. 2002.

[29] O. Younis, and S. Fahmy, “ Heed: A hybrid, energy-efficient, distributed clustering approach for ad-hoc sensor networks”, IEEE Transactions on Mobile Computing, vol. 3, no. 4, pp. 366-379, Oct.-Dec. 2004.

[30] P. Wang, C. Li, and J. Zheng, “Distributed minimum-cost clustering protocol for underwater sensor networks (UWSNs)”, in Proceedings of 2007 IEEE International Conference on Communications (ICC ’07), Glasgow, UK, pp. 3510-3515, June 2007.

[31] S. Banerjee, and S. Khuller, “A clustering scheme for hierarchical control in multi-hop wireless networks”, in Proceedings of IEEE INFOCOM ’01, Anchorage, AK, pp. 1028-1037, Apr. 2001.

[32] Jun Zheng, and Abbas Jamalipour, Wireless Sensor Networks: A Networking Perspective, Wiley Publishers, Sept. 2006.

[33] Y. Li, S. S. Panwar, and S. Mao, “A wireless biosensor network using autonomously controlled animals”, IEEE Network, vol. 20, no. 3, pp. 6-11, May-June 2006.

[34] H. Nakayama, N. Ansari, A. Jamalipour, Y. Nemoto, and N. Kato, “Fault-resilient sensing in wireless sensor networks”, Computer Communications, vol. 30, no. 11, pp. 2375-2384, Sept. 2007.

[35] L. B. Ruiz, J. M. Nogueira, and A. A. F. Loureira, “Sensor network management”, SMART DUST: Sensor Network Applications, Architecture, and Design (edited), CRC Press, Boca Raton, FL, 2006.

[36] Ian F. Akyildiz et al., “A survey on sensor networks”, IEEE Communications Magazine, vol. 40, no. 8, pp. 102-114, Aug. 2002.

[37] C. Shen, C. Srisathapornphat, and C. Jaikaeo, “Sensor information networking architecture and applications”, IEEE Personal Communications, vol. 8, no. 4, pp. 52-59, Aug. 2001.

[38] W. Su, Ö. B. Akan, and E. Cayirci, “Communication protocols for sensor networks”, Wireless Sensor Networks (edited), Kluwer Academic Publishers, Norwell, MA, 2004.

[39] E. Shih, S. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, and A. Chandrakasan, “Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks”, in Proceedings of ACM Mobicom ’01, Rome, Italy, pp. 272-286, July 2001.

[40] A. Woo and D. Culler, “A transmission control scheme for media access in sensor networks”, in Proceedings of ACM Mobicom ’01, Rome, Italy, pp. 221-235, July 2001.

[41] J. M. Kahn, R. H. Katz, and K. S. J. Pister, “Next century challenges: Mobile networking for smart dust”, in Proceedings of ACM Mobicom ’99, Washington, DC, pp. 271-278, 1999.

[42] F. Zhao, and L. Guibas, Wireless Sensor Networks: An Information Processing Approach, Morgan Kaufmann Publishers, San Francisco, CA, 2004.

[43] K. Kredo and P. Mohapatra, “Medium access control in wireless sensor networks”, Computer Networks, Vol. 51, Issue 4, pp. 961-994, March 2007.

[44] R. C. Shah, H. M. Rabaey, Energy aware routing for low energy ad hoc sensor networks, IEEE WCNC, Orlando, FL, USA, March 2002.

[45] J. Chang and L. Tassiulas, Maximum lifetime routing in wireless sensor networks, IEEE/ACM Transactions on Networking, Vol. 12, Issue 4, pp. 609-619, August 2004.

[46] Pei Liu, Zhifeng Tao, Zinan Lin, Elza Erkip, and Shivendra Panwar, “Cooperative Wireless Communications: A Cross-Layer Approach”, IEEE Wireless Communications, pp. 84-92, August 2006.

[47] M. C. Vuran, O. B. Akan, and I. F. Akyildiz, “Spatio-Temporal¨ Correlation: Theory and Applications for Wireless Sensor Networks”, Computer Networks Journal (Elsevier), Vol. 45, No. 3, pp. 245-261, June 2004.

[48] O. B. Akan, Ian F. Akyildiz, “Event-to-Sink Reliable Transport in Wireless Sensor Networks”, IEE/ACM Transactions on Networking, vol. 13, no. 5, pp. 1003-1016, Oct. 2005.

[49] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, “Directed diffusion for wireless sensor networking”, IEEE/ACM Trans. on Networking, vol. 11, no. 1, pp. 2-16, February 2003.

[50] K. Seada, M. Zuniga, A. Helmy, and B. Krishnamachari, “Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks”, in Proc. ACM Sensys 04, November 2004.

[51] L. van Hoesel, T. Nieberg, J. Wu, and P. J. M. Havinga, “Prolonging the lifetime of wireless sensor networks by cross-layer interaction”, IEEE Wireless Communications, vol. 11, no. 6, pp. 78-86, December 2004.

[52] C. Bennila Thangammal, P. Rangarajan and J. Raja Paul Perinbam, “Maximization of Wireless Sensor Network’s Lifetime using Losningen Cross-layer Approach”, Asian Journal of Scientific Research, 5, pp. 133-142, May 2012.

[53] J. Hapoola, Z. Shelby, C. Pomalaza-Raez, and P. Mahonen, “Cross-layer energy analysis of multi-hop wireless sensor network”, in EWSN’ 05, pp. 33-44, 2005.

[54] Luca Catarinucci, Riccardo Colella, Giuseppe Del Fiore, Luca Mainetti, Vincenzo Mighali, Luigi Patrono, and Maria Laura Stefanizzi, “A Cross-Layer Approach to Minimize the Energy Consumption in Wireless Sensor Networks”, International Journal of Distributed Sensor Networks, vol. 2014, Jan. 2014.

[55] Yuebin Bai, Shujuan Liu, Mo Sha, Yang Lu, and Cong Xu, “An Energy Optimization Protocol Based on Cross-Layer for Wireless Sensor Network”, Journal of Communications, vol. 3, No. 6, pp. 27-34, Nov. 2008.

[56] Chia-Hung Tsai, Tsu-Wen Hsu, Meng-Shiuan Pan, and Yu-Chee Tseng, “Cross-Layer, Energy-efficient Design for Supporting Continuous Queries in Wireless Sensor Networks: A Quorum-Based Approach”, Wireless Personal Communications, vol. 51, Issue 3, pp. 411-426, July 2009.

[57] Mehmet C. Vuran, Ian F. Akyildiz, “XLP: A Cross-Layer Protocol for Efficient Communication in Wireless Sensor Networks”, IEEE Transactions on Mobile Computing, vol. 9, No. 11, pp. 1578-1591, Nov. 2010.

[58] Ian F. Akyildiz, Mehmet C. Vuran, and Ozgur B. Akan, “A Cross-Layer Protocol for Wireless Sensor Networks”, Proc. Conf. Information Sciences and Systems (CISS ’06), pp. 1102-1107, Mar. 2006.

[59] Wei Ye, John Heidemann, and Deborah Estrin, "Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks", IEEE/ACM Transactions on networking, vol. 12, No. 3, pp. 493-506, June 2004.

[60] LAN MAN Standards Committee of the IEEE Computer Society, Wireless LAN medium access control (MAC) and physical layer (PHY) specification. IEEE Std. 802.11, IEEE, 1997.

[61] Ken Tang, and Mario Gerla, "Mac Reliable Broadcast in Ad Hoc Networks", MILCOM 2001, McLean, Virginia, October 2001.

[62] Fred Stann, and John Heidemann, "RMST: Reliable Data Transport in Sensor Networks", Appearing in 1st IEEE International Workshop on Sensor Net Protocols and Applications (SNPA), Anchorage, Alaska, USA., pp. 1-11, May 11, 2003.

[63] Muhammad Mahtab Alam, Olivier Berder, Daniel Menard, Thomas Anger, and Olivier Sentieys, "A Hybrid Model for Accurate Energy Analysis of WSN Nodes", EURASIP Journal on Embedded Systems, vol. 2011, pp. 1-17, January 2011.

[64] Andrew S. Tanenbaum and David J. Wetherall, "Computer Networks" (5th edition), page. 303-304.

[65] TinyOS, http://www.tinyOS.net, 2010.

Appendix A

(a) API Code

```
$ns_ node-config -adhocRouting $opt(adhocRouting)
-llType $opt(ll)
-macType $opt(mac)
-ifqType $opt(ifq)
-ifqLen $opt(ifqlen)
-antType $opt(ant)
-propInstance [new $opt(prop)]
-phyType $opt(netif)
-channel [new $opt(chan)]
-topoInstance $topo
-wiredRouting OFF
-agentTrace ON
-routerTrace OFF
-macTrace OFF
```

(b) Creating Wireless Sensor Node

```
for { set j 0 } { $j < $opt(nn)} {incr j} {
set node_($j) [ $ns_ node ]
$node_($i) random-motion 0 ;  # disable random motion
}
```

(c) Creating Node attribute

```
$ns_ node-config -energyModel $energymodel \
                 -rxPower $p_rx \
                 -txPower $p_tx \
                 -initialEnergy $initialenergy
```

Appendix B

Journal Paper:

1. Md. Imran Hossain Jony and Mohammad Rakibul Islam, "ENERGY EFFICIENT CROSS-LAYER APPROACH FOR WIRELESS SENSOR NETWORKS", International Journal of Computer Applications (IJCA), vol. 127, no. 10, pp. 31-37, October 2015.