ENERGY EFFICIENT MAC PROTOCOLS

FOR

WIRELESS SENSOR NETWORKS

Dissertation submitted to the Jawaharlal Nehru University
in Partial fulfillment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY

In

COMPUTER SCIENCE AND TECHNOLOGY

By
SUJATA JATAV

UNDER THE SUPERVISION OF
Mr. SUSHIL KUMAR

SCHOOL OF COMPUTER AND SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110067, INDIA
JULY, 2009

Dedicated to

My Parents and Sister

जवाहरलाल नेहरू विश्वविद्यालय

SCHOOL OF COMPUTER & SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110067, INDIA

CERTIFICATE

This is to certify that the dissertation entitled **"ENERGY EFFICIENT MAC PROTOCOLS FOR WIRELESS SENSOR NETWORKS"** being submitted by **Miss. Sujata Jatav** to the **School of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi,** in partial fulfillment of the requirements for the award of the degree of **Master of Technology** in **Computer Science and Technology,** is a record of bonafide work carried out by her under the supervision of **Mr. Sushil Kumar, Assistant Professor**.

This work has not been submitted in part or full to any university or institution for the award of any degree or diploma.

21-07-09

Mr. Sushil Kumar
(Supervisor)
School of Computer and System Sciences
Jawaharlal Nehru University
New Delhi-110067

Dean
S. Minz
School of Computer and System Sciences
Jawaharlal Nehru University
New Delhi-110067

जवाहरलाल नेहरू विश्वविद्यालय

SCHOOL OF COMPUTER AND SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110067, INDIA

DECLARATION

This is to certify that the dissertation entitled **"ENERGY EFFICIENT MAC PROTOCOLS FOR WIRELESS SENSOR NETWORKS"** is being submitted to the **School of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi,** in partial fulfillment of the requirements for the award of the degree of **Master of Technology** in **Computer Science and Technology,** is a record of bonafide work carried out by me.

The matter embodied in the dissertation has not been submitted in part or full to any university or institution for the award of any degree or diploma.

SUJATA JATAV
M. Tech (2007-2009)
SC&SS, JNU,
New Delhi-110067

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my supervisor, **Mr. Sushil Kumar** for his encouragement, enthusiasm and direction provided to me during the entire duration of the proposed work. Without his valuable thoughts, recommendation, and patience, I would have never been able to complete my work.

I wish to thank all my fellow students for their help and support. I gratefully acknowledge the unselfish help given to me by them who inspired me greatly through many interesting discussions, support, and feedbacks.

Also, I am thankful to my parents for their dogged persistence and motivations in getting me complete my degree.

Finally, I want to thank the almighty whose choicest blessings have sailed me through this research effort.

Sujata Jatav

ABSTRACT

Because of the difficulty in recharging or replacing the batteries of each node in a Wireless Sensor Network, the *energy efficiency* of the system is a major issue in the area of network design. Other critical parameters such as delay, adaptability to traffic conditions, scalability, system fairness, throughput and bandwidth utilization are mostly dealt as secondary objectives. Some sensor network applications adopt IEEE 802.11-like MAC protocol, which is however, not a good solution for sensor network applications because it suffers from energy inefficiency problem. The adaptive Sensor-MAC (S-MAC) proposes enhanced schemes such as periodic sleep and overhearing avoidance to provide a better choice for different sensor network applications.

In this dissertation we propose an energy efficient MAC (EE-MAC) protocol, which is based on adaptive S-MAC with added transmission power control techniques. The main contribution of our work is to introduce a controlled power transmission of RTS, CTS, DATA and ACK frames according to the adaptive S-MAC protocol.

We simulate our proposed protocol i.e., EE-MAC protocol using ns-2.33 simulator for two parameters *energy consumption* and *throughput,* for determining the behavior of the proposed protocol. The simulation results show that our proposed EE-MAC protocol performs better than adaptive S-MAC protocol in terms of energy consumption and throughput.

CONTENTS

List of Figures and Tables

LIST OF FIGURES

LIST OF TABLES

Chapter 1

WIRELESS SENSOR NETWORKS

1.1 INTRODUCTION

Wireless sensor networks (WSNs) have gained world-wide attention in recent years due to the current advancements in the area of digital circuitry (i.e., micro-electro-mechanical systems (MEMS) technology) that has facilitated the development of smart sensors. The sensor networks are highly distributed networks of small, lightweight wireless nodes, deployed in large numbers to sense, measure, and gather information about some physical parameters such as temperature, pressure, or relative humidity, and characteristics of objects and their motion from the environment and based on some local decision process; they can transmit the sensed data to the user.

Each node of the sensor network consists of following four subsystems:

- **Sensor Subsystem:** It senses the environment.
- **Processing Subsystem:** It performs local computations on the sensed data.
- **Communication Subsystem:** It is responsible for message exchange between neighboring sensor nodes.
- **Power Subsystem:** It provides power for accomplishing above tasks.

Unlike, traditional networks, a WSN have its own design and resource constraints. Design constraints are application dependent and are based on the monitoring environment. Resource constraints include a limited amount of energy, short communication range, low bandwidth, limited processing and storage in each node. The distinct features that makes wireless sensor network different from traditional wireless ad hoc networks are outlined below:

- The number of sensor nodes in a sensor network can be of several orders of magnitude higher than the nodes in an ad hoc network.
- Sensor nodes are usually densely deployed.
- Sensor nodes are more prone to failures.
- Network topology changes very frequently.
- Sensor nodes mainly use broadcast communication pattern whereas most ad hoc networks are based on point-to-point communications.
- Sensor nodes are limited in power, computational capacities, and memory.

1.2 APPLICATIONS

WSN applications [1] can be classified into two categories: *monitoring and tracking* (*fig. 1.1*). Monitoring applications include battlefield surveillance and monitoring under military applications, indoor/outdoor environmental monitoring such as forest fire and flood detection, and habitat exploration of animals, health and wellness monitoring such as patient diagnosis, and remote monitoring of physiological data, power monitoring, inventory location monitoring, seismic and structural monitoring.

Tracking applications include tracking objects, animals, humans and vehicles. Sensors will soon get their way into a number of military applications at home and in industries. Sensor nodes can be built into a variety of applications at home, such as ovens, refrigerators, and vacuum cleaners, which enable them to interact with each other and be remote-controlled. The home can provide a "smart environment" which adapts itself according to the user's tastes for example; the music and mood in the room can be automatically set according to user's preferences, similar control is useful in office building too, where the airflow and temperature of different parts of the building can be automatically controlled.

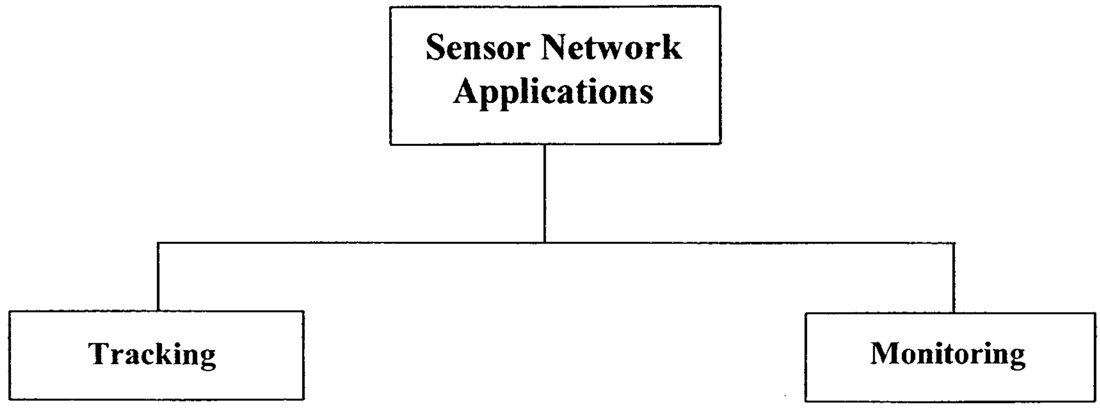

Fig. 1.1: Classification of Sensor Network Applications

So, we can say that the applications of sensor network are endless, limited only by the human imagination.

1.3 CHALLENGES

Sensor networks are a special category of ad hoc wireless sensor networks that are used to provide a wireless communication infrastructure among the sensors deployed in a specific application domain. Sensor networks present certain design challenges/issues due to the following reasons [2]:

- Sensor nodes usually do not require any kind of human intercession because they are randomly deployed. Therefore the sensor network must have self organizing capabilities. Hence the setup and maintenance of the sensor network should be entirely autonomous.

- Sensor nodes should be able to coordinate their actions with each other, so that the ordering of detected events can be performed without any uncertainty.

- Since sensor nodes use batteries for their operation which are difficult to recharge or replace when consumed (remember that often sensor nodes are deployed in remote and hostile environments), it is critical to design algorithms and protocols in such a way to utilize minimal energy.

- The network should be scalable and flexible for the enlargement of the network's size. The communication protocols must be designed in such a way that deploying more nodes in the network does not affect routing and clustering. Rather, the protocols must be adapted to the new topology and behave as expected i.e., the network must preserve its stability. Furthermore, introducing more nodes into the network means that additional communication messages will be exchanged, so that these nodes are integrated into the existing network. This must be done in a way that a minimum number of messages need to be exchanged among the sensor nodes, and thus battery is not wasted unreasonably.

- The communication over sensor networks must be made secure; because Security solutions are constrained when applying them to sensor networks. For example, cryptography requires complex processing to provide encryption to the transmitted data. Secure routing; secure discovery and verification of location, key establishment and trust setup, and attacks against sensor nodes are some of the many issues that need to be addressed in a security context.

- Many applications which track an object require the knowledge of the exact or approximate physical location of a sensor node in order to link sensed data with the objects under investigation. Location discovery protocols must be designed in such a way that minimum information needs to be exchanged among nodes to discover their location. Since sensor nodes are energy constrained, solutions are energy constrained, solutions like GPS are not recommended.

- If a sensor node fails due to a technical problem or consumption of its battery, the rest of the network must continue its operation without a problem. Researchers must design adaptable protocols so that new links are established in case of node failure or link congestion. Furthermore, appropriate mechanisms should be designed to update topology information immediately after the environment changes so as to minimize unnecessary power consumption.

- Sensor networks must support real-time communication through the prerequisite of guarantees on maximum delay, minimum bandwidth, and other QoS parameters.

1.4 PROBLEM STATEMENT

The S-MAC protocol saves some amount of energy but suffers from the latency problem. For reducing latency, adaptive S-MAC protocol is introduced which saves almost equal energy than S-MAC. For further enhancement in energy conservation a new MAC layer protocol i.e., energy efficient MAC protocol is proposed, which employs transmission power control techniques in the adaptive S-MAC protocol. The proposed work sets the following objectives to meet the goal of the problem:

1 Modification of the energy model
2 Implementation of the energy efficient MAC (EE-MAC) protocol
3 Simulation and analysis of the performance of the proposed protocol.
4 Comparison of the results of EE-MAC with that of adaptive-SMAC.

1.5 ORGANISATION OF THE DISSERTATION

Chapter 1 gives a brief introduction of Wireless Sensor Networks, its applications and various challenges it presents as compared to traditional wireless networks. This is followed by the problem statement and objectives to be met in this work.

Chapter 2 describes the fundamentals and taxonomy of MAC protocols through various improvements. Major advantages and disadvantages are also pointed out.

Chapter 3 reviews the previous work done related to our proposed work.

Chapter 4 discusses the modifications to be made in the existing adaptive S-MAC protocol as suggested in the objective.

Chapter 5 presents the implementation details of proposed protocol in ns-2 simulator. Simulation details and results of experiments are also discussed in this chapter.

Chapter 6 concludes the work presented in the dissertation and the scope of further extension of this work.

MEDIUM ACCESS CONTROL

2.1 INTRODUCTION

Communication among wireless sensor nodes is usually achieved by means of a unique radio channel. It is the characteristic of the channel that only a single node can transmit a message at any given time, because a radio channel cannot be accessed simultaneously by two or more nodes that are in a radio intervention range — neighboring nodes may cause conflict at some nodes if transmitting at the same time on the same channel.

In wireless sensor networks, controlling access to the channel is usually known as *Medium access control (MAC)*. Therefore, the objective of the MAC protocol is to regulate access to the shared wireless medium to ensure fair and efficient sharing and communication resources among the nodes, such that the network lifetime can be maximized. MAC protocols in sensor networks must create a network infrastructure to establish communication links among the thousands of randomly scattered sensors.

2.2 FUNDAMENTALS OF MAC PROTOCOLS FOR WSNs

For maximizing the network lifetime, the proposed MAC protocol must be energy efficient. This can be achieved by reducing the potential energy wastes. Types of communication patterns present the behavior of the sensor network traffic that has to be handled by a given MAC protocol; the properties that must be possessed by a MAC protocol to suit a sensor network environment are outlined below.

2.2.1 COMMUNICATION PATTERNS

Three types of communication patterns are defined in wireless sensor networks [3], *broadcast, converge-cast, and local gossip. Broadcast* type of communication pattern is generally used by a base station (sink) to transmit some information to all the sensor nodes of the network. Broadcasted information may include queries of sensor query-processing architectures, program updates for sensor nodes, control packets for the whole system. In some scenarios, the sensors that detect an intruder need to communicate with each other locally. This kind of communication pattern is called *local gossip*, where a sensor node sends a message to its neighboring nodes within a range. The sensors that detect the intruder, then, need to send what they perceive to the information center. That communication pattern is called *converge-cast*, where a group of sensors communicate to a specific sensor. The destination node could be a cluster-head, data fusion center, base station. In protocols that include clustering, cluster-heads communicate with their members and thus the intended receivers may not be the neighbors of the cluster-head, but just a subset of the neighbors. To serve for such scenarios, we define a fourth type of communication pattern; *multicast*, where a sensor sends a message to a specific subset of sensors.

2.2.2 PERFORMANCE REQUIREMENTS

Following is a brief discussion of the performance metrics [4]:

- **DELAY:** *Delay* refers to the amount of time spent by a data packet in the MAC layer before it is transmitted. In sensor networks, the importance of delay depends on the application, network traffic load, and the design choices of the MAC protocol. For time-critical applications, the MAC protocol is required to support delay-bound guarantees through careful message scheduling both locally, within a communicating node and globally among all nodes in the network.

- **SCALABILITY:** *Scalability* refers to the ability of a communication system to meet its performance characteristics regardless of the size of the network or the number of competing nodes. In WSNs the number of sensor nodes may be very large. Some nodes may die over time; some new nodes may join later; some nodes may move to different locations. A good MAC protocol should accommodate such changes gracefully. Scalability and adaptability to changes in size, density and topology are important attributes, because sensor networks are deployed in an ad hoc manner and often operate in uncertain environments.

- **THROUGHPUT:** *Throughput* (often measured in bits or bytes per second) refers to the amount of data successfully transferred from a sender node to a receiver node in a given amount of time. An important objective of a MAC protocol is to maximize the channel throughput while minimizing message delay. A related attribute is *good-put,* which refers to the throughput measured only by data received by the receiver without any errors.

- **BANDWIDTH UTILIZATION:** It reflects how well the entire bandwidth of the channel is utilized in communication. It is also referred to as channel utilization or channel capacity. It is an important issue in cell phone systems or wireless LANs, since the bandwidth is the most valuable resource in such systems and service providers want to accommodate as many users as possible.

- **FAIRNESS:** It reflects the ability of different users, nodes or applications to share the channel equally. It is an important attribute, since each user desires an equal opportunity to send or receive data for their own applications. However, at any particular time, one node may have dramatically more data to send than some other nodes. Thus rather than treating each node equally, success is measured by the performance of the application as a whole and per-node or per-user fairness becomes less important.

- **ENERGY EFFICIENCY:** A sensor node is equipped with one or more integrated sensors, embedded processors with limited capability, and short-range radio communication ability. These sensor nodes are powered using batteries with small capacity. Unlike in standard wireless networks, wireless sensor nodes are often deployed in unattended environments, making it difficult to change their batteries. Furthermore, recharging sensor batteries by energy scavenging is complicated and volatile. These severe constraints have a direct impact on the lifetime of a sensor node. As a result, energy conservation becomes of paramount importance in WSNs to prolong the lifetime of sensor nodes.

2.2.3 SOURCES OF ENERGY WASTE

The major sources of energy waste [5] in a MAC protocol are:

- **COLLISION:** It occurs when two or more sensor nodes attempt to transmit simultaneously and the retransmissions of these packets due to collision are required, which increases the energy consumption.

- **CONTROL PACKET OVERHEAD:** Sending and receiving control packets consumes energy too, and less useful data packets are transmitted.

- **IDLE LISTENING:** Listening to an idle channel in order to receive possible traffic.

- **OVERHEARING:** A node listens for a message that is sent to another node which is not the destination.

- **OVEREMITTING:** It is caused by the transmission of a message when the destination node is not ready to receive it.

In short, the above attributes reflects the characteristics of a MAC protocol. For WSNs, the primary factors are effective collision avoidance, energy efficiency, scalability and adaptability to densities and number of nodes. Other attributes are normally secondary.

Therefore to meet the requirement of wireless sensor network deployments and monitoring, we translate them to a set of goals for the media access protocol. Our design goals for a MAC protocol for WSN applications [2] are:

- Low power operations
- Effective collisions avoidance
- Simple implementation
- Efficient channel utilization
- Scalable to large number of nodes

2.3 TAXANOMY OF MAC PROTOCOLS

Many surveys and introductory papers on several MAC protocols for wireless sensor networks are available in the literature [4, 5, 6, 7]. To understand the differences and commonalties of different WSN-specific MAC protocols, we will classify (*fig. 2.1*) them according to how nodes organize energy efficient access to the shared radio channel [6].

2.3.1 FIXED ALLOCATION BASED ACCESS

Fixed allocation based MAC protocols share the common channel through a predetermined fixed amount of channel resources. Each node uses its allocated resources without competing with other nodes. These protocols provide a bounded delay for each node and are suitable for sensor networks that continuously monitor and generate deterministic data traffic.

Two types of fixed allocation based access mechanism are there:

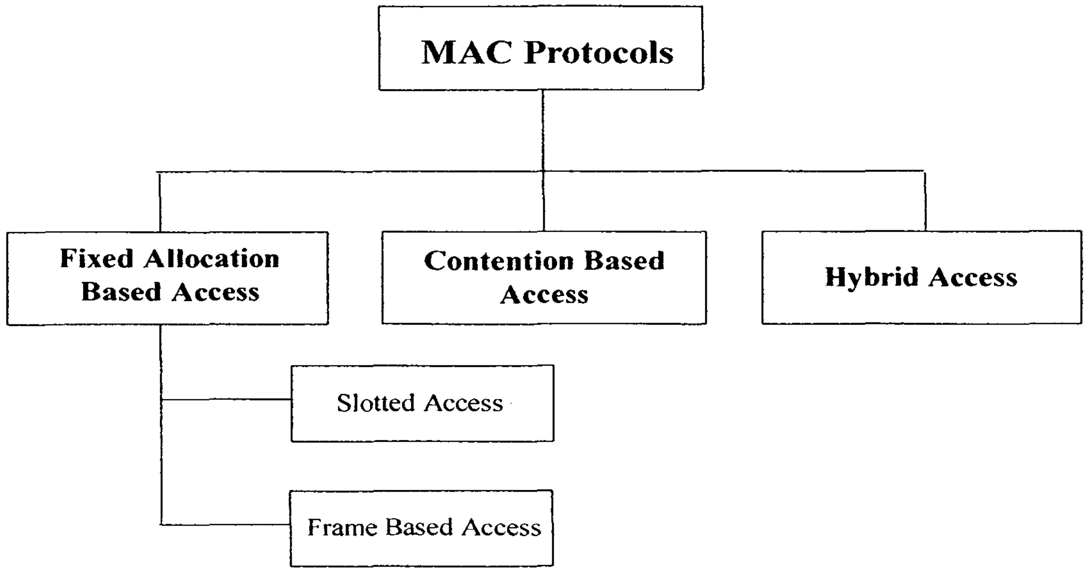

Fig. 2.1: Taxonomy of MAC Protocols

2.3.1.1 SLOTTED ACCESS

Slotted access requires nodes to synchronize on some common time of references such that they can wake up collectively at the beginning of each slot and exchange messages when available; then go back to sleep for the remainder of the slot. One of the most important protocols of this category is S-MAC [8], which is a low power RTS-CTS based MAC protocol; introducing a low duty cycle operation in a multi-hop wireless sensor networks, where the nodes spend most of their time in sleep mode to reduce energy consumption. Detailed study of this protocol is given in *chapter 3*. The static sleep-listen periods of S-MAC result in high latency and lower throughput, so for overcoming these problems Timeout MAC (T-MAC) [28] was introduced to improve the design of S-MAC by listening to the channel for only a short period of time after the synchronization phase, and if no data is received during this window, the node returns to sleep mode. If data is received, the node remains awake until no further data is received or the wake up period ends. T-MAC is aggressive in shutting down the radio, leaving messages queued for the next slot, which effectively increases latency and reduces throughput.

2.3.1.2 FRAME BASED ACCESS

The access mechanism time is divided into frames containing a fixed number of slots. This access mechanism differs in how slots are assigned to nodes; hence collision does not occur and idle listening and overhearing can be reduced. For example, in Traffic Adaptive Medium Access (TRAMA) [29] protocol nodes are synchronized and time is divided into two cycles. In 1st cycle during, random access periods nodes exchange neighbor's information to learn about their two-hop neighbors and in 2nd cycle, scheduled access periods nodes exchange schedules using schedule exchange protocol. Using adaptive election protocol, transmitter, receiver and stand-by nodes are elected for each transmission slot and nodes without traffic are removed from election.

Another MAC protocol called Lightweight Medium Access Control (LMAC) [30] is proposed to reduce the radio state transitions and protocol overhead. LMAC uses a distributed time slot election mechanism in which each node is able to send out a collision-free message since it owns an exclusive time slot in a two-hop neighborhood. However, all the nodes have to wake up at every time slot, in order to not miss the incoming messages. The main drawback of LMAC is the fixed length of the frame, which has to be specified prior to deployment, and may be problematic. To solve this problem an Adaptive Information-centric LMAC (AI-LMAC) [31] is proposed, so that the slot assignment can be more adapted to the actual traffic needs.

2.3.2 CONTENTION BASED ACCESS

In this access mechanism also known as random access, nodes do not organize time and contend access to the radio channel; hence there is a greater possibility of collisions and lots of energy is wasted due to idle listening. Therefore random access mechanism is not suitable for real time/delay sensitive traffic. One of the most popular contention-based MAC protocol is Berkeley MAC (B-MAC) [26], which is a low complexity and low power MAC protocol whose goal is to provide a few core functionalities and an energy efficient mechanism for channel access. First, B-MAC implements basic channel access control features: a back-off scheme, an accurate channel estimation facility and optional acknowledgements. Second, to achieve a low duty cycle B-MAC uses an asynchronous sleep/wake scheme based on periodic listening called Low Power Listening (LPL), also called preamble sampling. When a sender has data to send, it transmits a preamble that is at least as long as the sleep period of the receiver. The receiver will wake up, detect the preamble, and stay awake to receive the data. This allows low power communication without the need for explicit synchronization between the nodes. The receiver only wakes for a short time to sample the medium, thereby limiting idle listening.

Another low power listening protocol i.e., X-MAC [27] is introduced, which employs a *short preamble* to further reduce energy consumption and to reduce latency. The first idea is to embed address information of the target in the preamble so that non-target receivers can quickly go back to sleep, thereby addressing the overhearing problem. The second idea is to use a *strobed preamble* to allow the target receiver to interrupt the long preamble as soon as it wakes up and determines that it is the target receiver. This short strobed preamble approach reduces the time and energy wasted during waiting for the entire preamble to complete.

2.3.3 HYBRID ACCESS

If above mentioned access mechanisms are strictly followed by any protocol; they strictly provide tighter control of who is communicating when, but at the cost of being less flexible to accommodate changing conditions. Therefore, several hybrid protocols have been developed in recent times which were aim to combine the strengths of the above mentioned organizations while offsetting their weaknesses to better address the special requirements of wireless sensor network MAC protocols. The greatest advantage of the hybrid MAC protocols comes from its easy and rapid adaptability to traffic conditions, which can save a large amount of energy.

For these reasons recent researches in MAC area are mainly focused on the combination of their access mechanism (e.g., CSMA/TDMA) rather than stick to one

One of the most interesting hybrid protocols is Zebra MAC (Z-MAC) [32] which combines contention based access and schedule based access mechanism by dynamically switching between the two, thus widening the scope of applications that can be supported by a WSNs. Z-MAC starts with an initial setup phase in which neighbor discovery process builds a list of two-hop neighbors for each node. Then a distributed slot assignment algorithm is applied to ensure that no transmission from a node to any of its one-hop neighbor interferes with any transmission from its two-hop neighbors. Next, nodes must contend for access when wanting to send a message. By default a node may contend for any slot, but an owner gets priority by contending first. Another way of combining CSMA with TDMA is followed by Pattern MAC (PMAC) [33], which schedules (schedules are decided based on a node's own traffic and that of its neighbors) receiving slots so that a node can either be sleep or wake up in its own slot to check for incoming traffic, instead of, checking in every slot.

Chapter 3

RELATED WORK

A concise survey of recently proposed several MAC protocols for wireless sensor networks can be found in [8]. In this section we first give a brief idea of the IEEE 802.11, the SMAC, and the transmission power control techniques as they present the basic building blocks of our proposed protocol i.e., EE-MAC.

3.1 IEEE 802.11

IEEE 802.11 [9] is the standard MAC layer protocol which is proposed for wireless LANs. This scheme is a contention-based protocol which employs RTS/CTS control packets in order to reduce collision which may occur due to hidden and exposed node problem [2]. Beside, this, it uses both physical and virtual carrier sense mechanism to indicate the free channel. Now we define the terms *transmission range* and *carrier sensing zone* which are used for clear understanding of the proposed protocol.

- *Transmission range:* It is the distance between sender node and its neighboring nodes that can receive and correctly decode the packets coming from the sender node. The level of power used in transmission and radio propagation properties (i.e., attenuation) determines the transmission range.

- *Carrier sensing zone:* When a node is within the carrier sensing zone, it can sense the signal but cannot decode it correctly. It does not include transmission range because nodes in transmission range can definitely sense the transmission as well as decode it correctly.

Fig. 3.1 shows the transmission range (it may not be circular in reality) and carrier sensing zone for node M. When node M transmits a packet, node R and S can receive and decode it correctly since they are in transmission range. However, node P and Q only sense the signal and cannot decode it correctly because they are in the carrier sensing zone.

However, the IEEE 802.11 technique suffers from energy inefficiency problem. This problem is solved by S-MAC [8][10] protocol which combines the features of both Contention based as well as TDMA protocols which have been considered as the best provider for energy saving. The TDMA approach schedule transmission times of neighboring nodes to occur at different times. However, the major disadvantage of this technique is that it is not well adaptable to topology changes.

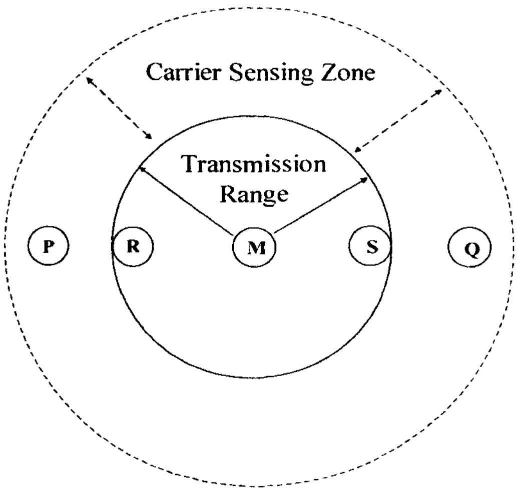

Fig. 3.1: Nodes in the transmission range can receive and decode packets correctly, whereas nodes in the carrier sensing zone can sense a transmission, but cannot decode it correctly.

3.2 S-MAC PROTOCOL

The S-MAC [8] protocol is considered to be the first standard MAC protocol which is proposed for WSNs in order to reduce energy consumption by all the sources of energy wastage i.e., idle listening, collision, control overhead, overhearing, and overemitting that were discussed earlier. In exchange the protocol incurs some performance reduction in per-hop fairness and latency. The S-MAC uses multiple techniques to reduce energy consumption, control overhead, and latency, in order to improve application-level performance.

1. Periodic listen and sleep: In S-MAC the energy lost caused by idle listening is reduced by letting nodes to go to in sleep mode periodically (as shown in below *fig. 3.2*), instead of constantly listening to an idle channel. When a node is in sleep mode its radio will be turned off, thus conserving energy.

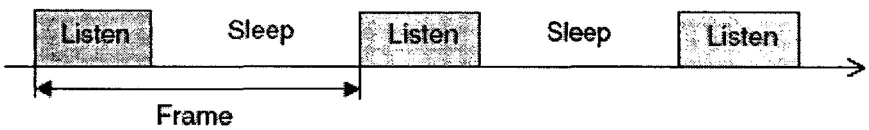

Fig. 3.2: Periodic listen and sleep

We call a complete cycle of the listen and sleep intervals, a frame. So,

$$T_{Frame} = T_{Listen} + T_{Sleep}$$

2. Synchronizing schedules: Synchronization between neighboring nodes is accomplished by means of broadcasting SYNC packets which are exchanged periodically between neighboring nodes. In this way, nodes make a cluster sharing in the same schedule. This mechanism allows S-MAC to reduce control packets overhead. Each node in the network is free to choose its own or other nodes schedule by following the steps mentioned below:

- Listen to sufficient amount of time to hear the existing schedule, if not, then, the node chooses its own schedule and broadcasts to its neighbors.
- If a schedule is reserved, it will be followed by the node, then broadcast after a random amount of time to avoid collisions between SYNC packets issued from nodes which select the same schedule.
- Two schedules will be followed if a new schedule is received, and another is followed and broadcasted.

3. Overhearing avoidance: radios are turned off while the shared media is used for transmission between other nodes. However, when a node hears a RTS or a CTS packet and doesn't present the data recipient, it goes to sleep until the current transmission ends.

4. Message passing: This scheme is used to reduce contention latency for sensor network applications requiring store-and-forward processing as data are moved through the network. In S-MAC, the RTS packet allocates the shared medium for transmitting the entire message instead of reserving the medium just for the first fragment like in IEEE 802.11. In later, each fragment and its ACK play the role of RTS/CTS sequence for the next fragment.

A closer look at the periodic listen and sleep scheme reveals that a message may incur increased latency as it is stored and forwarded between adjacent network nodes. To address this shortcoming, the protocol uses a technique referred to as adaptive listening [10]. The basic idea is to let the node which overhears its neighbor's transmissions (ideally only RTS or CTS) wake up for a short period of time at the end of the transmission. In this way, if the node is the next-hop node, its neighbor is able to immediately pass the data to it instead of waiting for its scheduled listen time. If the node does not receive anything during the adaptive listening, it will go back to sleep until its next scheduled listen time.

3.3 TRANSMISSION POWER CONTROL (TPC)

To further reduce the energy consumption TPC techniques [14, 15, 16, 17, 19] need to be applied. It is important in WSN for at least two reasons [18]:

1. It can affect the battery life of the nodes.
2. It can affect the traffic carrying capacity of the network.

To clearly understand the above points, let us consider the *fig. 3.3*. There is no need for node A to send a packet to the neighboring node B with 20mW power because node B is within the range of node A. Now suppose another node C also wishes to broadcasts a packet to node D at the same time with 2mW power, then, both the transmissions can be successfully received simultaneously, since neither node B is in the range of node C nor node D is in the range of node A. However, if node A broadcasts at 20mW, then it will interferes with node D's reception from node C, and hence only one packet is transmitted from node A to node B. Thus power control can enhance the traffic carrying capacity.

TPC techniques improve the performance of the network in several aspects [22]:

1. It improves the reliability of a link by increasing the transmission power upon detecting that the link reliability is below a certain threshold.
2. Only nodes which share the same space will contend to access the medium thereby decreasing the amount of collisions, latency, hidden and exposed terminal in the network while enhancing network utilization.
3. By using a higher transmission power, the physical layer increase the bandwidth in the presence of heavy traffic and by decreasing, it maximizes the energy savings.

A power control mechanism that can be incorporated into the IEEE 802.11 RTS-CTS handshake is proposed in [14] to perform the handshake at the maximum initial power

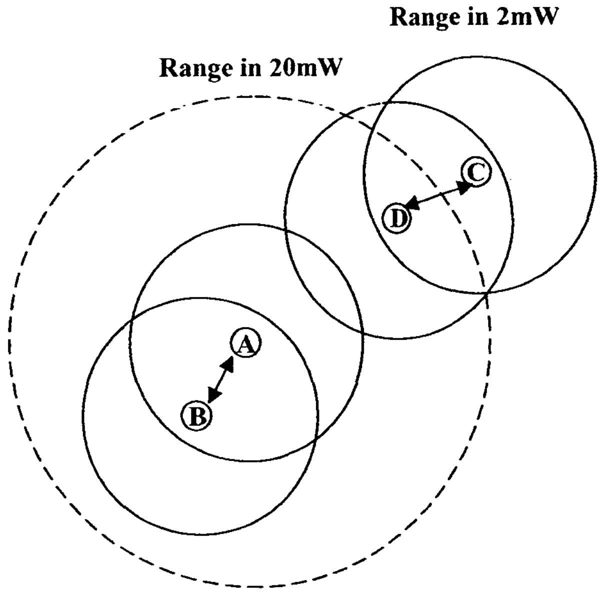

Fig. 3.3: The need for power control

level to avoid packet collision from the interfering nodes. However DATA and ACK packet may be sent at a lower power level. We refer to this scheme as the BASIC power control MAC protocol. BASIC scheme consumes less energy than IEEE 802.11 MAC protocol. Jung et al. [19] proposes a power control MAC (PCM) protocol that uses the BASIC scheme but periodically transmits a DATA frame with the maximum transmit power. PCM inherits IEEE 802.11's shortcomings except that it consumes less energy. A power control [15] protocol is also similar to the BASIC scheme. It maintains tables for a minimum transmit power necessary to communicate with neighboring nodes. However, different power levels among the nodes result in asymmetric links, which cause collision.

Several other TPC techniques [23, 24] are available for the purpose of topology control. It is also used to establish energy efficient spanning trees for multicasting and broadcasting [25].

Uptil now, the primary goal for wireless sensor networks in general and mainly in MAC has been energy efficiency. Although TPC techniques present an effective mechanism to reduce energy consumption they are still not implemented in any existing MAC protocol because of highly imprecise nature of transceiver readings and limited resources of sensor nodes. This enables us to propose a new *Energy Efficiency MAC (EE-MAC)* protocol which uses the techniques employed in adaptive SMAC with added transmission power control functionality to combine the strengths while offsetting their weaknesses.

Chapter 4

PROPOSED ENERGY EFFICIENT MAC (EE-MAC) PROTOCOL

4.1 DESIGN OVERVIEW

It is based on adaptive S-MAC protocol [10] with transmission power control mechanism. It consist of following steps-

1. Firstly node A in the network broadcast SYNC packet with maximum transmission power i.e. P_{TX_MAX} in order to synchronize its schedule for reducing control overhead. After choosing its schedule either by receiving SYNC-REC packet from one of its neighbors or according to conditions [21]. However, here we assume that each time the node selects its own schedule.
2. When synchronization is done, we divide the transmission power to different levels i.e., from P_{TX_MIN} to P_{TX_MAX}; node A sends an RTS packet to node B, using maximum power.

The power received by destination node separated from sender node by distance *d* is calculated by two-ray ground or two-path model [2] which is as follows:

$$P_r = P_t G_t G_r h_t^2 h_r^2 / d^4 L$$

Where,

P_t = Transmission power

P_r = Receiving power

G_t = Antenna gains of transmitter

G_r = Antenna gains of receiver

h_t = Height of transmitting antenna

h_r = Height of receiving antenna

L = System loss

The minimum transmission power P_{TX_MIN} [20] that can be received by a node must satisfy the equation (1) and (2):

$$P_{TX\_MAX}/P_r = P_{TX\_MIN}/P_{r\_threshold} \text{ -- } (1)$$

$$P_{TX\_MIN} = (P_{TX\_MAX} * P_{r\_threshold})/P_r \text{ ------------------------------------ } (2)$$

Where,

$P_{r\_threshold}$ is the minimum necessary received signal strength,

P_{TX_MAX} is the maximum power transmitted, and

P_{TX_MIN} is the minimum power transmitted.

3. The receiver upon receiving the RTS packet calculates the minimum required transmission power level of it i.e., P_{TX_MIN} by using equation (2).The receiver sends the CTS frame with maximum power along with P_{TX_MIN}. Thus nodes M and N that are located in the carrier sensing zones of nodes A and B defer their transmission for a sufficient period of time i.e., EIFS so as not to interfere with the RTS-CTS exchange as illustrated in *fig. 4.1*.

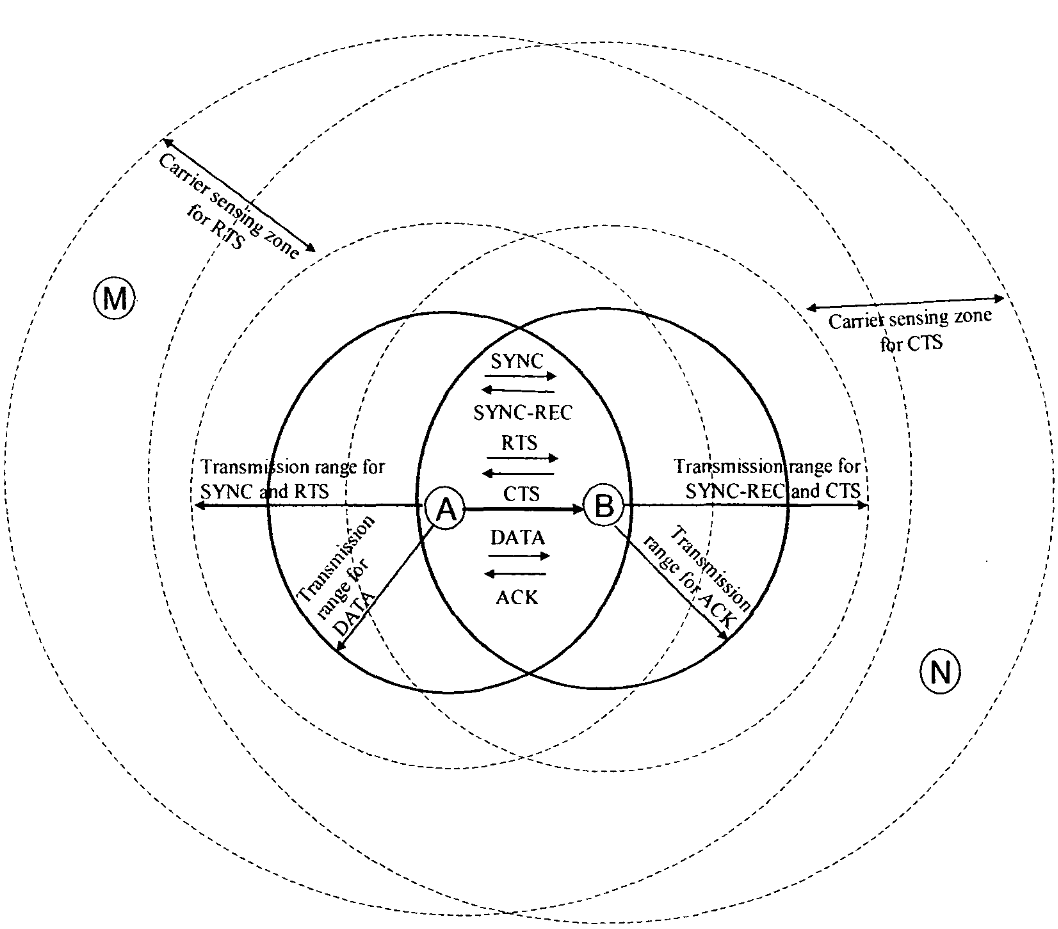

Fig. 4.1: Packet Transmission in EE-MAC Protocol

4. After the RTS-CTS handshake the sender node has the information of the required minimum transmission power. Therefore the DATA and ACK frames are transmitted at minimum required power level.

When source and destination nodes transmit RTS and CTS packets, nodes in the transmission range correctly receive these packets and set their NAVs for the duration of the whole packet transmission. However, nodes in the carrier sensing zone only sense the signal and cannot decode it correctly, so these nodes set their NAVs for EIFS duration (when they sense the channel changing state from busy to idle). The purpose of EIFS is to protect an ACK frame at the source node. The NAV setting and the power level changes for SYNC–SYNC-REC–RTS–CTS–DATA–ACK transmissions are depicted in *fig. 4.2*.

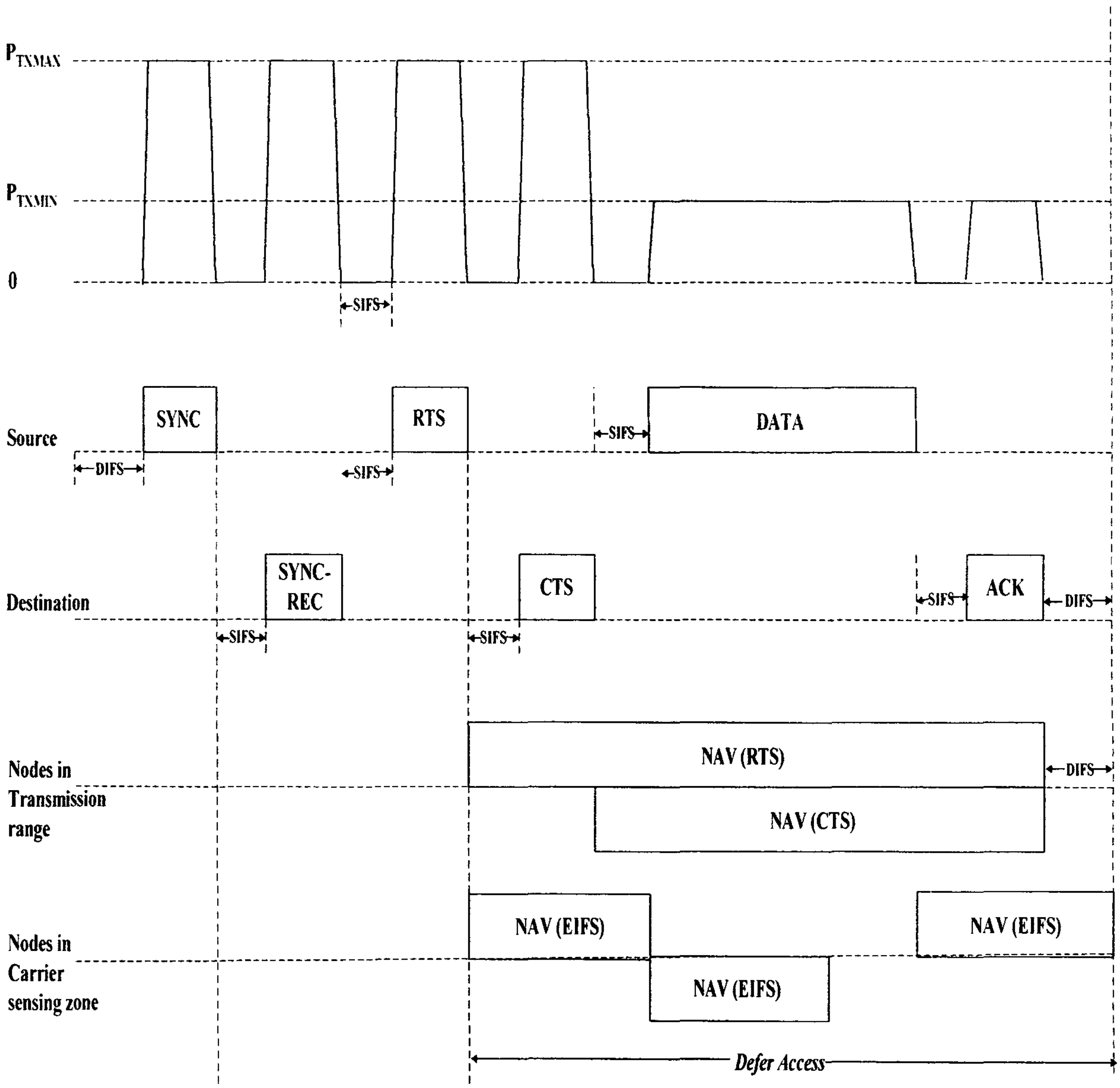

Fig. 4.2: NAV setting and Transmission Power Pattern in EE-MAC Protocol

5. ***Building and Updating Power Control Table***: Each sensor node stores the values of P_{TX_MIN} for its neighbor and the corresponding P_{TX_MIN} to transmit DATA and ACK frames in the next communication cycle. For this we need power control table [11, 15] as shown in *fig. 4.3* below:

neigh_node_id	power_level

Fig. 4.3: Transmission Power Control Table

The power control table contains two fields: its neighbor's node ID (neigh_node_id) and the corresponding transmission power level (power_level). Due to the variations in WSNs the power control table needs to be updated in each transmission cycle. The update algorithm is discussed below:

a) In the beginning, a node, say A, transmits SYNC and RTS packet with maximum power. The neighbors after receiving the RTS packet calculate the corresponding power levels separately and send the power level back to the sender's node through CTS frame. Node, A then receiving the CTS packet records the receiver's ID, corresponding power level in its power control table.

b) Node A repeats process (a) until it records all its neighbor's IDs and the corresponding power levels in its power control table.

c) If the RTS/CTS handshake fails, go to step (d). Otherwise the node A uses the latest power level to update its power control table.

d) RTS-CTS failure occurs due to two situations: 1) when the RTS arrives, the receiver node is in sleep state. For solving this problem the sender sends RTS packet for a pre-defined number of times. 2) If the sender has not received the CTS packet after a pre-defined number of times, it will assume that the transmission channel and/or receiver state has changed; senders then repeat the above process until they receive the CTS by gradually increasing the power level. If the sender still can't receive the CTS, it will delete the node from its power control table because the destination node is no longer within its transmission range or runs out of battery.

e) When the number of nodes becomes less than threshold value, i.e. some nodes are out of battery, go to step (a) to rebuild the power control table.

4.2 ENERGY ANALYSIS

In this section we will give an analytical model of energy consumption for adaptive SMAC (Sensor-MAC) described in [10]. Then we will modify and analyze the model by adding the SYNC packet and transmission power control mechanism.

As referred in [8] when a node has a packet to transmit, carrier sense delay (t_{CS}), back-off delay (t_{BO}), transmission delay, propagation delay, processing delay and queuing delay have to be considered. In addition to these delays, a sleep delay which is caused by the sleep of the receiver also occurs.

As described in [8], the sleep delay can be calculated by assuming that a packet arrives at the sender with equal probability

$$t_{SD} = T_{frame}/2$$

where,

$$T_{frame} = T_{listen} + T_{sleep}$$

During the sleep delay, nodes will stay in listening mode. Therefore the total energy consumption for transmitting a packet over a period of time denoted by *t* can be expressed as described in [12]

$$E(t) = N_T(t)E_T + N_R(t)E_R + T_S(t)P_S + T_I(t)P_I \text{ --------------------(3)}$$

Where the notations are illustrated in table I below.

TABLE I

ENERGY MODEL NOTATIONS

Notations	**Parameters**
N_T	Number of times that a node transmit
N_R	Number of times that a node receives
E_T	Energy consumption when transmitting
E_R	Energy consumption when receiving
T_S	Number of times the node spend in sleep
T_I	Number of times the node spend in idle
P_S	Power consumption for sleep mode
P_I	Power consumption for idle mode

The energy consumption for transmitting a packet can be evaluated as:

$$E_T = P_{Tx}(t_{RTS} + t_{data}) + P_{Rx}(t_{CS} + t_{BO} + t_{SD} + t_{CTS} + t_{ACK} + 3t_{SIFS} + t_{DIFS}) \text{ --(4)}$$

Similarly, the expected energy for receiving a packet is given by:

$$E_R = P_{Tx}(t_{CTS} + t_{ACK}) + P_{Rx}(t_{RTS} + T_{data} + 3t_{SIFS} + t_{DIFS}) \text{ -- (5)}$$

Where the used parameters are summarized in table II.

TABLE II

USED PARAMETERS

Notations	**Parameters**
P_{Tx}	Power consumption for transmitting
P_{Rx}	Power consumption for receiving
t_{RTS}	Time spent in sending RTS
t_{data}	Time spent in sending data
t_{CS}	Time spent in carrier sense
t_{BO}	Back-off delay
t_{SD}	Sleep delay
t_{CTS}	Time spent in receiving CTS
t_{ACK}	Time spent in receiving ACK
t_{SIFS}	SIFS time
t_{DIFS}	DIFS time

This work is also carried out in [13] where the energy consumed during idle and sleep period are not taken into account. Therefore the equation (3) becomes,

$$E(t) = N_T(t)E_T + N_R(t)E_R \quad \text{--(6)}$$

Also, this paper [12][13] doesn't consider the energy consumed while transmitting the SYNC packets for synchronization. Initially nodes in the sensor network send SYNC packets to neighboring nodes during synchronization period for synchronizing their sleep and listen periods. Here we assume that every time node selects its own schedule and broadcasts it through SYNC packet during synchronization period. Hence, the equations (5) and (6) can be modified as

$$E_T = P_{Tx}(t_{SYNC} + t_{RTS} + t_{data}) + P_{Rx}(t_{CS} + t_{BO} + t_{SD} + t_{CTS} + t_{ACK} + 3t_{SIFS} + t_{DIFS}) \quad \text{---(7)}$$

$$E_R = P_{Tx}(t_{CTS} + t_{ACK}) + P_{Rx}(t_{SYNC\_REC} + t_{RTS} + T_{data} + 3t_{SIFS} + t_{DIFS}) \text{ --(8)}$$

Where,

t_{SYNC} is the time spent in broadcasting SYNC packets and t_{SYNC_REC} is the time spent in receiving SYNC packets.

Now, we add the power control mechanism to this energy consumption model. In power control mechanism we send the SYNC, RTS and CTS packets with maximum power P_{TXMAX}, P_{RXMAX} and DATA and ACK packets with minimum power P_{TXMIN} and P_{RXMIN} respectively. Therefore the equations (7) and (8) can be modified as

$$E_T = P_{TXMAX}t_{SYNC} + P_{TXMAX}t_{RTS} + P_{TXMIN}t_{data} + P_{RX}(t_{CS} + t_{BO} + t_{SD}) + P_{RXMAX}t_{CTS} + P_{RXMIN}t_{ACK} + P_{RX}(3t_{SIFS} + t_{DIFS}) \text{ --(9)}$$

Where the notations are illustrated in table III.

TABLE III

NEW ADDED EQUATION PARAMETERS

Notation	**Parameters**
P_{TXMAX}	Maximum power transmitted by transmitter
P_{RXMAX}	Maximum power transmitted by receiver
P_{TXMIN}	Minimum power transmitted by transmitter
P_{RXMIN}	Minimum power transmitted by receiver

The minimum power transmitted by transmitter and receiver can be calculated using equation (1) and (2).

Similarly at the receiver side,

$$E_R = P_{TXMAX}t_{CTS} + P_{TXMIN}t_{ACK} + P_{RXMAX}t_{RTS} + P_{RXMIN}t_{data} + P_{RX}(3t_{SIFS} + t_{DIFS} + t_{SYNC\_REC}) \text{ ------------------------------ } (10)$$

Hence, the total energy consumed (by putting E_T and E_R values from equations (9) and (10) into equation (6)),

$$E_T(t) = N_T(t)[\, P_{TXMAX}t_{RTS} + P_{TXMIN}t_{data} + P_{TXMAX}t_{SYNC} + P_{RX}(t_{CS} + t_{BO} + t_{SD}) + P_{RXMAX}t_{CTS} + P_{RXMIN}t_{ACK} + P_{RX}(3t_{SIFS} + t_{DIFS})] + N_R(t)[\, P_{TXMAX}t_{CTS} + P_{TXMIN}t_{ACK} + P_{RXMAX}t_{RTS} + P_{RXMIN}t_{data} + P_{RX}(3t_{SIFS} + t_{DIFS} + t_{SYNC\_REC})] \text{ ------------------------------ } (11)$$

Chapter 5

IMPLEMENTATION DETAILS AND SIMULATION RESULTS

5.1 SIMULATOR USED

Network Simulator (version 2) [34, 35, 36] is an object-oriented, discrete event driven network simulator developed at UC Berkeley. It is written in C++ and uses OTcl (Tcl script language with Object-oriented extensions developed at MIT) as command and configuration interface *ns v2* has three substantial changes from *ns v1*:

1. More complex objects in *ns v1* have been decomposed into simpler components for greater flexibility.
2. The configuration interface is now OTcl, which is an object oriented version of Tcl.
3. The interface code to the OTcl interpreter is separated from the main simulator.

Fig. 5.1 shows the simplified user's view of ns-2. ns-2 is an Object-oriented Tcl (OTcl) script interpreter that has a simulation event scheduler, network component object libraries, and network setup module libraries. To setup and run a simulation network, a user should write an OTcl script that initiates an event scheduler, sets up the network topology using network objects and setup functions in the library, and tells the traffic sources when to start and stop transmitting packets through the event scheduler.

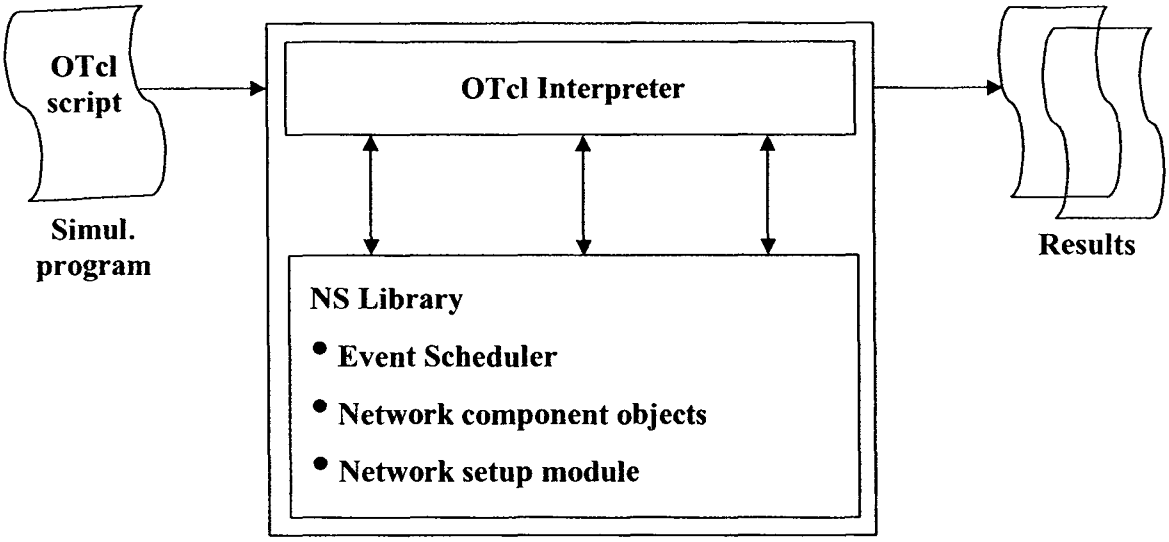

Fig. 5.1: Simplified User's View

For obtaining *ns* simulation results, when our simulation is finished (from *fig. 5.1*), ns-2 produces one or more text-based output files which contain detailed simulation data, if specified in OTcl script. This data can be used for simulation analysis or as an input to the graphical simulation display tool called Network Animator (NAM).

5.2 MODIFIED ENERGY MODEL

To support the simulation of our proposed EE-MAC protocol, we have modified the extended energy model [37]. Major changes have been made in the physical layer (ns-2/mac/wireless-phy.{h, cc}) and the energy model (ns-2/mobile/energy-model.{h, cc}). Energy model is a node attribute that has an initial value which represents the level of energy, the node has at the beginning of the simulation known as *initialEnergy_*. It also gives energy usage for every packet it transmits and receives, known as *txPower_* and *rxPower_* respectively.

The energy model is used through the node configuration API which consists of the type of addressing structure used in the simulation, defining the network components for mobile nodes, turning on/off trace options at Agent/Router/MAC levels, selecting the type of adhoc routing protocol wireless nodes or defining their energy model. An example for defining a node configuration API is given below:

	OPTION_TYPE	*AVAILABLE OPTION_VALUES*
	-------------------	---
$ns_ node-config	*-addressingType*	*flat or hierarchical or expanded*
	-adhocRouting	*DSDV or DSR or TORA or AODV*
	-llType	*LL*
	-macType	*Mac/802_11*
	-propType	*Propogation/TwoRayGround*
	-ifqType	*Queue/DropTail/PriQueue*
	-ifqLen	*50*
	-phyType	*Phy/WirelessPhy*
	-antType	*Antenna/OmniAntenna*
	-channelType	*Channel/WirelessChannel*
	-topoInstance	*$topo_instance*
	-wiredRouting	*ON or OFF*
	-mobileIP	*ON or OFF*
	-energyModel	*EnergyModel*
	-initialEnergy	*(in Joules)*
	-rxPower	*(in W)*
	-txPower	*(in W)*
	-agentTrace	*ON or OFF*
	-routerTrace	*ON or OFF*
	-macTrace	*ON or OFF*
	-MovementTrace	*ON or OFF*
	-reset	

The default values for all the above options are NULL except -addressingType whose default value is flat. The option reset can be used to reset all the node-config parameters to their default values.

The modified Tcl script for updated energy model is given below. The lines starting with '#' symbol are considered as comments for clear understanding of the script:

```
set opt(chan)           Channel/WirelessChannel
set opt(prop)           Propagation/TwoRayGround
set opt(netif)          Phy/WirelessPhy
set opt(mac)            Mac/SMAC        ;# MAC type
set opt(ifq)            Queue/DropTail/PriQueue
set opt(ll)             LL
set opt(ant)            Antenna/OmniAntenna
set opt(x)              1000            ;# X dimension of the topography
set opt(y)              1000            ;# Y dimension of the topography
set opt(cp)             "../mobility/scene/cbr-50-10-4-512"
set opt(sc)             "../mobility/scene/scen-670x670-50-600-20-0"
set opt(ifqlen)         50              ;# max packet in ifq
set opt(tr)             eemac.tr        ;# trace file
set opt(nam)            eemac.nam       ;# animation file
set opt(rp)             DSDV            ;# routing protocol
set opt(lm)             "off"           ;# log movement
set opt(energymodel)    EnergyModel     ;
set opt(radiomodel)     RadioModel      ;
set opt(initialenergy)  1000        ;# Initial energy in Joules
set opt(nn)             2           ;# number of nodes
set opt(distance)       22          ;# distance in meters between two mobile nodes.
set opt(seed)           0.0
set opt(begin)          50.0        ;# simulation start time
set opt(stop)           750.0       ;# simulation time

Mac/SMAC set syncFlag_ 1
Mac/SMAC set dutyCycle_ 10
```

 # The values are based on the specification of Orinoco 11b Card [38]. The Antenna height of transmitting and receiving antenna is 1.5 meters and the parameters are calculated under TwoRayGround model by tools from ns-2.

```
Antenna/OmniAntenna set Gt_ 1                      ;# transmit antenna gain
Antenna/OmniAntenna set Gr_ 1                      ;/# receive antenna gain
Phy/WirelessPhy set L_ 1.0                         ;# system loss factor
Phy/WirelessPhy set freq_ 2.472e9                  ;# channel-13. 2.472GHz
Phy/WirelessPhy set bandwidth_ 11Mb                ;# data rate
Phy/WirelessPhy set Pt_ 0.031622777                ;# transmit power (15dBm)
Phy/WirelessPhy set CPThresh_ 10.0                 ;# collision threshold
Phy/WirelessPhy set CSThresh_ 5.011872e-12         ;# carrier sense threshold
Phy/WirelessPhy set RXThresh_ 5.82587e-09          ;# receive power threshold;

Mac/SMAC set dataRate_ 11Mb                        ;# rate for Data Frames
Mac/SMAC set basicRate_ 2Mb                        ;# rate for Control Frames

set ns_        [new Simulator]
set topo       [new Topography]
set tracefd    [open $opt(tr) w]
set namtrace   [open $opt(nam) w]
set prop       [new $opt(prop)]

$topo load_flatgrid $opt(x) $opt(y)
ns-random 1.0
$ns_ trace-all $tracefd

create-god $opt(nn)
```

 # Global node setting

```
    $ns_ node-config -adhocRouting DSDV \
```

```
                        -llType $opt(ll) \
                        -macType $opt(mac) \
                        -ifqType $opt(ifq) \
                        -ifqLen $opt(ifqlen) \
                        -antType $opt(ant) \
                        -propType $opt(prop) \
                        -phyType $opt(netif) \
                        -channelType $opt(chan) \
                        -topoInstance $topo \
                        -agentTrace OFF \
                        -routerTrace OFF \
                        -macTrace ON \
                        -energyModel $opt(energymodel) \
                        -idlePower 1.0 \
                        -sleepPower 0.001 \
                        -transitionPower 0.2 \
                        -transitionTime 0.005 \
                        -initialEnergy $opt(initialenergy)

# -rxPower 1.0 \
# -txPower 1.0 \

$ns_ set WirelessNewTrace_ ON

for {set i 0} {$i < $opt(nn) } {incr i} {
                set node_($i) [$ns_ node]
                $node_($i) random-motion 0 ;# disable random motion
                $node_($i) set X_ [expr {$i * $opt(distance)}]
                $node_($i) set Y_ 0.0
                $node_($i) set Z_ 0.0
puts "Position of Node ($i) :[expr {$i * $opt(distance)}]"
        }
```

```
set udp_(0) [new Agent/UDP]
$ns_ attach-agent $node_(0) $udp_(0)
set null_(0) [new Agent/Null]
$ns_ attach-agent $node_([expr $opt(nn)-1]) $null_(0)

set cbr_(0) [new Application/Traffic/CBR]
$cbr_(0) set packetSize_ 512
$cbr_(0) set interval_ 1.0

# We are changing the cbr interval to generate data for analysis.

$cbr_(0) set random_ 1
$cbr_(0) set maxpkts_ 150
$cbr_(0) attach-agent $udp_(0)
$ns_ connect $udp_(0) $null_(0)

$ns_ at $opt(begin) "$cbr_(0) start"
#$ns_ at 177.000            "$node_(0) set ifqLen"

# Tell all the nodes when the simulation ends.

for {set i 0} {$i < $opt(nn) } {incr i} {
   $ns_ at $opt(stop) "$node_($i) reset";
}

$ns_ at $opt(stop) "puts \"NS EXITING...\" ; $ns_ halt"

puts "Starting Simulation..."
$ns_ run
```

The commands used in the above energy model are given in the appendix.

5.3 SIMULATION AND RESULTS

A detail of simulation methodology used for our proposed EE-MAC protocol is explained in the following sections:

5.3.1 SIMULATION ENVIRONMENT

We have used ns-2 as a simulation tool to evaluate the performance of our proposed algorithm under different varying cbr intervals i.e., 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0. In our simulation, nodes are spread in a terrain of 1000×1000. The nodes are randomly deployed within the simulation area. Each node has the same 22 meters of transmission range.

Results of simulation are expressed using following metrics:

- Energy Consumption
- Throughput

Energy Consumption is expressed in terms of total energy consumed by all the nodes in the transmission as well as in the reception of the packets.

Throughput is defined as the ratio of total packets sent per second (packets/s).

5.3.2 SIMUALTION PARAMETERS

The simulation parameters are summarized in the table below:

TABLE IV

SIMULATION PARAMETERS

Parameters Name	Values
ns-2 version	2.33
Simulation time	700 seconds
Traffic	cbr
Routing Protocol	DSDV
Transmission range	22 meters
Transmitting and receiving antenna height	1.5 meters
Transmitting and receiving antenna gain	Gt=1, Gr=1
System loss	L=1
Bandwidth	11 Mb
Minimum transmission power	0.031622777 W
Carrier sense power	5.011872e-12 W
Receive power threshold	5.82587e-09 W
Initial Energy	1000 Joules

5.3.3 RESULTS AND ANALYSIS

The results obtained in terms of two metrics i.e., energy consumption and throughput, are represented in the form of graphs, which are then compared and justified.

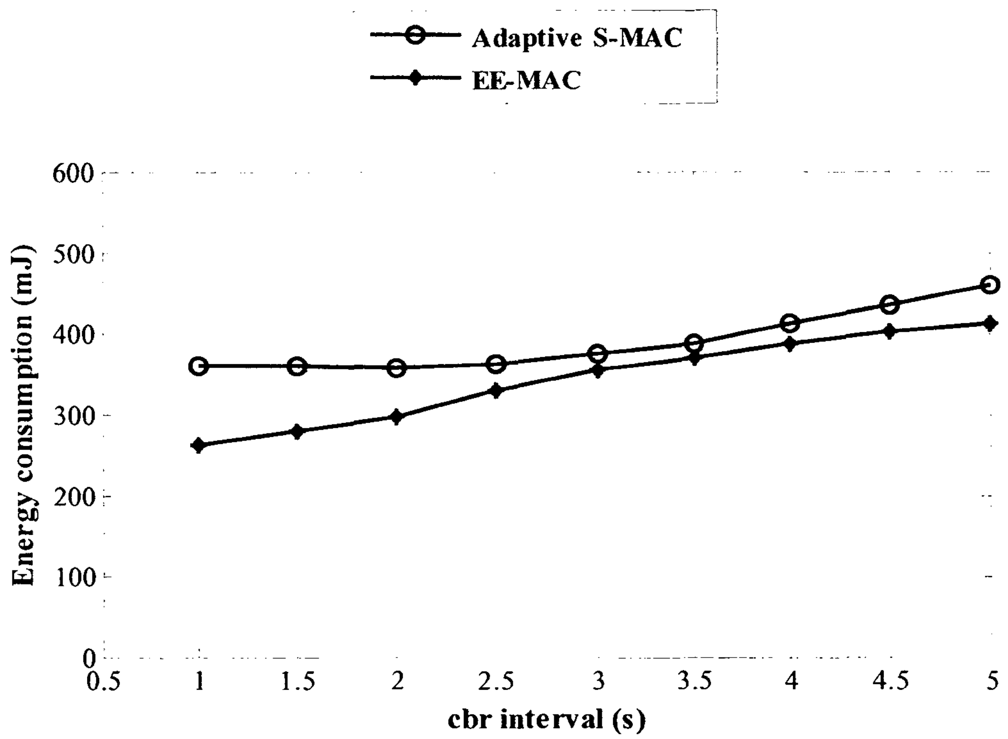

Fig. 5.2: Comparison of energy consumption between EE-MAC and adaptive S-MAC

Fig. 5.2 shows the node energy consumption (mJ) versus cbr interval (s) for our proposed protocol and the adaptive S-MAC protocol, and is plotted by varying the cbr intervals which impact the traffic load on the source node. For this experiment, each node has initially 1000 Joules of energy. The figure indicates that our proposed energy efficient MAC (EE-MAC) protocol presents an improvement over adaptive S-MAC protocol, which can be explained by the use of minimum transmission power required for transmitting DATA and ACK packets.

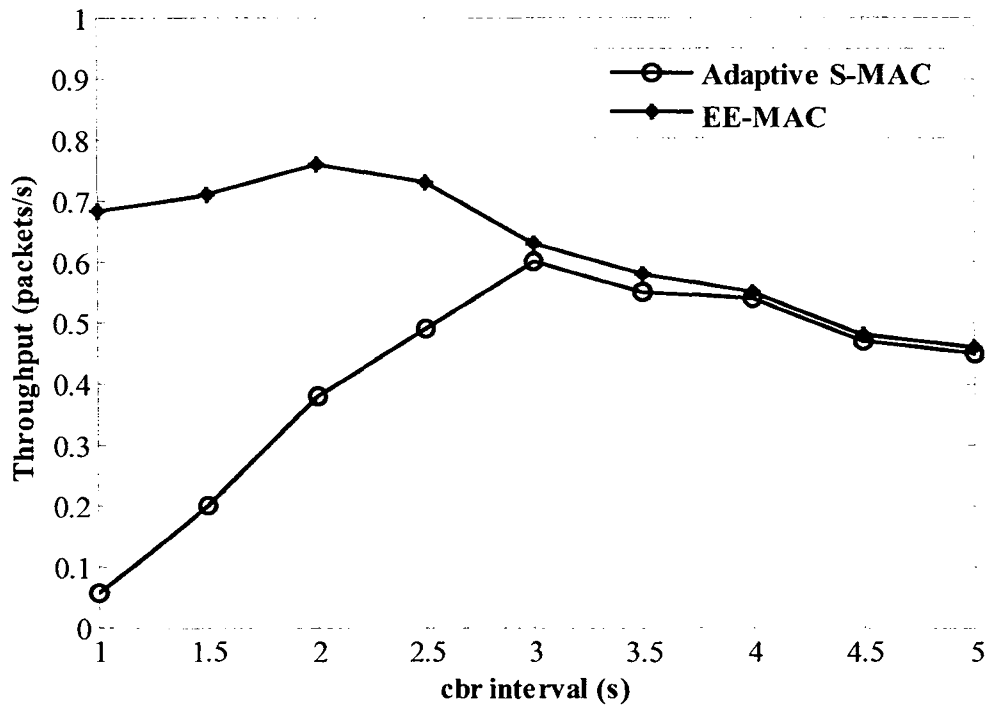

Fig. 5.3: Throughput comparison between EE-MAC and adaptive S-MAC

Another important parameter is the network throughput. *Fig. 5.3* shows that our proposed EE-MAC protocol outstands the adaptive S-MAC protocol at network throughput by two times. Initially from 1.0 to 3.0 cbr interval, our proposed EE-MAC protocol shows highly improvement as compared to adaptive S-MAC. Although, after 3.0 cbr interval, our proposed EE-MAC protocol performs almost in the same way as adaptive S-MAC protocol.

As described above, compared to the adaptive S-MAC protocol, our proposed EE-MAC protocol has improved in energy consumption and throughput of the networks. Thus, the addition of transmission power control techniques, improves the overall performance of the existing adaptive S-MAC protocol.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

Wireless sensor networks must be designed keeping energy efficiency in mind. Several protocols have been developed for this purpose, but none can actually fulfill the changing application needs. The S-MAC protocol which is a standard protocol, has received considerable attention due to its energy saving schemes designed for sensor network. The S-MAC protocol saves energy but sacrifices latency. To improve latency, adaptive S-MAC protocol has been developed. In order to further increase the energy savings, we add the transmission power control to the adaptive S-MAC protocol. The adjustment of the transmission power, performed by TPC protocols, is a technique to lessen energy consumption in the communication.

Keeping the same idea in mind, we propose an energy efficient MAC (EE-MAC) protocol which saves energy by broadcasting RTS and CTS packets using full transmission power whereas DATA and ACK packets are sent using minimum required transmission power. For this purpose, we change the energy model of the adaptive S-MAC protocol. We simulate our proposed EE-MAC protocol using ns-2. Our experimental results show that in comparison to adaptive S-MAC protocol, EE-MAC protocol achieves more energy saving and higher throughput.

The work carried out in this dissertation gives insight into the performance of the adaptive S-MAC protocol after modifying it by adding transmission power control technique. It can be concluded that current work can contribute to the knowledge in a modest way by simulating and realizing that adaptive S-MAC protocol, indeed will perform better by using controlled transmission power, both by saving energy and increasing throughput.

6.2 FUTURE WORK

We modified the adaptive S-MAC protocol by adding transmission power control technique and evaluated its performance in a limited scenario. However, further modifications can be made to evaluate its performance in a larger network context. Some of the modifications are suggested below:

- The number of nodes can be increased to investigate the protocol behavior under realistic sensor network environments.
- The proposed protocol can be evaluated for mobile scenarios.
- TPC algorithm can be devised in mobile application environment.

BIBLIOGRAPHY

1. J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," Copyright © 2008 Elsevier, pp. 2292-2330.

2. C. Shiva Ram Murthy, and B. S. Manoj, "Ad-hoc Wireless Networks: Architectures and Protocols," Copyright © 2004 by Pearson education, Inc.

3. S. S. Kulkarni, "TDMA Services for Sensor Networks," Proc. 24th Int'l. Conf. Distrib. Comp. Sys. Wksps., Mar. 2004, pp. 604-09.

4. Wei Ye, John Heidmann, "Medium Access Control in Wireless Sensor Networks" USC/ISI TECHNICAL REPORT ISI-TR-580, OCTOBER 2003.

5. I. Demirkol, C. Ersoy, F. Alagoz. "MAC protocols for Wireless Sensor Networks: A Survey" IEEE communication magazine, pages 115-121, April 2006.

6. Koen Langendoen, "Medium access control in wireless sensor networks," Vol. II: Practice and Standards edited by H. Wu and Y. Pan, published by Nova Science Publishers © 2007.

7. P. Naik, K. Sivalingam, "A Survey of MAC Protocols for Wireless Sensor Networks," in: C. Raghavendra, K. Sivalingam, T. Znati (Eds.), Wireless Sensor Networks, Kluwer Academic Publishers, 2004.

8. W. Ye, J. Heidemann, and D. Estrin, "An energy efficient MAC protocol for Wireless Sensor Networks," in Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communication Societies (INFOCOM 2002), New York, USA, June 2002, pp. 1567-1576.

9. IEEE 802.11, Wireless LAN media access controls (MAC) and physical layer (PHY) specifications, 1999.

10. W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks," in IEEE/ACM Transactions on Networking, June 2004, pp. 493-506.

11. Jian Xiao, and Fengqi Yu, "An Efficient Transmission Power Control Algorithm in Wireless Sensor Networks," in Proceedings of, 2007, pp. 2767-2770.

12. H. W. Tseng, S. H. Yang, P. Y. Chuang, E. H. Kuang Wu and G. H. Chen, "An Energy Consumption Analytic Model for A Wireless Sensor MAC Protocol", in Proceedings of the IEEE Vehicular Technology Conference (VTC'2004), September 2004, pp. 4533-4537.

13. T. Ezzedine, M. Miladi, and R. Bouallegue, "An Energy-Latency-Efficient MAC Protocol for Wireless Sensor Networks," in Proceedings of International Journal of Electronics, Communications and Computer Engineering, January 2009, pp. 30-35.

14. J. Gomez and A.T. Campbell, M. Naghshineh, and C. Bisdikian, "Conserving Transmission Power in Wireless Ad Hoc Networks," in ICNP'01, November 2001.

15. S. Agarwal, S. Krishnamurthy, R. H. Katz, and S. K. Dao, "Distributed Power Control in Ad-hoc Wireless Networks," in PIMRC, 2001.

16. M. B. Pursley, H. B. Russell, and J. S. Wysocarski, "Energy Efficient Transmission and Routing Protocols for Wireless Multiple-hop Networks and Spread-Spectrum Radios" in EUROCOMM, 2000.

17. P. Karn, "MACA-A New Channel Access Method for Packet Radio" in Proceedings of 9th ARRL Computer Networking Conference, 1990.

18. S. Narayanaswamy, V. Kawadia, R. S. Sreenivas, and P. R. Kumar, "Power Control in Ad-Hoc Networks: Theory, Architecture, Algorithm and Implementation of the COMPOW Protocol" in European Wireless 2002, February 2002.

19. E. S. Jung and N. H. Vaidya, "A Power Control MAC Protocol for Ad-Hoc Networks" in MOBICOM 2002, September, 2002.

20. Z. Zhao, X. Zhang, P. Sun, and P. Liu, "A Transmission Power Control MAC Protocol for Wireless Sensor Networks" in Proceedings of 6th International Conference on Networking (ICN'07), © 2007.

21. M. Miladi, T. Ezzedine, and R. Bouallegue, "Latency of Energy Efficient MAC Protocols for Wireless Sensor Networks" © 2006 IEEE.

22. L. H. A. Correia, D. F. Macedo, A. L. dos Santos, A. A. F. Loureiro, J. M. S. Nogueira, "Transmission Power Control Techniques for Wireless Sensor Networks" in COMNET 51, © 2007 Elsevier, pp. 4765-4779.

23. R. Ramanathan and R. Rosales-Hain, "Topology Control of Multihop Wireless Networks using Transmit Power Adjustment" in INFOCOM 2000, volume 2, pp. 404-413, March 2000.

24. R. Wattenhofer, L. Li, P. Bahl, and Y. M. Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad-Hoc Networks" in INFOCOM 2001, volume 3, pp. 1388-1397, April 2001.

25. J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks" in INFOCOM 2000, pp. 585-594, March 2000.

26. J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for sensor networks" in Proceedings of 2nd ACM Conference on Embedded Networked Sensor Systems (SenSys), November 3-5, 2004.

27. M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks" in Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys 2006), Boulder, Colorado, USA, 2006, pp. 307-320.

28. T. V. Dam, and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks" in 1st ACM Conf. n Embedded Networked Sensor Systems (SenSys 2003), Los Angeles, CA, Nov. 2003, pp. 171-180.

29. V. Rajendran, K. Obraczka, and Garcia-Luna-Aceves, "Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks" in Proceedings of ACM SenSys 03, Los Angeles, CA, November 2003.

30. L. V. Hoesel, and P. Havinga, "A lightweight medium access protocol (LMAC) for wireless sensor networks" in Proceedings of 1st International Workshop on Networked Sensing Systems (INSS 2004), Japan, June 2004.

31. S. Chatterjea, L. V. Hoesel, and P. Havinga, "AI-LMAC: an adaptive, Information- centric and lightweight MAC protocol for wireless sensor networks" in Proceedings of 2nd International Conference on Intelligent Sensors, Sensor

Networks and Information Processing (ISSNIP 2004), Melbourne, Australia, December 2004.

32. I. Riiee, A. Warrier, M. Aia, and J. Min, "Z-MAC: A hybrid MAC for wireless sensor networks," in 3rd ACM Conf. on Embedded Networked Sensor Systems (SenSys 2005), San Diego, CA, Nov. 2005, pp. 90-101.

33. T. Zheng, S. Radhakrishnan, and V. Sarangan, "PMAC: An adaptive energy-efficient MAC protocol for wireless sensor networks," in 19th IEEE Int. Parallel and Distributed Processing Symposium (IPDPS 2005), Denver, CO, Apr. 2005.

34. The ns manual, http://www.isi.edu/nsnam/ns/ns-documentation.html.

35. Jae Chung, and Mark Claypool, "ns by example" http://nile.wpi.edu/NS/.

36. Marc Greis, "ns tutorial" http://www.isi.edu/nsnam/ns/tutorial.

37. http://www.isi.edu/ilense/software/smac/ns2_energy.

38. Proxim, ORiNOCO 11b Client PC Card Specifications, 2004.

ABBREVIATIONS

ACK	Acknowledgment
cbr	Constant Bit Rate
CSMA	Carrier Sense Multiple Access
CSMA/CA	CSMA with Collision Avoidance
CTS	Clear-to-Send
DIFS	Defer Inter-Frame Spacing
DSDV	Distance Sequenced Distance-Vector
EE-MAC	Energy-Efficient Medium Access Control
EIFS	Extended Inter-Frame Spacing
FDMA	Frequency Division Multiple Access
IEEE	Institute of Electrical and Electronics Engineers
IFS	Inter-Frame Spacing
LAN	Local Area Network
MAC	Medium Access Control
MACA	Multiple Access Collision Avoidance
NAV	Network Allocation Vector

QoS	Quality of Service
RSSI	Received Signal Strength Indicator
RTS	Request-To-Send
SIFS	Short IFS
S-MAC	Sensor-MAC
SYNC	Synchronization
SYNC-REC	Synchronization-Receive
TDMA	Time Division Multiple Access
TPC	Transmission Power Control
WLAN	Wireless LAN
WSN	Wireless Sensor Network

APPENDIX

COMMANDS AT A GLANCE

set ns_ [new simulator]

This command creates an instance of the simulator object.

$ns_ node

This command is used to create a simple node. This returns a handle to the node instance created.

$topo load_flatgrid <X> <Y> <optional:res>

This initializes the grid for the topology object. <X> and <Y> are the x-y co-ordinates for the topology and are used for sizing the grid. The grid resolution may be passed as <res> whose default value is 1.

$ns_ trace-all <tracefile>

This command is used to set up tracing in ns. All traces are written in the <tracefile>.

create-god <num_nodes>

This command is used to create a God instance. The number of mobile nodes is passed as argument which is used by God to create a matrix to store connectivity information of the topology.

$ns_ node-config -<config-parameter> <optional-value>

This command is used to configure nodes. The different config-parameters are addressingType, different type of the network stack components, whether tracing will be turned on or not, mobileIP flag is turned on or not, energy model is being used or not, etc.

set udp0 [newAgent/UDP]

This command creates an instance of the UDP agent.

set cbr1 [newApplication/Traffic/CBR]

This command creates an instance of the CBR traffic.

$node attach-agent <node> <agent>

This command attaches the <agent> to the <node>. Here we assume that the <agent> has already been created. An agent is typically been created by *set agent [new Agent/AgentType]* where Agent/AgentType defines the class definition of the specified agent type.

$ns_ connect <src> <dst>

Sets up a connection between the src and dst agents.

$ns_ run

This command starts the simulator