MEDIUM ACCESS CONTROL(MAC) LAYER DESIGN AND DATA QUERY PROCESSING FOR WIRELESS SENSOR NETWORKS

by

QINGCHUN REN

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2007

ACKNOWLEDGEMENTS

Though only my name appears on the cover of this dissertation, a great many people have contributed to its production. I owe my gratitude to all those people who have made this dissertation possible and because of whom my graduate experience has been one that I will cherish forever.

I would like to gratefully and sincerely thank my advisor Dr. Qilian Liang. I have been amazingly fortunate to have an advisor who gave me the freedom to explore on my own, and at the same time the guidance to recover when my steps faltered. Dr. Liang taught me how to question thoughts and express ideas. His patience and support helped me overcome many crisis situations and finish this dissertation. His mentorship was paramount in providing a well rounded experience consistent my long-term career goals. I am not sure many graduate students are given the opportunity to develop their own individuality and self-sufficiency by being allowed to work with such independence. For everything you've done for me, Dr. Liang, thank you very much.

I would like to extend my special appreciation to Dr. Frank L. Lewis, Dr. Saibun Tjuatja, Dr. Dan Popa, Dr. Sajal Das and Dr. Gautam Das for their interest in my research for taking time to serve in my dissertation committee.

I would also like to thank all the lab mates: Liang Zhao, Lingming Wang, Haining Shu, Jing Liang and Xinsheng Xia, and all friends in UTA, where we have had a good time to share research experience, to help each other not only in research area.

Finally, my deepest gratefulness is devoted to my family. My husband Xinli is always standing by me and his love, knowledge and encouragements support me all the

time in my Ph. D degree studies. I thank my parents and parents-in-law, whose unselfish contribution makes me can concentrate on my work.

September 12, 2007

ABSTRACT

MEDIUM ACCESS CONTROL(MAC) LAYER DESIGN AND DATA QUERY PROCESSING FOR WIRELESS SENSOR NETWORKS

Publication No. _____

Qingchun Ren, Ph.D.

The University of Texas at Arlington, 2007

Supervising Professor: Qilian Liang

In this dissertation, there are four main aspects included: energy reservation on MAC layer, secure improvement for DoS attacks on MAC layer, query processing with uncertainty for sensor database systems, and throughput maximization on MAC layer for ultra wideband communication systems.

In energy reservation, we proposed asynchronous MAC (A-MAC) protocol and asynchronous schedule-based MAC (ASMAC) protocol. A-MAC and ASMAC protocols are attractive due to their suitabilities for multihop networks and capabilities of removing accumulative clock-drifts without any network synchronization. Moreover, we build a traffic-strength- and network-density-based model to adjust essential algorithm parameters adaptively.

In secure problem, we proposed a secure MAC protocol for WSNs. Our algorithm-FSMAC firstly design multiple indicators for intrusion detection according to the classification of popular DoS attacks on MAC layer of WSNs, and innovatively utilizes fuzzy logic theory to implement making decision on intrusion.

In query processing, we proposed a quality-guaranteed and energy-efficient (QGEE) algorithm. QGEE utilizes in-network query processing method to task WSDSs through declarative queries, and confidence interval strategy to determine the accuracy of query answers. Given a query, the QGEE algorithm can reduce disturbance from measurements with extreme error and minimize energy consumption, while providing satisfying service for various applications.

In MAC protocol design for ultra wideband, we proposed a throughput maximized MAC protocol (TM-MAC). In TM-MAC, we implement concurrent multiuser access scheme. For multiuser interference, we establish a model to adaptively adjust the data transmission rate to generate the expected signal to interference noise ratio (SINR) at the receiver side for reliable communications. In subset formation, we propose a general analytical framework that captures the unique characteristics of shared wireless channel and throughput variance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Wireless Sensor Networks

A wireless sensor network (WSN) can be thought as an *ad hoc* network consisting of sensor nodes linked by a wireless medium to perform distributed sensing tasks. WSNs have recently come into prominence because they hold the potential to revolutionize many segments of our economy and life, from environmental monitoring and conservation, to manufacturing and business asset management, to automation in the transportation and health-care industries[2].

Such networks are often deployed in resource-constrained environments, for instance with battery operated sensor nodes running untethered. Unlike other traditional systems, a WSN is subject to an unique set of challenges:

- *Limited hardware:*

  Each sensor node has limited processing, storage, and communication capabilities, and limited energy supply and bandwidth.

- *Limited support for networking:*

  The network is peer-to-peer, with a mesh topology and dynamic, mobile, and unreliable connectivity. There are no universal routing protocols or central registry services. Each sensor node acts both as a router and as an application host

- *Limited support for software development:*

  The tasks are typically real-time and massively distributed, involving dynamic collaboration among sensor nodes, and handling multiple competing events. Global properties can be specified only via local instructions.

1.2 Wireless Medium Access Control

Sensor nodes (or short, nodes)in a WSN share a common broadcast radio channel. Inside this network, every node may transmit at any point in space, which could cause contention over the radio channel. Access to this shared medium should be controlled in such a manner that all nodes receive a fair share of the available bandwidth, and that the bandwidth is utilized efficiently. The medium access control (MAC) protocol is responsible for deciding when competing nodes can access the share medium and ensuring that there is no nodes interfering with other's communications.

Numerous MAC protocols have been proposed for wireless networks. They can be classified into two basic categories: schedule-based MAC protocols and contention-based MAC protocols. In schedule-based MAC protocol, such as FDMA, TDMA, CDMA, FCDMA and SDMA [3], a central authority specifies when, and how long, each controlled node may transmit over the shared medium. Hence, there is no collision during transmission process. While in contention-based MAC protocol, such as CSMA, CSMA/CA, 802.11 DCF, WMACA [3], all nodes access the radio channel through competition. Note that for these existed MAC protocols, the overarching goal is ensuring the fairness and efficiency of medium access.

In CSMA/CA, an example of contention-based MAC protocol, a node senses the channel for a small period of time before transmitting a packet. If it does not detect any traffic, it assumes that the channel is idle and starts transmission. Moreover, CSMA/CA introduces a three-way handshake to make hidden nodes aware of upcoming transmissions, so collisions at common neighbor can be avoided. The working process is:

- The sender initiates the handshake by transmitting a Request-To-Send (RTS) control packet announcing its intended data transmission;
- The receiver responds received RTS with a Clear-To-Send (CTS) packet, which informs all neighbors of the receiver of the upcoming transfer;

- The final data packet is now guaranteed to be transmitted collision-free.

Fig. 1.1 shows the timing of successful data packet transmissions.

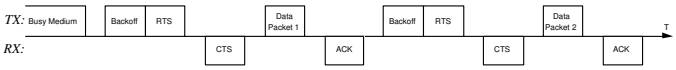


Figure 1.1. CSMA/CA timing of successful data packet transmissions.

In TDMA, an example of schedule-bases MAC protocol, each node has its own view of the world around it, and maintains a database and time base that is its own but is compatible with its community. Fig. 1.2 is the perspective for a particular node, node i. There are several general types of time slots assigned. These are:

- network entry/access slots
- nodes own broadcast-to-its-neighbors slot
- slots for this nodes neighbors to broadcast to their neighbors
- slots for point-to-point communication between this node and its neighbors (both directions)
- slots that are assigned for other purposes that do not involve this node but cannot be used by this node (i.e., they are too close for spatial reuse)
- specially assigned slots for temporary high bandwidth needs, such as beamforming
- all of the rest of the bandwidth, which is currently unassigned and therefore free to be seized for new purposes as they arise

1.3 Network Synchronization

For many digital communication engineers, the term synchronization is familiar in a somewhat restricted sense, meaning only the acquisition and the tracking of a clock in a receiver with reference to the periodic timing information contained in the received signal.

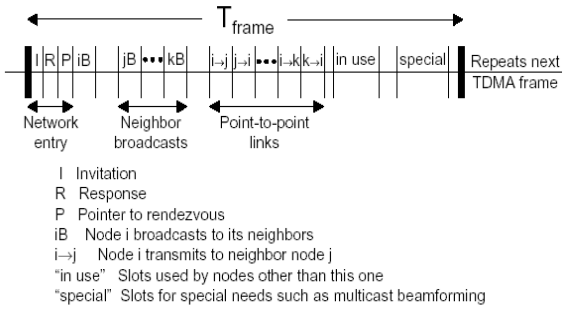


Figure 1.2. TDMA frame structure (simplied).

More properly speaking, this should be referred to as carrier or symbol synchronization. Summarily, there are eight types of synchronization mainly applied to telecommunication networks, i.e., carrier synchronization, symbol synchronization, frame synchronization, bit synchronization, packet synchronization, network synchronization, multimedia synchronization, and synchronization of real-time clocks [4]. Network synchronization is one of targets in this paper.

Network synchronization deals with the distribution of time and frequency over a network spread over an even wider geographical area. The goal is to align time and frequency scales of all clocks by using the communication capacity of links interconnecting them. Some well known applications for network synchronization are synchronization of clocks located at different multiplexing and switching points in a digital telecommunication network, synchronization of clocks in a telecommunication network that requires some form of time-division multiplexing multiple access and range measurement between two nodes in a network.

Over the years, many protocols have been designed for maintaining synchronization of physical clocks over telecommunication networks [5][6][7]. Some wireless standards such as 802.11 have similar time-synchronization beacons built into MAC layer. Network time protocol (NTP) stands out by virtue of its scalability, self-configuration for creating a global timescale in multihop networks, robustness to various types of failure, security in

the face of deliberate sabotages, and ubiquitous deployments. Other algorithms, such as time-diffusion synchronization protocol (TDP) and reference broadcast synchronization (RBS) are proposed in [8][9].

1.4 Issues on Energy-Efficient MAC Protocol Design

1.4.1 Energy Constraint Problem

One crucial challenge for WSN designers is to develop a system that will run for years unattendedly, which calls for not only robust hardware and software, but also lasting energy sources. However, currently, sensor nodes are powered by battery, whose available energy is limited. Moreover, replacing or recharging battery, in many cases, may be impractical or uneconomical. Even though, future sensor nodes may be powered by ambient energy sources (such as sunlight, vibrations, etc.) [10], the provided current is very low. From both perspectives, protocols and applications designed for WSNs should be highly efficient and optimized in terms of energy.

In general, a sensor node consists of a microprocessor, a data storage, sensors, analog-to-digital converters (ADCs), a data transceiver, an energy source, and controllers that tie those pieces together [2]. Communications, not only transmitting, but also receiving, or merely scanning channel for communication, can use up to half energy [11]. Thus, recently, some researchers have begun to study the energy efficiency problem through reducing power consumption on wireless interface.

1.4.2 Accumulative Clock-Drift Problem

As a matter of fact, the quality of a node's clock usually boils down to its frequency's stability and accuracy [9]. Generally speaking, as frequency stability and accuracy increase, so do its required power, size and cost, which are all troublesome for general nodes. Moreover, the frequency generated by a quartz oscillator is also affected by a number of

environmental factors: voltage applied to it, ambient temperature, acceleration in space, and so forth. Low-cost oscillators commonly have nominal frequency accuracy on the order of 10^4 to 10^6. That is, two similar but un-calibrated oscillators will drift apart from 1 to 100 microseconds every second [9]. As time goes by, oscillators will drift apart farther and farther. We call it accumulative clock-drift in this paper.

The basic idea of most energy-efficient MAC protocols is to power on/off their radios alternately to implement communication and to reduce energy consumption. This active/sleep scheme requires matching operation among nodes (i.e., source-destination pairs switch between active and sleep states coincidently) to ensure the low-power radio schedule to work successfully. Hence, for general WSNs, it is necessary to develop effective and efficient methods to resolve the mismatching problem caused by accumulative clock-drift. Network synchronization is one of existing approaches for this issue, in which a common timescale is necessary. While, is it the only or the best choice? If a system dose not provide network synchronization service, is there any alternative solution? Furthermore, although the strategies exploited by existing network synchronization schemes are various, the working load for carrying out network synchronization is mainly located at user sides (or data-collection node sides), which we call user-exhaustion schemes. Obviously, user-exhaustion scheme is not a wise choice, since data-collection nodes are typically subjected to strict energy constraint while data-gathering nodes are not.

1.4.3 Heterogeneous Problem

The traffic of WSN, in general, has a heterogeneous nature [12] (i.e., the traffic arrival rate for different sensor nodes and even for the same sensor node at different time fluctuates considerably during the network lifetime). Consequently, according to time-variant situation of system, how to adjust essential parameters adaptively is another important task for protocol designers. As a matter of fact, we notice that the power-

on/off duration is tightly related to system performance in terms of energy saving, time delay and system throughput. That is, with the increase of power-off duration, there is more chance for buffer overflowing, longer waiting-time for data packets and fewer data packets transmitted during a period of time. While there is more energy reserved for avoiding excessive idle listening. On the other hand, with the increase of power-on duration, there are more data packets transmitted, then there is less chance for buffer overflowing and shorter waiting-time for data packets. However, there is more energy wasted by idle listening. Nevertheless, little work is done on how to determine those essential parameters.

1.5 Secure Problems Caused by Denial of Service Attack

Since WSNs always have critical tasks, security will be important for most applications. Most WSNs actively monitor their surroundings, therefore it is easy to deduce information other than the data monitored[13]. Such unwanted information leakage often results in privacy breaches. Moreover, the wireless communication schemes employed by sensor networks facilitate eavesdropping and packet injection by adversaries. To ensure operation safety, secrecy of sensitive data and privacy for people in sensor environments, security algorithms are eagerly needed for WSNs.

Generally, denial of service (DoS) attacks[14] aim at disabling normal functions by wasting network resources that could have been utilized to provide useful services to legitimate clients. They can be carried out either by flooding victim nodes with network traffic or sending requests that cause victim nodes to behave unpredictably. DoS attacks on WSNs can not only be carried by attackers within an organization for having the authority to access the network, but also be raised by attackers outside the organization. For DoS attacks, the target resources may be file system space, process space, network bandwidth and network connections. There are two main kinds of DoS attack: system-

oriented DoS attack and congestion-based DoS attack. System-oriented DoS attacks exploit vulnerabilities in operating systems and the implementations of protocol stack. Congestion-based DoS attacks take advantage of the weakness inside the network design. While, according to the number of attack source, there are single-source DoS attack that is originated only at one host, and multi-source DoS attack that floods the victim with a barrage of attack packets coming from multiple hosts. DoS attacks on WSNs may be carried at network layer and MAC layer. On network layer, DoS attacks will result in a disruption of routing functionalities. While on MAC layer, DoS attacks can potentially disrupt channel access and cause the resource wastage in terms of bandwidth and energy.

Currently, agent-based intrusion detection[15][16], encryption[17][18] and authentication schemes[19][20] are popular and effective security methods for networks. They can cooperate together, or work individually to improve system security. But, these secure methods are disabled or degraded when they are applied to WSNs to defend DoS attacks. The reasons are: (a) Complex cryptography, which is energy, memory and CPU consuming, is too onerous to be completed by capacity-limited sensor nodes; (b) The cooperation among sensor nodes within a distributed WSN will consume extra resource; (c) It is impractical to employ particular sensor nodes to conduct intrusion detection in WSNs; (d) It is very easy for enemies to compromise some sensor nodes to get the access authority.

DoS attacks are increasingly common and critical in computer networks in recent years. Without proper security mechanisms, computer networks will be confined to limited and controlled environments, and negate much of promises they hold. However, DoS attacks have been studied extensively for wire-line networks, e.g., Internet, there is lack of equivalent researches for WSNs. Moreover, security in WSNs is complicated by the constrained capabilities of sensor nodes' hardware and the properties of their deployment. That is,

- Wireless links make WSNs more susceptible to link attacks ranging from passive eavesdropping, active impersonating, message replaying to message distorting;
- Sensor nodes, roaming in a hostile environment with relatively poor physical protection, have the non-negligible possibility of being compromised;
- WSNs, unlike fixed networks, have dynamic topology and group membership, which can dramatically increase the complexity to administrate authorization.

1.6 Ultra Wideband Systems

Within the past forty years, advances in analog, digital electronics and ultra-wideband (UWB) signal theory have enabled system designers to propose some practical UWB communication systems such as in [21]. Currently, numerous companies and government agencies are investigating the potential of UWB to deliver on its promises. A wide range of UWB applications have been demonstrated[22][23]. According to Federal Communications Commission (FCC), a UWB system is defined as any radio system that has a 10-dB bandwidth larger than 20 percent of its center frequency, or has a 10-dB bandwidth equal to or larger than 500 MHz[24]. Two different UWB communications systems - impulse-based systems (I-UWB) and multi-carrier systems (MC-UWB)) - have been pursued recently. UWB has several features that differentiate it from conventional narrowband systems[25]:

- Large instantaneous bandwidth enables fine time resolution for network time distribution, precision location capability, or use as a radar;
- Short duration pulses are able to provide robust performance in dense multipath environments by exploiting more resolvable paths;
- Low power spectral density allows coexistence with existing users and has a Low Probability of Intercept (LPI);
- Data rate may be traded for power spectral density and multipath performance.

Shannon channel capacity theory defines the theoretical maximum data transmission rate, at which error-free digits can be transmitted over a bandwidth-limited channel in the presence of noise, usually expressed in the form as following:

$$C = Wlog_2(1 + \frac{S}{N}) \tag{1.1}$$

where C is the channel capacity in bits per second, W is the bandwidth in hertz, and $\frac{S}{N}$ is the signal-to-noise ratio (SNR).

In UWB communication systems, even though bandwidth is finite, the bandwidth is at least $500MHz$ or the transmitted signal has a $10dB$ bandwidth larger than 20 percent of its center frequency based on FCC standard. Compared with narrowband system, information-theoretic results in [26] and [27] show that a Shannoncapacity of a multipath fading with Additive White Gaussian Noise (AWGN) wideband channel is a linear function of SNR as shown in (1.2).

$$C = \xi \times SNR \tag{1.2}$$

Moreover, for I-UWB system, data transmission rate R can be formulated as following:

$$R = \frac{1}{N_s N_h T_c} \tag{1.3}$$

where N_s is the number of pulses transmitted at the pulse repetition time, N_h is the number of frames (pulses) per information bit and T_c is the bin duration.

Note that, UWB is flexible in the reconfiguration process of transmission data rate and power, due to the availability of a number of transmission parameters, such as N_s, N_h and T_c, which can be tuned to better match the requirements of data flow. Therefore, for a given level of interference at a receiver, a sender can tune its rate by adjusting the code in order to achieve the desired bit error rate (BER) in UWB systems.

1.7 Sensor Database Systems

In WSNs, the widespread deployment of sensor nodes is transforming the physical world into a computing platform. Sensor nodes not only respond to physical signals for producing data, but also are equipped with computing and communicating capabilities. They are thus able to store, process locally, and transfer data produced. From data storage and process point of view, a WSN can be regarded as a kind of database - distributed wireless sensor database system (WSDS). Compared with traditional database systems, WSDSs store data within system and allow queries to be injected anywhere through query processing operators.

In a WSDS, the query execution usually starts from front-end nodes that issue queries into the system. Sensor nodes are viewed as data sources that provide relevant information for query processing operators. Generally, queries in a WSDS can be classified into three categories depending on the type of data (past, present, or future) requested:

- Historical Queries

  This type of query is mainly used for the analysis of historical data stored at front-end nodes. For example, "what was the temperature two hours ago in the northwest quadrant?"

- One-Time Queries

  This type of query gives a snapshot view of a system. For example, "what is the temperature in the northwest quadrant?"

- Persistent Queries

  This type of query is mainly used to monitor a system over a time interval with respect to some parameters. For example, "report the temperature in the northwest quadrant for next two hours."

Warehousing and on-demand approaches are two conventional ways of handling queries[28]. In the warehousing approach, base stations collect and store data, periodically, depending on a set of predefined parameters. There are two specific limitations for this approach: users can only query base stations, and query processing is very expensive as it utilizes valuable resources like channel bandwidth and energy. In the on-demand way, data is collected based on users' requests. However the drawbacks are that the delay for queries is unacceptable for time critical data, and the flooding of the entire system might be wasteful for one-time queries.

1.8 Related Works

1.8.1 Energy-Efficient MAC Protocols for Wireless Sensor Networks

In contrast to typical MAC protocols of WLAN, MAC protocols designed for WSNs usually trade off performance (such as latency, throughput, fairness) and cost (such as energy efficiency, reduced algorithmic complexity). However, it is not clear what is the best trade-off and various designs differ significantly.

An energy-efficient MAC protocol - power aware multi-access protocol with signaling (PAMAS) [29] for ad hoc networks is proposed in 1999. PAMAS reserves battery power by intelligently powering off users that are not actively transmitting or receiving packets. In this algorithm, two separated channels - control channel and traffic channel - are needed. Following PAMAS, some other solutions for WSNs are put forward. Energy-efficient MAC protocols for WSNs can be classified into three main categories according to strategies applied to channel access: contention-based protocols, TDMA-based protocols and slotted protocols.

As a contention-based energy-efficient MAC protocol, 802.11 [30] standard is based on carrier sensing (CSMA) and collision detection (through acknowledgements). A node

intended to transmit must test the channel whether it is free for a specified time (i.e., DIFS). In [31], Hill and Culler developed a low-level carrier sensing technique that effectively turn radios off repeatedly without losing any incoming data. This technique operates at the physical layer and concerns the layout of PHY pre-pended header of packet. However, energy consumption by collision, overhearing and idle listening is still an unresolved problem. Nevertheless, TDMA-based MAC protocols (i.e., TDMA) have the advantage of avoiding all those energy wastes, since TDMA scheme is inherently collision-free and schedules notify each sensor node when it should be active and, more importantly, when not.

As a TDMA-based energy-efficient MAC protocol, traffic-adaptive medium access (TRAMA) [32] employs a traffic-adaptive and distributed election scheme to allocate system time among nodes. EMACS [33] reduces idle time by forcing nodes to go into dormant mode and to wake up for announcing their presence at the schedule time only. Other TDMA-based energy-efficient MAC protocols such as bit-map-assisted (BMA) protocol and GANGS MAC protocol are described in [34][35]. However, the price to be paid is the fixed costs (i.e., broadcasting traffic schedules) and the reduced flexibility to handle traffic fluctuations and topology change. The third type of energy-efficient MAC protocol - slotted MAC protocols - is proposed and organizes sensor nodes into a slotted system (much like slotted-ALOHA), which strikes a middle ground between the first two ones.

As a slotted energy-efficient MAC protocol, S-MAC [36] is a low power RTS-CTS protocol for WSNs inspired by PAMAS and 802.11. S-MAC includes four major components: periodic listening and sleeping, collision avoidance, overhearing avoidance and message passing. In S-MAC, periodically listening and sleeping are designed to reduce energy consumption during the long idle time. T-MAC [37] improves S-MAC on energy usage by using a quite short listening window at the beginning of active period. To

achieve ultra low power operation, effective collision avoidance and high channel utilization, B-MAC [38] provides a flexible interface and employs an adaptive preamble sampling scheme to reduce duty cycle and to minimize idle listening. While, synchronization among sensor nodes is a strict premise for this kind of protocol.

Besides above works, battery aware MAC (BAMAC(k)) protocol is proposed in [39]. BAMAC(k) is a distributed battery aware MAC scheduling scheme, where nodes are considered as a set of batteries and scheduled by a round-robin scheduler. BAMAC(k) tries to increase node lifetime by exploiting the recovery capacity of batteries. Their work showed how battery awareness influences throughput, fairness and other factors which indicate system performance. In [40], a power control MAC protocol - proposed power control MAC (PCM) is put forward. PCM allows nodes to vary transmission power on the packet basis, which does not degrade throughput and yields energy saving comparing to some simple modifications of IEEE802.11.

1.8.2 Secure MAC protocols for Wireless Sensor Networks

In literature, some secure protocols have been proposed for wireless network. In [41], wired equivalent privacy (WEP) is added to IEEE 802.11[30] standard to bring security into wireless networks. It employs the well-known and believed-secure RC4[42] cipher to bring the security level of wireless systems close to that of wired ones. Perrig proposed a security protocol for WSNs: SPINS[43]. It has two secure building blocks - SNEP and μTESLA. SNEP includes data confidentiality, two-party data authentication, and evidence of data freshness. μTESLA provides authenticated broadcast for severely resource-constrained environments. But, WEP and SPINS cannot effectively deal with the security problems in WSNs caused by DoS attacks. Even though some intrusion detection algorithms for wireless networks are proposed in [44][45][46][47], they are all agent-based. Therefore, additional nodes are needed to execute intrusion detection.

1.8.3 Data Query Processing

The goal of monitoring through sensor nodes is to infer information about objects from measurements made from remote locations. Moreover, inference processes are always less than perfect. Consequently, the problem of uncertainty, which stands for the quality of query answer, is central to monitoring applications. Hence, to build useful information systems, it is necessary to learn how to represent and reason with imperfect information[48]. As a result, Motro[49] is interested in how imperfect information may be represented in a database system. In [50] and [51], they proposed methods of dealing with the uncertainty of moving objective databases. In [52], they presented a unified fuzzy-probabilistic framework for modelling processes of medical diagnosis. In their work, the belief computation is related to diagnostic inference. The final conclusion of inference is the diagnosis with the greatest belief value. It is also shown how their membership functions and basic probability assignments are estimated on the basis of experimental data.

In general, imperfect information is typically handled by attaching a number to it, which represents a subjective measure of the certainty according to the observer. The way, in which the number is manipulated, depends upon the theory that underlies the number. There are possibilistic databases[53] and probabilistic databases[54][55][56]. Probabilistic approach has begun to be used by WSDSs to process query with limited information[57]. In [58], they discussed how to handle aggregate operations in probabilistic databases. They devised a formal linear programming method, analyzed its complexity, and introduced several families of approximation algorithm that run in polynomial time. In [59], a new semantics for database queries is introduced, which supports uncertain matches and ranked results through combining probabilistic relational algebras and models for belief. However, those existing works do not consider the energy constraint problem, nor how to reason and represent the uncertainty introduced by network natures.

Most algorithms for determining query processing strategy for WSDSs are static in nature. In [60], Bodorik proposed the aborted join last (AJL) method to substitute static mechanisms with an adaptive one that owns low overhead and delay to decide when to correct a strategy. In AJL, the decision to correct is computationally simple and, moreover, a corrective strategy is already existed when it is decided to correct. In [61], the phases of adaptive query processing (AQP) is decoupled, and generic framework is constructed. Their work advocated an approach based on self-monitoring algebraic operators. This approach is shown to be generic, independent of any specific adaptation mechanism, easily implementable, and portable.

Considering energy constraint issues, some energy efficient solutions are proposed. Query processing based on random walk technique[62] is an alternative scheme to implement failure recovery in dynamic environments. The robustness of this approach under dynamic situation follows the simplicity of processing, which only requires the connectivity of moving neighbors. In [63] and [64], they proposed an indexing method that supports content-based retrieval queries on a wireless data stream and a tree-structured index to increase the energy efficiency of query processing. To reduce network traffic when accessing and manipulating data, in-network query processing requires placing not only a tree of query processing operators such as filters and aggregations, but also the correlation of nodes. In [65], an adaptive and decentralized algorithm is proposed. This algorithm progressively refines the placement of query processing operators by walking through neighbors. Thus, an initial arbitrary placement of query processing operators can be progressively refined toward an optimal placement.

Existing query processing systems for WSDSs, including Directed Diffusion[66], TinyDB[67], U-DBMS[68] and Cougar[69], provide high-level interfaces that allow users to collect and process such continuous streams. Note that they are especially attractive as ways to efficiently implement monitoring applications without forcing users to write

complex, low-level codes for managing multihop network topology and acquiring samples from sensor nodes. TinyDB, Directed Diffusion and Cougar are relatively mature research prototypes that give some ideas on how future query processing system will function for WSDSs.

1.8.4 MAC Protocols for Ultra Wideband Communication Systems

Conventional wireless MAC protocols assume that simultaneous transmissions result in transmission errors and thus employ mutual exclusion mechanisms to avoid them. Moreover, from layered architecture aspect, the functions executed by MAC should be defined without taking into account the underlying physical layer, which is seen by MAC as a black box offering the service of transferring bits in the form of signals appropriately for the channel. Following this approach, there are some MAC protocols appeared for UWB communication systems. In the European funded project U.C.A.N (Ultra wideband Concepts for Ad-hoc Networks)[70], a TDMA-based MAC protocol is proposed. This MAC protocol is an adaptation for UWB from IEEE 802.15.3[71] draft standard for narrow-band wireless personal area network (WPAN). Ding et al[72] studied the impact of channel acquisition time with different MAC protocols including centralized TDMA and distributed CSMA/CA methods. In [73], a single transceiver approach for UWB and a companion MAC layer based on busy tone multiple access (BTMA) was proposed. BTMA reduces the time and energy spent on collision as compared to handshaking protocols.

However, the design of an efficient MAC protocol for UWB systems should investigate some possible MAC enhancements that will take into account the inherent advantages of UWB technology. In [74], a scheme providing distributed medium access through pulse sense was proposed, which is similar to carrier sense in narrowband system. Moreover, UWB is flexible in the reconfiguration process of transmission data rate

and power. Thus another approach for MAC protocol design is proposed. Opposite to mutual exclusion MAC protocols, this kind of MAC protocol for UWB systems tries to allow simultaneous transmission and to adapt to multiuser interference. Cuomo et al. [75] outlined key issues to design a multi access scheme based on UWB. They selected a distributed mechanism to handle radio resource sharing, and presented a general framework of radio resource sharing to UWB wireless ad hoc networks. An UWB-tailored MAC algorithm - the uncoordinated, wireless, baseborn medium access for UWB communication networks ($(UWB)^2$) is proposed in [76]. $(UWB)^2$ takes advantage of the multiple access capabilities offered by time hopping (TH) codes and relies for the access to a common channel on the high multiple user interference (MUI) robustness provided by the processing gain of impulse radio (IR).

CHAPTER 2

SECURE MAC PROTOCOL

2.1 Denial of Service Attacks on MAC Layer of Wireless Sensor Networks

CSMA/CA protocol mainly concentrates on how to utilize common channel efficiently and fairly when it is designed. It also assumes that each node accesses the common channel following the same multiple access scheme strictly. This assumption is the premise to guarantee the common channel be shared successfully and efficiently among nodes. However, it leaves chances for DoS attacks on MAC layer through violating this rule on purpose.

Analyzing the working mechanism of CSMA/CA protocol, the characteristics of DoS attacks and the limited capabilities of WSNs, potential DoS attacks on MAC layer of WSNs can be classified into three categories: collision attack, unfairness attack and exhaustion attack. Basing on the classification of DoS attacks, we correspondingly choose our indicators for intrusion detection.

2.1.1 DoS Attack Classification

1. Collision Attack

   In collision attacks, adversaries conduct attacks through sending attack packets onto the busy channel. Those attack packets will collide with control packets (RTS, CTS and ACK) and data packets from normal sensor nodes. In Fig.2.1, grey rectangles stand for packets sent by collision attacker. Collision is supposed to happen when there is an overlap between a normal sensor node's transmission and

a collision attacker's transmission. In this case, data transmissions of normal sensor nodes fail due to collision.

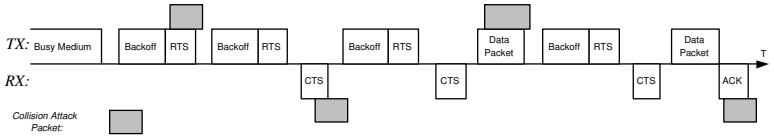


Figure 2.1. Failure transmissions due to collision attacks.

2. Unfairness Attack

In unfairness attack case, attackers have higher chance to access common channel than other normal sensor nodes. Unfairness attackers send attack packets to the common channel immediately once sensing channel is idle. In that sense, unfairness attackers prevent other normal sensor nodes from transmitting. In Fig. 2.2, grey rectangles stand for packets coming from unfairness attackers.

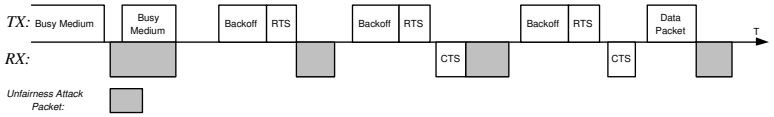


Figure 2.2. Failure transmissions due to unfairness attacks.

3. Exhaustion Attack

Exhaustion attacks are supposed to happen when adversaries abnormally send a great amount of RTS to normal sensor nodes. RTS/CTS-based MAC protocols are transmitter invitation MAC protocols. That means, when a receiver receives a RTS successfully, it must respond that invitation with a CTS. Moreover, adversaries are compromised from normal sensor nodes, which makes the receiver cannot tell whether RTS is sent by normal sensor nodes or by adversaries. Under this condition,

if adversaries send RTS to normal sensor nodes repeatedly, normal sensor nodes have to respond those RTS incessantly with CTS, which will lead to the exhaustion of battery resources of those normal sensor nodes.

2.1.2 Intrusion Indicator Design

When collision attacks intrude the network, attackers send massive packets into the common channel when detecting channel is busy. As a result, more RTS, CTS, ACK and even data packets may be destroyed due to collisions caused by attackers (See Fig.2.1). Moreover, the average latency of data packet is prolonged because of more retransmitting RTS packets and data packets.

Under normal condition, without any attacks, it is fair for each node to transmit data over the common channel from a long-term statistic view, since nodes have to wait for a random time before sending RTS to try to hold the common channel (see Fig.1.1), and only the first successful one can be allowed to transmit data over the common channel. Data packets of normal sensor nodes have to wait longer time at MAC layer, when unfairness attackers prevent other normal sensor nodes from transmitting by holding the channel ahead of time.

Differing from collision attack and unfairness attack, exhaustion attackers work almost same as other normal sensor nodes, except for sending RTS repeatedly to some normal sensor nodes. As a result, the arrival rate of RTS at victim nodes will increase dramatically. Besides this, data packets should wait longer time, since the common channel is more utilized for transmitting RTS by attackers.

From above analysis, we can see that intrusions may be detected by monitoring abnormal alterations of some sensitive network parameters. They are:

- A great number of RTS packets are received by victim nodes for exhaustion attacks;

- Average waiting time becomes very long for both unfairness attacks and collision attacks;
- Collision takes place considerably often in collision attacks.

In our algorithm, we choose arrival rate (R_{RTS}), average waiting time (T_w) and collision rate (R_c) as our intrusion indicators.

- Collision Rate (R_c): R_c is the collision number detected by a node per second.
- Average Waiting-Time (T_w): T_w is the waiting time of data packet at MAC layer.
- RTS Arrival Rate (R_{RTS}): R_{RTS} is the number of RTS received successfully by a node per second.

Fixing a network's traffic strength, node density, node capacity, etc., we add collision attack, unfairness attack and exhaustion attack respectively into this network, which runs on CSMA/CA protocol. Table 2.1 displays a set of mean and variance values for R_{RTS}, T_w and R_c. There is no DoS attacks for normal scenario. collision attack, unfairness attack and exhaustion attack are introduced into collision attack scenario, unfairness attack scenario and exhaustion attack scenario separately.

Table 2.1. Mean and variance values for R_{RTS}, T_w and R_c

$Mean/Variance$	R_{RTS}	$T_w(ms)$	R_c
$Normal\ Scenario$	9.9/2.14	4/0.0	0.07/0.19
$Collision\ Attack\ Scenario$	6.0/4.78	290/370	146/157.19
$Unfairness\ Attack\ Scenario$	5.9/4.72	180/370	0.15/0.35
$Exhaustion\ Attack\ Scenario$	22.1/14.3	270/410	1.8/2.16

Note that different attacks result in different abnormal changes on R_{RTS}, T_w, R_c:

- When a collision attacker intrudes a network, the mean values of R_c is about 2084 times bigger and T_w is about 71 times longer than the normal value;

- For unfairness attack, the mean value of T_w is about 44 times longer than the normal value;
- For the exhaustion attacks, the mean values of R_{RTS} is about 1 times larger and T_w is about 66 times longer than the normal value.

Those sharp changes in Table 2.1 verify the feasibility of our intrusion detection algorithm. Thus, once we detect some unusual changes on R_{RTS}, T_w or R_c, we can infer, with certain reliability, that there are DoS attacks intruding the network. Based on this analysis, we design our FLS to implement intrusion detection.

2.2 Fuzzy Logic Based Secure MAC (FSMAC) Algorithm Description

The structure of our proposed secure MAC algorithm is shown in Fig.2.3. We improve the security of MAC layer through adding two special modules - intrusion detection module and intrusion defense module - into the original MAC layer. Each sensor node has its own security system.

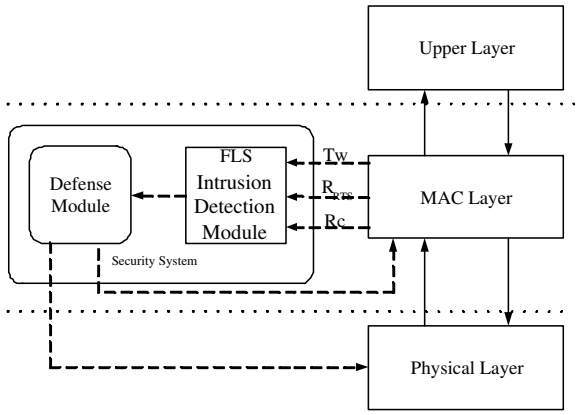


Figure 2.3. Secure MAC algorithm model.

In FSMAC, intrusion module of each sensor node monitors R_{RTS}, T_w and R_c, and periodically judges whether intrusion exists or not. If intrusion is found, the defense

module of this sensor node will be trigged by FLS intrusion detection module. Defense module will inform physical layer and MAC layer to switch to different RF band to make transmission, or pause for a while. Then, after a period of time, sensor node will switch back to the original RF band, or restarts information exchanging over the network. Intrusion detection will be resumed also.

2.2.1 Intrusion Detection Strategy

From Table 2.1, we notice that various intrusion indicators are not identically sensitive to different DoS attacks. Collision attacks extremely increase data average waiting time and data packet collision rate, unfairness attacks extremely increase data average waiting time only and exhaustion attacks increase RTS arrival rate and data average waiting time so much. It looks like the abnormal change on the data average waiting time T_w can indicate those three kinds of DoS attacks. However, measurements are usually corrupted by noise; hence, they are uncertain. To increase the accuracy of our intrusion detection, we choose all of three intrusion indicators, i.e., R_{RTS}, T_w and R_c, to make the decision.

Moreover, since the distributed DoS attacks (DDoS)[77] can use many computers to launch a coordinated DoS attack against one or more targets, it is possible for more than one type of DoS attacks appeared within a system. Through monitoring R_{RTS}, T_w and R_c simultaneously, we can make intrusion decision without classifying the attack type.

Based on above considerations, it is an essential part that how to combining those three intrusion indicators together. There are two main approaches: linear combination methods and nonlinear combination schemes. For linear combination, the probability of intrusion found P_{if} can be expressed as $P_{if} = \alpha \times R_{RST} + \beta \times T_w + \gamma \times R_c$. The key task is to determine the values for combination factors α, β and γ for optimizing the

performance in terms of higher detection probability and lower false alarm rate. There are maximum likelihood (ML)[78], equal gain (EG) and maximal-ratio combining (MRC) methods[79].

For nonlinear combination approach, FLS optimize combination through rules and fuzzy logic. FLS nonlinearly maps a set of input data into a output; the output case decomposes into a collection of independent multi-input/sign-output systems. The richness of fuzzy logic is that there are enormous numbers of possibilities that lead to lots of different mappings. Moreover, a FLS is well-known for being able to handle linguistic characterizations and uncertainty of data. As an extending of our previous work, we design a fuzzy logic model for intrusion detection in this paper.

2.2.2 Fuzzy Logic Intrusion Detection Module Design

Designing a FLS can be viewed as given a set of input-out pairs, tuning is essentially equivalent to determining a system that provides an optimal fit to the input-output pairs, with respect to a cost function. There exists a multitude of design methods that can be used to construct FLSs that have different properties and characteristics. There are data-intensive ways, computational simplicity approaches, recursive methods, offline ways and application specific approaches[80].

Each sensor node monitors R_{RTS}, T_w and R_c, which are the inputs (or antecedents) of FLS intrusion detection module (See Fig.2.3). Then according to the output (or consequent) - possibility of intrusion found by FLS intrusion detection module, the security system makes a decision whether intrusion exists or not.

There are three main steps for our FLS design:

- Fix the shapes and parameters of all the antecedent and consequent membership functions ahead of time. The data establish the rules and no tuning is used;

- Fix the shapes and parameters of the antecedent membership functions ahead of time. Use the training data to tune the consequent parameters;
- Fix the shapes of all the antecedent and consequent membership functions ahead of time. Use the training data to tune the antecedent and consequent parameters.

Based on the preceding steps, for our FLS, the linguistic variables used to represent R_{RTS} and R_c are divided into two levels: Low and $High$; and those that are used to represent T_w are divided into two levels: $Short$ and $Long$. The result - the possibility that a sensor node finds intrusion - is divided into 5 levels, $Very\ Low$, Low, $Moderate$, $High$ and $Very\ High$. We show the membership functions in Fig.2.4(a) and Fig.2.4(b). From these two figures we can see that every antecedent has 2 fuzzy subsets, so we need to set up $2^3 = 8$ (because every antecedent has 2 fuzzy subsets, and there are 3 antecedents) rules for this FLS.

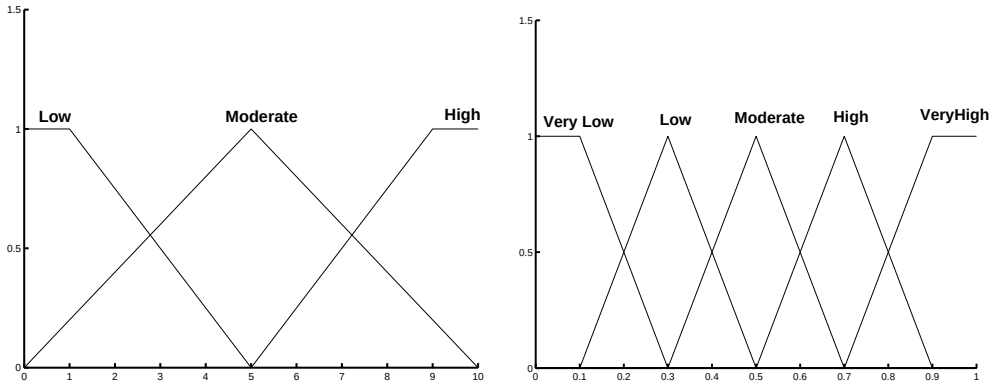


Figure 2.4. (a)Antecedent and (b)Consequent Membership Function.

Based on the fact that when any type of DoS attacks (i.e., collision attack, unfairness attack or exhaustion attack) intrudes a network, R_{RTS}, T_w or R_c becomes sharply high/long. The sharper the change is, the more the possibility of detecting intrusion is. We design a fuzzy logic system using rules such as:

R^l: IF *collision rate* (x_1) *of a node detected* is *High*, *average waiting time* (x_2) *of a node collected* is *Long*, and *RTS arrival rate* (x_3) *of a node received* is *Low*, THEN *the possibility* (y) *that a intrusion found by this node* is *High*.

Where $l = 1,2,\ldots,8$. We summarize all the rules in Table 3.1. Ante1 is collision rate, Ante2 is average waiting time, Ante3 is RTS arrival rate. Consequent is the possibility that intrusion is found.

Table 2.2. Rules for intrusion detection

Rule	*Ante*1	*Ante*2	*Ante*3	*Consequent*
1	*Low*	*Short*	*Low*	*VeryLow*
2	*Low*	*Short*	*High*	*Moderate*
3	*Low*	*Long*	*Low*	*Moderate*
4	*Low*	*Long*	*High*	*High*
5	*High*	*Short*	*Low*	*Moderate*
6	*High*	*Short*	*High*	*Low*
7	*High*	*Long*	*Low*	*High*
8	*High*	*Long*	*High*	*VeryHigh*

Defuzzification produces a crisp output for the FLS from the fuzzy sets that appear at the output of the inference block. Many defuzzifiers have been proposed in the literature, such as centroid defuzzifier, center-of-sums defuzzifier, height defuzzifier and center-of-sets defuzzifier[81]. One criterion for the choice of a defuzzifier is computational simplicity. Based on this criterion, in our design, for every input(x_1,x_2,x_3), the output is computed using

$$y(x_1, x_2, x_3) = \frac{\sum_{l=1}^{8} \bar{y}^l \mu_{\mathrm{F}_1^l}(x_1)\mu_{\mathrm{F}_2^l}(x_2)\mu_{\mathrm{F}_1^3}(x_3)}{\sum_{l=1}^{8} \mu_{\mathrm{F}_1^l}(x_1)\mu_{\mathrm{F}_2^l}(x_2)\mu_{\mathrm{F}_3^l}(x_3)} \tag{2.1}$$

The height of the five fuzzy sets depicted in Fig.2.4(b) are $\bar{y}_1$=0.1, $\bar{y}_2$=0.3, $\bar{y}_3$=0.5, $\bar{y}_4$=0.7, $\bar{y}_5$=0.9.

Since the normal behaviors of system are often changed in WSNs, to keep our FLS always be optimal tuning are needed periodically. That is, after a period of time, named coherent time for a system in which the system status is relatively fixed, the parameters of all antecedent and consequent and the rules are tuned based on the newest training data. That dynamic adjustment operation can be done offline, then doing system updating.

2.2.3 Defense Module Design

When intrusions are found, the defense module is triggered, taking some countermeasures to reduce the effects of attackers on the network. In fact, it is an energy waste or unsafe action for normal sensor nodes to transmit or receive information during the intrusion period. That is, transmission or reception is almost unsuccessful or spied by attackers when enemies attack a network. Thus it is an appropriate and effective choice for normal nodes to switch to different RF bands to make transmission or to pause for a while.

In this paper, we focus on intrusion detection. So just as an example, we choose pausing for a while to implement defense. There is no information on attacks, thus we can't know the duration of attacks. Therefore, after a period of sleep, nodes should wake up to resume data communication and intrusion detection. Nodes will stay at this state until intrusion is found again. At sleep mode, there is no transmitting or receiving, but sensing (such as collecting data of temperature, humidity, or degree of air pollution) still continues.

In order to make our secure algorithm available for general WSNs, in which there is no center control, our defense scheme, like intrusion detection scheme, is also distributed.

2.3 Simulation and Performance Analysis

We used the simulator OPNET to run simulations. A network with 30 nodes is set up and the radio range (radius) of each node is 50m. Those nodes are randomly deployed in an area of $100 \times 100 m^2$ and have no mobility. In order to simplify the analysis about the performance of our security MAC algorithm, we exclude the factors coming from physical layer and network layer in our experiments. Table 3.3 summarizes the parameters used by our simulations. Packet size is 1000 bytes. The destination for each node's traffic is randomly chosen from its neighbors. As in [28][82], data packets arrive according to a Poisson process with certain rate in our simulations.

Table 2.3. Physical layer parameters

W_{wmin}	32	W_{wmax}	1024
MAC Header	34 bytes	ACK	38 bytes
CTS	38 bytes	RTS	44 bytes
SIFS	10 μsec	DIFS	50 μsec
ACK-Timeout	212 μsec	CTS-Timeout	348 μsec

All sensor nodes are set initially with energy of 2J. We use the same energy consumption model as in [83] for the radio hardware. To transmit an l-symbol message a distance d, the radio expends:

$$E_{Tx}(l,d) = E_{Tx-elec(l)} + T_{Tx-amp(l,d)} = l \times E_{elec} + l \times e_{fs} \times d^2 \tag{2.2}$$

and to receive this message, the radio expends:

$$E_{Rx} = l \times E_{elec} \tag{2.3}$$

The electronics energy, E_{elec}, as described in [83], depends on factors such as coding, modulation, pulse-shaping and matched filtering. The amplifier energy, $e_{fs} \times d^2$ depends

on the distance to the receiver and the acceptable bit error rate. In this paper, we choose: $E_{elec} = 50nJ/sym$, $e_{fs} = 10pJ/sym/m^2$.

The attacker is an abnormal sensor node, which has been captured and reprogrammed by enemies successfully before the system starts to work. In our simulations, attackers start the intrusion randomly, and each attack lasts for a random length of period.

We define a parameter metrics to testify the performance of our algorithm. It is

- Possibility of detection (P_d): the possibility of nodes making correct detection when there is attack;
- False alarm rate (P_{fd}): the possibility of nodes making false detection when there is no attack;
- Data packet successful transmission rate (R_{st}): the rate of successfully transmitted data packets to all data packets transmitted;
- Average energy consumption (E_{av})(J/Pk): the energy consumed per successful transmission data packet;
- Time of first node dead (T_s)(Second): the time that the first node, in the network, runs out of power. When the energy of a node is used up, we assume this node is dead.

We separately test the influences of each DoS attack - collision attack, unfairness attack and exhaustion attack. In each experiment, there is only one type of attack introduced. We compared our FSMAC against original CSMA/CA protocol. In Fig.2.5, we plot the simulation time versus the successful transmission rate. Observe that data packet successful transmission rate for FSMAC has been increased about 25% for each DoS attack. In Fig.2.6, we compared the average energy consumption for successful data transmission of these two schemes. Observed that, our FSMAC outperforms the original CSMA/CA for about 5% less energy consumption per data packet. In Table

2.4, FSMAC extends the time of first node dead about one time, compared to original CSMA/CA scheme.

For CSMA/CA without any attacks, the time of first node dead is about 200s. But Table 2.4 even shows FSMAC has longer lifetime than CSMA/CA without attacks case. The reason is that some sensor nodes switch to sleep state when intrusion found. At sleep state, some energy is reserved for no transmitting and receiving. For three different types of attack, the false alarm rate of our algorithm is 0, and the probability of detection is 100%.

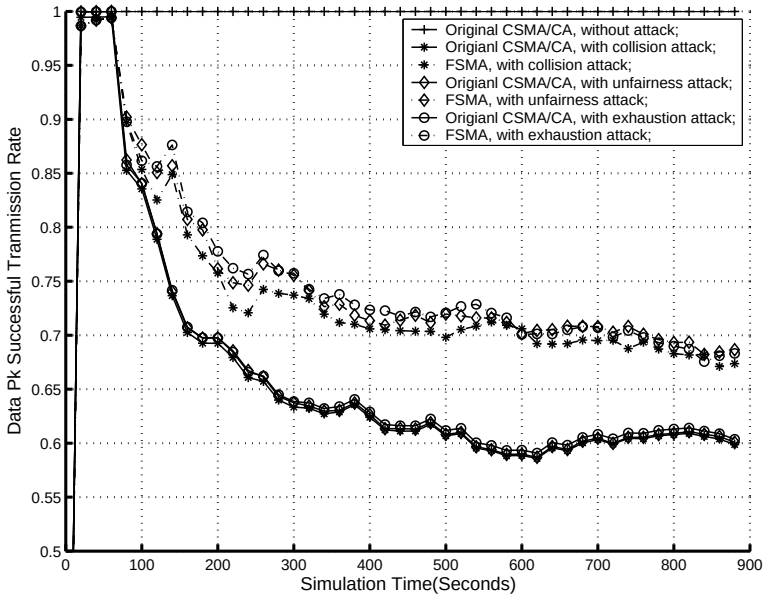


Figure 2.5. Rate of Data Packet Successful Transmission.

Table 2.4. Time that the First Node Dead

	Collision Attack		Unfairness Attack		Exhaustion Attack	
	CSMA/CA	FSMAC	CSMA/CA	FSMAC	CSMA/CA	FSMAC
Ts	126s	271s	107s	229s	126	275s

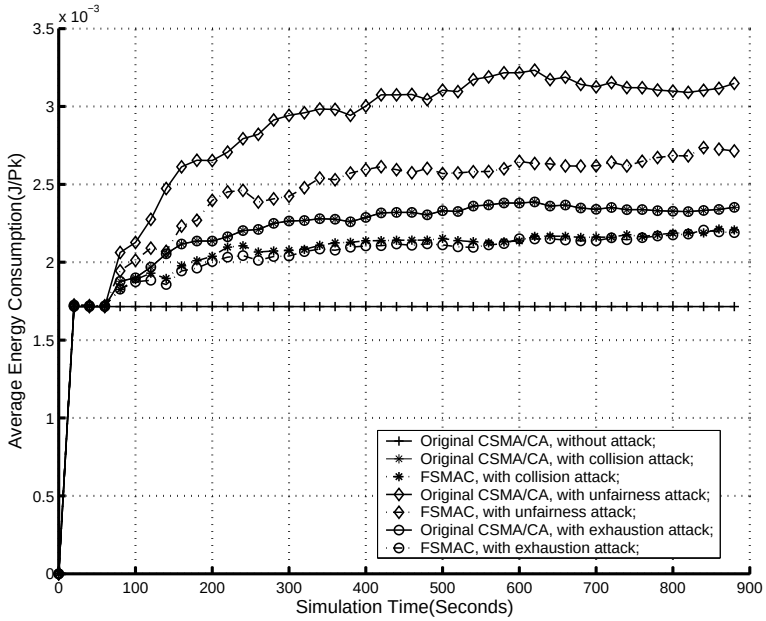


Figure 2.6. Average Energy Consumption.

We also test the impact of combining DoS attack. That is, three types of DoS attacks intrude the network together(See Fig. 2.7 and 2.8).

2.4 Conclusion

In this chapter, we classify popular DoS attacks on MAC layer of WSNs into three categories: collision attack, unfairness attack and exhaustion attack. We firstly choose RTS arrival rate, average waiting-time and collision rate as our indicators for intrusion detection. Based on the classification of DoS attacks and selection of intrusion indictors, we propose a secure MAC protocol for WSNs - FSMAC, a fully distributive method. In our algorithm, a fuzzy logic system is designed for making decision on intrusion. The simulation results verify that:

- Our algorithm can detect all intrusions without any false alarm;
- Failure data transmission, caused by DoS attacks, could be alleviated 25% by our secure method;

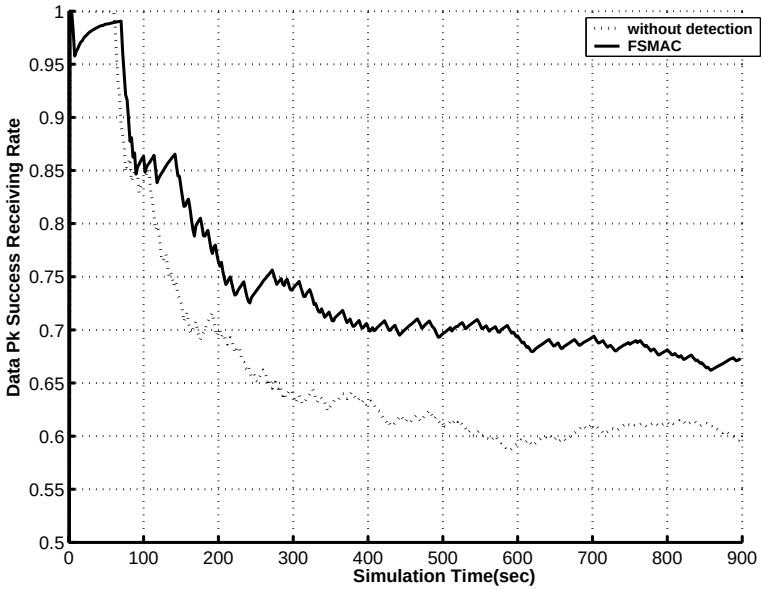


Figure 2.7. Rate of Data Packet Successful Transmission.

- half energy is reserved through reducing the energy waste for failure transmission caused by DoS attacks.

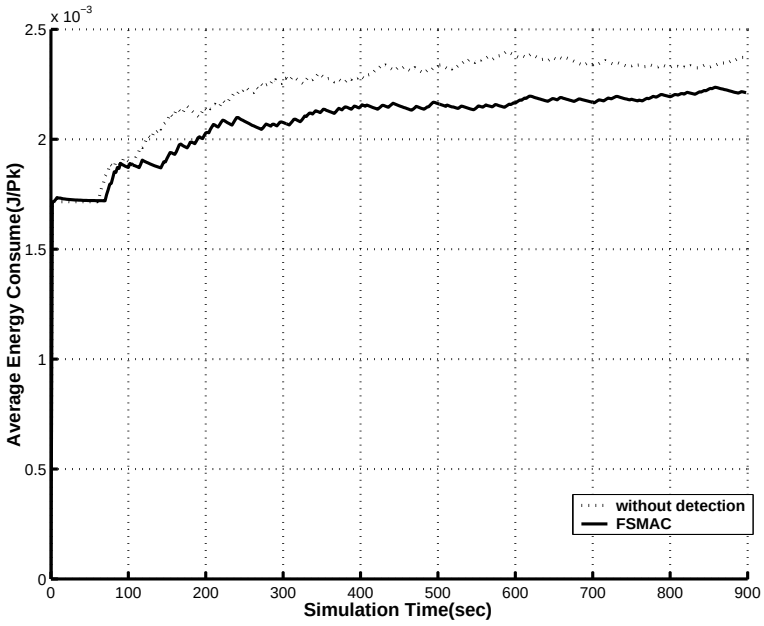


Figure 2.8. Average Energy Consumption.

CHAPTER 3

ENERGY-EFFICIENT MAC PROTOCOLS FOR WSNS

Commonly, a distributed WSN is composed of a set of low-end data-gathering sensor nodes and high-end data-collection sensor nodes. In kinds of network, data-collection sensor nodes collect the data about a physical phenomenon and send them to related data-gathering sensor nodes that act as lead-sensor or fusion center over wireless links. For example, in [84][85], such network model was employed for investigating the energy efficiency of distributed coding and signal processing. Similar model was employed in [86] to develop a collaborative and distributed tracking algorithm for energy-aware WSNs.

In a WSN with hierarchical topology, communications can be divided into three main categories based on communication terminals, i.e., communication between data-collection nodes, communication between data-gathering nodes, as well as communication between data-collection nodes and data-gathering nodes. In this chapter, we mainly focus on how to design energy-efficient MAC protocols to organize the communication between data-collection nodes, which suffer from power constraint strictly. This type of communication is quite common in general WSNs. For instance, data-collection nodes exchange their collected information before sending it to data-gathering nodes to reduce information redundancy caused by position correlation of nodes. Another example is given in [87], in which to implement V-BLAST based virtual multiple-input multiple-output (MIMO) communication, data-collection nodes share their collected information with each other before transmission.

3.1 Asynchronous MAC (A-MAC) protocol

Asynchronous MAC (A-MAC) protocol divides system time into four phases: TRFR-Phase, Schedule-Phase, On-Phase and Off-Phase (Fig. 3.1).

- TRFR-Phase is preserved for data-collection nodes to send Traffic-Rate & Failure-Rate (TRFR) messages to data-gathering nodes;
- Schedule-Phase is preserved for data-gathering nodes to locally broadcast phase-switching schedules;
- Off-Phase is preserved for data-collection nodes to power off their radios. In this phase, there is no communication, but data storing and sensing may happen;
- On-Phase is preserved for data-collection nodes to power on their radios to carry on communication.

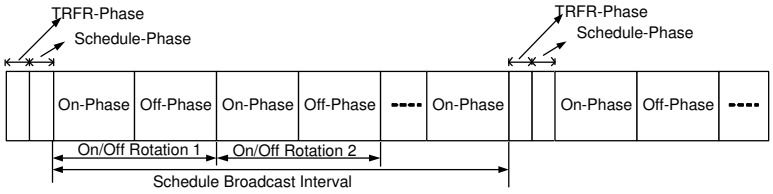


Figure 3.1. Time scheme structure for A-MAC.

In our system, at the end of On-Phase, nodes go to "vacation" - Off-Phase, for a period of time. Thus, new arrivals during an On-Phase can be served in first-in-first-out (FIFO) order. While, new arrivals during an Off-Phase, rather than going into service immediately, wait until the end of this Off-Phase, then they are served in On-Phase and in FIFO order. Interarrival time and service time for data packets are independent and follow general distribution $F(t)$ and $G(s)$ individually. For average interarrival time $\frac{1}{\lambda}$, we have $0 < \frac{1}{\lambda} = \int_0^\infty t dF(t)$. Similarly, for average service time μ, we have $0 < \mu = \int_0^\infty s dG(s)$.

3.1.1 Essential Parameter Design

3.1.1.1 *O*ff-*P*hase Duration (T_f)

We treat each node as a single-server queuing system during our analysis on the waiting-time of data packets. Note that most data packets arrive during either *O*ff-*P*hase or *O*n-*P*hase. The waiting-time of data packet w_{ij} can be expressed as

$$w_{ij} = T_{f,j} - \sum_{l=1}^{i} t_{lj} + \sum_{l=1}^{i-1} s_{lj} \tag{3.1}$$

where

- t_{ij} denotes the interarrival time for the i-th arrived data packet for node j, and $t_{1j}, t_{2j}, \cdots, t_{Nj}$ are independent and identically distributed (iid) random variables.
- s_{ij} denotes the service time for the i-th arrived data packet for node j, and $s_{1j}, s_{2j}, \cdots, s_{Nj}$ are iid random variables also.
- N is the total number of data packets arrived during one On/Off Rotation, and n is the number of data packets arrived only during an *O*ff-*P*hase, i.e., $\sum_{l=1}^{n} t_{ij} \leq T_f < \sum_{l=1}^{n+1} t_{ij}$ and $\sum_{l=n+1}^{N} t_{ij} \leq T_n < \sum_{l=n+1}^{N+1} t_{ij}$.

Note that w_{ij} is a function of t_{ij}, s_{ij} and $T_{f,j}$. $T_{f,j}$ is a constant for each On/Off Rotation during a schedule-broadcast interval. While t_{ij} and s_{ij} are random variables with probability distribution function (PDF) $f_j(t_i) = f_j(t) = F_j'(t)$ and $g_j(s_i) = g_j(s) = G_j'(s)$ respectively. In this case, the average waiting-time $\bar{w}_{ij}$ can be formulated as:

$$\begin{aligned}
\bar{w}_{ij} &= \int_0^\infty \cdots \int_0^\infty \int_0^\infty \cdots \int_0^\infty w_{ij} h(t_{1j}, t_{2j}, \cdots, t_{ij}, s_{1j}, s_{2j}, \cdots, s_{i-1j}) dt_{ij} \cdots dt_{1j} ds_{i-1j} \cdots ds_{1j} \\
&= \int_0^\infty \cdots \int_0^\infty \int_0^\infty \cdots \int_0^\infty w_{ij} \prod_{l=1}^{i} f_j(t_l) \prod_{l=1}^{i-1} g_j(s_l) dt_{ij} \cdots dt_{1j} ds_{i-1j} \cdots ds_{1j}
\end{aligned} \tag{3.2}$$

where $h(t_{1j}, t_{2j}, \cdots, t_{ij}, s_{1j}, s_{2j}, \cdots, s_{i-1j})$ is the joint PDF of $t_{1j}, t_{2j}, \cdots, t_{ij}$ and $s_{1j}, s_{2j}, \cdots, s_{i-1j}$.

Considering λ, μ and (3.1), we can rewrite (3.2) as following:

$$\bar{w}_{ij} = T_{f,j} - \frac{i}{\lambda_j} + (i-1)\mu_j \tag{3.3}$$

Note that, when fixing data arrival rate λ_j and service time μ_j, the longer *O*ff-*P*hase duration $T_{f,j}$ is, the longer data packets waiting-time $\bar{w}_{ij}$ is. Moreover, based on (3.3), the difference of average waiting-time between the i-th arrived packet and the k-th arrived packet for node j is shown in (3.4).

$$\Delta\bar{w}_j(ik) = (k-i)(\frac{1}{\mu_j - \lambda_j}) \quad (k \geq i) \tag{3.4}$$

Obviously, the earlier arrived data packet waits longer time than the later ones if the queuing system is not overloaded (i.e. $\mu\lambda < 1$) and is served in FIFO. In order to keep data packets up to date, $\bar{w}_{ij}$ should be no longer than the maximum acceptable waiting-time W_{max}, which is specified by applications. So $T_{f,j}$ should satisfy

$$T_{f,j} - \frac{1}{\lambda_j} \leq W_{max} \tag{3.5}$$

T_f is the power-off duration for all nodes within a cluster. In order to ensure data packets from all nodes up to date, it is reasonable to choose the shortest duration of $T_{f,j}$ as a cluster's sleep duration. Then we have (3.6) for T_f.

$$T_f \leq \min_j(W_{max} + \frac{1}{\lambda_j}) \tag{3.6}$$

The expected duration, denoted by t, within which node j's buffer will be fully loaded, is given by $t = \frac{k_j}{\lambda_j}$, where k_j is the buffer size for node j. So $T_{f,j}$ should also satisfy the following constraint to avoid buffer overflowing.

$$T_{f,j} \leq t = \frac{k_j}{\lambda_j} \tag{3.7}$$

Since there are multiple nodes that have various buffer size and traffic arrival rate within a cluster, the power-off duration of a cluster should ensure no buffer overflow for

all nodes. Hence setting T_f equal to the shortest duration of $T_{f,j}$ determined by (3.7) can satisfy this criterion.

$$T_f \leq \min_j(\frac{k_j}{\lambda_j}) \tag{3.8}$$

Combining (3.8) and (3.6), the optimum value of T_f can be obtained through (3.9).

$$T_f = \min\left\{\min_j(W_{max} + \frac{1}{\lambda_j}), \min_j(\frac{k_j}{\lambda_j})\right\} \tag{3.9}$$

3.1.1.2 *O*n-*P*hase Duration (T_n)

During *O*n-*P*hase, data-collection nodes start to send data packets through competition. The contention process is similar to 802.11 DCF scheme. In A-MAC algorithm, a transmission is treated as an unsuccessful one when retransmission time exceeds a threshold - N_{ret}. We utilize the same model to calculate the value of N_{ret} as in [88], and only data packets will do retransmission.

If we let the duration of *O*n-*P*hase for node j be $T_{n,j}$, according to Little's Theorem [89], the total number of data packets (N_j) arrived during an On/Off Rotation is given by

$$N_j = \lambda_j(T_f + T_{n,j}) \tag{3.10}$$

In our *O*n-*P*hase duration and *O*ff-*P*hase duration design, we not only try to extend the power-off duration to reserve energy (by avoiding excessive idle listening), but also to ensure data packets up to date. So the optimum value for $T_{n,j}$ is

$$\frac{T_{n,j}}{\mu_j} = \lambda_j(T_{n,j} + T_f) \tag{3.11}$$

T_n is the power-on duration for a cluster. In order to ensure all nodes to have enough time to send buffered data packets out, we choose the longest duration of T_{nj} as a cluster's active duration.

$$T_n = \max_j\left\{\frac{\lambda_j T_f \mu_j}{1 - \mu_j\lambda_j}\right\} \tag{3.12}$$

For 802.11 DCF scheme, the service time for data packets consists of back-off time and transmission time as in (3.13).

$$\mu_j = \bar{T}_{B,j} + \frac{L_d}{R_j} \tag{3.13}$$

where R_j is the data transmission rate for node j. L_d is the size of data packet.

Researches in [90][91] showed the theoretic result on average backoff time ($\bar{T}_B$) of data transmission in 802.11 DCF scheme. Under the assumption that stations always have a packet available for transmission, in other words, the system operates in saturation condition, $\bar{T}_B$ is determined by (3.14).

$$\bar{T}_B = \sum_{k=1}^{\infty}(\sum_{l=1}^{k}\frac{\alpha}{2}w_l)\,(1-q)^{k-1}\,q - \frac{\alpha}{2q} + \frac{1-q}{q}t_c \tag{3.14}$$

Where

- q is the conditional successful probability;
- $\alpha = \sigma p_i + t_c p_c + t_s p_s$;
- p_s is the probability of successful transmission. p_i is the probability of the channel being idle. p_c is the probability of collision. Moreover, p_s+p_i+p_c=1;
- σ is the time during which the channel is sensed idle. t_c is the average time during which the channel is sensed busy due to a collision in the channel. t_s is the average time that the common channel is sensed busy due to a successful transmission;
- w_l is the contention window size at the l-th backoff stage.

In A-MAC, for node j, the probability q_j that a packet is successfully transmitted at the end of a backoff stage is linear with its traffic strength. That is, $q_j = \frac{q\lambda_j}{\sum_{l=1}^{N}\lambda_l}$. We assume there are N nodes within this cluster, and each node accesses the common channel following 820.11 DCF scheme. Moreover the duration of On-Phase is designed to be just

long enough to let all arrived data packets be sent out. In this case, the assumption for (3.14) is still held, but the average backoff time for our A-MAC is modified as following:

$$\bar{T}_{B,j} = \sum_{k=1}^{\infty}(\sum_{l=1}^{k}\frac{\alpha}{2}w_l)\,(1-q_j)^{k-1}\,q_j - \frac{\alpha}{2q_j} + \frac{1-q_j}{q_j}t_c \tag{3.15}$$

3.1.1.3 *T*RFR-*P*hase Duration

At the beginning of TRFR-Phase, nodes estimate their data arrival rate, service time, transmission failure rate and buffer overflowing rate over on/off rotations independently. That information will be forwarded to data-gathering nodes through TRFR messages (Fig. 3.2).

T (2 bits)	SRC (8 bits)	AR_d (8 bits)	SR_d (8 bits)	FR (8 bits)	OR (8 bits)

T	packet type	SR_d	data service rate
SRC	source address	FR	transmission failure rate
AR_d	data arrival rate	OR	buffer overflowing rate

Figure 3.2. TRFR message format.

In this situation, data-gathering nodes become bottlenecks in increasing the chance for TRFR messages being successfully transmitted. Our strategy is to make the transmission time for each TRFR message comply with an uniform distribution and carrier sensing is done before sending. Since hidden-problem is accessible for our system, the performance will be worse compared with using CSMA/CA scheme. Following experiments show the chance for a TRFR message being successfully transmitted.

Fixing the duration of TRFR-Phase from 5 to 30 seconds and increasing the number of node within a cluster from 5 to 30, we obtain a branch of curves on successful transmission rate for TRFR messages (Fig. 3.3). Note that, TRFR's successful transmission rate is impacted by node density (which is defined as how many nodes is there over

an area) and the length of TRFR-Phase. From experimental results, we can choose a suitable duration for TRFR-Phase to ensure that data-gathering nodes can acquire necessary information from data-collection nodes to determine system schedule successfully.

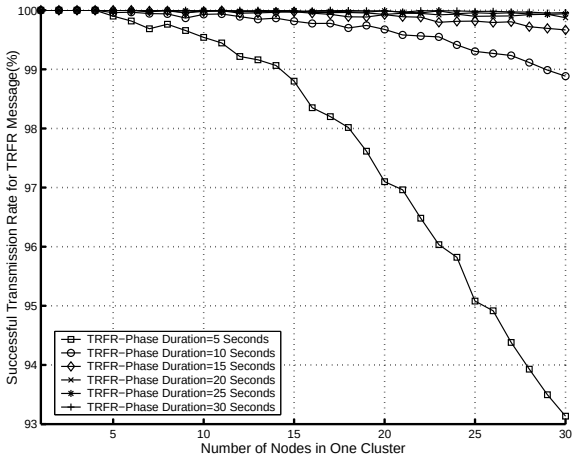


Figure 3.3. Successful transmission rate for TRFR message.

3.1.2 Matching Schedule Establishment and Maintenance

According to received schedule messages (Fig. 3.4), nodes set up their own phase-switching schedules, which ensure them to switch to the same phase simultaneously. To simplify the schedule setting up process, we consider, firstly, the scenario in which there is no clock-drift and traffic is time-invariant.

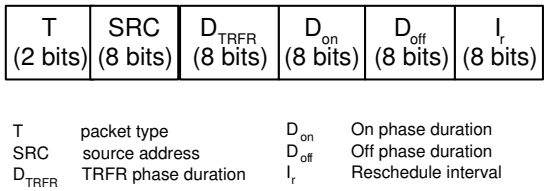


Figure 3.4. Schedule message packet format for A-MAC.

We utilize two techniques to make our scheme robust and feasible to use free-running timing method [9], which allows nodes to run on their own clocks and makes contribution to save the energy used by setting up and maintaining the global or common timescale. Firstly, schedule messages are broadcasted. Leveraging the property of broadcast, schedule messages can reach all data-collection nodes at the same time, once we ignore the difference of propagation time of them (it is reasonable since the propagation time within a cluster, which is between 0.1 and 1 microseconds). Moreover, nodes go to On-Phase immediately after receiving schedule messages. Secondly, in a schedule message, all time references, such as On-Duration and Off-Duration, are relative values rather than absolute values. This property can eliminate errors introduced by sending time and access time. Hence each node within a cluster is synchronized to a reference packet (schedule message) that is injected into the physical channel at the same instant. Furthermore, after a same period of time specified by T_n, all nodes switch to Off-Phase and stay there for a T_f period. Finally, all nodes switch back to On-Phase. Phase is circulatedly switched like this way (See Fig. 3.5).

Note that, based on schedule messages and nodes' local clocks, phase-switching schedules are supposed to be established at each node to ensure matching operations if no clock-drift. Obviously, there is no global or common timescale in our system.

As we mentioned earlier, however, mismatching operations among nodes are unavoidable, since there are always clock-drifts caused by unstable and inaccurate frequency standards. So it is possible that transmitters have powered on their radios to send message, but receivers' radios are still powered off. Those mismatching operations cause communication to fail. Moreover, with the accumulative clock-drift becoming bigger and bigger, the impact on communications turns to be more and more serious.

Our solution is to re-broadcast schedule message, which forces data-collection nodes to remove accumulative clock-drifts and to re-establish matching schedules. While, how

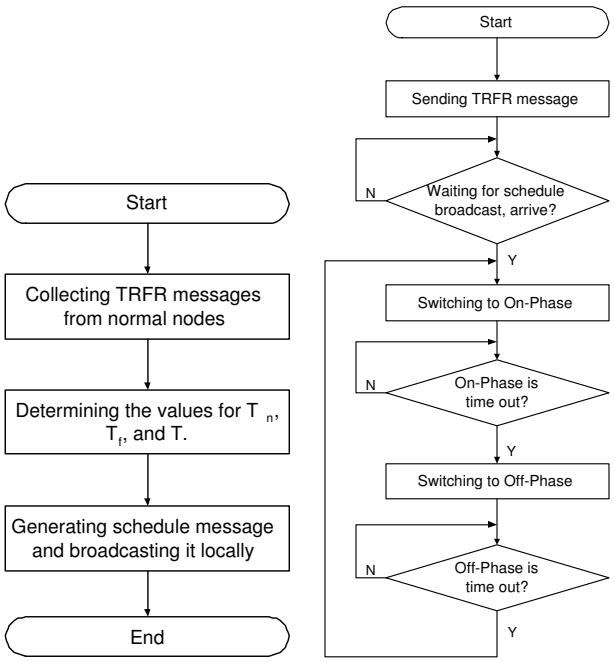


Figure 3.5. (a)for data-gathering nodes and (b)for data-collection nodes.

can data-collection nodes know the time of next schedule broadcast so as to power on their radios? The solution is that we include reschedule interval information into schedule messages. How to pre-estimate the value of schedule interval is another main contribution in this paper. The detail is described in Section 3.1.3. Flow charts for data-gathering nodes and data-collection nodes are modified as in Fig. 3.6, in which clock-drift is added and time-variant traffic is considered.

Nevertheless, this scheme may lose efficiency in a special situation. That is, data-collection nodes start *S*chedule-*P*hase later than their data-gathering nodes for accumulative clock-drifts. Consequently, schedule broadcast will be missed and those nodes cannot be synchronized or know the latest schedule. This kind of node is named synchronization-losing node. For this issue, we design an on-demand strategy. That is, when the last *O*n-*P*hase is over, synchronization-losing nodes proactively send requests to their data-gathering nodes. Related data-gathering nodes will response those requests with

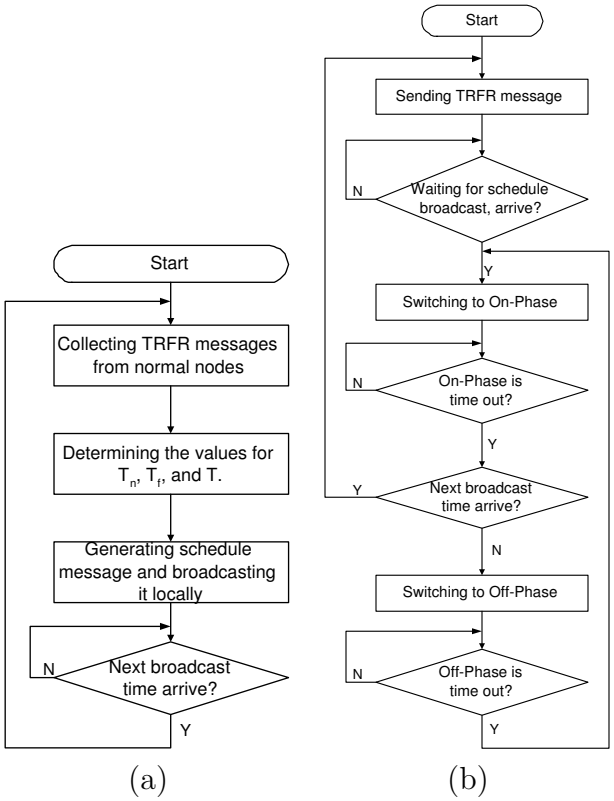


Figure 3.6. (a)for data-gathering nodes and (b)for data-collection nodes.

the latest schedule and the information on the next *O*n-*P*hase's starting time. Then, synchronization-losing nodes can be synchronized and re-establish their phase-switching schedules.

3.1.3 Schedule Interval Design

Above discussions show that the matching operation among nodes can avoid unsuccessful transmission caused by accumulative clock-drifts. However, we also argue that it is unnecessary to offer matching operation at all time and for all nodes. For instance, two nodes, which have little information to exchange, need not to switch phases coincidently, since their mismatching operation has little influence on communications. Hence, some nodes could be allowed to go out of coincidence and to be rescheduled only if necessary.

Furthermore, from (3.9) and (3.12), we note that the durations of *O*n-*P*hase and *O*ff-*P*hase are tightly related to nodes' traffic strength and service capability, which are heterogeneous for WSNs as we discussed above. Thus, besides on-demandly removing accumulative clock-drifts and informing phase-switching schedules, additional function for schedule broadcasts is to acquire more suitable values for essential parameters according to system situation. This property enables our algorithm to be an adaptive scheme in terms of node density and traffic strength.

How to combine all factors to adjust the length of schedule interval correctly is a complicated and vague task, which impacts the performance in terms of energy reservation and successful communication significantly. Since FLS is outstanding in dealing with uncertain problems, we design a rescheduling-FLS to monitor the influence of accumulative clock-drifts, the variance of traffic strength and service capability on communications. Then we can adjust schedule interval and power-on/off duration adaptively. We use

$$T_i = \xi_i \times T_{i-1} \tag{3.16}$$

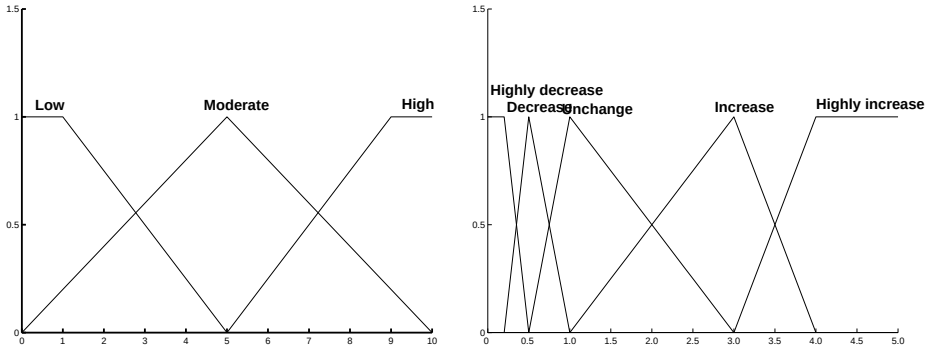
as our interval adjustment function. Where T_i is the interval for the i-th schedule broadcast, ξ_i is the i-th adjustment factor determined by our rescheduling-FLS.

In our rescheduling-FLS, there are three antecedents:

- Ratio of node with overflowing buffer (R_{of}): the percentage of node having buffer overflowing within a cluster;
- Ratio of node with high unsuccessful transmission rate (R_{hf}): the percentage of node whose unsuccessful transmission rate is higher than a threshold within a cluster;
- Ratio of node experiencing unsuccessful transmission (R_{sr}): the percentage of node having failure transmission within a cluster.

R_{of} reflects traffic strength. R_{hf} and R_{sr} reflect the influence of accumulative clock-drifts on communications from depth and width aspects individually.

The consequent is the adjustment factor (ξ_i) for the schedule-broadcast interval. The linguistic variables representing R_{of}, R_{hf} and R_{sr} are divided into three levels: *Low*, *Moderate* and *High*. ξ_i is divided into 5 levels: *Highly Decrease*, *Decrease*, *Unchange*, *Increase* and *Highly Increase*. We use trapezoidal membership functions (MFs) to represent *Low*, *High*, *Highly Decrease* and *Highly Increase*, and triangle MFs to represent *Moderate*, *Decrease*, *Unchange* and *Increase*. We show those MFs in Fig. 3.7(a) and Fig. 3.7(b).



(a) (b)

Figure 3.7. (a)Antecedent MFs and (b) Consequent MFs.

The schedule interval should be shortened when there are many data packets missing due to accumulative clock-drifts and/or unsuitable *O*ff-*P*hase duration, otherwise the schedule interval should be extended to reduce the energy consumption on scheduling. Based on this fact, we design our rescheduling-FLS using rules summarized in Table 3.1. Ante1 is R_{of}. Ante2 is R_{hf}. Ante3 is R_{sr}. Consequent is ξ.

Table 3.1. Rules for rescheduling-FLS

Rule	Ante1	Ante2	Ante3	Consequent
1	Low	Low	Low	Highly Increase
2	Low	Low	Moderate	Increase
3	Low	Moderate	Moderate	Decrease
4	Low	Moderate	High	Decrease
5	Moderate	Low	Moderate	Increase
6	Moderate	Low	High	Unchange
7	Moderate	Moderate	Moderate	Decrease
8	Moderate	Moderate	High	Highly Decrease
9	Low	High	High	Decrease
10	Moderate	High	High	Highly Decrease
11	High	Low	Moderate	Increase
12	High	Low	High	Unchange
13	High	Moderate	Moderate	Decrease
14	High	Moderate	High	Decrease
15	High	High	High	Highly Decrease

For every input (R_{of},R_{hf},R_{sr}), the output is defuzzified using (3.17). The height of the five fuzzy sets depicted in Fig. 3.7(b) are $\bar{\xi}_1$=0.2, $\bar{\xi}_2$=0.5, $\bar{\xi}_3$=1.0, $\bar{\xi}_4$=3.0, $\bar{\xi}_5$=4.0.

$$y(R_{of}, R_{hf}, R_{sr}) = \frac{\sum_{l=1}^{15} \bar{\xi}^l \mu_{\mathrm{F}_1^l}(R_{of}) \mu_{\mathrm{F}_2^l}(R_{hf}) \mu_{\mathrm{F}_3^l}(R_{sr})}{\sum_{l=1}^{15} \mu_{\mathrm{F}_1^l}(R_{of}) \mu_{\mathrm{F}_2^l}(R_{hf}) \mu_{\mathrm{F}_3^l}(R_{sr})} \tag{3.17}$$

The inputs of rescheduling-FLS are acquired from TRFR messages. Prior to each schedule broadcast, rescheduling-FLSs located in data-gathering nodes individually estimate the influence degree of the accumulative clock-drift and the change of traffic strength on communications. After obtaining ξ_i, data-gathering nodes determine the value for the next schedule-broadcast interval according to (3.16). This operation can not only save energy through avoiding unnecessary schedule broadcasts and idle listening, but also ensure an adequate data successful transmission rate.

3.2 Asynchronous Schedule-Based MAC (ASMAC) protocol

Asynchronous Schedule-Based MAC (ASMAC) is similar to A-MAC. ASMAC's system time is also divided into four phases: TRFR-Phase, Schedule-Phase, On-Phase and Off-Phase (Fig. 3.8). Same TRFR message and TRFR-Phase duration design method are used by ASMAC. While, On-Phase is further divided into super-time-slots, which are composed of several normal time-slots, and one source-destination pair continuously occupies one super-time-slot. We add ACK message as the acknowledgment for receiving data packets successfully. A transmission is defined as an unsuccessful one once the transmitter does not receive ACK after a certain period of time. The format of schedule message is shown in Fig. 3.9.

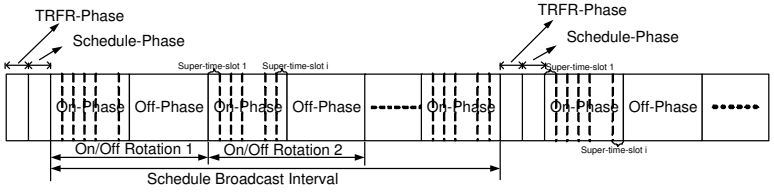


Figure 3.8. System Time scheme structure for ASMAC.

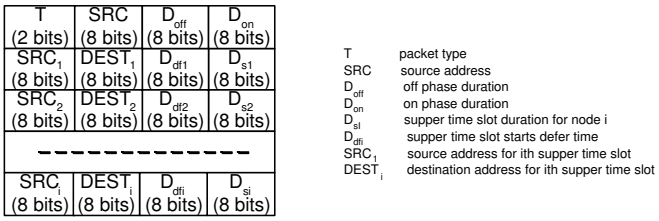


Figure 3.9. Schedule message packet format for ASMAC.

Matching schedule establishment, maintenance and schedule interval design mechanisms of ASMAC are as same as in A-MAC, but power-on/off duration design is somewhat different. A new task: time-slot allocation is added into ASMAC.

3.2.1 *O*n-*P*hase/*O*ff-*P*hase Duration (T_n/T_f) Design

In ASMAC, nodes perform communication in their own super-time-slots, and turn off their radios to save energy in *O*ff-*P*hase and other nodes' super-time-slots. Hence, nodes carry out communications orderly and contention-freely. Same criteria are utilized by ASMAC for *O*n-*P*hase and *O*ff-*P*hase duration design: trying to save more energy, keeping information up to date and avoiding losing information due to buffer overflowing. The optimum values for T_f and T_n can be calculated

$$\begin{array}{cc} \text{1) when} \quad 2W_{max} < \min_j(\frac{k_j}{\lambda_j}) & \text{2) when} \quad 2W_{max} \geq \min_j(\frac{k_j}{\lambda_j}) \\ \begin{cases} T_f = \frac{2W_{max}}{(1+\sum_{l=1}^{N}\frac{\mu_l\lambda_l}{1-\mu_l\lambda_l})} \\ T_n = \frac{2W_{max}\sum_{l=1}^{N}\frac{\mu_l\lambda_l}{1-\mu_l\lambda_l}}{(1+\sum_{l=1}^{N}\frac{\mu_l\lambda_l}{1-\mu_l\lambda_l})} \end{cases} & \begin{cases} T_f = \min_j(\frac{\frac{k_j}{\lambda_j}}{1+\sum_{l=1}^{N}\frac{\mu_l\lambda_l}{1-\mu_l\lambda_l}}) \\ T_n = \min_j(\frac{\frac{k_j}{\lambda_j}\sum_{l=1}^{N}\frac{\mu_l\lambda_l}{1-\mu_l\lambda_l}}{1+\sum_{l=1}^{N}\frac{\mu_l\lambda_l}{1-\mu_l\lambda_l}}) \end{cases} \end{array} \tag{3.18}$$

3.2.2 Time-Slot Assignment

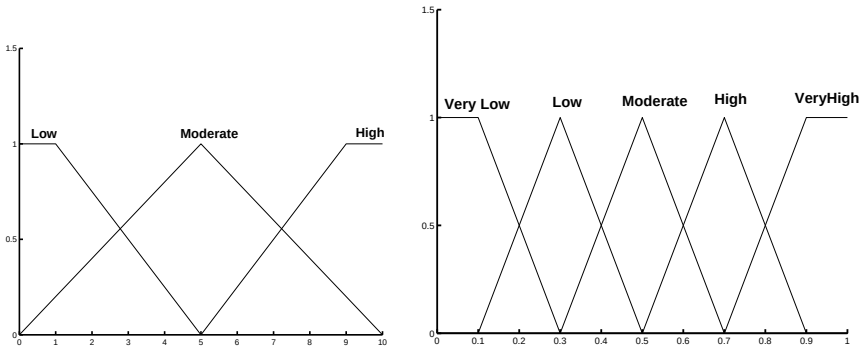
For classic TDMA systems, system time is divided into slots, and each user occupies cyclically repeating time-slots - a buffer-and-burst method. Thus, high quality network synchronization method is needed. Unfortunately, this premise is troublesome for our ASMAC scheme. However, we note that as the length of time-slots increases, more transmissions are done successfully under the same mismatching situation. Hence, contrast to buffer-and-burst method, we design a buffer-and-continue method to enhance the tolerance on accumulative clock-drifts, in which the same communication pairs occupy sets of continuous time-slots.

In ASMAC, we design an allocation-FLS to correspondingly quantify transmission priorities for each node. There are two antecedents to our allocation-FLS:

- Traffic arrival rate (R_a), and
- Transmission failure rate (R_{us}).

The consequent is the priority of a node performing transmission (P_t).

The linguistic variables used to represent R_a and R_{us} are divided into three levels: Low, $Moderate$ and $High$. P_t is divided into 5 levels: $Very\ Hign$, $High$, $Moderate$, Low and $Very\ Low$. We use trapezoidal membership functions (MFs) to represent Low, $High$, $Very\ Low$ and $Very\ High$, and triangle MFs to represent $Moderate$, Low and $High$. We show those MFs in Fig. 3.10(a) and Fig. 3.10(b).



(a) (b)

Figure 3.10. (a)Antecedent MFs and (b)Consequent MFs.

The transmission priority of a node should be higher when there are more data packets waiting for transmitting and/or its transmission failure rate is high. Based on this fact, we design our allocation-FLS using rules summarized in Table 3.2. Ante1 is the traffic arrival rate. Ante2 is the unsuccessful transmission rate. Consequent is the priority of a node performing transmission. With the allocation-FLS, data-gathering nodes leverage the information acquired from TRFR messages to quantify priorities for nodes. The

node owning the highest priority is the earliest one to perform communications during On-Phase.

Table 3.2. Rules for allocation-FLS

Rule	Ante1	Ante2	Consequent
1	Low	Low	Moderate
2	Low	Moderate	High
3	Low	High	Very High
4	Moderate	Low	Low
5	Moderate	Moderate	Moderate
6	Moderate	High	High
7	High	Low	Very Low
8	High	Moderate	Low
9	High	High	Moderate

3.3 Simulations and Performance Evaluation

We used the simulator OPNET to run simulations. A network with 30 nodes is set up and the radio range (radius) of each node is 30m. Those nodes are randomly deployed in an area of $100 \times 100m^2$ and have no mobility. This network can be treated as one cluster in a large-scale system. In order to simplify the analysis about the impact of accumulative clock-drifts on communications and the performance of our MAC algorithms, we exclude the factors coming from physical layer and network layer in our experiments. The clock-drift rate of frequency oscillators varies from 1 to 100 microseconds every second. Table 3.3 summarizes the parameters used by our simulations. Packet size is 1000 bytes. The destination for each node's traffic is randomly chosen from its neighbors.

As in [28][82], data packets arrive according to a Poisson process with certain rate in our simulations. Moreover, in every 10 seconds the traffic will be held for 5 seconds to

Table 3.3. Physical layer parameters

W_{wmin}	32	W_{wmax}	1024
MAC Header	34 bytes	ACK	38 bytes
CTS	38 bytes	RTS	44 bytes
SIFS	10 μsec	DIFS	50 μsec
ACK-Timeout	212 μsec	CTS-Timeout	348 μsec

simulate bursting traffic. In our simulations, we substitute statistic average values with time average values for data packet arrival rate and service time.

All nodes are set initial energy of 15J. We use the same energy consumption model as in [83] for radio hardware. To transmit an l-symbol message for a distance d, the radio expends:

$$E_{Tx}(l,d) = E_{Tx-elec(l)} + T_{Tx-amp(l,d)} = l \times E_{elec} + l \times e_{fs} \times d^2 \tag{3.19}$$

and to receive this message, the radio expends:

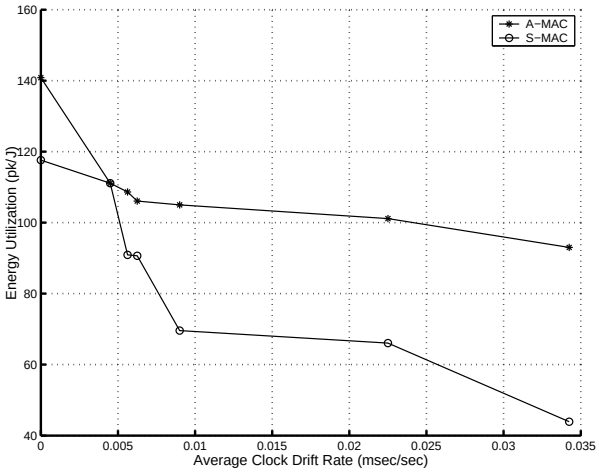
$$E_{Rx} = l \times E_{elec} \tag{3.20}$$

The electronics energy, E_{elec}, as described in [83], depends on factors such as coding, modulation, pulse-shaping and matched filtering. The amplifier energy, $e_{fs} \times d^2$ depends on the distance to the receiver and the acceptable bit error rate. In this paper, we choose: $E_{elec} = 50nJ/syn$ and $e_{fs} = 10pJ/sym/m^2$. When a node receives packets but the destination is not for it, those packets will be discarded. This kind of useless receiving, i.e., idle listening, uses the same model in (3.20) to calculate energy consumption.

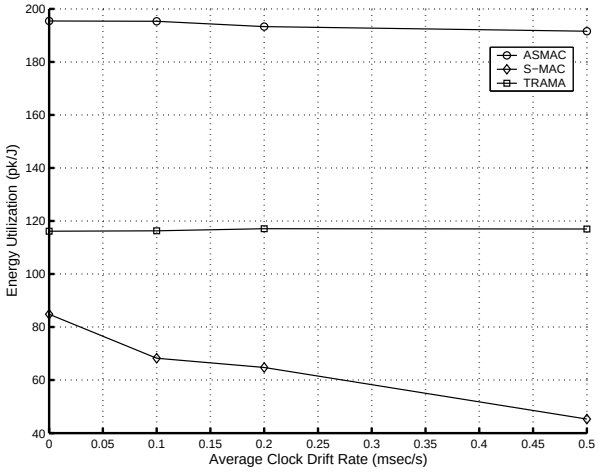
3.3.1 A-MAC Vs S-MAC

We compared our A-MAC against S-MAC [36] without network synchronization function. In our simulations, energy utilization is assessed by the number of successfully transmitted data packets per J, and the unit is pk/J. Energy utilization versus clock-drift

rate is plotted in Fig. 3.11(a). Observe that A-MAC can send 17.85% to 33.33% more packets per J. Therefore, with same available energy and traffic strength, the lifetime of a network will be extended about 0.2 to 0.4 times when using our algorithm A-MAC instead of S-MAC scheme. This result demonstrates that A-MAC can implement energy reservation task successfully.



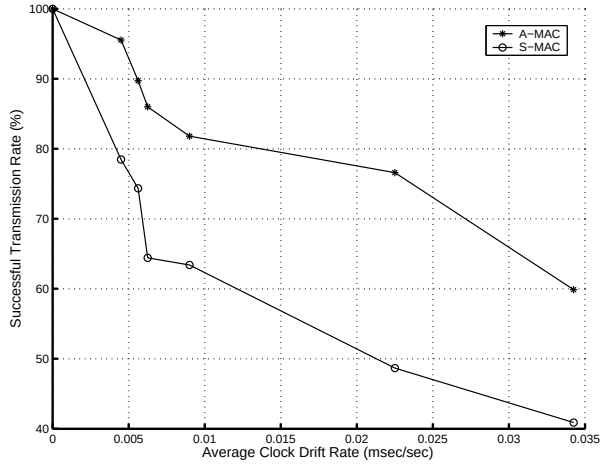
(a)



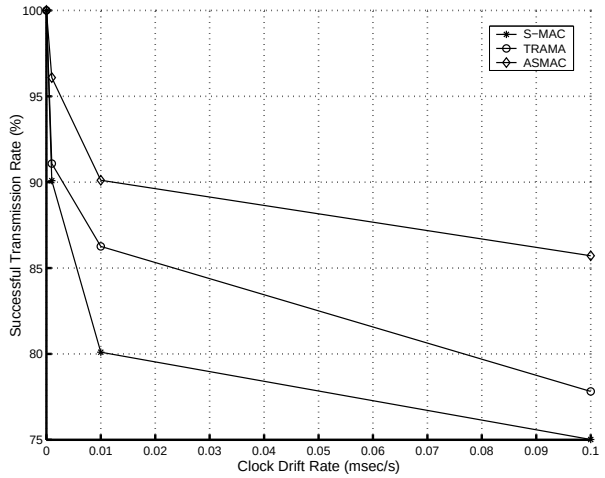
(b)

Figure 3.11. Energy utilization for (a)A-MAC and (b)ASMAC.

Saving energy is one of the main aims of A-MAC. While, we should not achieve this goal through sacrificing data successful transmission rate, since communication is the ultimate role for communication systems. In Fig. 3.12(a), we plot data successful transmission rate versus clock-drift rate. Observe that A-MAC can transmit data packets successfully about 30.77% more than S-MAC. The reasons are: Firstly, failure transmissions are reduced, because clock-drifts among nodes are removed effectively; secondly, more energy is utilized by transmitting data packets.



(a)



(b)

Figure 3.12. Successful transmission rate for (a)A-MAC and (b)ASMAC.

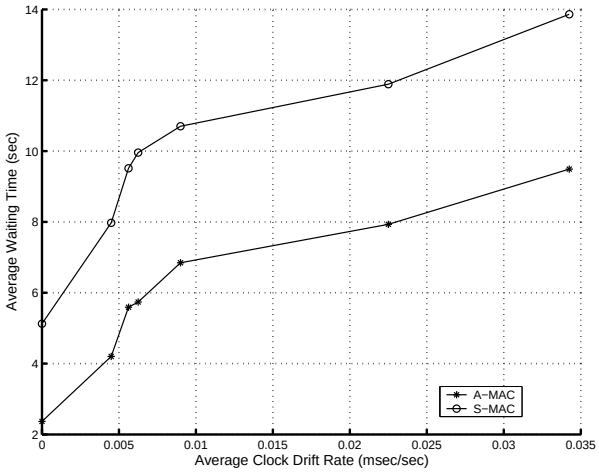
In Fig. 3.13(a), we plot average waiting-time versus clock-drift rate. It is shown that A-MAC has about 33.3% shorter waiting-time than that of S-MAC. Moreover, we set W_{max} to 12 seconds for average waiting-time, we found that average waiting-time for A-MAC is always shorter than 12 seconds even at different clock-drift rates. However for S-MAC, the average waiting-time is longer than 12 seconds when clock-drift rate is longer than 0.0225 msec/sec. That is, there are many out-of-date packets received when using S-MAC. This result demonstrates our claim that our algorithm A-MAC is a waiting-time aware method.

3.3.2 ASMAC vs. S-MAC and TRAMA

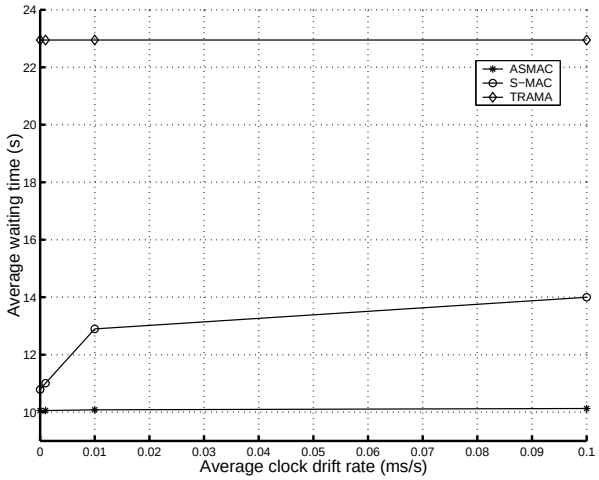
We compared our ASMAC against S-MAC and TRAMA [32] without network synchronization function. Energy utilization versus clock-drift rate is plotted in Fig. 3.11(b). Observe that ASMAC can send 41.176% to 56.14% more packets per J. Therefore, with same available energy and traffic strength, the lifetime of a network will be extended about 0.4 to 0.6 times when using our algorithm ASMAC instead of S-MAC and TRAMA schemes. This result demonstrates that ASMAC can also implement energy reservation task successfully.

In Fig. 3.12(b), we plot data successful transmission rate versus clock-drift rate. Observe that ASMAC can transmit data packets successfully about 12.5% more than S-MAC, and about 4.65% more than TRAMA.

In Fig. 3.13(b), we plot average waiting-time versus clock-drift rate. It is shown that ASMAC has about 56.178% shorter waiting-time than TRAMA, and about 8.648% than S-MAC. We found that the average waiting-time for ASMAC are also shorter than $W_{max} = 12sec$ at different clock-drift rates. While for TRAMA and S-MAC, the average waiting-time is longer than that threshold.



(a)



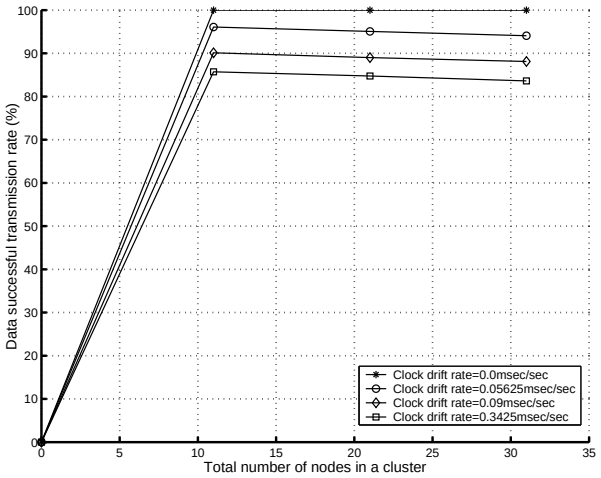
(b)

Figure 3.13. Waiting-time for (a)A-MAC and (b)ASMAC.

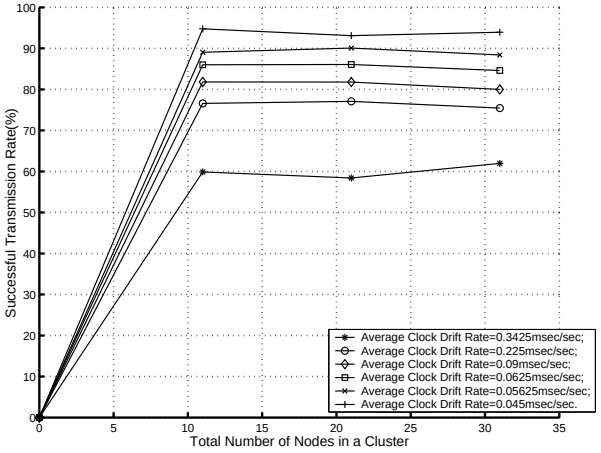
3.3.3 Adaptation of ASMAC and A-MAC

We investigate the influences of node density and traffic strength on system performance of our algorithms. We change node density and traffic strength individually at a set of clock-drift situation. In Fig. 3.14(a), we plot number of nodes, changed from 10 to 30, in a cluster versus successful transmission rate of data packet for ASMAC. Notice that, for each clock-drift rate, the vibration of successful transmission rate with the change of node density is less than $85.714\% - 83.606\% = 2.108\%$. The same experiment

is done for A-MAC. We can see that the vibration of successful transmission rate are less than $61.96\% - 59.87\% = 2.09\%$ (Fig. 3.14)(b).



(a)

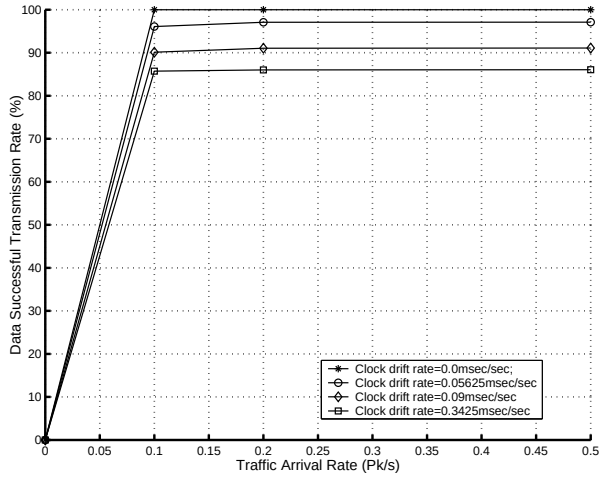


(b)

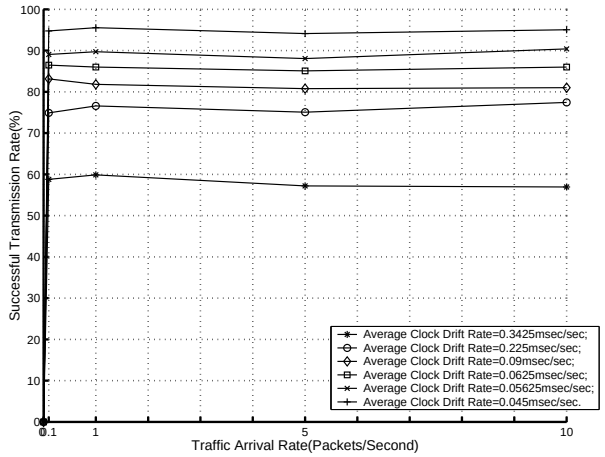
Figure 3.14. Network density adaptation for (a)ASMAC and (b)A-MAC.

In Fig. 3.15(a), we compare successful transmission rate at different traffic arrival rates, varied from 0.1 to 0.5 pk/sec. It shows that, for each clock-drift, the vibration of successful transmission rate with the change of node number is less than $97.099\% -$

96.087% = 1.012% for ASMAC. The vibration of A-MAC is less than 60% − 58.79% = 1.21% (Fig. 3.15)(b).



(a)



(b)

Figure 3.15. Traffic intensity adaptation for (a)ASMAC and (b)A-MAC.

These two experiments show that, with the variance of node density and traffic strength, network throughput can keep almost stable through using our A-MAC and ASMAC protocols. The reason is that essential parameters - reschedule interval, *O*n-*P*hase and *O*ff-*P*hase durations - are adaptively adjusted with system situation.

3.4 Conclusion

Leveraging the characteristics of free-running timing method and the advantages of fuzzy logic system on uncertain problems, we propose two energy-efficient MAC protocols for WSNs: asynchronous MAC (A-MAC) protocol and asynchronous schedule-based MAC (ASMAC) protocol. Our timing-rescheduling scheme and time-slot allocation algorithm provide an approach to remove the tight dependency on network synchronization for energy-efficient MAC protocols, which is a critical constraint for network upgrading and expanding (Fig. 3.16). Within A-MAC and ASMAC protocols, no common timescale is needed any more, which will free the energy for setting up and maintaining it.

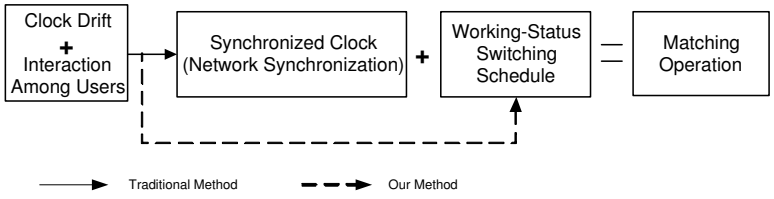


Figure 3.16. Motivation of our energy-efficient MAC protocols.

Furthermore, considering the heterogeneous nature of WSN, we build a traffic-strength- and network-density-based designing model. This model equips the system with the capability to determine essential algorithm parameters adaptively, which greatly influence system performance in terms of energy reservation and communication capability. Those algorithm parameters include power-on/off duration, schedule-broadcast interval, as well as super-time-slot size and order. In addition, static approaches may be far from optimal because they deny the opportunity to reschedule operations if system situation is changed, thus we apply adaptive methods for parameter adjustment.

Opposition to existing network synchronization schemes, A-MAC and ASMAC are control-center-exhaustion schemes. It is data-gathering nodes, whose energy are more

abundant and easier to be recharged than data-collection nodes, that are in charge of most working load to form matching operation among nodes.

CHAPTER 4

QUERY PROCESSING FOR WSDS

As a motivation for the quality-guaranteed and energy-efficient query processing algorithm, we describe a scenario:

- A great multitude of temperature sensor nodes are randomly deployed in an interested region . Individual sensor nodes (or in short, nodes) are connected to other nodes in their vicinity through wireless communication interface, and use a multihop routing protocol to communicate with nodes that are spatially distant. All nodes are interconnected to at least one gateway directly or through intermedial nodes. Gateways are in charge of relaying data to a powered PC (front-end node) and, on the opposite direction, disseminating queries to related nodes. Within this WSDS, each node owns equal computing and sensing capability, but measurement quality for sensor parts might not be identical.

  This scenario involves such a kind of region-based query:

- *Environmental Temperature Monitoring:* With p confidence, tell the average temperature of nodes in the region defined by a rectangle (a,b,c,d).

  Written in SQL-like language[92] or MeshSQL[93], this query is shown in Fig. 4.1.

```
SELECT          AVG(temp)
FROM            sensors
WHERE           loc in (a,b,c,d) AND PROB ≥ p
SAMPLE PERIOD   100 seconds;
```

Figure 4.1. Average Temperature Query in SQL Form.

4.1 Problem Description

4.1.1 Sources of Imperfect Information

Imperfect information is ubiquitous (almost all information that we have about the real world is not certain, complete or precise). In most occasions, there are four types of imperfection in an information system: uncertainty, incompleteness, ambiguity and imprecision[94][95]. Incompleteness arises from the absence of values, imprecision is caused by the existence of values that cannot be measured with suitable precision, ambiguity is introduced by vague statements, and uncertainty arises from the fact that an agent has constructed a subjective opinion on the truth of a fact that it does not know for certain.

In the context of analyzing and understanding the uncertainty of query answer, one significant challenge is how to correctly understand the nature and source of uncertainty of the information derived from remote sensing. Image chain approach[96] is one of the most important and useful models for describing remote sensing process. It can identify steps in remote sensing process (or links in chain) completely and illustrate the interrelation nature of those steps.

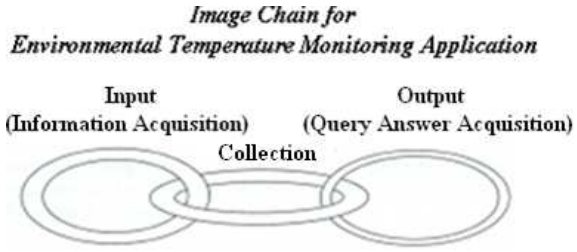


Figure 4.2. Image Chain Model.

Analyzing the working process of this temperature monitoring application, there are three main kinds of imperfect information source: measurement quality, point spread

function (PSF) and link quality. Fig. 4.2 illustrates the image chain model we exploit. Links in the chain represent various steps from nodes collecting related information (Input), to flowing data records back to related front-end nodes (Collection) at run-time, to obtaining query answers through processing all collected information at front-end nodes (Output).

- *Measurement Quality*

  As we know, the quality of nodes' sensing part usually boils down to their measurement stability and accuracy. In general, as measurement stability and accuracy increase, so do their power requirement and cost, which are all troublesome for general nodes. Therefore, inaccurate measurements generated by sensing parts are very common. Measurement quality of nodes introduces uncertainty and imprecise information into query answers.

- *Point Spread Function (PSF)*

  Temperature monitoring application is interested in the temperature over a region instead of one point in space. However considering the operation feasibility, as well as the cost on hardware and time, sampling method is widely used rather than completely measuring. In this aspect, another imperfect information source - PSF is raised. PSF is caused by nodes' nonuniform sensitivity within their monitoring space. In general, nodes exhibit sensitivity variation following Gaussian distribution. That is, nodes are more sensitive to the center than toward the edge of their regions. PSF of nodes introduces ambiguity into query answers.

- *Link Quality*

  The dynamic and lossy nature of wireless communication pose the major challenge to reliable, self-organized WSNs. Failure transmissions may happen during data transmission because of collision, node dying out (no battery), node being busy, or node's mobility. Moreover, in physical layer, sensor node's mobility generates

channel fading during data transmission, which degrades the performance in terms of bit error rate (BER) and frame error rate (FER). Packet loss due to poor link quality introduces incompleteness into query answers.

In a set of temperature monitoring experiments, such as finding the highest temperature, the lowest temperature and the average temperature over a region, we vary measurement quality, PSF and link quality respectively to check the impact of those imperfect information sources on the confidence of query answers. During those experiments, we fix other parameters, such as the number of nodes, node density, communication range, sensing range and network coverage. Assuming true results are known beforehand. Experiment results are shown in Table 4.1. RMSE stands for root mean square of error of query answers. σ is the standard deviation of nodes' measurements. r is the radius utilized for modelling nodes' sensing space. LBR stands for link break rate.

Table 4.1. Error sources

RMSE	Measure Quality		PSF (σ^2=1)		Link Quality	
	σ=0.01	σ=0.1	r=1.65m	r=1.96m	LBR=0.1	LBR=0.5
MAXIMUM	0.2537	2.5618	22.882	25.17	0.2069	0.7132
MINIMUM	0.2541	2.5416	22.905	25.195	0.1291	0.5895
AVERAGE	0.0102	0.0944	3.0802	3.5202	0.0936	0.3127

We observe the following:

- *Measure Quality*

  When employing nodes with poor measurement quality, whose measurements have 0.1 standard deviation, the error of query answers is much larger (around ten times) than the one when using nodes whose measurements just have 0.01 standard deviation.

- *PSF*

  When using large disks to cover a monitoring region, i.e., the radius r of disks is 1.96 meter, query answers deviate from the true values much more than those, in which disks with 1.65 meter radius are used. Note that, more information is missed since less nodes are employed, thus large error is introduced into query answers.

- *Link Quality*

  When using more reliable links to forward collected data, in which there is only 0.1 percentage of links broken during information transmission, more data reach query processing operators. Thus, the more missing information is caused by link broken, the more the difference between the true values and the acquired values becomes.

In conclusion, with measurement errors, misrepresent errors, and/or missing information increased, the error introduced into query answers is obviously increased, consequently the confidence of query answers is reduced. Even though those conclusions are derived from the temperature monitoring application, they are still valuable for other remote sensing applications, such as humidity monitoring, hurt gas monitoring, etc.

4.1.2 Source of Energy Consumption

Sensor nodes are very limited in their processing, computation, communication, storage and power supply. For example, a typical Crossbow MICA mote MPR300CB[97] has a low-speed 4MHz processor equipped with only 128KB flash, 4KB SRAM and 4KB EEPROM. It has a maximal data rate of 40kbps, a transmission range of about 100 feet, and is powered by two AA batteries. However, communication, not only transmitting, but also receiving, or merely scanning channel, can use up to half energy[11]. Therefore wireless transmitting and receiving are the most energy consuming operations inside nodes. Recently, some researchers have begun studying the problem of reducing power consumption on wireless interface. One approach is reducing energy consumption for

transmitting/receiving each bit[98], the other is reducing the amount of information exchanged over networks.

Low cost on devices makes it feasible to generate high density network. This kind of dense deployment can help to improve a WSDS's reliability. Somehow, within a WSDS, not all available nodes provide useful/related information that improves the accuracy of query answers. For example, if nodes are far away from the interested region, the information collected by those nodes is less useful/lower correlative. Furthermore, some information might be redundant because nodes closing to each other would have similar data. From those prospects, collecting raw readings from all nodes involves large amounts of readings, which will lead to shorter network lifetime.

For the energy reservation issue on data collection, previous networking researches approach data aggregation[10] as an application specific technique to reduce the amount of data sent over a network. But where the aggregation should be carried out is a very essential and tough problem, which relates to the correctness and effectiveness of operations.

4.2 Quality-Guaranteed and Energy-Efficient Query Processing

Keeping those two primary problems: quality-required and energy-constraint in mind, we propose a quality-guaranteed and energy-efficient (QGEE) query processing algorithm for WSDSs. Acquisitional query processing (ACQP)[67] is employed to task network through declarative queries. Compared with typical methods, QGEE focuses on betaking the significantly new opportunity risen in WSDSs: smart sensor nodes have the capability to control over where, when, and how often data is physically acquired (i.e., sampled) and delivered to query processing operators.

In QGEE, only a subset of nodes will be chosen to acquire readings or samplings corresponding to the fields or attributes referenced in queries. During data's forward-

ing, nodes' mobility, link quality and battery level are considered to establish multiple paths. The overall goals of this approach are reducing disturbance from measurements with extreme error, decreasing information loss from link failure and minimizing energy consumption, but still providing satisfying service for various applications. Furthermore, according to the analysis and classification on the sources of imperfect information, probabilistic method is employed to formulate their distributions. Finally probabilistic query answers are acquired on uncertain data. The probability corresponding with query answers can be used to determine the amount of confidence the users should have.

4.2.1 Confidence Control for Query Answer

4.2.1.1 Query Vector Space Model (VSM) Design and Node Selection

In information retrieval, VSM is one of efficient methods to quantify the correlation between a query and all candidate documents. If treating sensor nodes as candidate documents for a query, the correlation between a query and a node can be quantified by utilizing the same principle. However, redesign on the VSM vector is needed to implement quality and energy control. Based on the study mentioned above, following factors are considered:

- Location

  Given a piece of space, number and location of nodes determine the monitoring coverage. In order to employ as few as possible nodes to cover as large as possible area, those nodes, located at some special locations called optimal locations, should be selected. The detail on determining optimal locations is presented in Section 4.2.1.2.

- Measurement Quality

  Since the cost and measurement quality of sensor node are related to each other,

sensor nodes owning a variety of qualities are always deployed simultaneously in a WSN for economical reasons. Moreover, through a query, database users supply not only what information they are interested in, but also the expectation on answers' quality, i.e., the confidence of answers. In this case, suitable nodes should be selected to response queries rather than all of them.

- Battery

  Remaining battery capacity of sensor nodes is the third factor, but not the least important one. When the battery of a node is used up, the uncertainty of answers, at some degree, will increase caused by missing data. It inspires us to select those nodes with high remaining battery capacity, so that all expected data could be collected with best effort.

In the QGEE algorithm, VSM is employed to incorporate above all factors for electing the most related nodes (called active nodes) for query processing. Query vector Υ is designed as $\Upsilon = [R_l, A_d, B_m]$.

- R_l stands for location relativity. It is the indicator of the distance between the location of a sensor node at (x,y) and the optimal location at (x_0,y_0).

$$R_l = 1 - \frac{\sqrt{(x-x_0)^2 + (y-y_0)^2}}{L} \tag{4.1}$$

  where L is the factor to ensure R_l to be a positive numeric value. For instance, L can be equal to the maximum distance of two nodes within a network. Small value of R_l indicates a node closing to an optimum location.

- A_d stands for measurement quality. A_d is equal to the confidence of measurement bias. For example, for speed detecting sensor nodes, CXM539[97], the bias is $\pm$1mGauss and owns 0.95 confidence. In this case, $A_d = 0.95$.

- B_m stands for remaining battery capacity.

When a query is submitted, the related top-end node fixes the optimal locations, and translates this query into a query VSM vector Υ_0=[1,$A_{d,0}$,$B_{m,0}$]. For instance, Υ_0=$[1, p, 5]$ according to the query given in Fig. 4.1. Assuming the maximum battery capacity for nodes is $5J$. Υ_0 and information on optimal locations will be flooded over the network. Then, nodes update their query VSM vectors ($\Upsilon_i = [R_{l,i}, A_{d,i}, B_{m,i}]$ (i=1,2,$\cdots$, n)). Assuming there are n nodes in this network. For node i the node location $R_{l,i}$ is defined as:

$$R_{l,i} = \max_j \{r_{l,i,j}\} \tag{4.2}$$

where $r_{l,i,j} = 1 - \frac{\sqrt{(x_i - x_{0,j})^2 + (y_i - y_{0,j})^2}}{L}$, ($x_i$, y_i) is the position of node i, and ($x_{0,j}$, $y_{0,j}$) is the position of the jth optimal location.

A query correlation indicator (QCI) is designed, which is referred as ζ, to represent the correlation between individual nodes and a query. ζ is formulated as follows.

$$\begin{aligned} \zeta_{\Upsilon_0,\Upsilon_i} &= sim_{vs}(\Upsilon_0, \Upsilon_i) = \Upsilon_0 \cdot \Upsilon_i \\ &= \frac{1 \times R_{l,i} + A_{d,0} \times A_{d,i} + B_{m,0} \times B_{m,i}}{\sqrt{(1 + A_{d,0}^2 + B_{m,0}^2)(R_{l,i}^2 + A_{d,i}^2 + B_{m,i}^2)}} \end{aligned} \tag{4.3}$$

Observe that, QCI $\zeta_{\Upsilon_0,\Upsilon_i}$ is a function of query quality $A_{d,o}$, node's energy $B_{m,i}$, measurement quality $A_{d,i}$ and node location $R_{l,i}$. Moreover, $\zeta_{\Upsilon_0,\Upsilon_i}$ is computed by nodes through a distributed way.

ζ directly indicates the similarity between a query and a node. Inspired by this, the criterion for active node choosing is formed: the decision - which nodes are active to respond queries - is based on nodes' QCIs. That is, nodes with highest QCI among their one-hop neighbors are chosen to participate in related query processing. In QGEE, active nodes are chosen locally leveraging cooperation among nodes. Assuming that each node knows QCIs of its one-hop neighbors, which can be achieved by requiring each node to broadcast its QCI initially. This distributed calculation balances the working

load for QCIs over the entire network so that the overhead introduced by VSM for each node is extremely reduced. Furthermore, according to (4.3), the computing complexity is only around O(16). Therefore, it is safe to claim that the overhead caused by VSM is minor, even though the processing capability for general nodes is limited (i.e., 4 MIPS for Berkeley MICA Mote[10]).

4.2.1.2 Optimal Location Determination

Modelling the problem - determining optimal locations for a query - as a k-partial set cover problem. This problem is defined as follows: Let n be the number of all sensor nodes, n' be a given positive integer so that $n' \leq n$. If we have k disks with radius r, the k-partial set cover problem tries to solve whether and which k disks can cover at least n' nodes. In this paper, sensor nodes on a plane (the dimension is 2) are only considered.

Unfortunately, this kind of k-partial set cover problem is a NP problem. At present, all known algorithms for NP problem require time that is exponential to the problem size. It is unknown whether there is any faster algorithm. Therefore, to solve a NP problem for any nontrivial problem size, one of approaches is approximation algorithm, which can acquire solution during polynomial time. The SETCOVER algorithm[99] is a good approximation method to determine the value of k and locations of those disks on a plane. QGEE chooses the centers of those k disks as the optimal locations, and lets $n' = n$. Therefore, for QGEE, nodes locating at the centers of those disks can monitor the entire interested region.

According to users' requirements, QGEE, considering the impact of PSF on the uncertainty of query answers, adaptively adjusts the radius r instead of fixing it. Assuming PSF $g(d)$ of nodes in a WSDS is defined by (4.4) and the confidence of query answer

is required to be at least equal to p as shown in the example in Fig. 4.1. Here d is the distance between a point and the center on a disk.

$$g(d) = e^{-\pi d^2} \tag{4.4}$$

It is obviously that, among various locations on a disk, measurements own the lowest sensitivity/confidence when they stand for the situation at a disk's edge. This nature inspires us to design a criterion to calculate suitable value for r. That is, if the sensitivity/confidence is equal to or higher than p at the edge of disks, it is ensure that the measurements of all active nodes can represent the situation within their disks, at least, with p confidence. Deriving (4.5) from (4.4) to determine the suitable value for r.

$$r = -\sqrt{\frac{1}{\pi} lnp} \tag{4.5}$$

Note that, r is a function of standard deviation σ of PSF and quality requirement p. If fixing σ, r will decrease with the increase of p. That means, with higher query quality, smaller disks should be used to search the optimum locations and more active nodes are needed for a query.

4.2.1.3 Sample Size Determination and Query Answer Acquisition

A set of nodes has been chosen to respond a query. While, "How many measurements should be included in one sample (any subset of a population)?" is the question that will be answered in this Section. Sample size determination refers to the process of determining exactly how many samples should be measured/observed in order that the sampling distribution of estimators meets users' pre-specified precision[81].

As a matter of fact, nodes' readings are subject to errors caused by limitations of devices' hardware and environmental noise. Consequently, uncertainty is inherent regarding a true value. In this paper, the formula for a node's reading x is:

$$x = v + e_m + \eta \tag{4.6}$$

where v is the true value, e_m is the measurement error introduced by limitations of device's hardware, and η is the environmental noise considered as white Gaussian noise in this paper and $\eta \sim N(0, \frac{N_0}{2})$. Based on central limit theorem[100], the probability distribution of measurement errors complies with a normal distribution. That is, $e_m \sim N(0, \sigma_e^2)$. Generally, in product's technical datasheet, manufactories supply the information on measurement errors. For example, as mentioned above, the bias for CXM539 is ±1Gauss with 0.95 confidence. In this case, $\sigma_e^2 = 0.1302$. For general cases, if knowing the maximum bias Δx and its confidence p, the general expression of σ_e^2 can be obtained. That is

$$\sigma_e^2 = \frac{\Delta x^2}{[Q^{-1}(\frac{1-p}{2})]^2} \tag{4.7}$$

where $Q(x)$ stands for Q-Function, defined as $Q(x) \stackrel{\triangle}{=} \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{y^2}{2}} dy$

Moreover, e_m and n_o are independent. Therefore, node reading also complies with a Gaussian distribution with μ_x-mean and σ_x-standard deviation given as follows.

$$\mu_x = v \qquad and \qquad \sigma_x = \sqrt{\sigma_e^2 + \frac{N_0}{2}} \tag{4.8}$$

Therefore the PDF of node reading x is

$$f_X(x) = \frac{1}{\sqrt{2\pi(\sigma_e^2 + \frac{N_0}{2})^2}} e^{-\frac{(x-v)^2}{2(\sigma_e^2 + \frac{N_0}{2})^2}} \tag{4.9}$$

Using sample mean to estimate the mean of a random variable is an unbiased estimation, in which the estimator aims at the true value or the correct average[101].

Thus, an unbiased estimation on the true value v is done when choosing sample mean as the estimator, i.e. $\hat{v}_n = \frac{1}{n}\sum_{j=1}^{n} x_j$. Here n is sample size. In this case, the probability density function of $\hat{v}_n$ ($f_{\hat{V}_n}(\hat{v}_n)$) is similar to $f_X(x)$ with $\mu_{\hat{v}_n} = \mu_x$ and $\sigma^2_{\hat{v}_n} = \frac{1}{n}\sigma^2_x$. That is

$$f_{\hat{V}_n}(\hat{v}_n) = \frac{\sqrt{n}}{\sqrt{2\pi(\sigma_e^2 + \frac{N_0}{2})}} e^{-\frac{n(\hat{v}_n - v)^2}{2(\sigma_e^2 + \frac{N_0}{2})}} \tag{4.10}$$

QGEE lets Δx as the error margin between the estimator $\hat{v}_n$ and the true value v to reflect the target precision of query answers, and specifies the capability for ensuring this error not to be smaller than p. The criterion for sample size determination is simply stated as:

$$P_r\{|\hat{v}_n - v| \leq \Delta x\} \geq p \tag{4.11}$$

The probability of the estimation error not larger than Δx is

$$P_r\{|\hat{v}_n - v| \leq \Delta x\} = 1 - 2Q(\frac{\Delta x}{\sqrt{\frac{\sigma_e^2 + \frac{N_0}{2}}{n}}}) \tag{4.12}$$

Solving (4.11) and (4.12) for sample size n, we obtain

$$n \geq \frac{(\sigma_e^2 + \frac{N_0}{2})[Q^{-1}(\frac{1-p}{2})]^2}{\Delta x^2} \tag{4.13}$$

Since a statistic measurement on samples can rarely, if ever, be expected to be exactly equal to a parameter, it is important for estimations to be accompanied by statements that describe their precision. Confidence interval[102] provides a method of stating both how close the value of a statistic being likely to be value of a parameter and the chance of being close. A confidence interval of an attribute denoted by U_i is a interval $[l_i, h_i]$ such that l_i and h_i are real-valued, and the condition $h_i \geq l_i$ holds.

Note that (4.11) is the same statement made when defining a $100 \times p\%$ confidence interval, and Δx is about half of the width of this confidence interval. Using the sample size computed by (4.13) to estimate the true value (v), we have

$$P_r\{\hat{v}_n - \Delta x < v < \hat{v}_n + \Delta x\} \geq p \tag{4.14}$$

With (4.14), a bounded value: $v \in [\hat{v}_n - \Delta x, \hat{v}_n + \Delta x]$ is obtained, which is called "semi-manufactured" query answer.

Since heterogeneity is one of natures of general WSDSs, that is, measurement quality p_i, sample size n_i and confidence interval U_i for individual nodes might vary. For node i, accompanying confidence p_i, n_i and U_i are given as follows.

$$n_i = \lfloor \frac{(\sigma_{e,i}^2 + \frac{N_0}{2})[Q^{-1}(\frac{1-p_i}{2})]^2}{\Delta x_i^2} \rfloor \quad and \quad v_i \in [\hat{v}_{n_i} - \Delta x_i, \hat{v}_{n_i} + \Delta x_i] \tag{4.15}$$

4.2.1.4 Data Collection

After active nodes being chosen, a data centric routing algorithm, EM-GMR[103] is employed, which is a multipath, power-aware and mobility-aware routing scheme. It is used to establish route-tree from active nodes to front-end nodes for query answers' return. EM-GMR uses reactive networking approach, in which a route is found only when a message is to be delivered from a source to a destination.

In EM-GMR, *distance to the destination*, *remaining battery capacity*, and *mobility* of each sensor node are considered. The geographical locations of destination node are known to source nodes (as in [104]), and the physical location of each node can be estimated easily if the locations of three nodes (within a communication range) are known in a WSN. This scheme is a fully distributed approach where each sensor only needs the above three parameters, and fuzzy logic systems (FLS) are utilized to handle those three parameters. The EM-GMR scheme consists of route discovery phase, route reconstruction phase and route deletion phase.

- *Route Discovery Phase*

  The source node uses a fuzzy logic system (FLS)[81] to evaluate all eligible nodes (closer to the destination location) based on the parameters of each node: distance to the destination, remaining battery capacity, and degree of mobility. It chooses

the top M nodes based on the degree of the possibility (output of FLS). The source node sends a Route Notification (RN) packet to each desired node, and each desired node would reply using a REPLY packet if it is available. After a certain period of time, if the source node does not receive REPLY from some desired nodes, it will pick the node with the $M + 1$st degree of selection possibility. In the second hop, selected nodes in each path will choose its next hop node using a FLS.

- *Route Reconstruction Phase*

  Because each node is mobile, it may be possible that some nodes move out of the communication range or some nodes die out, which will lead to link failure. Then a route reconstruction phase is started. The last-hop of the failure node in the routing path will apply FLS to determine the selection possibility for all of its eligible neighbor nodes, and choose the top one degree node (via RNREPLAY procedure). The new node will determine its next node accordingly.

- *Route Deletion Phase*

  Energy, mobility and physical location of each node are changing. It may be possible that a node (in path) observes that its next node is not the optimal after a while, and then this node will initiate a route deletion phase. This node will send an RN packet to the optimal node via common control channel, and this RN packet will also be received by the original relay node, which will notice that the original path is deleted.

In EM-GMR for M-path routing, the source node selects M nodes in its communication range for the first hop relay. Assume there are N ($N > M$) nodes in its communication range, nodes who are further to the destination node than the source node are not considered. Choosing M nodes from remaining eligible nodes is based on a FLS. Starting the second hop, each node in the M-path selects its next hop node also using a FLS. For example, as illustrated in Fig. 4.3, node B needs to choose one node

from eligible nodes C, D, E, F, H based on their three parameters, and sends RN packet to the selected node and waits for REPLY. If the top one node is unavailable (selected by another path or busy), then the top second node will be selected. By this means, M paths can be set up.

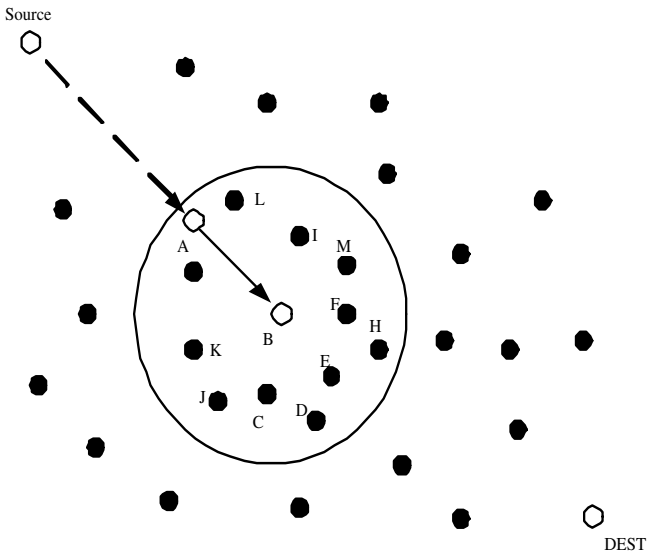


Figure 4.3. Illustration of node selection.

Note that EM-GMR considers distance, remaining battery capacity, and mobility of each node during route path setting up. This scheme could tremendously reduce frame loss rate and link failure rate since mobility is considered, so that incompleteness information caused by poor link quality can be reduced at certain degree.

4.2.2 Energy Consumption Control for Query Processing

In energy consumption control, three strategies are employed. There are active node number control, sample size control and link quality control.

Firstly, in query SVM design, nodes' location is considered besides measurement quality and remaining battery capacity, since it is directly related to the number of active nodes. Through solving optimal location problem, as few as possible nodes can be employed to cover as large as possible monitoring space so as to carry out energy reservation task.

However, too large sample size implies a waste of resources, and too small sample size will diminish the utility of results. QGEE achieves (4.15) to specify the value of sample size during information sensing, so that it can acquire enough samples to meet users' pre-specified precision, and reduce energy consumption for data sensing.

Third, frame loss rate and link failure rate are tremendously reduced through choosing more suitable nodes to set up route-tree for queries. With this improvement, energy consumption is reduced for route-tree maintenance and information retransmission.

4.2.3 Final Query Answer Acquisition

Once query processing strategies have been optimized, disseminated, and semi-manufactured query answers have been acquired from active nodes, query processing operators continue to acquire final query answers. Aggregations are required in many database applications. Common functions applied to collections of numeric value include SUM, AVERAGE, MAXIMUM, and MINIMUM. This paper will specifically discuss how to obtain final query answers, as examples, which focus on those most often used aggregation operations: MAXIMUM, MINIMUM, and AVERAGE.

Returned semi-manufactured answers have a confidence interval form, i.e., $[\hat{v}_{n_i} - \Delta x_i, \hat{v}_{n_i} + \Delta x_i]$ with confidence p_i $(i = 1, 2 \cdots, \psi)$. For simplified reason, letting $l_i = \hat{v}_{n_i} - \Delta x_i$, $h_i = \hat{v}_{n_i} + \Delta x_i$. Assuming there are ψ active nodes for a query.

4.2.3.1 MAXIMUM/MINIMUM Aggregation

The cumulative distribution function (CDF) of $\hat{v}_{n_i}$ is given as follows according to (4.10).

$$F_{\hat{V}_{n_i}}(\hat{v}_{n_i}) = Q(\frac{\sqrt{n_i}(\hat{v}_{n_i} - \mu_{x_i})}{\sigma_{x_i}}) \tag{4.16}$$

Since measurements from individual active nodes are independent with each other, the CDFs for $Z_{max} \triangleq \max_i(\hat{v}_{n_i})$ and $Z_{min} \triangleq \min_i(\hat{v}_{n_i})$ $(i = 1, 2, \cdots, \psi)$ are given as follows for MAXIMUM and MINIMUM.

$$F_{Z_{max}}(z) = \prod_{i=1}^{\psi} F_{\hat{V}_{n_i}}(z) = \prod_{i=1}^{\psi} Q(\frac{\sqrt{n_i}(z - \mu_{x_i})}{\sigma_{x_i}}) \tag{4.17}$$

And

$$F_{Z_{min}}(z) = 1 - \prod_{i=1}^{\psi}\{1 - F_{\hat{V}_{n_i}}(z)\} = 1 - \prod_{i=1}^{\psi}\{1 - Q(\frac{\sqrt{n_i}(z - \mu_{x_i})}{\sigma_{x_i}})\} \tag{4.18}$$

For the MAXIMUM aggregation, the final query answer is $Z_{max} \in [l_{max}, h_{max}]$ with p_{max} confidence in bounded probability form. Here

$$l_{max} = arg\max_i\{l_i\} \quad and \quad h_{max} = arg\max_i\{h_i\} \qquad p_{max} = \frac{1}{\psi}\sum_{i=1}^{\psi} p_i \tag{4.19}$$

For MINIMUM aggregation, the final query answer is $Z_{min} \in [l_{min}, h_{min}]$ with p_{min} confidence in bounded probability form. Where

$$l_{min} = arg\min_i\{l_i\} \quad and \quad h_{min} = arg\min_i\{h_i\} \qquad p_{min} = \frac{1}{\psi}\sum_{i=1}^{\psi} p_i \tag{4.20}$$

4.2.3.2 *AVERAGE Aggregation*

In this aggregation operation, a derivative value over a group of active nodes' data is returned. The PDF for $Z_{avg} \triangleq \frac{1}{\psi}\sum_{j=1}^{\psi}\hat{x}_{n_j,j}$ $(f_{Z_{avg}(z)})$ has the similar distribution to $f_{\hat{X}_n}(\hat{x}_n)$. But the mean and variance are $\frac{1}{\psi}\Sigma_{j=1}^{\psi}\hat{\mu}_{n_j,j}$ and $\frac{1}{n\psi^2}\sum_{j=1}^{\psi}\hat{\sigma}^2_{n_j,j}$ individually.

For AVG aggregation, the final query answer is $Z_{avg} \in [l_{avg}, h_{avg}]$ with p_{avg} confidence in bounded probability form. Where

$$l_{avg} = \frac{1}{\psi}\sum_{j=1}^{\psi} l_i \quad and \quad h_{avg} = \frac{1}{\psi}\sum_{j=1}^{\psi} h_i \qquad p_{avg} = \frac{1}{\psi}\sum_{i=1}^{\psi} p_i \tag{4.21}$$

4.3 Simulations and Performance Evaluation

In the simulations, nodes are randomly deployed in an area $10 \times 10m^2$, and sensing range for individual nodes is 1m. Initial energy of nodes uniformly distributes within [0,5]J. Monte Carlo simulations are run to remove the randomicity of simulation results. We compare QGEE against the query processing method without any query optimization.

The energy consumption model for data sensing is shown as follows.

$$E_q = E_{se} * S_t \tag{4.22}$$

where E_q is the total energy consumed by sensing. E_{se} is the energy consumed by sensing unit sample. S_t is the sample period. In this simulation, $E_{se} = 5nJ/sample$.

Same energy consumption model is used as in [83] for radio hardware. To transmit an l-symbol message for a distance d, the radio expends:

$$E_{Tx}(l,d) = E_{Tx-elec(l)} + T_{Tx-amp(l,d)} = l \times E_{elec} + l \times e_{fs} \times d^2 \tag{4.23}$$

and to receive this message, the radio expends:

$$E_{Rx} = l \times E_{elec} \tag{4.24}$$

The electronics energy, E_{elec}, as described in [83], depends on factors such as coding, modulation, pulse-shaping and matched filtering. The amplifier energy, $e_{fs} \times d^2$ depends on the distance to the receiver and the acceptable bit error rate. In this paper, $E_{elec} = 50nJ/syn$ and $e_{fs} = 10pJ/sym/m^2$.

4.3.1 Energy Reservation Performance

In this simulation, before active nodes selection, nodes individually update their VSM vectors, i.e., determining latest values for R_l, A_d and B_m. Based on this query VSM vector, its OCI ζ can be determined. Through collaboration among nodes, each node can acquire their neighbors' OCIs. Then active nodes can be distributedly selected according to those OCIs. Data Sensing, packet transmitting and receiving are considered for energy consumption. Except for active nodes, other nodes in a network will enter energy saving mode - sleeping mode. Assuming that in sleep mode there is no data sensing nor communication. Moreover, a node is excluded from experiments when its power is used up. Same number of nodes is used for different algorithms and queries are submitted to the network until all nodes are dead.

In Fig. 4.4, we plot query index versus nodes dead time. We can see that after processing about 20 queries, all nodes, without query optimization, use up their energy. But for QGEE, the whole network is not down until 53 queries are completed. Therefore, QGEE can reserve 50% of energy on processing same number of queries.

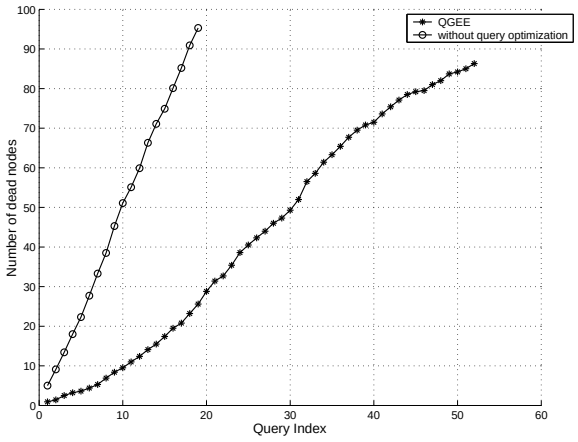


Figure 4.4. Nodes Dead Time.

In Fig. 4.5, we compare the coverage of these two schemes. Observed that, QGEE employs $70 - 45 = 25$ less nodes to cover 90% area interested. This simulation result illustrates the reason why QGEE can implement energy reservation. That is, about $\frac{25}{70} \times 100 = 35.71\%$ nodes switch to energy saving model during query processing.

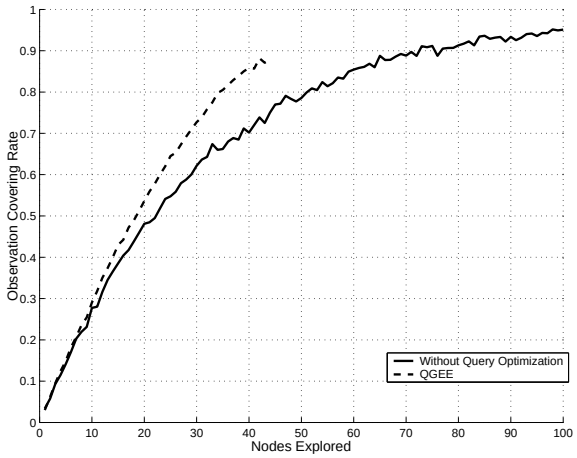


Figure 4.5. Observation Coverage Rate.

By employing QGEE, the energy is saved and the network lifetime is extended. But the cost to achieve this improvement is the decrease of coverage. Fig. 4.6 shows that the biggest decrease of monitoring coverage is 16.6% for QGEE.

4.3.2 Quality Guarantee Performance

In this simulation, various requirements on query answers' confidence are given (i.e., various value for p). To simplify the simulation scenarios, enough nodes are set up to satisfy measurement quality requirements. For MAXIMUM, MINIMUM and AVERAGE aggregation operation, we check the probability of true values locating within the confidence intervals acquired at front-end nodes (see Table 4.2). p_1 is the pre-specified value for p. p_2 is the acquired value for p using QGEE.

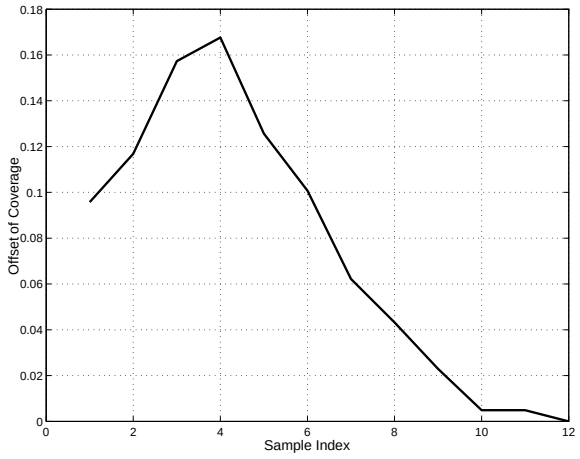


Figure 4.6. Decrease of Coverage.

Table 4.2. Confidence for query answers

p_1	MAXIMUM	MINIMUM	AVERAGE
	p_2	p_2	p_2
0.8	0.8133	0.8133	0.8112
0.9	0.9139	0.9151	0.9144
0.9050	0.912	0.9122	0.9144
0.9246	0.93	0.9318	0.9301
0.9334	0.948	0.942	0.9428
0.95	0.958	0.9561	0.9551
0.99	0.9939	0.9924	0.9938

Note that QGEE can successfully obtain suitable confidence intervals to guarantee the true value located within them with a probability p_2, which is equal to or larger than the pre-specified probability p_1. In reality, maybe, all available nodes do not own enough measurement quality to satisfy users' requirement. Then, even using QGEE, expected results still cannot be acquired. However, QGEE processes query with best effort.

4.3.3 EM-GMR Performance

We compare EM-GMR against the geographical multipath routing (GMR)[104] scheme where only distance to the destination is considered. We run the simulations using OPNET. 60 nodes in total are deployed initially. There are 5 couples of source and destination nodes communicating at the same time in this network. Each node (including source and destination nodes) has moving speed ranging from 0-10m/s, and its moving speed changes in every 10s. The frame length is 512 symbols and symbol rate is 9.6ksym/sec in this simulation. Each sensor locally broadcasts a beacon message in every 2s to keep link alive, so that neighbor table can be updated (including new neighbor joins in, and old neighbors expire). That information is used for route discovery, reconstruction, and deletion. The coherence time is set as 10s.

In Fig. 4.7, we plot the simulation time versus the number of nodes dead. Observe that when 50% nodes (30 nodes) die out, the network lifetime for EM-GMR has been extended about $\frac{175-125}{125} = 40\%$. In Fig. 4.8, we compare the frame loss rate of these two scheme. Observe that EM-GMR outperforms GMR by about 20% on frame loss. The average latency during transmission (end-to-end) is 419.68ms for EM-GMR and 407.5ms for GMR, and link failure rate for EM-GMR is 5.68%, but for GMR it is 10.42%.

4.4 Conclusion

This chapter proposes a quality-guaranteed and energy-efficient (QGEE) algorithm for WSDSs. It employs an in-network query processing method to task WSDSs through declarative queries, and uses confidence interval strategy to determine the accuracy of a query answer. In QGEE, the correlation between a query and a node is calculated by vector space model (VSM), and a query correlation indicator (QCI) is designed to quantify the priority of becoming active for individual nodes. Given a query, QGEE will

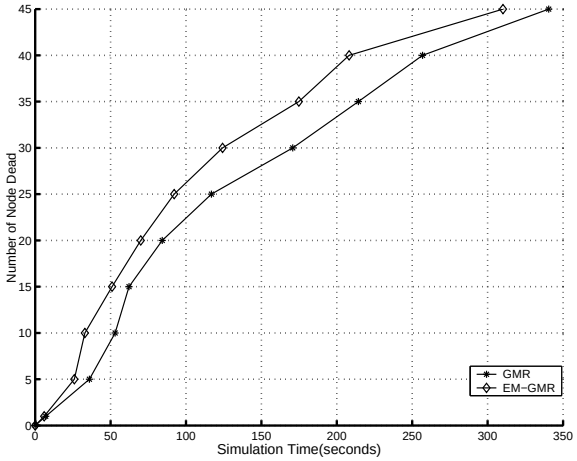


Figure 4.7. Simulation time versus number of nodes dead.

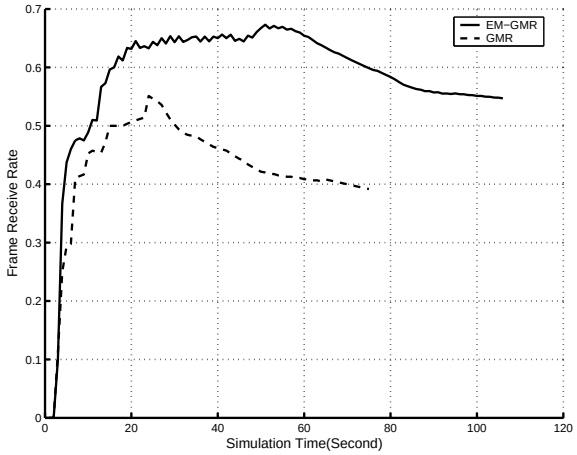


Figure 4.8. Simulation time versus frame loss rate.

adaptively form an optimal query plan in terms of energy efficiency and quality awareness. This approach can reduce the disturbance from measurements with extreme error and minimize energy consumption, while providing satisfying service for various applications. Furthermore, probabilistic method is employed to formulate the distribution of imperfect information and the accuracy of each query answer. The probability corresponding with each query answer can be used to determine the amount of confidence the users should have.

Simulation results demonstrate that QGEE can reduce resource usage by about 50% and frame loss rate by about 20%. Moreover, the confidence of acquired query answers is always higher than, or equal to, users' pre-specified precision.

CHAPTER 5

MAC PROTOCOL FOR UWB WSNS

One approach implements energy reservation through reducing the amount of information exchanged over network and considers the problem of reducing power consumption on wireless interfaces. The research in [35] points out that a bit rate of $100Kbps$ over 5 meters with no more than 1 mW power consumption. Hence, UWB is the best choice in terms of energy efficiency for energy-constraint WSNs. For energy constraint problem for generic WSNs, we have done some work in [98] and [105]. Our proposed energy-efficient MAC protocols not only reserve energy to extend network lifetime but also own the capability to solve accumulative clock drift problem without network synchronization. In this paper, we extend our previous work to UWB communication systems.

5.1 Theoretic Analysis on the Variance of Network Throughput

5.1.1 Network Model

Due to the limited transmission range of current UWB signals, UWB technology is considered for short-range communications. The network model assumed in this paper is that of "piconet" or clustered architecture. A piconet is a collection of devices consisting of one master or piconet controller (PNC) device and remaining slave devices. The PNC is responsible for scheduling the communication between slaves. Each piconet can have only one PNC and up to 255 slaves. A device can be a PNC in only one piconet but it can be a slave in multiple piconets simultaneously. In this network, we assume that all nodes have the same communication and computing capabilities. Since clustering method design is not our target of this paper, we assume that we can set up this hierarchical

clustering topology for a randomly deployed network through certain existing methods, such as the energy-efficient self-organization (ESO)[106] algorithm.

5.1.2 Physical Layer Model

The UWB physical model, on which the design of our protocol bases, is described in this section. The most common and traditional way of emitting an UWB signal is by radiating pulses that are very short in time. IR transmits extremely short pulses giving rise to wide spectral occupation in the frequency domain (bandwidth from near DC to a few gigahertz). The way by which the information data symbols modulate the pulses may vary. Pulse Position Modulation (PPM) and Pulse Amplitude Modulation (PAM) are commonly adopted modulation schemes[107][108]. In addition to modulation, in order to shape the spectrum of the generated signal, data symbols are encoded using pseudorandom or pseudnoise (PN) codes. Fig. 5.1 reports an example of signal waveform, each characterized by a TH code word. Here T_f is the frame duration and $T_f = N_h T_c$.

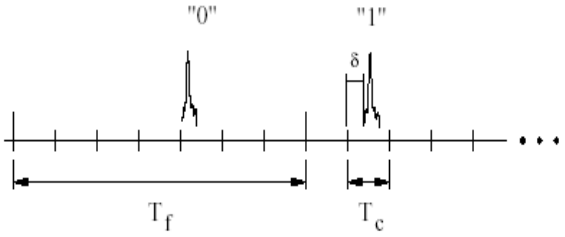


Figure 5.1. UWB signal waveform with PPM[1].

During information propagation, we consider the multipath-affected UWB radio channel. The presence of multiple paths between the transmitter and the receiver makes the channel exhibit time-variant properties. Due to the distortion, the received signal

often has little resemblance with the transmitted waveform. Two popular UWB channel models are used in most of researches. They are S-V model[109] and IEEE 802.15.3a[110] model. We use IEEE 802.15.3a model in this paper. The impulse response of channel is described as:

$$h(t) = \sum_{j=1}^{L(t)} \alpha_j(t)\delta(t - \tau_j(t)) \tag{5.1}$$

where $\delta(t)$ is the Dirac function. $\alpha_j(t)$ and $\tau_j(t)$ are the channel gain and the delay measured at time t for the jth path. $L(t)$ is the number of path observed at time t.

Then, the signal $r(t)$ at the receiver side can be expressed as follows:

$$\begin{aligned} r(t) &= x(t) * h(t) + \omega(t) + n(t) \\ &= \sum_{j=1}^{L(t)} \alpha_j(t)x(t - \tau_j(t)) + \omega(t) + n(t) \end{aligned} \tag{5.2}$$

where $x(t)$ is the transmitted signal. $\omega(t)$ is the multiuser interference. $n(t)$ is AWGN noise.

Supposing N pairs of communicating UWB terminals, and each pair consisting of one transmitter and one receiver and using one pseudorandom code, N links are active and the $SINR$ at the ith link's receiver is formed as following[75]:

$$SINR_i = \frac{P_i g_{ii}}{R_i(\eta_i + T_f\sigma^2 \sum_{k=1,k\neq i}^{N} P_k g_{ki})} \quad (i = 1, 2, \cdots, N) \tag{5.3}$$

where R_i is the binary bit rate of the ith link; P_i is the average power emitted by the ith link's transmitter; g_{ij} is the path gain from the ith link's transmitter to the jth link's receiver; η_i is the background noise energy plus interference from other non-UWB systems; σ^2 is a dimensional parameter depending on the shape of monocycle.

Even though, the parameters characterizing the channel impulse response in (5.2) are time-varying, we can generally assume, however, that this rate of variation is slow compared to the pulse rate. In other words, we can assume that the channel to be

stationary within an observation time T, which is larger than the average pulse repetition period. Under this assumption, (5.2) can be rewritten as follows:

$$r(t) = \sum_{j=1}^{L(t)} \alpha_j x(t - \tau_j(t)) + \omega(t) + n(t) \tag{5.4}$$

In this case, the total multi-path gain g, which measures the total amount of energy collected over $L(t)$ pulses, is determined as following:

$$g = \sum_{j=1}^{L(t)} \mid \alpha_j \mid^2 \tag{5.5}$$

Note that $g \leq 1$ and is related with the attenuation suffered by the transmitted pulses during propagation. In multi-path environments, g decreases with distance according to the path-loss model[111] as following:

$$g = \frac{g_0}{d^\beta} \tag{5.6}$$

where g_0 is the reference value for power gain evaluated at $d_0 = 1m$ and β is the exponent of power or energy attenuation law.

5.1.3 Impact of Simultaneous Transmissions on Throughput

Within a network, piconets can be treated as independent with each other, since the distance among piconets is long enough to permit us to ignore the interference among them (piconet formation scheme can ensure this assumption to be held). Thus, the analysis on the change of network throughput can be simplified into a set of independent analysis on the change of throughput for individual piconets. Then combining all results together linearly, we can obtain the impact of simultaneous transmission on the aggregate throughput. In the following parts, the discussions focus on the impact of simultaneous transmissions on the throughput within a piconet.

According to the noisy channel coding theorem: if the data transmission rate is less than the channel capacity, there exist channel codes (and decoders) that make it possible to achieve reliable communication. Otherwise, it is not possible to make the probability of error tend toward zero with any code. In this paper, the theoretic maximum channel capacity (C_{t-max}) is defined as the largest channel capacity implied by channel state, and the achievable maximum channel capacity (C_{a-max}) is defined as the largest channel capacity which is acquired using the highest data rate to make reliable communication. From (1.2) and (5.3), it is noted that C is in inverse proportion to R, i.e., $C \propto \frac{1}{R}$. We derive the data rate referring to C_{a-max}, noted as R_{a-max}, to achieve our goal - trying to not only enhance data rate to shorten the transmission latency but also ensure reliable communication. The value of R_{a-max} is calculated as following:

$$R_{a-max} = \sqrt{\frac{\xi P g}{\eta + T_f \sigma^2 \sum_{k=1,k\neq i}^{N} P_k g_{ki}}} \tag{5.7}$$

For estimating the throughput of a piconet, in which there are N pairs of communication terminals making communication simultaneously with data packets R_{a-max}, the throughput (TH_{put}) of this piconet is calculated as following:

$$TH_{put} = \sum_{i=1}^{N} \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + T_f \sigma^2 \sum_{k=1,k\neq i}^{N-1} P_k g_{ki}}} \tag{5.8}$$

For an existing piconet, if we add m more pairs of communication terminals, what is the influence on the throughput of this piconet? Through analyzing the change of throughput when adding different number of communication pairs or picking up same number of communication pairs but located at different position, we try to obtain the criterion for the formation of simultaneous transmission subset.

In order to simplify the description, we let $U_i \triangleq T_f \sigma^2 \sum_{k=1,k\neq i}^{N} P_k g_{ki}$. Without losing generality, we let the newly added communication pairs be pair $N+1$ to pair

$N+m$. For existing communication pairs (i.e., pair 1 to pair N), the new achieved channel capacity $C'_{a-max,i}$ is:

$$C'_{a-max,i} = \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + U_i + T_f \sigma^2 \sum_{k=N+1}^{N+m} P_k g_{ki}}} \quad (i = 1, \cdots, N) \tag{5.9}$$

And the achieved channel capacity for new communication pairs $C_{a-max,i}$ is:

$$C_{a-max,i} = \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + T_f \sigma^2 \sum_{k=1,k\neq i}^{N+m} P_k g_{ki}}} \quad (i = N+1, \cdots, N+m) \tag{5.10}$$

Based on results in (5.9), (5.10), and (5.8), the influence of adding more simultaneous transmissions on throughput of a piconet is expressed as the change of piconet throughput, noted as $\triangle TH_{put}$ and calculated using (5.11).

$$\begin{aligned} \triangle TH_{put} &= TH'_{put} - TH_{put} \\ &= \sum_{i=1}^{N} \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + U_i + T_f \sigma^2 \sum_{k=N+1}^{N+m} P_k g_{ki}}} + \sum_{i=N+1}^{N+m} \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + T_f \sigma^2 \sum_{k=1,k\neq i}^{N+m} P_k g_{ki}}} \\ &- \sum_{i=1}^{N} \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + U_i}} \end{aligned} \tag{5.11}$$

We also obtain the relationship between the new data rate $R'_{max,i}$ and the original data rate $R_{max,i}$ for node i ($i = 1, \cdots, N$) as shown in (5.12). Note that, adding some communication pairs into an existing network will decrease the highest data rate which ensures reliable communication, since $\frac{T_f \sigma^2 \sum_{k=N+1}^{N+m} P_k g_{ki}}{\xi P_i g_{ii}}$ is always positive.

$$\frac{1}{R'^2_{a-max,i}} = \frac{1}{R^2_{a-max,i}} + \frac{T_f \sigma^2 \sum_{k=N+1}^{N+m} P_k g_{ki}}{\xi P_i g_{ii}} \tag{5.12}$$

From (5.11) and (5.12), we observe that, when adding more communication pairs, the change of piconet throughput is related with the negative influence caused by degrading the highest data rate for existing communication pairs and the positive influence due to permitting more communication pairs work concurrently. Thus, from networkwide

perspective, letting more nodes work concurrently does not definitely mean upgrade or degrade the throughput of a piconet. There is a watershed for it. That is, when the negative influence equals to or is smaller than the positive influence of adding some communication pairs into an existed net, the network performance in term of throughput will be improved, or at least not be degraded. Consequently, we can formulate the criterion that guarantees the network throughput to be improved when allowing more simultaneous communication pairs as following:

$$\begin{aligned} & \sum_{i=1}^{N} \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + U_i}} - \sum_{i=1}^{N} \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + U_i + T_f \sigma^2 \sum_{k=N+1}^{N+m} P_k g_{ki}}} \\ \leq & \sum_{i=N+1}^{N+m} \sqrt{\frac{\xi P_i g_{ii}}{\eta_i + T_f \sigma^2 \sum_{k=1, k \neq i}^{N+m} P_k g_{ki}}} \end{aligned} \tag{5.13}$$

First, we consider the $N = 1$ and $m = 1$ scenario as our analysis basis. We assume that the distance between transmitters and receivers is same for each communication pair (i.e., $d_{ii} = d$), thus each communication pair can use same power to make transmission. Moreover, the noise floor for communication is fixed. Thus (5.11) is changed into

$$\begin{aligned} \triangle TH_{put} &= \sqrt{\frac{\xi P g_0}{\eta d_{22}^{\beta} + T_f \sigma^2 P g_0 (\frac{d_{22}}{d_{12}})^{\beta}}} + \sqrt{\frac{\xi P g_0}{\eta d_{11}^{\beta} + T_f \sigma^2 P g_0 (\frac{d_{11}}{d_{21}})^{\beta}}} \\ &- \sqrt{\frac{\xi P g_0}{\eta d_{11}^{\beta}}} \end{aligned} \tag{5.14}$$

In this case, $\triangle TH_{put}$ is a function of the distance between the original existed communication pair's transmitter and the added communication pair's receiver (d_{12}), the distance between the added communication pair's transmitter and the original existed communication pair's receiver (d_{21}), i.e., $\triangle TH_{put} = f(d_{12}, d_{21})$. We define the throughput change rate of a piconet as $\frac{\triangle TH_{put}}{TH_{put}}$. Based on (5.14), we plotted the piconet throughput change rate versus the strength of multiuser interference. Fig. 5.2 shows the hypersurface for $\forall d_{12}, d_{21} \in [0,\ 20]$. Table 5.1 shows the parameters we use.

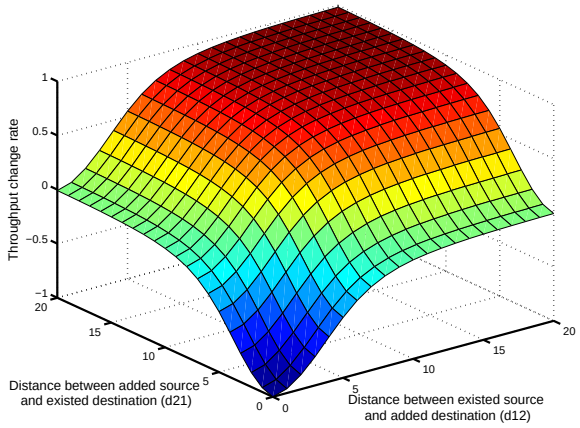


Figure 5.2. Relationship between the throughput and the multiuser interference.

Table 5.1. typical values for parameters in (5.14)

Parameter	Typical Value	Parameter	Typical Value
T_f	100 ns	σ^2	1.9966×10^{-3}
η	$2.568 \times 10^{-21}\ V^2 s$	P	$1\ \mu W$
d	1 m	ξ	$\log_2^e$
g_0	7.9433×10^{-6}		

Note that, to ensure the throughput to be enhanced when adding one or more communication pairs, the shortest distance for d_{12} and d_{21} should be at least equals to a threshold d_{min}, which is related with the choice of transmission power (P), environment noise strength (η), environment exponential (β), symbol time (T_f), monocycle shape (σ^2) and power gain (g_0) evaluated at $d_0 = 1m$, but the distance between the transmitter and the receiver of originally existing communication pair. That is:

$$d_{min} = \{(\frac{1}{(\sqrt{\frac{1}{\eta}} - \sqrt{\frac{1}{\eta + T_f \sigma^2 P \frac{g_0}{d_{max}^{\beta}}}})^2} - \eta)(g_0 T_f \sigma^2 P)^{-1}\}^{\frac{1}{\beta}} \tag{5.15}$$

5.2 Throughput-Maximized MAC Protocol (TM-MAC) Design

We have proposed an energy-efficient MAC protocol, ASCEMAC in [98] for WSNs. ASCEMAC intelligently forces nodes power on and off their batteries alternately to implement communication, as well as to reduce energy consumption on collision and idle listening to extend network lifetime. However, for UWB communication systems, TDMA-based and contention-based mutual exclude media access control schemes are not the best choice any more in terms of spectrum efficiency. Motivated by this challenge, we propose our algorithm: throughput-maximized MAC protocol specially for UWB communication systems.

TM-MAC divides the system time into four phases: TRFR (Traffic-rate & Failure-rate)-Phase, Schedule-Broadcast-Phase, On-Phase and Off-Phase. TRFR message, TRFR-Phase duration design, matching schedule establishment and maintenance, schedule interval design and time-slot allocation mechanisms for TM-MAC are the same as in ASCEMAC. While, in TM-MAC, during On-Phase the further divided super-time-slots are occupied by subsets individually. TM-MAC, based on the network topology, is responsible for further dividing a network into a set of subsets, in which communication pairs can make communication simultaneously. Consequently, TM-MAC can not only inherits the advantages of ASCEMAC, but also maximizes network throughput.

5.2.1 Optimizing Piconet Throughput

The network is represented as a directed graph $G = (V, E)$. V is the set of nodes in a piconet. $e = (u, v)$ is an edge in E iff nodes u and v are transmitter and receiver of a communication pair. Fig. 5.3 shows an example, in which nodes $A, C, E, G, , I, K$ are transmitters, B, D, F, H, J, L are receivers. If a receiver locates within another transmitter's communication range, interference will not avoid when they communicate simulta-

neously. The interference coming from other communication pairs is denoted by dotted lines.

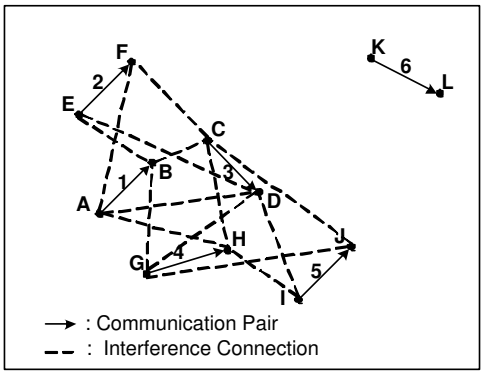
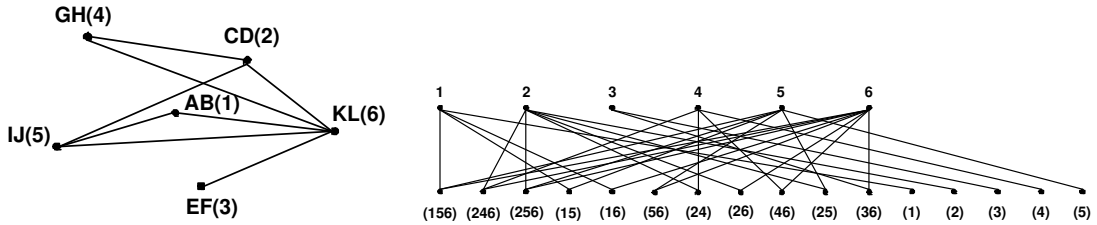


Figure 5.3. Network graph G.

In our algorithm, the interference caused by the newly added communication pairs is untolerable when the created throughput is smaller than the original one. Otherwise, it is tolerable. We consider all communication pairs in a piconet to generate the interference tolerable graph $G' = (V', E')$ according to (5.15). $V' \subset E$, i.e., each point in G' is a communication pair in G. $e' = (u', v')$ is an edge in E' iff the achieved throughput is bigger than the throughput generated by those two terminals separately. According to the network topology, we utilize the conclusion acquired in (5.15) to detect the untolerable interference from other communication pairs. Fig. 5.4(a) presents the inference tolerable graph for the graph in Fig. 5.3.

We generate the potential subset to forme graph G''. $G'' = (V_1, V_2, E'')$ is a bit-partite graph such that $V_1 = V'$, and each point in V_2 presents all cliques and sub-cliques in G'. $e'' = (u'', v'')$ is an edge in E'' iff $u'' \in V_1, v'' \in V_2$, and u'' belongs to one of cliques in G' represented by v''. Fig. 5.4(b) represents the potential group formation graph for the inference tolerable graph shown in Fig. 5.4(a).



(a) (b)

Figure 5.4. (a)Interference tolerable graph G' and (b)Potential Group graph G''.

Each clique in G' represents a potential subset of communication pairs making simultaneous communication to enhance throughput. We represent each point in V_1 as an one-off source which is granted and must to be used once in a super-timeslot. Then, the subset formation in a piconet represents the optimal classification for all nodes within a piconet, and a point in V_2 is permitted to occupy the channel mutually excluded if and only if it can make contribute to achieve higher throughput and one-off source is still available.

Let I_{ij} be an indicator function such that $I_{ij} = 1$ if the point $j \in V_2$ is allocated with channel by point $i \in V_1$, and $I_{ij} = 0$ otherwise. Let $TH'_{put,j}$ be the sub-throughput generated by point j. Then the channel allocation problem can be represented as a set of the following linear constraints.

$$\begin{aligned} &\forall i, \quad \sum_j I_{ij} = 1 \\ &\forall j, \forall i, \quad r_j = TH'_{put,j} \times \sum_i I_{ij} \end{aligned} \tag{5.16}$$

Where r_j is the sub-throughput generated by point j in V_2, and the unit for it is pks/sec. Note that this set of constraints captures the location-dependent interference on piconet throughput characteristics of UWB communication systems.

The utility function $U(r)$ for a throughput r is defined as:

$$U(r) = r \tag{5.17}$$

Since our goal for subset optimization is to improve piconet throughput as much as possible, maximizing piconet throughput problem can be modelled by the following equations:

$$Maximize \quad \sum_{j} U(r_j)$$

Subject to

$$\begin{aligned} &\forall i, \quad \sum_{j} I_{ij} = 1 \\ &\forall j, \forall i, \quad r_j = TH'_{put,j} \times \sum_{i} I_{ij} \end{aligned} \tag{5.18}$$

5.2.2 Essential Parameters Design

The study in [98] and [105] have shown that essential algorithm parameters, such as power-on/off duration and schedule-broadcast interval, which greatly influence system performance in terms of energy reservation and communication capability. How to adaptively determine the value for those essential parameters of TM-MAC is the main task in this section.

5.2.2.1 *O*ff-*P*hase Duration (T_f)

It is now recognized[112][113] that traffic in wired and wireless communication networks is better described by heavy-tailed distributions rather than by Poisson, Gaussian or other classical distributions with exponentially decreasing tails. In this paper, we model network arrivals as Pareto distribution, a heavy-tailed distribution. The probability mass function is given in (5.19).

$$f(x) = \alpha k^{\alpha} x^{-\alpha-1}, \quad 1 < \alpha < 2, \quad k > 0, \quad x \geq k \tag{5.19}$$

and its cumulative distribution function is given by:

$$F(x) = 1 - (\frac{k}{x})^{\alpha} \tag{5.20}$$

where k represents the smallest value the random variable can take.

In this case, we establish an embedded-Markov chain to express the packet arrive process for each user. $N(t)$ is the number of data packets in buffer at time t. N_n^- stands for the queue length when the $n-th$ data packet arrives (the current arrival data packet not included). Even though the queue length of each user does not own Markov property any more, $N_n^-, n \geq 0$ forms a Markov chain, an embedded-Markov chain. The average arrival interval ($\frac{1}{\lambda}$) is given:

$$\frac{1}{\lambda} = \int_k^\infty x dF(x) = \frac{\alpha k}{\alpha - 1} \tag{5.21}$$

During Off-Phase, there are about $T_{f,i} \times \lambda_i$ data packets arrived at node i. We assume the buffer size for node i is B_s. Then the duration, denoted by t_i, within which node i's buffer can be fully filled with arrived data packets is given by $t_i = \frac{B_s}{\lambda_i}$. Considering the first criteria, $T_{f,i}$ for node i should not longer than t_i. In this algorithm, we let

$$T_{f,i} = t_i = \frac{B_s}{\lambda_i} = B_s(\frac{\alpha_i k_i}{\alpha_i - 1}) \tag{5.22}$$

We assume B_s is a constant, $\alpha_i = \alpha$ for all users. While k_i follows a uniform distribution at range [0, k^*]. Note that, the cumulative density function for $T_{f,i}$ is given:

$$F_{T_{f,i}}(t_{f,i}) = P\{T_{f,i} \leq t_{f,i}\} = F_K(\frac{t_{f,i}(\alpha - 1)}{\alpha B_s}) \tag{5.23}$$

Since

$$F_K(k) = \frac{k}{k^*} \tag{5.24}$$

then

$$F_{T_{f,i}}(t_{f,i}) = \frac{t_{f,i}(\alpha - 1)}{\alpha k^* B_s} \tag{5.25}$$

Since, within a cluster, there are multiple nodes which have various traffic arrival rate, the duration for all nodes will not be same. If we let the Off-Phase duration for an

entire cluster $T_{f,tot}$ equal to $i-th$ user, that is $T_{f,tot} = T_{f,i}$. Moreover, since a new arrival to an idle system, rather than going into service immediately, waits for the end of the vacation period, and arrivals are served following a first-come-first-in order. Therefore, the longer $T_{f,tot}$ is, the longer for data packets waiting in buffer for transmission is. We leverage the $GI^*/G/1$ with vacation model to model our system. Through analysis, we try to get the relationship between the average waiting time ($\bar{W}_j$) for $j-th$ user and $T_{f,tot}$, that is $\bar{W}_j = f_j(T_{f,tot})$.

Based on this conclusion, we try to get the probability (p_j) for data packets out of date for $j-th$ when Off-Phase duration equals to $T_{f,tot}$ is given in (5.27).

$$p_j = P\{\bar{W}_j \geq W_{max}\} = 1 - F_{T_{f,tot}}(f_j^{-1}(W_{max})) \tag{5.26}$$

Since $T_{f,tot} = T_{f,i}$, (5.27) is rewritten as follows:

$$p_{ij} = P\{\bar{W}_{ij} \geq W_{max}\} = 1 - F_{T_{f,i}}(f_j^{-1}(W_{max})) \tag{5.27}$$

where p_{ij} is the probability of data packets out of date for $j-th$ user when letting $i-th$ user's Off-Phase duration for the entire cluster.

For a system with a Off-Phase duration $T_{f,i}$ and total data packets out of date probability $\Sigma_j p_j$, we represent our objective function as

$$arg\max_i \quad J(T_{f,i}) = arg\max_i \{\beta T_{f,i} - \gamma \sum_j p_{ij}\} \tag{5.28}$$

where β and γ are systems parameters that respectively represent the "latency constant" and the "penalty constant", which are tuned to achieve the desired trade-off between maximizing energy reservation period and minimizing buffer overflowing rate.

5.2.2.2 *O*n-*P*hase Duration (T_n)

During *O*n-*P*hase, normal nodes start to send data packets through competition. Users, who have data packets to send, access the channel to make communication. If we let the duration for *O*n-*P*hase of nodes be $T_{n,tot}$, the total number of data packets (N_i) arrived is given in (5.29), since the traffic arrival process is independent with the data transmission process. Generally, there are two parts for N_i: one is the data packets arrived during *O*ff-*P*hase, denoted by $N_{f,i}$; the other is the data packets arrived during *O*n-*P*hase, denoted by $N_{n,i}$.

$$N_i = N_{f,i} + N_{n,i} = \lambda_i(T_{f,tot} + T_{n,tot}) \tag{5.29}$$

In our *O*n-*P*hase duration and *O*ff-*P*hase duration designing, we not only try to extend the power off time to reserve energy (through more idle listening avoided), but also need to ensure data packets up to date. Considering that the active duration ($T_{n,i}$) should be long enough for all received data packets to be sent out, then we have

$$R_{b,i}T_{n,i} = \lambda_i(T_{n,i} + T_{f,tot}) \tag{5.30}$$

Solving (5.30) for $R_{b,i}$ we get

$$R_{b,i} = \frac{\lambda_i(T_{f,i} + T_{n,tot})}{T_{n,i}} \tag{5.31}$$

From another aspect of obtaining satisfied data successful transmission rate to acquire data rate, that is $\hat{R}_{b,i}$ for node i acquired in (5.7). We define the objective function for T_f is:

$$U(T_{n,i}) = \sum_j |R_{b,j} - \hat{R}_{b,j}| \tag{5.32}$$

Then the optimum task is shown in follows:

$$T_{n,tot} = arg \min_i U(T_{n,i}) \tag{5.33}$$

5.3 Simulations and Performance Evaluation

We performed extreme simulations to evaluate the performance of our algorithm: TM-MAC. A network with amount of communication pair, which is composed of a transmitter and a receiver, is set up and the radio range (radius) of nodes is 20m. For each communication pair, the distance between the transmitter and the receiver is fixed at 1m. Those communication pairs are deployed randomly in an area of $50 \times 50m^2$ and have no mobility. This network can be treated as one piconet in a large-scale system. Other parameters for physical layer are same as in Table 5.1.

We deploy 5 to 40 communication pairs separately in the same region. Then, we form simultaneous transmission subsets using our TM-MAC. We observe that the number of various subset generated, such as 1-pair subset, 2-pair subset, 3-pair subset and 4-pair subset, which means in a subset there is one communication pair, two communication pairs, three communication pairs and four communication pairs. We run Monte Carlo simulations and make average operation on those results to remove the randomness of simulation results. The results are shown in Table 5.2. We also observed the chance for each communication pair being classified into different subset (See Table 5.3).

Note that, from Table 5.2 and Table 5.3, three to four percent of communication pairs being classified into 1-pair, 2-pair or 3-pair subset. With node density increase, there is more chance to form 4-pair subset, i.e., there is 10% higher probability for 40 communication pairs scenario than for 5 communication pairs scenario.

Within a piconet, we made simulations to check the actual throughput achieved by different subset. Under various node densities from 5 to 40 communication pairs, we plot throughput versus total communication pairs within the same piconet we set up (See Fig.5.5(a)). Note that, subset, within which there are more communication pairs making communication simultaneously acquires higher throughput. In 40 communication pairs scenario, the throughput achieved by 4-pair subset is $119.309kbps$, which is around

Table 5.2. Number of subset generated with various node density

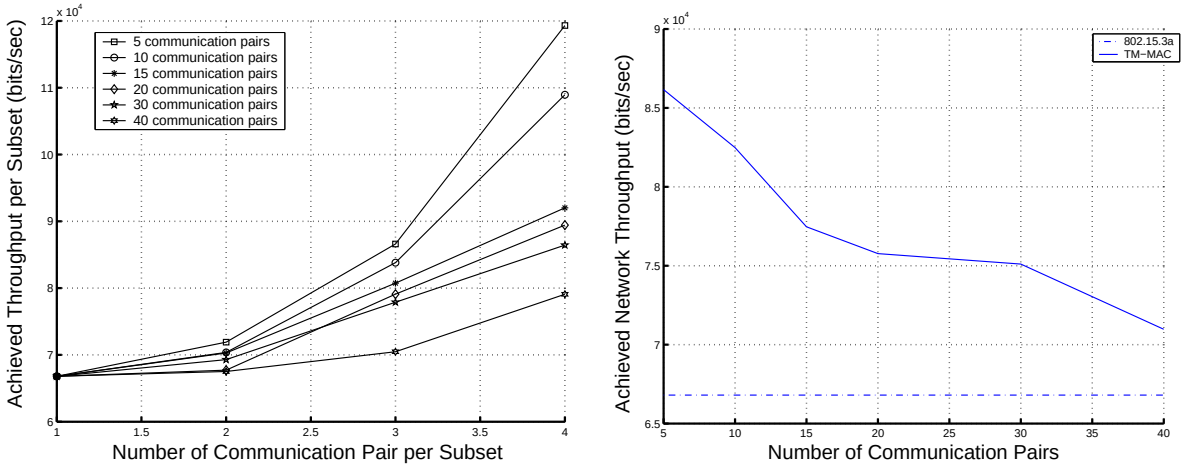
	5 pairs	10 pairs	15 pairs	20 pairs	30 pairs	40 pairs
1-pair subset	1.648	2.964	4.198	5.782	8.398	10.635
2-pairs subset	0.824	1.368	2.175	3.058	5.081	7.293
3-pairs subset	0.464	0.62	0.856	1.134	1.908	2.677
4-pairs subset	0.078	0.61	0.771	1.105	1.429	1.687

Table 5.3. Percentage of communication pair being classified into various subsets

%	5 pairs	10 pairs	15 pairs	20 pairs	30 pairs	40 pairs
1-pair subset	32.96	29.64	27.987	28.91	27.993	26.587
2-pair subset	32.96	27.36	29	30.58	33.873	36.465
3-pair subset	27.84	18.6	17.12	17.01	19.08	20.078
4-pair subset	6.24	24.4	25.893	23.5	19.053	16.87

two times of the throughput achieved by 1-pair subset. Moreover, with the node density increase, the largest throughput which can be achieved is decreased since the interference coming from other users is increased.

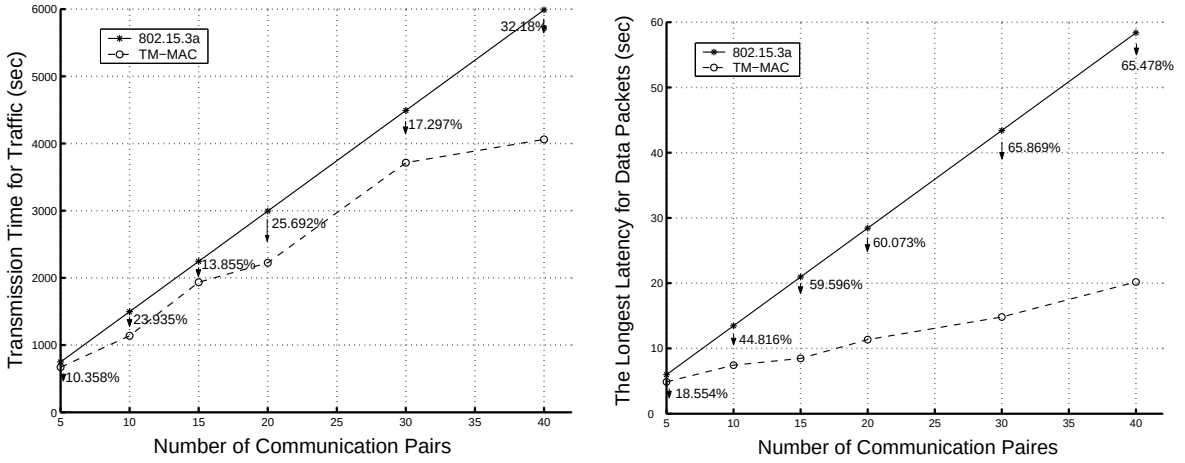
We compared our TM-MAC against IEEE802.15.3a, which uses a mutual excluded scheme to implement media access control - TDMA scheme. We check the achieved throughput, the transmission time needed for certain traffic load and the longest latency for data packet for our TM-MAC and IEEE802.15.3a (See Fig.5.5(b), Fig.5.6(a) and Fig.5.6(b)). Note that, our TM-MAC can achieve higher throughput than IEEE802.15.3a around 5.97% to 25.358%. Given same amount of traffic to networks which run TM-MAC and IEEE802.15.3a separately, the transmission time needed for TM-MAC is shorter than the one for 80.215.3a. The reduced ratio for various node densities from 40 communication pairs to 5 communication pairs locates within the range from 10.358% to 32.18%. Since IEEE802.15.3a uses the mutual excluded scheme for media access control, the communication for various communication pairs is carried out serially. While, for TM-MAC,



(a) (b)

Figure 5.5. (a)Throughput per Subset and (b)Average Throughput.

some communication pairs can make communication simultaneously. The longest latency for TM-MAC is shorter than IEEE802.15.3a. The decreased ratio is from 18.554% to 65.869%.



(a) (b)

Figure 5.6. (a)Transmission Duration and (b)Longest Latency of Data.

5.4 Conclusion

In this chapter, we proposed a MAC protocol: throughput maximized MAC protocol (TM-MAC). In TM-MAC, we implemented multiuser access way instead of mutual exclusion method such as TDMA and random access. Since cross-layer design approach is a good solution to reach the target of highly energy-efficient WSNs, we combined MAC layer and physical layer together to optimize throughput of network. On MAC layer, according to the network topology, TM-MAC re-divides each piconet into several subsets, in which communication pairs can make communication simultaneously and achieve the maximum throughput using the highest data rate. For subset formation, we proposed a general analytical framework that captures the unique characteristics of shared wireless channel and throughput variance, as well as allows the modeling of a large class of systemwide throughput maximizing models via the specification of per-link utilization functions. On physical layer, we analyzed the relationship among the theoretical maximum channel capacity, the achievable maximum channel capacity and the data rate. For multiuser interference, we established a model to adaptively adjust the data rate to ensure certain SNR at the receiver side.

Simulation results demonstrate that TM-MAC can implement throughput maximization to achieve short latency and transmit same amount of data packets over a network during shorter period.

APPENDIX A

VECTOR SPACE MODEL

Vector Space Model (VSM)[114][115] is a way to represent documents through words that they contain. VSM has been widely used in traditional information retrieval (IR) field[116][117]. Most search engines also use similarity measures based on this model to rank Web documents. VSM creates a space in which both documents and queries are represented by vectors. For a fixed collection of documents, an m-dimensional vector is generated for each document and query from sets of terms with associated weights. Then, a vector similarity function such as the inner product can be used to compute the similarity between a document and a query.

In VSM, weights associated with the terms are calculated based on the following two numbers:

- term frequency, f_{ij}, the number of occurrence of term y_i in document x_i; and
- inverse document frequency, $g_i = log(N/d_j)$, where N is the total number of documents in a collection and d_j is the number of documents containing term y_i.

The similarity $sim_{vs}(q, x_i)$ between a query q and a document x_i can be defined as an inner product of query vector Q and document vector X_i:

$$sim_{vs}(q, x_i) = Q \cdot X_i = \frac{\sum_{j=1}^{m} v_j \cdot w_{ij}}{\sqrt{\sum_{j=1}^{m}(v_j^2) \cdot \sum_{j=1}^{m}(w_{ij}^2)}} \tag{A.1}$$

where m is the number of unique terms in a document collection. Document weight $w_{i,j}$ and query weight v_j are

$$w_{ij} = f_{ij} w_{ij} = f_{ij} log(N/d_j) \quad and$$
$$v_j = \begin{cases} log(N/d_j) & y_j \, is \, a \, term \, in \, q \\ 0 & otherwise. \end{cases} \tag{A.2}$$

APPENDIX B

K-PARTIAL SET COVER PROBLEM

Covering problems are widely studied in discrete optimization. Basically, these problems involve picking a least-cost collection of sets to cover elements. Classical problems in this framework include general set cover problem and partial covering problem. k-partial set cover problem[99] as a partial covering problem is about how to choose a minimum number of sets to cover at least n elements, and which k elements should be chosen. k-partial set cover problem can be formulated as an integer program as following:

MINIMIZE:

$$\sum_{j=1}^{m} c(S_j) \cdot x_j \tag{B.1}$$

SUBJECT TO:

$$\begin{aligned} & y_i + \sum_{j:t_i \in S_j} x_j \geq 1 \\ & \sum_{i=1}^{n} y_i \leq n - k \\ & x_j \geq 0 \quad j = 1, 2, \ldots, m, \quad \textit{and} \quad y_i \geq 0 \quad i = 1, 2, \ldots, n, \end{aligned} \tag{B.2}$$

Where $x_i \in \{0,1\}$ corresponds to each $S_j \in S$. Iff set S_j belongs to the cover, then $x_j = 1$. Iff set t_j is not covered, then $y_i = 1$. $t_i \in \Gamma$.

APPENDIX C

FUZZY LOGIC SYSTEMS

Fig.C.1 shows the structure of a fuzzy logic system (FLS) [118]. When an input is applied to a FLS, the inference engine computes the output set corresponding to each rule. The defuzzifier then computes a crisp output from these rule output sets. Consider a p-input 1-output FLS, using singleton fuzzification, *center-of-sets* defuzzification [119] and "IF-THEN" rules of the form [81]

$$R^l : \text{IF } x_1 \text{ is } \mathrm{F}_1^l \text{ and } x_2 \text{ is } \mathrm{F}_2^l \text{ and } \cdots \text{ and } x_p \text{ is } \mathrm{F}_p^l, \text{ THEN } y \text{ is } \mathrm{G}^l.$$

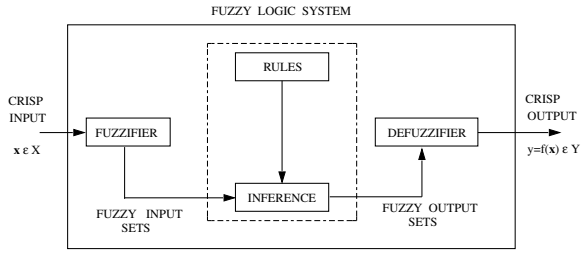


Figure C.1. Structure of a fuzzy logic system.

Assuming singleton fuzzification, when an input $\mathbf{x}' = \{x_1', \ldots, x_p'\}$ is applied, the degree of firing corresponding to the l-th rule is computed as

$$\mu_{\mathrm{F}_1^l}(x_1') \star \mu_{\mathrm{F}_2^l}(x_2') \star \cdots \star \mu_{\mathrm{F}_p^l}(x_p') = \mathcal{T}_{i=1}^{p} \mu_{\mathrm{F}_i^l}(x_i') \tag{C.1}$$

where $\star$ and $\mathcal{T}$ both indicate the chosen t-norm. There are many kinds of defuzzifiers. In this paper, we focus, for illustrative purposes, on the height defuzzifier [81]. It computes a crisp output for the FLS by first computing the height, $\bar{y}^l$, of every consequent set G^l, and, then computing a weighted average of these heights. The weight corresponding to the l-th rule consequent height is the degree of firing associated with the l-th rule, $\mathcal{T}_{i=1}^{p} \mu_{\mathrm{F}_i^l}(x_i')$, so that

$$y_h(\mathbf{x}') = \frac{\sum_{l=1}^{M} \bar{y}^l \mathcal{T}_{i=1}^{p} \mu_{\mathrm{F}_i^l}(x_i')}{\sum_{l=1}^{M} \mathcal{T}_{i=1}^{p} \mu_{\mathrm{F}_i^l}(x_i')} \tag{C.2}$$

where M is the number of rules in the FLS.

In [120], there is a survey on the computation complexity of fuzzy logic system. Because a key element of fuzzy logic is its characteristic trait that transforms the binary world of digital computing into a computation based on continuous intervals, true fuzzy logic must be emulated by a software program on a standard microcontroller/processor. Inform Software Corp. has pioneered the fuzzy logic development tool market with its "fuzzyTECH microkerne" software architecture that provides implementation of fuzzy logic much more efficiently than previous emulation technologies. Now, the same example of a small fuzzy logic system running on a standard 8051 requires about one millisecond only for computation.

APPENDIX D

PUBLICATION LIST

[1] Q. Ren and Q. Liang, "Throughput and energy-efficiency aware ultra wideband communication in wireless sensor networks: a cross-layer approach", accepted by IEEE Transaction on Mobile Computing

[2] Q. Ren, Q. Liang, "Energy and quality aware query processing in wireless sensor database systems," Information Sciences (Elsevier), vol. 177, no. 10, pp. 2188-2205, May 2007

[5] Q. Ren, Q. Liang, "Energy-Efficient Medium Access Control Protocols For Wireless Sensor Networks," EURASIP Journal on Wireless Communications and Networking, vol. 2006, pp. 1-17, Article ID 39814, 2006

[6] Q. Liang, L. Wang and Q. Ren, "Fault-Tolerant and Energy Efficient Cross-Layer Design for Wireless Sensor Networks", International Journal of Sensor Networks 2007 - Vol. 2, No.3/4 pp. 248 - 257

[7] Q. Ren and Q. Liang, "Performance analysis of spectrum sensing for cognitive radio wireless networks: considering exposed and hidden problems", submitted to IEEE Transaction on Wireless Communications

[8] Q. Ren and Q. Liang, "Query processing optimization through sample size and monitoring coverage controlling in wireless sensor networks", in Proc. of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON2006)

[9] Q. Ren and Q. Liang, "An energy-efficient MAC protocol for ultra-wideband communication over wireless sensor networks", in Proc. of IEEE Global Telecommunications Conference 2006 (GLOBECOM 2006), vol. 1, pp.1-5

[10] Q. Ren and Q. Liang, "An energy-efficient MAC protocol for wireless sensor networks", in Proc. of IEEE Global Telecommunications Conference 2005 (GLOBECOM'2005), vol. 1, pp. 157-161

[11] Q. Ren and Q. Liang, "A VSM-based and quality-aware query processing protocol for wireless sensor networks", in Proc. of IEEE Wireless Communications and Networking Conference 2007 (WCNC 2007)

[12] Q. Ren and Q. Liang, "A contention-based energy-efficient MAC protocol for wireless sensor networks", in Proc. of IEEE Wireless Communications and Networking Conference 2006 (WCNC 2006)

[13] Q. Ren and Q. Liang, "A quality-guaranteed and energy-efficient query processing algorithm for sensor networks", in Proc. of IEEE Wireless Communications and Networking Conference 2006 (WCNC 2006)

[14] X. Xia and Q. Ren and Q. Liang, "Cross-layer design for mobile ad hoc networks: energy, throughput and delay-aware approach", in Proc. of IEEE Wireless Communications and Networking Conference 2006 (WCNC 2006)

[15] Q. Liang and Q. Ren, "Energy and mobility aware geographical multi-path routing for wireless sensor networks", in Proc. of IEEE Wireless Communications and Networking Conference 2005 (WCNC 2005), vol. 3, pp. 1867-1871

[16] Q. Ren and Q. Liang, "Asynchronous energy-efficient MAC protocols for wireless sensor", in Proc. of IEEE Military Communications Conference 2005 (MILCOM2005)

[17] X. Xia, Q. Liang and Q. Ren, "Bottom-up cross-layer optimization for mobile ad hoc networks", in Proc. of IEEE Military Communications Conference 2005 (MILCOM2005)

[18] Q. Ren and Q. Liang, "Performance Analysis of Energy Detection for Cognitive Radio Wireless Networks"', in Proc. of International Conference on Wireless Algorithms, Systems and Applications (WASA'07)

[19] Q. Ren and Q. Liang, "Fuzzy logic-optimized secure media access control (FSMAC) protocol for wireless sensor networks", in Proc. of the 2005 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety (CIHSPS 2005), pp. 37-43

[20] Q. Ren and Q. Liang, "Secure media access control (MAC) in wireless sensor networks: intrusion detections and countermeasures", in Proc. of the 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2004), vol. 4, pp. 3025-3029

[21] Q. Ren, Q. Liang, X. Cheng, "A MAC Protocol for Underwater Acoustic Sensor Networks", submitted to IEEE International Conference on Communications 2007 (ICC2007)

REFERENCES

[1] M. Win and R. Scholtz, "Ultra-wide bandwidth time-hopping spread-spectrum impulse radio for wireless multiple-access communications," *IEEE Trans. Commun.*, vol. 48, no. 4, pp. 679–691, Dec. 2000.

[2] D. Culler, D. Estin, and M. Srivastava, "Guest editors' introduction: overview of sensor networks," *IEEE Comput.*, vol. 37, no. 8, pp. 41–49, 2004.

[3] C. K. Toh, *Ad Hoc Mobile Wireless Networks: Protocols and Systems.* Prentice-Hall, 2002.

[4] S. Bregni, *Synchronization of Digital Telecommunications Networks.* NY: Wiley, 2002.

[5] F. Cristian, "Probabilistic clock synchronization," *Distributed Comput.*, vol. 3, no. 3, pp. 146–158, 1989.

[6] R. Gusell and S. Zatti, "The accuracy of clock synchronization achieved by tempo in berkeley unix 4.3 bsd," *IEEE Trans. Software Eng.*, vol. 15, no. 7, pp. 847–853, July 1989.

[7] T. K. Srikanth and S. Toueg, "Optimal clock synchronization," *J. of the ACM.*, vol. 34, no. 3, pp. 626–645, July 1987.

[8] S. Weilian and A. F. Ian, "Time-diffusion synchronization protocol for wireless sensor networks," *IEEE/ACM Trans. Networking*, vol. 13, no. 2, pp. 384–397, Apr. 2005.

[9] J. E. Elson, "Time synchronization in wireless sensor networks," in *Dissertation of Computer Science Dept., Univ. of California Los Angeles*, 2003.

[10] F. Zhao and L. Guibas, *Wireless Sensor Networks: An Information Processing Approach.* Morgan Kaufmann, 2004.

[11] M. Stemm and R. H. Katz, "Measuring and reducing energy consumption of network modules in hand-held devices," *IEICE Trans. Commun.*, vol. E80-B, no. 8, pp. 1125–1131, Aug. 1997.

[12] C. Ma, M. Ma, and Y. Yang, "Data-centric energy efficient scheduling for densely deployed sensor networks," in *Proc. IEEE International Conference on Communications*, June 2003, pp. 3652–3656.

[13] E. Shi and A. Perrig, "Designing secure sensor networks," *IEEE Wireless Commun. Mag.*, vol. 11, no. 6, pp. 38–43, 2004.

[14] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *IEEE Trans. Comput.*, vol. 35, no. 10, pp. 54–62, 2002.

[15] D. Xiao, C. Chen, and G. Chen, "Intrusion detection based security architecture for wireless sensor networks," in *Proc. IEEE International Symposium on Communications and Information Technology 2005 (ISCIT2005)*, Oct. 2005, pp. 1412–1415.

[16] S. Banerjee, C. Grosan, and A. Abraham, "Ideas: intrusion detection based on emotional ants for sensors," in *Proc. 5th International Conference on Intelligent Systems Design and Applications 2005 (ISDA '05)*, Sept. 2005, pp. 344–349.

[17] Y. Zhen and G. Yong, "A robust group-based key management scheme for wireless sensor networks," in *Proc. IEEE Wireless Communications and Networking Conference, 2005 (WCNC2005)*, Mar. 2005, pp. 1915–1920.

[18] R. Du, H. Tu, and W. Song, "An efficient key management scheme for secure sensor networks," in *Proc. 6th International Conference on Parallel and Distributed Computing, Applications and Technologies 2005 (PDCAT2005)*, Dec. 2005, pp. 279–283.

[19] N. Engelbrecht and W. T. Penzhorn, "Secure authentication protocols used for low power wireless sensor networks," in *Proc. IEEE International Symposium on Industrial Electronics 2005 (ISIE 2005)*, June 2005, pp. 1777–1782.

[20] P. Taejoon and K. G. Shin, "Soft tamper-proofing via program integrity verification in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 4, no. 3, pp. 297–309, 2005.

[21] P. K. Dutta, A. K. Arora, and S. B. Bibyk, "Towards radar-enabled sensor networks," in *Proc. 5th International Conference on Information Processing in Sensor Networks, 2006 (IPSN2006)*, Apr. 2006, pp. 467–474.

[22] R. J. Fontana, A. Ameti, E. Richley, L. Beard, and D. Guy, "Recent advances in ultra wideband communications systems," in *Proc. 2002 IEEE Conference on Ultra Wideband Systems and Technologies*, May 2002, pp. 129–133.

[23] R. J. Fontana, "Recent system applications of short-pulse ultra-wideband (uwb) technology," *IEEE Trans. Microwave Theory Tech.*, vol. 52, no. 9, pp. 2087–2104, 2004.

[24] "First report and order in the matter of revision of part 15 of the commissions rules regarding ultra-wideband transmission systems, federal communications commission (fcc 02-48)," in *Std., ET Docket 98-153*, Apr. 2002.

[25] J. H. Reed, *An Introduction to Ultra Wideband Communication Systems.* NJ: Prentice-Hall, 2005.

[26] I. E. Teletar and D. N. C. Tse, "Capacity and mutual information of wideband multipath fading channels," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1384–1400, July 2000.

[27] S. Verdu, "Spectral efficiency in the wideband regime," *IEEE Trans. Commun.*, vol. 48, no. 6, pp. 1319–1343, June 2002.

[28] A. Manjeshwar, Q. Zeng, and D. P. Agrawal, "An analytical model for information retrieval in wireless sensor networks using enhanced apteen protocol," *IEEE Trans. Parallel Distrib. Syst.*, vol. 13, no. 12, pp. 1290–1302, Dec. 2002.

[29] S. Singh and C. Raghavendra, "Pamas: power aware multi-access protocol with signaling for ad hoc networks," in *Proc. ACM SIGCOMM Computer Communication Review*, July 1998, pp. 5–26.

[30] P802.11, "Ieee standard for wireless lan medium access control (mac) and physical layer (phy) specifications," Nov. 1997.

[31] J. Hill and D. Culler, "Mica: a wireless platform for deeply embedded networks," in *Proc. IEEE MICRO*, Nov. 2002, pp. 12–24.

[32] V. Rajendran, K. Obraczka, J. Juslin, and O. Koukousoula, "Energy-efficient, collision-free medium access control for wireless sensor networks," in *Proc. ACM SenSys'03*, Nov. 2003, pp. 181–192.

[33] L. F. W. van Hoesel, T. Nieberg, H. J. Kip, and P. J. M. Havinga, "Advantages of a tdma based, energy-efficient, self-organizing mac protocol for wsns," in *Proc. IEEE Vehicular Technology Society 2004 (VCT'04)*, May 2004, pp. 1598–1602.

[34] J. Li and G. Y. Lazarou, "A bit-map-assisted energy-efficient mac scheme for wireless sensor networks," in *Proc. 4th Information Processing in Sensor Networks (IPSN'04)*, Apr. 2004, pp. 55–60.

[35] S. Biaz and Y. D. Barowski, "Gangs: an energy efficient mac protocol for sensor networks," in *Proc. 42nd Annual Southeast Regional Conference (ACMSE'04)*, Apr. 2004, pp. 82–87.

[36] W. Ye, J. Herdemann, and D. Estin, "An energy-efficient mac protocol for wireless sensor networks," in *Proc. IEEE INFOCOM'02*, June 2002, pp. 1567–1576.

[37] T. V. Dam and K. Langendoen, "Adaptive energy-efficient mac protocol for wireless sensor networks," in *Proc. ACM SenSys'03*, Nov. 2003, pp. 171–180.

[38] J. Polastre, J. Hillm, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. ACM SenSys'04*, Nov. 2004, pp. 95–107.

[39] S. Jayashree, B. S. Manoj, and C. S. R. Murthy, "On using battery state for medium access control in ad hoc wireless networks," in *Proc. 10th Annual International Conference on Mobile Computing and Networking (MobiCom'04)*, Sept. 2004, pp. 360–373.

[40] E. Jung and N. H. Vaidya, "A power control mac protocol for ad hoc networks," in *Proc. 8th Annual International Conference on Mobile Computing and Networking (MobiCom'02)*, Sept. 2002, pp. 36–47.

[41] J. Walker, "Unsafe at any key size: an analysis of the wep encapsulation," in *IEEE 802.11 doc 00-362*, Oct. 2000.

[42] R. L. Rivest, "The rc4 encryption algorithm," in *Technical Report of RSA Data Security, Inc.*, Mar. 1992.

[43] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "Spins: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[44] H. Chan and A. Perrig, "Security and privacy in sensor networks," *Computer*, vol. 36, no. 10, pp. 103–105, 2003.

[45] M. K. Chirumanilla and B. Ramamurthy, "Agent based intrusion detection and response system for wireless lans," in *Proc. IEEE International Conference on Communications 2003 (ICC '03)*, May 2003, pp. 492–496.

[46] J. E. Dickerson, J. Juslin, O. Koukousoula, and J. A. Dickers, "Fuzzy intrusion detection," in *Proc. Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, July 2001, pp. 2165–2170.

[47] O. Kachirski and R. Guha, "Effective intrusion detection using multiple sensors in wireless ad hoc networks," in *Proc. 36th Hawaii International Conference on System Sciences (HICSS'03)*, Jan. 2003.

[48] H. R. Turtle and W. B. Corft, "Uncertainty in information retrieval systems," in *Proc. 2nd Workshop on Uncertainty Management and Information Systems: From Needs to Solutions*, Sept. 1992, pp. 93–111.

[49] A. Motro, "Sources of uncertainty in information systems," in *Proc. 2nd Workshop on Uncertainty Management and Information Systems: From Needs to Solutions*, Sept. 1992, pp. 1–18.

[50] G. Trajcevski, K. Hinrichs, and S. Chamberlain, "Managing uncertainty in moving objects databases," *JACM Trans. on Database Syst.*, vol. 29, no. 3, pp. 463–507, 2004.

[51] H. K. Park, J. H. Son, and M. H. Kim, "Dynamic histograms for future spatiotemporal range predicates," *Information Sciences*, vol. 172, no. 1-2, pp. 195–214, 2005.

[52] E. Straszecka, "Combining uncertainty and imprecision in models of medical diagnosis," *Information Sciences*, vol. 178, no. 20, pp. 3026–3059, 2006.

[53] H. Prade and C. Testemal, "Generalizing database relational algebra for the treatment of incomplete or uncertain information and vague queries," *Information Sciences*, vol. 34, no. 2, pp. 115–143, 1984.

[54] R. Cavallo and M. Pittarelli, "The theory of probabilistic databases," in *Proc. 13th Very Large Database Conference*, Sept. 1987, pp. 71–81.

[55] M. Pittarelli, "An algebra for probabilistic databases," *IEEE Trans. Knowledge Data Eng.*, vol. 6, no. 2, pp. 293–303, 1994.

[56] ——, "Probabilistic databases for decision analysis," *International J. Intell. Sys.*, vol. 5, no. 4, pp. 209–236, 1990.

[57] R. Biswas, S. Thrun, and L. J. Guibas, "A probabilistic approach to inference with limited information in sensor networks," in *Proc. 3rd International Symposium on Information Processing in Sensor Networks (IPSN'04)*, Apr. 2004, pp. 269–276.

[58] R. Ross, V. S. Subrahmanian, and J. Grant, "Aggregate operators in probabilistic databases," *Journal of the ACM*, vol. 52, no. 1, pp. 54–101, 2005.

[59] N. Dalvi and D. Suciu, "Efficient query evaluation on probabilistic databases," *International J. of Very Large Data Bases*, 2006.

[60] P. Bodorik, J. S. Riordon, and J. S. Pyra, "Deciding to correct distributed query processing," *IEEE Trans. Knowledge Data Eng.*, vol. 5, no. 3, pp. 253–265, 1992.

[61] A. Gounaris, N. W. Paton, A. A. A. Fernandes, and R. Sakellariou, "Self-monitoring query execution for adaptive query processing," *Data & Knowledge Engineering*, vol. 51, no. 3, pp. 325–348, 2004.

[62] C. Avin and C. Brito, "Efficient and robust query processing in dynamic environments using random walk techniques," in *Proc. 3rd International Symposium on Information Processing in Sensor Networks (IPSN'04)*, Apr. 2004, pp. 277–286.

[63] Y. D. Chung, "An indexing scheme for energy-efficient processing of content-based retrieval queries on a wireless data stream," *Information Sciences*, 2006.

[64] J. Kim, "Advanced structural joins using element distribution," *Information Sciences*, vol. 176, no. 22, pp. 3300–3331, 2006.

[65] B. J. Bonfils and P. Bonnet, "Adaptive and decentralized operator placement for in-network query processing," in *Proc. 2nd International Symposium on Information Processing in Sensor Networks (IPSN'03)*, Apr. 2003, pp. 47–62.

[66] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proc. 6th Annual International Conference on Mobile Computing and Networking (MobiCOM 2001)*, Aug. 2000, pp. 56–67.

[67] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "Tinydb: an acquisitional query processing system for sensor networks," *ACM Trans. Database Syst.*, vol. 30, no. 1, pp. 122–173, Mar. 2005.

[68] R. Cheng, S. Singh, and P. S, "U-dbms: a database system for managing constantly-evolving data," in *Proc. the 31st VLDB Conference, Trondheim*, Aug. 2005, pp. 1271–1274.

[69] Y. Yao and J. Gehrke, "Query processing in sensor networks," in *Proc. 1st Biennial Conference on Innovative Data Systems Research*, Jan. 2003.

[70] "Ultra wideband concepts for ad-hoc networks (ucan)," in *[On Line] http://www.prorec-projekte.de/ucan/.*

[71] I. 802.15.3-2003, "Ieee standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks specific requirements part 15.3: wireless medium access control (mac) and physical layer (phy) specifications for high rate wireless personal area networks (wpans)," Sept. 2003.

[72] J. Ding, L. Zhao, S. R. Medidi, and K. M. Sivalingam, "Mac protocols for ultra-wide-band (uwb) wireless networks: impact of channel acquisition time," in *Proc. Int. Conference on Internet, Performance and Control of Networks (SPIE IT-COM)*, July 2002, pp. 97–106.

[73] N. J. August and D. S. Ha, "An efficient uwb radio architecture for busy signal mac protocols," in *Proc. IEEE Conference on Ultra Wideband Systems and Technologies*, May 2004, pp. 325–334.

[74] N. J. August, H. Lee, and D. S. Ha, "Pulse sense: a method to detect a busy medium in pulse-based ultra wideband (uwb) networks," in *Proc. IEEE Conference on Ultra Wideband Systems and Technologies*, May 2004, pp. 366–370.

[75] F. Cuomo, C. Martello, A. Baiocchi, and F. Capriotti, "Radio resource sharing for ad hoc networking with uwb," *IEEE J. Select. Areas Commun.*, vol. 20, no. 9, pp. 1722–1732, Dec. 2002.

[76] M.-G. D. Benedetto, L. D. Nardis, M. Junk, and G. Giancola, "$(uwb)^2$: Uncoordinated, wireless, baseborn, medium access control for uwb communication networks," *J. of Mobile Networks and Applicat. Special Issue on WLAN Optimization at the MAC and Network Levels*, vol. 10, no. 5, pp. 663–674, 2005.

[77] A. D. Wood and J. A. Stankovic, "Defeating distributed denial of service attacks," *IT Professional*, vol. 2, no. 4, pp. 36–42, 2000.

[78] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory.* NJ: Prentice-Hall, 1993.

[79] D. Brennan, "Linear diversity combining techniques," in *Proc. the Institute of Radio Engineers (IRE)*, June 1959, pp. 1075–1102.

[80] C.-T. Lin and C. S. G. Lee, *Neural Fuzzy Systems.* NJ: Prentice-Hall, 1996.

[81] J. M. Mendel, *Lessons in Estimation Theory for Signal Processing Communications, and Control.* NJ: Prentice-Hall, 1995.

[82] V. P. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff, "A minimum cost heterogeneous sensor network with a lifetime constraint," *IEEE Trans. Mobile Comput.*, vol. 4, no. 1, pp. 4–15, Jan. 2005.

[83] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

[84] J. Chou, D. Petrovice, and K. Ramachandran, "A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks," in *Proc. IEEE INFOCOM 2003*, Mar. 2003, pp. 1054–1062.

[85] M. L. Chebolu, V. K. Veeramachaneni, S. K. Jayaweera, and K. R. Namuduri, "An improved adaptive signal processing approach to reduce energy consumption in sensor networks," in *Proc. 38th Annual Conference on Information Science and System (CISS 04)*, Mar. 2004.

[86] S. Balasubramanian, I. Elangovan, S. K. Jayaweera, and K. R. Namuduri, "Distributed and collaborative tracking for energy-constrained ad-hoc wireless sensor networks," in *Proc. IEEE Wireless Communications and Networking Conference 2004 (WCNC 2004)*, Mar. 2004, pp. 1732–1737.

[87] S. K. Jayaweera, "An energy-efficient virtual mimo communications architecture based on v-blast processing for distributed wireless sensor networks," in *Proc. IEEE SENCOM 2004*, Oct. 2004, pp. 299–308.

[88] L. H. Bao and J. J. Garcia-Luna-Aceves, "Hybrid channel access scheduling in ad hoc networks," in *Proc. 10th IEEE International Conference on Network Protocols (ICNP'02)*, Nov. 2002, pp. 46–57.

[89] D. Bersekas and R. Gallager, *Data Networks*. Prentice-Hall, 1987.

[90] M. M. Carvalho and J. J. Garcia-Luna-Aceves, "Delay analysis of ieee 802.11 in single-hop networks," in *Proc. 11th IEEE International Conference on Network Protocols (ICNP'03)*, Nov. 2003, pp. 146–155.

[91] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," *IEEE J. Select. Areas Commun.*, vol. 18, no. 3, Mar. 2000.

[92] J. R. Groff, P. N. Weinber, and L. Wald, *SQL: The Complete Reference*. CA: McGraw-Hill/Osborne, 2002.

[93] B. S. Lee and R. Musick, "Meshsql: the query language for simulation mesh data," *Information Sciences*, vol. 159, no. 3-4, pp. 177–202, 2004.

[94] P. P. Bonissone and R. M. Tong, "Editorial: reasoning with uncertainty in expert systems," *IEEE Trans. Man-Mach. Syst.*, vol. 22, no. 3, pp. 241–250, 1985.

[95] P. Bosc and H. Prade, "An introduction to fuzzy set and possibility theory based approaches to the treatment of uncertainty and imprecision in database management systems," in *Proc. the Workshop on Uncertainty Management in Information Systems: From Needs to Solutions*, Dec. 1996, pp. 285–324.

[96] J. R. Schott, *Remote Sensing: The Image Chain Approach.* NY: Oxford University Press, 1997.

[97] "Products description," in *[On Line] WWW: http://xbow.com.*

[98] Q. Ren and Q. Liang, "An energy-efficient mac protocol for wireless sensor networks," in *Proc. IEEE Global Telecommunications Conference 2005 (GLOBECOM'2005)*, Dec. 2005, pp. 157–161.

[99] R. Gandhi, S. Khuller, and A. Srinivasan, "Approximation algorithm for partial covering problems," *J. of Algor.*, vol. 53, no. 1, pp. 55–84, Oct. 2004.

[100] A. Papoulis, *Probability, Random Variables, and Stochastic Processes.* NY: McGraw-Hill, 2002.

[101] P. Olofsson, *Probability, Statistics, and Stochastic Processes.* NJ: Wiley, 2005.

[102] J. D. Wilfrid and J. M. J. Frank, *Introduction to Statistical Analysis.* NY: McGraw-Hill, 1983.

[103] Q. Liang and Q. Ren, "Energy and mobility aware geographical multipath routing for wireless sensor networks," in *Proc. IEEE Wireless Communications and Networking Conference 2005 (WCNC 2005)*, Mar. 2005, pp. 1867–1871.

[104] R. Jain, A. Puri, and R. Sengupta, "Geographical routing using partial information for wireless sensor networks," in *Proc. IEEE Personal Communications*, Feb. 2001, pp. 48–57.

[105] Q. Ren and Q. Liang, "A contention-based energy-efficient mac protocol for wireless sensor networks," in *Proc. IEEE Wireless Communications and Networking Conference 2006 (WCNC2006)*, Apr. 2006.

[106] L. Zhao, X. Hong, and Q. Liang, "Energy-efficient self-organization for wireless sensor networks: a fully distributed approach," in *Proc. IEEE Global Telecommunications Conference 2004 (Globecom 2004)*, Nov. 2004, pp. 2726–2732.

[107] M. L. Welborn, "System considerations for ultra wideband wireless networks," in *Proc. IEEE Radio and Wireless Conference*, Aug. 2001, pp. 628–634.

[108] I. Guvenc and H. Arslan, "On the modulation options for uwb systems," in *Proc. IEEE Military Communications Conference 2003 (MILCOM'03)*, Oct. 2003, pp. 892–897.

[109] A. A. M. Saleh and R. A. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE J. Select. Areas Commun.*, vol. 5, no. 2, pp. 128–137, Feb. 1987.

[110] I. 802.15.SG3a, "Channel modeling sub-committee report final."

[111] M. D. Benedetto and G. Giancola, *Understanding Ultra Wide Band Radio Fundamentals*. NJ: Prentice-Hall, 2004.

[112] K. Park and W. Willinger, *Self-Similar Network Traffic and Performance Evaluation*. Wiley, 2000.

[113] G. Anandalingam and S. Raghavan, *Telecommunications Network Design and Management*. Springer, 2002.

[114] D. Chow and C. T. Yu, "On the construction of feedback queries," *J. of the ACM.*, vol. 29, no. 1, pp. 127–151, Jan. 1982.

[115] S. K. M. Wong, W. Ziarko, V. V. Raghavan, and P. C. N. Wong, "On modeling of information retrieval concepts in vector spaces," *ACM Trans. Database Syst.*, vol. 12, no. 2, pp. 299–321, June 1987.

[116] L. Gravano, H. Garcia-Molina, and A. Tomasic, "Gloss: Text-source discovery over the internet," *ACM Trans. Database Syst.*, vol. 24, no. 2, pp. 229–264, June 1999.

[117] D. Grossman, O. Frieder, D. Holmes, and D. Roberts, "Integrating structured data and text: a relational approach," *J. of the American Soc. for Inform. Sci.*, vol. 48, no. 2, pp. 122–132, Feb. 1997.

[118] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Prentice-Hall, 2001.

[119] E. H. Mamdani, "Applications of fuzzy logic to approximate reasoning using linguistic synthesis," *IEEE Trans. Comput.*, vol. 26, no. 12, pp. 1182–1191, 1977.

[120] C. V. Altrock, "Fuzzy logic design: methodology, standards, and tools," in *Proc. Electronic Engineering Times*, July 1996.

BIOGRAPHICAL STATEMENT

Qingchun Ren received her B.S. and M.S. degrees from University of Electrical Science and Technology of China, in 1997 and 2003 respectively, both in Electrical Engineering. She is working toward the Ph.D. degree in Electrical Engineering at University of Texas at Arlington.

Since Aug. 2003, she has been a Research Assistant in the Wireless Communication Network Group, University of Texas at Arlington. Her research interests are in sensor networks (energy efficiency, cross layer design, optimal sensor deployment, etc.), fuzzy logic systems, query processing for sensor database systems, cognitive radio systems and underwater acoustic sensor networks.