11-27-2007

IEEE 802.15.4 MAC Protocol Study and Improvement

Liang Cheng

IEEE 802.15.4 MAC PROTOCOL STUDY AND IMPROVEMENT

by

LIANG CHENG

Under the Direction of Prof. Anu G. Bourgeois

ABSTRACT

IEEE 802.15.4 is a standard used for low rate personal area networks (PANs). It offers device level connectivity in applications with limited power and relaxed throughput requirements. Devices with IEEE 802.15.4 technology can be used in many potential applications, such as home networking, industry/environments monitoring, healthcare equipments, etc, due to its extremely low power features.

Although the superframe beacons play the key role in synchronizing channel access in IEEE 802.15.4, they are sources for energy inefficiency. This research focuses on exploring how to optimize the beacons, and designing novel schemes to distribute the information that are supposed to be delivered to a subset of PAN devices. In this work, an acknowledgement based scheme is proposed to reduce the energy consumption in the distribution of guaranteed time slot (GTS) descriptors. Based on the observation that the superframe beacon frame has global impact on all PAN devices, an energy-efficient channel reservation scheme is presented to deliver the information (GTS descriptors and pending addresses). In addition, the problem of channel underutilization is studied in the contention free period. To address the problem, a new GTS allocation scheme is proposed to improve the bandwidth utilization.

INDEX WORDS: IEEE 802.15.4, energy-efficiency, superframe beacon, GTS

IEEE 802.15.4 MAC PROTOCOL STUDY AND IMPROVEMENT

By

LIANG CHENG

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2007

IEEE 802.15.4 MAC PROTOCOL STUDY AND IMPROVEMENT

By

LIANG CHENG

Major Professor: Dr. Anu G. Bourgeois
Committee: Dr. Yi Pan
Dr. Alex Zelikovsky
Dr. Johannes Hattingh

Electronic Version Approved:

Office of Graduate Studies
College of Arts and Sciences
Georgia State University
Dec 2007

I wish I had the Ring to wake up my mother-in-law

ACKNOWLEDGMENT

First of all, I would like to express my special and sincere appreciation to my advisor, Dr. Anu G. Bourgeois, who has been constant in her valuable guidance and encouragement in my Ph.D research. During the past four years, she has spent enormous time in answering my questions, offering supervision or suggestions on my research and polishing every paper submitted. In addition, generous financial support from her helped me survive. She is very considerate and supportive. I had very fortunately worked as her teaching assistant a couple of times. She always knows when to shift the workload to make you more comfortable. She is open-minded and willing to listen to other's ideas and opinions. Without her help, the dissertation could not be completed.

I would also like to extend my appreciation to Dr. Yi Pan, Dr. Alex Zelikovsky, and Dr. Johannes Hattingh for reviewing this dissertation and serving on my Ph.D committee. They provided valuable suggestions in my dissertation proposal, which help to improve the quality of the research. During the review, they sacrificed their precious time to go through the dissertation thoroughly, and provided many constructive suggestions and comments.

Computer science department has provided an ideal working environment during the years' study under the Chair, Prof. Yi Pan. I specially thank the Professors: Prof. Yi Pan, Prof. Alex, Prof. Raj Sunderrman, Prof. Sushil Prasad, Prof. Xiaolin Hu, Prof. King, Prof. Saeid Belkasim, and Prof. Preethy, who lead me to the field of Science from Engineering. Also I want to thank administrative and technical specialists for their great service and help: Tammie, Adrienne, Shaochieh, Celena and Venette. Many friends in our department deserve my credit for their support and encouragement. However, I could not list your name here due to space limit. Most of the research work is done in DiMoS lab in Yamacraw building (now called TSRB) and the manuscript is produced in LaTeX.

Finally, I want to thank my parents, Weiyuan Cheng and Kelan Sun. They raised me with the hearts and provided me the best education in those hard days. Their love and

encouragement enables me to brace challenges bravely in the past, now and the future. I also want to thank my elder brother, Xianmin Cheng, whose guidance and encouragement are vital at every turning point of my life. Without him, I could not be here, and I could not achieve what I have achieved. Lastly but absolutely not the least, I am deeply thankful to my wife, Xin Zhang, for her love, support as well as precious discussions about my research. She is always the first reader of my papers and provides most valuable suggestions/comments. Without her love and strong support, I could not focus on my Ph.D study through these years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACK	Acknowledgement
BE	Backoff Exponent
BO	Beacon Order
CAP	Contention Access Period
CCA	Clear Channel Assessment
CFP	Contention Free Period
CSMA-CA	Carrier Sense Multiple Access - Collision Avoidance
CTS	Clear-to-Send
CW	Contention Window
ED	Energy Detection
FFD	Full Function Device
GTS	Guaranteed Time Slot
IFS	Inter Frame Space
LIFS	Long Inter Frame Space
LQI	Link Quality Indication
MAC	Medium Access Control
NB	Number of Backoff
PAN	Personal Area Network
PHY	Physical layer
PDU	Protocol Data Unit
QoS	Quality of Service
RFD	Reduced Function Device
RTS	Request-to-Send
SDL	Specification and Description Language
SIFS	Short Inter Frame Space

SO	Superframe Order
TDMA	Time Division Multiple Access
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Network

CHAPTER 1

INTRODUCTION

Advances in micro–sensing technology, as well as numerous novel applications, have led to a substantial volume of research on wireless sensor networks (WSNs) [1]. Compared to traditional wireless ad hoc networks, WSNs have several distinct characteristics: limited or no mobility, extreme power constraints, very limited capability of computation, relaxed throughput and latency requirements, and fierce environments for deployment. Nodes in WSNs are typically powered by batteries and deployed where replacing the battery is not feasible or at uneconomical costs. Since a WSN normally consists of a large of distributed nodes, which coordinate to perform a common task, the problem of energy waste can cause node failure and eventually make the whole network unusable because of network partition. Hence the critical issue in WSNs is to reduce energy consumption and prolong the lifetime of these nodes. In a sensor node, with advance of microprocessors, the energy consumption of computing chips is falling sharply while the energy consumed by radio remains the same. Thus, the wireless interface is the primary consumer of energy in a WSN node. How to design an energy efficient medium access control (MAC) protocol attracts many research efforts in the sensor network community. Over the years, the very active research on communication protocols used in WSNs have resulted in more than 30 variations of MAC protocols [2]. Due to the use of different protocols and radios, nodes have difficulties in communicating with each other. The problem was not solved until IEEE 802.15.4 standard appeared.

The IEEE 802.15.4 standard (referred to as 802.15.4 hereinafter) defines both physical (PHY) and MAC specifications [3]. It targets a wide variety of applications that require simple short-range wireless communications with limited power and relaxed throughput needs. Among the applications are industrial and commercial control and monitoring,

home automation and networking, automotive sensing, health care and precision agriculture. Generally speaking, the applications can be placed in the following classifications:

- **Stick-on sensor:** These applications comprise wireless sensors that are powered by battery and are expected to function for months or years. The focus of these applications is for monitoring or remote diagnostics.

- **Virtual wire:** This refers to the set of monitoring and control applications that can only be enabled through wireless connectivity, in places where a wired communications link cannot be implemented, e.g. tire pressure monitoring, motor bearing diagnosis, and so on in the automotive industry.

- **Wireless hub:** These are applications in which a centralized wireless bridge is added over a wired network. A wireless hub acts as a gateway between a wired network and a wireless low rate wireless personal network.

- **Cable replacement:** These applications attempt to add value through the removal of wires in consumer electronic portable devices. The 802.15.4 technology offers a lower power and low cost solution. The applications of this category may use a continuous source of power or rechargeable battery.

The IEEE 802.15.4 offers device-level wireless connectivity at low cost. The "low cost" here means lower manufacturing cost, lower installation cost and lower maintenance cost. The MAC protocol of 802.15.4 supports both contention-based medium access (i.e. Carrier Sense Multiple Access Collision Avoidance (CSMA-CA)) and scheduled-based medium access (i.e. Time Division Multiple Access (TDMA)) simultaneously. It incorporates some energy efficient features to qualify it as a best choice of WSN MAC protocol.

In this research, we focus on 802.15.4 MAC protocols, especially on the prospect of energy efficiency. Before the discussion of improving the energy efficiency, we need to identify what are the main sources that cause inefficient use of energy in 802.15.4 MAC

protocol, and other MAC protocols used in WSNs. As summarized in [6] [7] [17], much energy is wasted due to the following several major sources:

- **Collision:** If two nodes transmit at the same time and interfere with each other's transmission, packets are corrupted. Follow-on retransmissions increase energy consumption.

- **Overhearing:** A node may pick up packets that are destined to other nodes since the radio channel is a shared medium.

- **Protocol overhead:** This includes overhead caused by MAC headers and signalling control packets. Application data are encapsulated into data frames by appending protocol headers. Signalling control packets do not contain application data. Sending and receiving headers and control packets consume energy. Overhead becomes significant when the traffic is light.

- **Idle listening:** A node has to keep its radio in receive mode at all times so that it does not miss any packets destined to it. However, if nothing is sensed, the node is idle for most of the time. Measurements have shown that idle listening consumes 50%-100% of the energy required for receiving.

However, as a part of a protocol, the procedure of exchange packets can also cause energy inefficiency. This kind of source for energy drain is often overlooked. It can be classified into the category *Protocol overhead*. In fact, this research particularly studies this problem in the context of 802.15.4. Through the study of energy efficiency in data transmission, beacons are identified as a large overhead. Thus, we propose a light beacon structure with unnecessary information stripped out. We also propose an innovated guaranteed time slot (GTS) descriptor (GTS allocation information) distribution mechanism that can reduce the energy consumption of a personal area network (PAN) effectively. Based on the observation of parameters carried within beacons destined to different groups of devices, an energy-efficient channel reservation scheme is proposed to disseminate pending

addresses and GTS descriptors.

The remainder of the dissertation is organized as follows. Chapter 2 presents an overview of 802.15.4 standard and related work. In Chapter 3, an IEEE 802.15.4 simulator based on GTNetS platform is presented. Chapter 4 examines the beacon structure and the procedure of transmitting *guaranteed time slot (GTS)* descriptors and pending addresses in the IEEE 802.15.4 standard. A one byte overhead caused by *GTS specification* incurs when no valid GTS information is included in beacon. The use of a reserved bit makes this field as an option, and it is included only when valid GTS descriptors are present. Also an acknowledgement (ACK) based scheme is proposed to reduce the number of times that GTS descriptors are included within beacons. In Chapter 5, an energy-efficient channel reservation scheme is proposed to deliver the information (descriptors and pending addresses). This could directly reduce the overall energy consumption of an entire PAN. Chapter 6 studies the GTS utilization problem. And a new GTS allocation is proposed to allow more PAN devices to share the limited number of GTSs and improve the channel utilization. Finally, Chapter 7 concludes the dissertation and outlines some future work.

CHAPTER 2

BACKGROUND AND RELATED WORK

Wireless communication has experienced exponential growth caused by the need for connectivity in recent years. The evolution starts from IEEE 802.11 Wireless Local Area Networks(WLAN), which was created as the wireless extension of the IEEE 802 wired local area network. The operating range of the IEEE 802.11b technology is about 100 meters, and data rate supported vary from 2 to 11 Mbps [8]. The developing trend goes to two directions from IEEE 802.11. One is toward larger networking range, higher data throughput and quality of service (QoS). It targets applications such as the Internet, e-mail, data file transfer and Internet Protocol Television (IPTV) in Wireless Metropolitan Area Networks (WMAN). The latest technology example in this category is WiMAX defined in IEEE 802.16 standard [9]. The other direction is toward smaller networking range and simple networks. It targets applications in Wireless Personal Area Network (WPAN). The WPANs are used to convey information over relatively short distances among the participant devices. The family standards of IEEE 802.15 are defined in this category. These standards are differentiated by data rate supported, battery drain and QoS. For example, IEEE 802.15.3 is suitable for multimedia applications that require very high QoS [15], while IEEE 802.15.1 and Bluetooth are designed for cable replacements for consumer electronic devices suitable for voice applications [16][14]. IEEE 802.15.4 is at the low end to serve applications not covered by other 802.15 technology.

In this chapter, we provide an overview of the 802.15.4 standard, followed by the discussion of related work.

2.1 IEEE 802.15.4 overview

IEEE 802.15.4 standard is designed for low rate wireless PANs [3] [19]. It includes the physical layer and medium access control layer specifications. The design of 802.15.4

keeps low power consumption in mind because it is often infeasible to replace the battery for devices in its targeted applications. The protocol is simple and easy to implement, while it is flexible enough to accommodate the needs of a wide variety of applications. The MAC protocol of 802.15.4 can operate in either beacon mode or beaconless mode. In the beaconless mode, it is a simple unslotted CSMA-CA protocol. Thus, this research focuses on beacon mode. The typical operating range of 802.15.4 is approximately 10 to 20 meters, and the raw data rate is 250kb/s in the 2.4GHz band. In this section we introduce the features of the MAC layer, followed by a brief overview of the physical layer.

Superframe structure

IEEE 802.15.4 supports low rate wireless personal area networks working in beacon mode by use of superframes. The structure of a superframe, bounded by beacon frames, is shown in Figure 1. The coordinator of the PAN defines the format of the superframe.

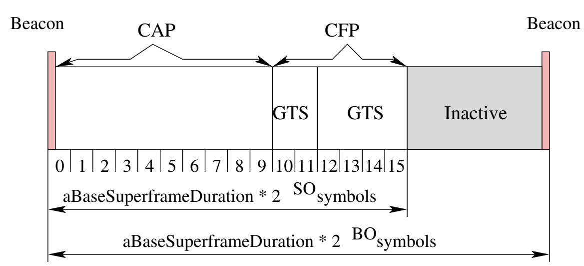


Figure 1. The typical structure of a superframe

The superframe beacons are transmitted periodically by the coordinator and are used to identify the PAN, synchronize the attached devices and describe the structure of superframes. The transmission frequency of beacons is determined by the *macBeaconOrder* ranged from 0 to 14. Within the superframe structure, the coordinator may choose to enter sleep mode during the inactive portion. The length of the inactive portion depends on the *macSuperframeOrder* (SO) and *macBeaconOrder* (BO). As shown in Figure 1, the active portion of the superframe is divided into 16 time slots. It may consist of two periods, namely contention access period (CAP) and contention free period (CFP). The CAP immediately follows the superframe beacon and completes before the CFP begins. All

frames except acknowledgement (ACK) frames are transmitted in this period using a slotted CSMA-CA mechanism. A transmission in the CAP shall be complete one inter frame space (IFS) period before the end of the CAP. If this is not possible, it defers its transmission until the CAP of the following superframe. In slotted CSMA-CA, the backoff period boundaries are aligned with the superframe slot boundaries of the PAN coordinator.

In order to support applications with particular bandwidth and latency requirements, 802.15.4 defines the second portion of the active period as CFP. Unlike CAP, channel access in CFP is based on reservations and is free of contention. So no transmissions within the CFP shall use a CSMA-CA mechanism. One part of the CFP is allocated for a particular device, which is denoted as a guaranteed time slot (GTS). The GTS direction, which is relative to the data flow from the device that owns the GTS, is specified as either transmit or receive. In other words, traffic flows from a device to the coordinator in a transmit GTS, while a device receives data from the coordinator in a receive GTS. A GTS may extend over one or more time slots. During the GTS, the wireless channel is dedicated exclusively for the communications between the particular device and the PAN coordinator. A maximum of seven GTSs are allowed to be allocated in a PAN. A device transmitting in the CFP shall ensure that its transmissions are complete one IFS period before the end of its GTS.

Inter frame space (IFS) time is the amount of time necessary to process the received packet by the physical layer (PHY). Transmitted frames shall be followed by an IFS period. The length of IFS depends on the size of the frame that has just been transmitted. Frames up to *aMaxSIFSFrameSize* in length shall be followed by a short IFS (SIFS) whereas frames of greater length shall be followed by a long IFS (LIFS).

The PANs that do not wish to use the superframe (referred to as a nonbeacon-enabled PAN) shall set both *macBeaconOrder* and *macSuperframeOrder* to 15. In this kind of network, a coordinator shall not transmit any beacons, all transmissions except the acknowledgement frame shall use unslotted CSMA-CA to access channel, and GTSs shall not be permitted.

CSMA-CA algorithm

If superframe structure is used in the PAN, the slotted CSMA-CA is used. The algorithm is implemented using units of time called backoff periods, which is equal to *aUnitBackoffPeriod* symbols. Each time a device wishes to transmit data or command frames during the CAP, it locates the boundary of the next backoff period.

Each device maintains three variables for CSMA-CA algorithm: NB, CW, and BE. NB is the number of times the CSMA-CA algorithm was required to backoff while attempting the current transmission. It is initialized to 0 before each new transmission. CW is the contention window length, which defines the number of backoff periods that need to be clear of activity before the transmission can start. It is initialized to 2 before each transmission attempt and reset to 2 each time the channel is assessed to be busy. BE is the backoff exponent, which is related to how many backoff periods a device shall wait before attempting to assess the channel.

In the slotted CSMA-CA, NB, CW, and BE are initialized and the boundary of the next backoff period is located. The MAC layer delays for a random number of backoff periods in the range of 0 to $2^{BE} - 1$, then requests the PHY to perform a clear channel assessment (CCA).

If the channel is assessed to be busy, the MAC layer increments both NB and BE by one, ensuring that BE shall be no more than *aMaxBE*. If the value of NB is less than or equal to *macMaxCSMABackoffs*, the CSMA-CA will start another around of delay for a random number of backoff periods. Otherwise, it declares channel access failure.

If the channel is assessed to be idle, the MAC sublayer ensures the contention window is expired before starting transmission. For this, CW is decremented by one first. If CW is not equal to 0, CCA is performed on backoff boundary. Otherwise, it starts transmission on the boundary of the next backoff period. Figure 2 illustrates the steps of the CSMA-CA algorithm.

Data Transfer model

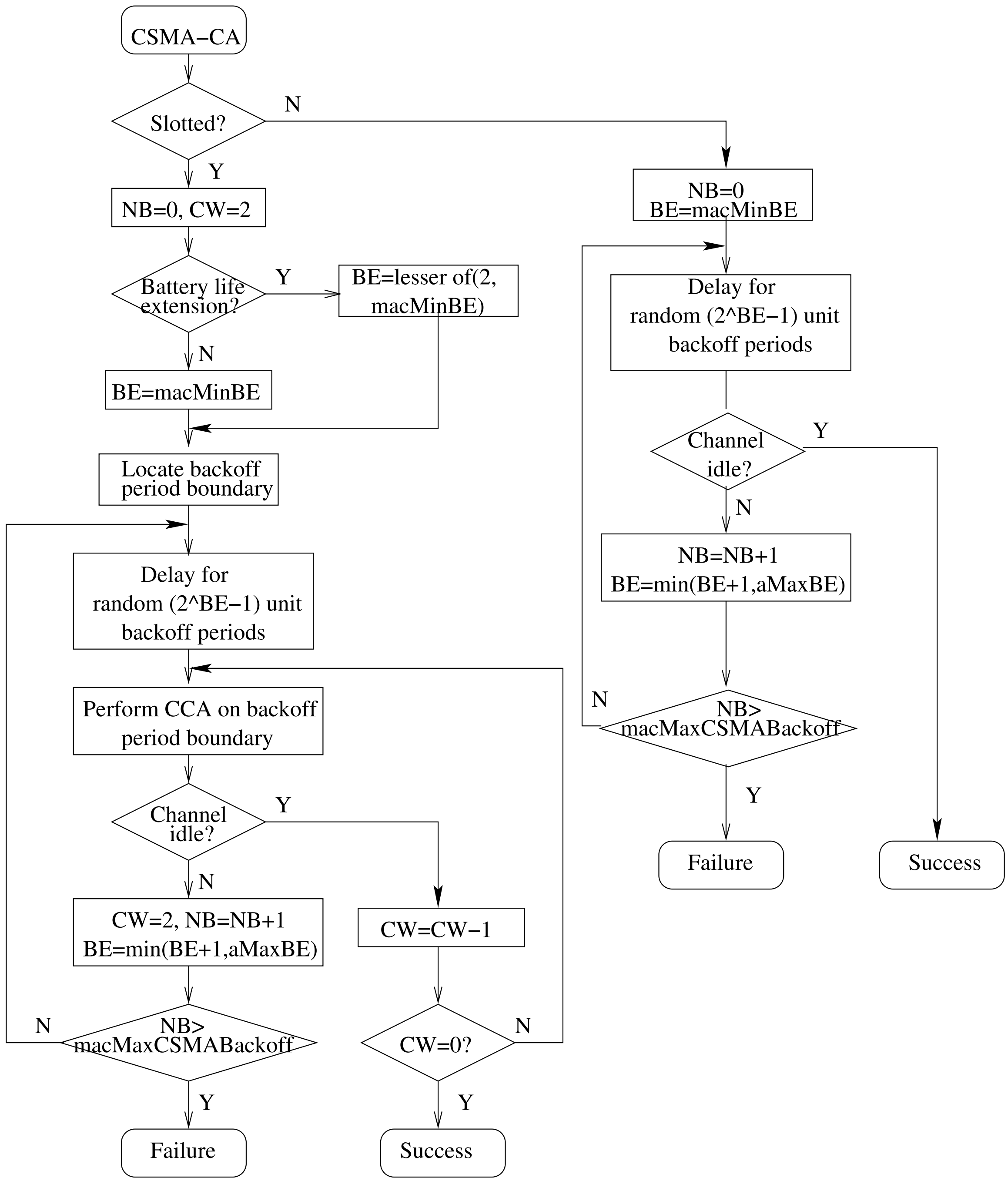


Figure 2. The CSMA-CA algorithm

Three types of data transfer models are supported in 802.15.4. The first one is the data transfer from a device to a PAN coordinator. In a beacon-enabled PAN, the device has to synchronize to the superframe structure. Then the device could transmit a data frame using slotted CSMA-CA at the appropriate time. The coordinator returns an acknowledgement when receiving the frame. The transaction is now completed. This type of transfer is called direct transfer model.

The second type is the data transfer from a coordinator to a device. In a beacon-enabled PAN, the PAN coordinator stores the data frame in a transaction list, and notifies the device through beacons. After the device decodes the beacon, it transmits a data request to the coordinator if there is a transaction pending for it. The coordinator acknowledges the reception of the request and the pending data frame is transmitted at the appropriate time using slotted CSMA-CA. The pending data frame is removed from the transaction list only after the acknowledgement for this frame is received or the data frame remains unhandled in the list over the maximal transaction persistence time. An example for this type of data transfer is the association response issued by a PAN coordinator. Because the data transfer is instigated from the destination (device) instead of the data source (coordinator), this type of transfer is called indirect transfer model.

The third type is the peer-to-peer data transfer between two devices in a peer-to-peer PAN. This type of transfer model allows devices to communicate with each other directly through unslotted CSMA-CA or some synchronization mechanisms.

Physical layer

The physical layer provides an interface between the MAC layer and the physical radio channel. It handles the data transmission request from the MAC layer and passes the incoming frame to the MAC layer. In order to provide these services efficiently, it supports activation and deactivation of the radio transceiver, energy detection (ED), link quality indication (LQI), and clear channel assessment (CCA). The feature of activation/deactivation of the radio transceiver is very important for energy conservation in battery-powered devices.

It allows devices to turn off their radio to avoid overhearing and idle listening. Receiver energy detection is intended for use by a network layer as part of channel selection algorithm. It is an estimate of the received signal power within the bandwidth of an 802.15.4 channel. Link quality indication measurement is a characterization of the strength and quality of a received packet. The measurement may be implemented using receiver energy detection, a signal-to-noise estimation or a combination of these methods. How to use the LQI is up to the network or application layers. Before the physical layer transmits a frame in the contention access period, it checks whether the channel is available. Thus CCA is performed by the physical layer using energy detection, carrier sense or a combination of both of them. In some sense, it is the front end of CSMA-CA in MAC layer.

2.2 Related work

In this section, the strategies for energy efficiency in other well-known MAC protocols are reviewed first. Then the related work specific to 802.15.4 are discussed.

Energy efficiency consideration in other MAC protocols

Due to extreme power constraints, either battery or ambient energy sources (sunlight, vibration, etc), the design of an energy-efficient MAC is the biggest challenge for designers of WSNs. How to reduce the energy consumption caused by *collisions, overhearing, protocol overhead*, and *idle-listening* is addressed in the design of WSN MAC protocols.

Historically, IEEE 802.11 protocol [8] forms the basis of all energy-efficient derivatives. The medium access control in the 802.11 standard is based on carrier sensing (CSMA) and collision detection through acknowledgement. And the hidden terminal problem in ad hoc networks is solved by the use of the collision avoidance handshake. The short Request-To-Send (RTS) and Clear-To-Send (CTS) control packets include a time field indicating the duration of the the upcoming data/acknowledgement frame. Other devices overhearing the control packet can defer their own transmission and switch off their radios during that period. Thus CSMA/CA effectively eliminates collisions and overhearing overhead.

However, the energy wasted by idle listening is not solved.

Low power listening was developed in [10]. It is a cross-layer design with CSMA at the MAC layer and a low-level carrier sense technique at the physical layer. The carrier sense effectively duty cycles the radio, i.e. turns it off repeatedly, without losing any incoming data. It works by prepending to the PHY header a preamble that is used to notify receivers of the upcoming transfer and allows them to adjust their circuitry to the current channel. If a receiver detects a preamble, it will continue listening until the message can be properly received. Otherwise, the radio is turned off until the next sample. This protocol does not have collision avoidance, thus the protocol overhead is saved. A similar concept was proposed in [11] and extended in [12]. IEEE 802.15.4 standard defines the similar carrier sense mechanism – CCA – in the physical layer.

The Sensor-MAC (S-MAC) introduces the concept of virtual clustering to allow nodes to synchronize on a common slot structure [1] [6]. To this end, nodes regularly broadcast SYNC packets at the beginning of a slot, so other nodes receiving these SYNC can adjust their clocks to compensate for drift and join the network. Then all nodes in such a network run the same schedule. Each slot contains three phases: *SYNC, active* and *sleep*. The length of the active period is fixed to 300 ms. During sleep periods nodes turn off their radio in order to save power. S-MAC addresses the idle-listening overhead, and it also includes collision avoidance (RTS/CTS handshake) and overhearing avoidance.

The Timeout-MAC protocol introduces an adaptive duty cycle to improve S-MAC [13]. In contrast to S-MAC, it operates with fixed length slots (615 ms) and uses a time-out mechanism to dynamically determine the end of the active period. The timeout value is set to span a small contention period and an RTS/CTS exchange. If a node does not detect any activity within the time-out interval, it can safely assume that no neighbor wants to communicate with it and goes to sleep. On the other hand, if the node engages or overhears a communication, it simply starts a new timeout after the communication finishes. To

[1]The concept of the slot is very similar to the superframe in 802.15.4

save energy, a node turns off its radio while waiting for other communications to finish (overhearing avoidance).

Unlike the above contention-based MAC protocols, schedule-based MAC protocols are collision free. Since nodes know exactly when to expect incoming data, it can avoid idle listening. This category of channel access is also called time division multiple access (TDMA). The classic example is Bluetooth [14]. In a PAN powered by Bluetooth, each device is allocated an up-link and a down-link time slot so that the device and the master can transmit packets alternatively. The disadvantage for Bluetooth is that it is complicated to support more than 7 devices in a PAN which usually requires the master parks some devices before activating others.

IEEE 802.15.4 MAC combines contention-based channel access with TDMA. The channel access in contention avoidance periods are through the CSMA-CA algorithm. Depending on application requirements, devices may apply for guaranteed time slots the in contention-free period. Therefore, 802.15.4 leaves much flexibility to applications, varied from latency insensitive applications to real-time applications. The related work on 802.15.4 is reviewed next.

Related work on 802.15.4

In the literature, a great number of existing 802.15.4 work focused on performance evaluation and analysis either through simulations or through experiments [20] [21] [22] [23] [24] [25] [55] [26] [56] [27]. Their evaluation metrics are throughput (or goodput), packet delivery ratio, energy consumption and packet delay. Among these works, beacon-tracking and non-beacon tracking modes were studied in [21]. One of their conclusions is that tracking beacons causes devices large energy cost. Compared with other application areas, medical applications and body area networks impose a strict requirement to 802.15.4 – reliability. Thus, in [22] [24], the performance analysis for medical applications and body area networks was performed.

Several other works studied 802.15.4 MAC from the prospect of power consumption.

In [28], an energy model of 802.15.4 was proposed, while the power consumption of slotted CSMA-CA was analyzed in [29]. In [47], based on their previous work, an energy model is derived to evaluate the total energy consumed to successfully transmit a packet in a peer-to-peer network. The model assumes programmable transmit power levels supported as in Chipcon CC2420, and considers the error rate because of path loss provided that the distance between the transmitter and the receiver is d. A peer-to-peer power control policy based on the energy efficiency was established to optimize transmit power.

The energy efficiency analysis of 802.15.4 was presented in [31]. It proposed an energy-aware radio activation scheme to minimize the energy consumption. Based on the energy breakdown, the authors suggested several possible ways, such as reducing the state transition time and transmitting larger packets, to improve the overall energy efficiency of 802.15.4 in wireless sensor networks. However, these approaches to improving energy efficiency either depend on the electrical characteristics of 802.15.4 RF transceivers, or are only applicable to certain application scenarios.

As mentioned in the Introduction, collisions can also cause energy waste. 802.15.4 effectively resolves the problem by employing clear channel assessment (CCA) with a random back-off. However, when the network load is heavy, the collision is likely to repeat once it happens because the nodes are adopting the same small contention window (CW) and back-off exponent (BE). In [30], a memorized backoff scheme was proposed to solve this problem. The main principle of this scheme is to record the CW value for the successful data delivery in the previous superframe, and use it to predict the initial value of CW for the current superframe. The scheme, thus, adjusts the parameter according to network load. It could effectively decrease the number of collisions, which results in the reduction of power consumption.

In monitoring applications, the parameter BO and SO may affect the performance of a WSN because it determines the duty cycle. A beacon order adaption algorithm was proposed in [32]. The algorithm requires the PAN coordinator to keep the number of packets

that each device generates in a superframe period. (Packets are generated only if changes occurred.) If the maximum of packets among all devices is below a threshold b_l, it means that this is a slower process and the network requires tuning. The coordinator increments BO. Similarly, the coordinator decrements BO if the maximum of packets among all devices exceeds a threshold b_u. The algorithm, thus, could adjust network duty cycles dynamically according to the monitoring process, which results in power saving.

All the research mentioned above pay attention to the slotted CSMA-CA channel access in 802.15.4. So far, only a few works studied GTSs. A GTS allocation analysis is performed using network calculus theory in [34]. They provided two models for the service curve of a GTS allocation and derived the delay bounds. Another work explores a mechanism of sharing GTS slots between multiple nodes in time-sensitive WSNs [35].

Unlike the existing work to improve energy efficiency by parameter tuning, this research tend to improve it by optimizing protocol overheads. In order to facilitate the ongoing research, a 802.15.4 network simulator is developed based on network simulator platform Georgia Tech Network Simulator (GTNetS). Based on the performance study of energy efficiency in the three supported transmission methods, an acknowledgement based energy-efficient scheme is proposed to replace the approach of distributing GTS descriptors in the 802.15.4 standard. After that, some special slots between CAP and CFP are reserved to transmit pending addresses and GTS descriptors. Lastly, the problem of GTS bandwidth utilization is studied, and a new GTS allocation scheme is proposed to improve the bandwidth utilization in the contention free period. Evaluation results show that the new GTS allocation scheme outperforms the standard scheme and another counterpart in [35].

CHAPTER 3

802.15.4 SIMULATOR IN GTNETS

In this chapter, we introduce a IEEE 802.15.4 simulator. The implementation detail of MAC layer and the physical layer is presented, followed by some experiments to demonstrate the correctness of the implementation.

3.1 Introduction

Simulation is one of the most important ways to conduct protocol research. Various limitations of the existing 802.15.4 simulation modules motivate us to develop a new simulation tool for the protocol evaluation. This chapter dedicates to our 802.15.4 simulator.

One of the earliest efforts to develop an 802.15.4 simulator is described in [21]. The simulator implemented some fundamental features of 802.15.4 such as superframe structure, carrier sense multiple access with collision avoidance (CSMA-CA) mechanism, and contention free period (CFP). However other important features, such as association, disassociation and channel scan, could not be simulated. Moreover, it only supports a star topology and also lacks some protocol details.

Another publicly available simulator presented in [20] implements the full set of primitives at the physical layer and MAC layer defined in the specification [3]. However, it does not provide an energy model, which is essential for performance evaluation of power sensitive protocols such as 802.15.4.

In [44], the effects of modeling detail on the accuracy or even correctness in wireless network simulations are demonstrated. For sensor nodes with low power consumption, it is critical to model the nodes and MAC protocol such as 802.15.4 in appropriate detail level. With that consideration, we model 802.15.4 elaborately so that simulations conducted with the simulation module could be as accurate as possible.

With some commercial 802.15.4 RF transceivers pushed into the market, some modifications are expected to incorporate the real experiment data in simulation modules in order to approximate the real protocol performance as much as possible. This is also included in the consideration of our developing of the 802.15.4 simulator.

Moreover, both of the above simulators are based on *NS2* [45]. Although *NS2* is an excellent and widely used network simulator, some deficiencies have been exposed over the time. Among them are the long learning curve due to using Tcl and C++, substantial memory requirements, and limited scalability of the network topology simulated.

All these limitations in existing 802.15.4 simulation tools motivate us to develop another simulator module. We choose *Georgia Tech Network Simulator* (GTNetS) [37] as our development platform. *GTNetS* is an extensible simulator that can support very large scale simulations. It is implemented entirely in C++, which facilitates testing and debugging. Moreover, it models networks very similar to real networks. There is a clear distinction between nodes, interfaces, links, and protocols. Thus it not only shortens the cycle of extending and learning, but also provides researchers an insight on how a real network works.

3.2 Simulator implementation

The implementation of 802.15.4 MAC and physical layer functionality takes the advantages of *GTNetS* and inherits the layered design architecture. On the other hand, our implementation of 802.15.4 simulation module closely follows the specification and description language (SDL) description in Annex D of [3]. Meanwhile, for discrepancies found between standard body and the SDL description, we choose to conform to the former. Some omissions in the SDL are also filled.

3.2.1 MAC layer implementation

Two kinds of device types are defined, a full-function device (FFD) and a reduced-function device (RFD) based on its complexity. An RFD is a kind of device with very limited

resources. Therefore, some of the protocol primitives are not supported. In contrast, an FFD is able to perform all the defined functionalities. Consequently, it could operate as a PAN coordinator, a coordinator, or a device. Sensor networks is a major application of 802.15.4. So a *sensorNode* class is defined in our simulator module. Each node contains an 802.15.4 interface. The structure of the interface has a field *opMode* to indicate whether the node is an FFD. The full set of the primitives in MAC and physical layers are implemented. Depending on the type of a device, some functionalities are not supported by checking the field. Users may configure a node as an RFD or FFD and designate the role of each node in simulations.

For ease of use and extension, in GTNetS the concept of protocol data unit (PDU) stack is introduced to construct a packet quickly. Thus, all packet headers of 802.15.4 modules are derived from class *PDU*. When a packet is generated and moved down to the MAC layer for further processing, the 802.15.4 packet header is pushed into the *PDU* stack. At the destination, the PDUs are popped out for processing at each layer.

A PAN consists of at least a coordinator and a number of devices. During the establishment procedure of the association between the coordinator and a device, a large amount of information, such as macState, numberOfBackoff and preMsg, are required to be saved. The use of *list* to store the information seems cumbersome. In a large network, a device may be a coordinator in one PAN and be a device in another PAN at the same time. Thus in the module, we maintain a main *macInstance* at each node. Each *macInstance* uniquely identifies the peer communication entity by *peerExtdAddr* and *peerShortAddr*. The PAN coordinator accepts the first association request with the main *macInstance*, and forks a new *macInstance* upon subsequent association requests from other devices. Each *macInstance* communicates with its peer MAC entity independently. The design avoids frequent switches of the communication contexts. In order to solve the problem of multiple *macInstance*s being stacked over an interface, the pointer to a *macInstance* must be passed to the physical layer whenever the operation initiator requires a notification. In addition, a simple

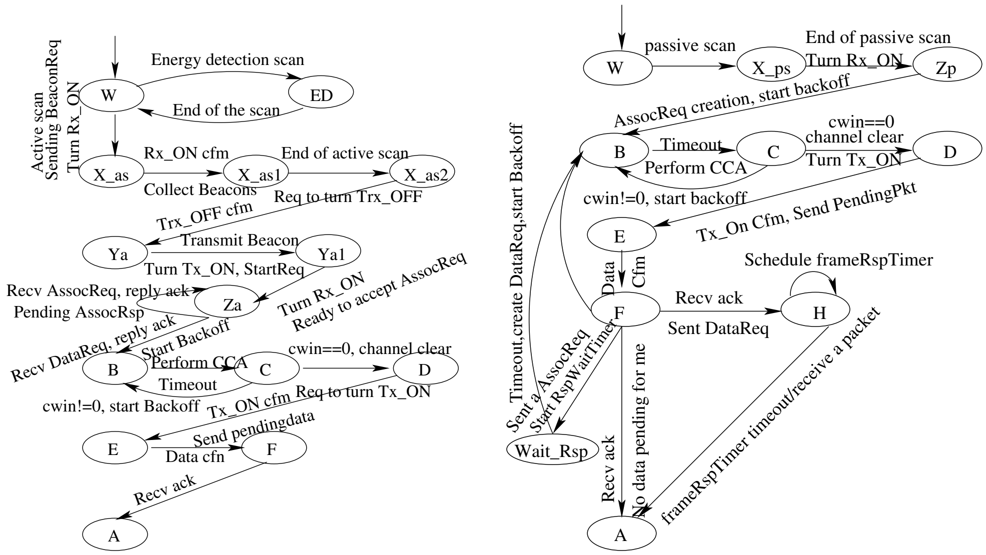
multiplex is implemented to pass incoming packets to an appropriate *macInstance*.

The mac *L2Proto802_15_4* class calls those methods defined in class *Interface_802_15_4* to perform ED, CCA, and data transmission. Also, it could obtain the physical layer attributes, parameters, and state by issuing corresponding requests directly. In order to support the notifications from the physical layer, the method *Notify* is defined in class *L2Proto802_15_4*. The notifications include confirmations such as set_trx_state_cfm, data_cfm, and cca_cfm, etc. A status is returned with the notification to indicate the result of requesting operation, *SUCCESS*, *PENDING* or *FAIL*.

An FFD configured as a PAN coordinator starts to perform an ED scan to search whether there exists another PAN operating in the current channel within the personal operating space of the device. It may switch to another channel to avoid interference after the phase of ED scan (8 symbols period). Once the operating channel is determined, the PAN coordinator enters the active scan phase by sending *BeaconReq* followed by turning its receiver on. If it has not received any beacons at the end of the active scan, it may request to turn the transceiver off or determine to start its own PAN. The method *MLME_StartReq* is called to start using a superframe configuration. If the *macBeaconOrder* is equal to 15, the PAN is configured as a beaconless one. Otherwise, it starts a beacon-enabled PAN, and the PAN coordinator transmits superframe beacons in predetermined intervals ranging from 15ms to about 250s. Devices in this PAN will use the beacons to align their starting time of transmission.

IEEE 802.15.4 devices (RFD or FFD) perform a passive scan at the beginning. During the scan period, it stores the potential PAN information (such as PAN id, coordinator extended address, and superframe) extracted from beacons. A device may attempt to join a specific PAN by sending an *AssocReq* to the PAN coordinator. The state transitions of the association process are shown in Figure 3(a) and Figure 3(b) for the PAN coordinator and PAN devices respectively.

Transmission of the response for association request is through indirect transfer mode.



(a) State machine for the PAN Coordinator

(b) State machine for PAN devices

Figure 3. State machine for the PAN coordinator and devices

The association response is stored in *pendingTransactionList* at the PAN coordinator, and the availability of the response is indicated in *Beacon*. The device requests the response by a *DataReq* addressed to the PAN coordinator. Once the association process is completed, devices and the coordinator compete for the channel access for data transmission in CAP period by CSMA-CA. In addition, a device may request the PAN coordinator to allocate a GTS.

In a beacon-enabled PAN, the CAP is divided into slots. The superframe structure is bounded by the beacons. Before a device in the PAN transmits, it has to perform CCA to check whether another device is currently transmitting. If the channel is busy due to another transmission, the device backs off a random number of time slots. The moment to schedule the backoff timer is not necessarily at the boundary of the slot. Thus the important aspect of the backoff is to locate the backoff boundary. The formula used to adjust the backoff

time is as follows:

$$
\begin{aligned}
align = T_{aBackoff} * symbolDuration \\
* \left\lceil \frac{now - beaconTxTime}{T_{aBackoff} * symbolDuration} \right\rceil \\
- (now - beaconTxTime)
\end{aligned}
\quad (1)
$$

, where $T_{aBackoff}$ is the unit backoff period expressed in terms of symbol period, *symbolDuration* is the period of a symbol (0.000016 sec in 2.4G band), *now* is the current time, and *beaconTxTime* is the most recent beacon transmitting time.

In MAC class *L2Proto802_15_4*, ten timeout events are defined and they are scheduled at appropriate times. Among these timeout events are BACKOFF_TIMEOUT, ACK_TIMEOUT, RESPONSEWAIT_TIMEOUT, FRAMERESPONSE_TIMEOUT, SUPERFRAME_TIMEOUT, BEACON_TIMEOUT, SCANDURATION_TIMEOUT, and other three that are infrequently used (RXON_TIMEOUT, DEFER_TIMEOUT and TRACKBEACON_TIMEOUT). RRESPONSEWAIT_TIMEOUT is for the PAN coordinator to prepare for association response. Upon the RESPONSEWAIT_TIMEOUT, the device initiating associate request attempts to extract association response by sending a *DataReq*. FRAMERESPONSE_TIMEOUT is used by a device in the case that the acknowledgement frame from the coordinator indicates there is a transaction pending for it. If the timer expires without receiving the transaction, the device concludes no data available.

For the convenience of study, the energy model in [28] is implemented. This model takes into account transition energy between the operational states of the RF transceiver CC2420 [40]. Additionally, the following structures are defined to collect the statistics on the energy and the packets.

```
struct zigbee_energy {
  float steady_idle;
  float steady_recv;
  float steady_tx;
  float trans_energy;//engery on transition
```

```
};
struct zigbee_stats {
  uLong ack_recv_cnt;
  uLong ack_sent_cnt;
  uLong data_recv_cnt;
  uLong data_sent_cnt;
  uLong beacon_recv_cnt;
  uLong beacon_sent_cnt;
  uLong dreq_recv_cnt;//data req
  uLong dreq_sent_cnt;
  //other commands sent and recvd
  uLong ocmds_recv_cnt;
  uLong ocmds_sent_cnt;
  uLong bytes_recv_cnt;
  uLong bytes_sent_cnt;
  //energy statistics
  zigbee_energy energy_stats;
};
```

3.2.2 Physical layer implementation

In 802.15.4 simulation module, class *Interface_802_15_4* is the physical layer functionality implementation. The class maintains all physical characteristics of a 802.15.4 interface such as transceiver state, device operation mode, physical information base (PIB), and radio range. Physical information base is defined as a structure *PhyPIBType*, comprising the following attributes: phyCurrentChannel, phyChannelsSupported, phyTransmitPower, and phyCCAMode. For convenience, aMaxPHYPacketSize and aTurnaroundTime are also defined. All thirteen primitives, including three data service and management service primitives, are implemented.

ED scan and CCA could be simulated. When the MAC layer requests to perform ED scan, *Interface_802_15_4* schedules a timer event *EDtimeout*. It reports to MAC layer when the *EDtimeout* event happens. Currently the simulation module supports CCA of mode 2.

The *Interface_802_15_4* reports a busy medium upon detection a signal with the modulation and spreading characteristics of 802.15.4, no matter whether the signal strength is above the threshold.

3.3 Experiment

In this section, a set of experiments are designed to demonstrate the correct implementation of the simulator. Simulation results are obtained by analyzing log files. Our simulation module allows users to enable an appropriate log level based on their interest. Alternatively, users can customize a built-in statistics tool easily to collect the desired output.

As pointed out in [57], there are three possible sources of the validation of a simulation model: expert intuition, real system measurements, and theoretical results. Here, expert intuition is used to validate the correctness of our simulator implementation. To quote Jain, "a fully validated model is a myth", so we choose to present some experiment results to validate the correctness.

A set of experiments is conducted with a simple beacon-enabled PAN, consisting of a PAN coordinator *node(0)* and a PAN device *node(1)*. Our goal is to demonstrate the correctness of our implementation. We choose BeaconOrder = 7 and SuperframeOrder = 7, thus the duty cycle is 100%. The active potion of the superframe is 1.98114 seconds. *Node(1)* is the traffic source with a simple exponential on-off pattern. The on period only occupies 5% of the time. The traffic rate is 1000 bits/sec and the packet size is 20 bytes. One hundred and eighty bytes of data flows from the traffic source *node(1)* to the destination *node(0)*.

This experiment illustrates the indirect transmission, which favors energy-sensitive devices. The devices could fetch the data at appropriate time after notified. Figure 4 shows an excerpt from the output file. The first two lines show the two devices and their roles in PAN. The lines showing "Send a Packet" is recorded when the MAC layer receives the notification from the interface indicating the packets have been successfully transmitted.

```
Device: 0x8d2c470 a Pan Coordinator
Device: 0x8d2ca10 a Pan staff..
....     ....   ..      ....         ..
0x8d2c470 3.96288 @@## Send a "@Packet@"
0x8d2ca10 3.96365 @@## Received a "@BEACON@"0x4f
....     ....   ..      ....         ..
0x8d2ca10 7.88058 @@## Send a "@Packet@"
0x8d2c470 7.88163 @@## Received a "@COMMAND@":assoc_req  0x4e
0x8d2c470 7.88163 @@## Send a "@Packet@"
0x8d2ca10 7.88202 @@## Received a "@ACK@"0x4e
0x8d2c470 7.89504 @@## Send a "@Packet@"
0x8d2ca10 7.89574 @@## Received a "@BEACON@" 0x51
0x8d2ca10 8.37594 @@## Send a "@Packet@"
0x8d2c470 8.37664 @@## Received a "@COMMAND@":data_req  0x4f
0x8d2c470 8.37664 @@## Send a "@Packet@"
0x8d2ca10 8.37702 @@## Received a "@ACK@" 0x4f
0x8d2c470 8.37837 @@## Send a "@Packet@"
 0x8d2ca10 set associated flag
0x8d2ca10 8.37955 @@## Received a "@COMMAND@":assoc_rsp  0xd1
0x8d2ca10 8.37955 @@## Send a "@Packet@"
0x8d2c470 8.37994 @@## Received a "@ACK@" 0xd1
 0x8d2c470 Setting associate flag
0x8d2c470 13.7933 @@## Send a "@Packet@"
0x8d2ca10 13.794 @@## Received a "@BEACON@" 0x52
....     ....   ..      ....         ..
0x8d2ca10 15.7622 @@## Send a "@Packet@"
0x8d2c470 15.7627 @@## Received a "@COMMAND@":data_req  0x50
0x8d2c470 15.7627 @@## Send a "@Packet@"
0x8d2ca10 15.7631 @@## Received a "@ACK@" 0x50
0x8d2c470 15.7678 @@## Send a "@Packet@"
0x8d2ca10 15.7686 @@## Received a "@DATA@" 0xd2
0x8d2ca10 15.7686 @@## Send a "@Packet@"
0x8d2c470 15.769 @@## Received a "@ACK@" 0xd2
0x8d2c470 17.7254 @@## Send a "@Packet@"
0x8d2ca10 17.7265 @@## Received a "@BEACON@" 0x54
0x8d2ca10 17.7289 @@## Send a "@Packet@"
0x8d2c470 17.7294 @@## Received a "@COMMAND@":data_req  0x51
0x8d2c470 17.7294 @@## Send a "@Packet@"
0x8d2ca10 17.7298 @@## Received a "@ACK@" 0x51
....     ....   ..      ....         ..
0x8d2c470 1099.07 @@## Send a "@Packet@"
0x8d2ca10 1099.07 @@## Received a "@BEACON@" 0x7a
....     ....   ..      ....         ..
```

Figure 4. An excerpt from the simulation log file

Therefore, no further information (such as packet type, seq no, etc) is written in the output file.

The output files show detailed information on the event of packet receiving. The information includes the destination of the packet, the time of receiving, packet type, and packet sequence number. The message sequences exchanged between the coordinator and the device confirms the correct implementation of 802.15.4 protocol.

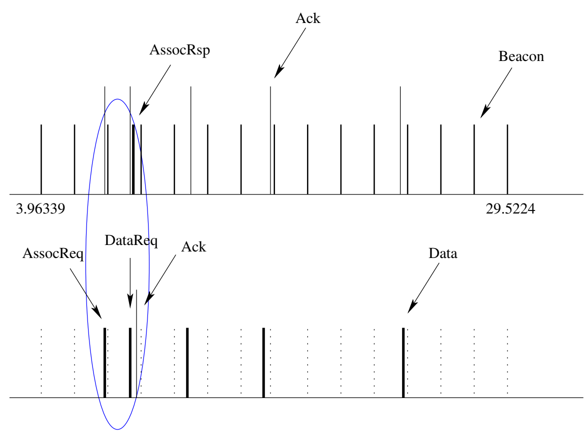


Figure 5. Messages exchanged between the coordinator and the device

The messages exchanged between the PAN coordinator and the device are shown in Figure 5. Before the device *node(1)* is allowed to transmit data, it has to join the PAN by establishing the association with the coordinator *node(0)*. The procedure of association shows exactly how the indirect transmission mode works. In order to see clearly the timeliness of the messages, the zoomed-in circle part of Figure 5 is reproduced in Figure 6.

After *node(1)* finds a beacon in its proximity, it acquires the coordinator information such as *address*, *superframeOrder*, *beaconOrder*, which are critical for communications afterwards. The association starts from *node(1)* sending *AssociateReq*. *Node(1)* enables its receiver in preparation for an acknowledgement. It schedules a *responseWaitTimer* after receiving the acknowledgement. When the period ends the device applies CSMA-CA mechanism before transmitting a *DataReq*. In this experiment, the device needs only two backoffs and CCAs because it is the only device in the PAN. As usual, an acknowledgement

is expected for the *DataReq*. Upon the reception of the acknowledgement, *frameResponseTimer* is scheduled. It is cancelled when the device *node(1)* receives *AssocRsp* from *node(0)*. The coordinator also uses CSMA-CA mechanism before the transmission of *AssocRsp*. Note: even in Figure 6, the period of *ackTimer* was so short that it is not visible.

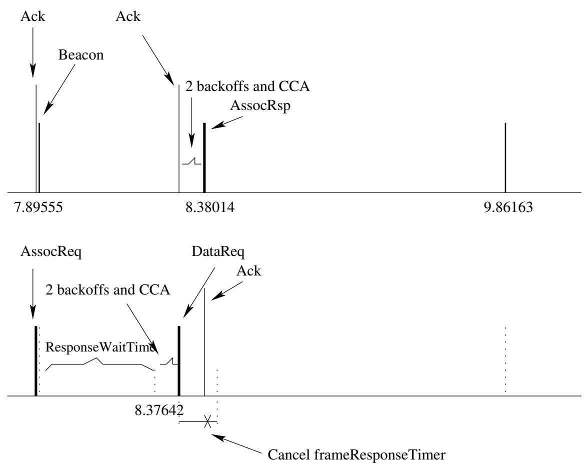


Figure 6. Zoom-in view of messages exchanged during association

The message sequences between the coordinator and the device confirms the correct implementation of the 802.15.4 simulation module.

3.4 Conclusion

IEEE 802.15.4 standard targets a wide variety of applications, requiring simple short-range wireless communications with limited power and relaxed throughput needs. It offers device-level wireless connectivity at low cost.

To investigate the performance of wireless sensor network applications running on top of the new standard, we develop a simulation module in *GTNetS*. The full set of physical and MAC layer primitives of 802.15.4 are implemented. Our aim is to provide an effective 802.15.4 simulation tool for a very large scale wireless sensor network with tens of thousands of nodes. In addition, we present a set of experiments designed to demonstrate the correct implementation.

The simulator is continuously under development. Enabling visualization is under consideration because it can help pinpoint incorrect protocol behaviors as suggested in [44]. Once the implementation is complete and fully tested, we will make our simulation module publicly available online to benefit the PAN research community.

CHAPTER 4

ENERGY-EFFICIENT BEACON STRUCTURE AND GTS DESCRIPTOR DISTRIBUTION

Although IEEE 802.15.4 standard is designed carefully with energy energy efficiency in mind, it is subject to improvements in protocol overheads. The goal of the work presented in this chapter is to reduce protocol overhead caused by superframe beacons. Particularly, an acknowledgement based GTS descriptor distribution scheme is presented in this chapter. Also unnecessary information in beacon structure is stripped to reduce energy consumption of a whole PAN.

4.1 Introduction

As discussed in Chapter 1, reduction of protocol overhead can result in energy savings in WSNs. This chapter presents an innovation scheme to reduce protocol overhead in IEEE 802.15.4.

The GTS in 802.15.4 can provide services with a particular bandwidth and latency requirements. However, the very limited number of GTSs has to be shared by multiple devices if highly demanded. The GTS descriptor is distributed within four superframe beacons mechanically in the 802.15.4 standard. In a PAN of larger size, this would result in unnecessary energy drain in other devices. For example, a home network can reasonably have more than 200 sensors (light switches, universal remote controllers, motion sensors, temperature sensors, gas presence, intrusion, alarm, a number of actuators, so on.). Based on the observation, an acknowledgement based GTS descriptor distribution scheme is proposed with unnecessary information in beacon structure stripped.

As discussed in [21], with GTS, a device can obtain dedicated bandwidth with the overhead of beacon frame receptions. However, [21] did not give detail on how much energy inefficiency is caused by beacon frames. This motivates us to evaluate the ideal

energy efficiency of the three transmission modes (direct, indirect, and GTS) of 802.15.4.

Two different device types can participate in an LR-WPAN network: a full-function device (FFD) and a reduced-function device (RFD). The FFD can serve as a PAN coordinator, or a device. An RFD is intended for applications that are extremely simple, such as a light switch or a passive infrared sensor; they do not have the need to send large amounts of data. Consequently, the RFD can only serve as devices. Due to this fact that devices (e.g. RFD) in a PAN are usually simpler and more power-constrained compared to the coordinator (i.e. FFD), we focus on the energy consumed at a device side when sending/receiving a data packet. Our goal is to determine schemes to reduce energy consumption further in order to extend the lifetime of devices. In the three transmission methods, a device has to track at least one beacon before it starts the procedure of transmission or transaction request. Data transmission/reception in GTS requires the device to apply for a GTS before use. Thus, compared with direct transmission and indirect transmission modes, the GTS request procedure is the overhead. Depending on the number of data transmissions, the overhead may appear quite low. We assume that the energy consumption in the GTS request is negligible in the case that the device has a substantial amount of data to transmit.

The 802.15.4 module was developed in the *Georgia Tech Network Simulator* (GTNetS) simulation platform as a part of the research [36] [37]. The experiments are conducted in a simple beacon-enabled PAN, consisting of a PAN coordinator *node(0)* and a PAN device *node(1)*. We choose BeaconOrder = 7 and SuperframeOrder = 7, thus the duty cycle is 100%. The active potion of the superframe is 1.98114 seconds. *Node(1)* is the traffic source with a simple exponential on-off pattern. The on period only occupies 5% of the time. This traffic pattern matches with most typical sensor applications, where the traffic is not continuous, and the traffic is generated only if the monitored parameter is changed. The traffic rate is 1000 bits/sec and the packet size is 20 bytes. One hundred and eighty bytes of data flows from the source to the destination.

The only variance is that the data source is different in direct, indirect and GTS data

transmission. In direct and GTS data transmission, *node(1)* is the data source while it is the data destination in indirect mode. For each method, we run 10 simulations with different seeds. The transceiver characteristics of the commercial transceiver CC2420 [40] as shown in Table 1 are used for energy calculations.

Table 1. Transceiver parameters

Item	Value
Voltage supply	1.8 v
Receiver current	19.7 mA
Idle current	0.426 mA
Transmitter current (-10dBm)	11 mA

We define the energy efficiency as

$$\rho = \frac{p_{data}}{\sum_{k=1}^{n} p_k} \tag{2}$$

, where p_{data} is the power spent on the raw data transmission, and p_k is the power consumed on the beacon, acknowledgement, idle state, data frame and other command frames. The energy consumed in turning on/off the transceiver is fixed, thus is excluded from the calculation. The energy efficiency obtained for the three modes are shown in Table 2.

Table 2. Power efficiency

Transmission method	Efficiency	Energy spent on tracking beacons
Direct transmission	38.08%	37.02%
Indirect transmission	40.87%	24.52%
Transmission through GTS	39.45%	38.35%

From Table 1, we can see that the cost of receiving one byte is relatively high compared to that of transmission of one byte. Moreover, the current in idle state is about 2% of the current in receive state and the backoff periods are very short. Thus, the indirect transmission has the highest efficiency although it involves more complicated procedures. However, the rank of its efficiency is conditional. The condition is that the device knows that a pending data packet for it is available in the upcoming superframe. Direct transmission method ranks the lowest in energy efficiency due to similar reasons. Transmission through GTS is

a very efficient method with guaranteed service although it may waste some bandwidth if no data transmission takes place in the allocated slots. Surprisingly, the cost of tracking beacons is almost 1/3 of energy consumption in all three transmission methods as shown in Table 2. Although the transmission of short data packets is a partial reason, tracking beacons is still a big overhead. Therefore, although superframe beacons from the PAN coordinator play the key role in channel access synchronization, they are also the overhead contributing to lower power efficiency. Periodical transmission and tracking of the beacons consume a substantial amount of energy. In next section, we will investigate how to improve energy efficiency. We acknowledge that the three transmission methods serve different purposes. Our objective is to show which method favors low power devices, and examine the procedures of each transmission method in order to propose new mechanisms to improve their efficiency.

4.2 Energy-efficient beacon structure and GTS descriptor distribution

As we discussed in the introduction, the indirect transmission method has the conditional highest energy efficiency. The condition is that a device turns on its transceiver upon knowing there is a data packet pending for it. In practice, this is usually impossible. Waking-up frequently would lower the efficiency. On the other hand, waking-up rarely would incur the risk of losing packets because the pending packet is discarded if not fetched within the maximal transaction persistence time. However, a suboptimal solution exists. Devices can wake up periodically (maximal transaction persistence time) to check whether there is a transaction pending. If there is, the devices will try to fetch the packet. And the devices should continue to track the next superframe beacon to check again. If there is no indication of packets pending, the device turns off its transceiver. Otherwise, the fetch procedure continues.

One similarity shared among the three transmission methods is that the device has to

(a)

Octs:2	1	4/10	2	1	0/1	var	var	var	2
Frame control	Seq No	Address –ing fields	Superframe Spec	GTS Spec	GTS directions	GTS list	Pending address fields	Beacon payload	FCS

(b)

Bits: 0–3	4–7	8–11	12	13	14	15
Beacon order	Superframe order	Final CAP slot	Battery life extension	Reserved	PAN coordinator	Association permit

Figure 7. Beacon frame structure and superframe specification

track at least one beacon before it can prepare for its data transmission. As shown in Table 2, the power consumption for receiving a beacon is about 37% in direct transmission mode, 24% in indirect transmission mode and 38% in transmission through GTS. In a large size PAN, multiple devices have to receive the same beacon and decode it although they do not have any data to transmit/receive, which causes a substantial amount of energy waste. By examining the beacon frame structure in Figure 7 closely, we find that it has much space to minimize its length. A beacon contains at least 1 byte GTS field – *GTS specification* all the time regardless of whether valid GTS information is included. Figure 7(b) shows there is a *Reserved* bit in the superframe specification. Therefore it can be used to make the GTS field as an option in the beacon frame. In this case, the GTS field is included only when necessary, so higher energy efficiency can be achieved. The modification and the interpretation are as follows:

GTSIndication(Reserved)	Description
0	no GTS spec
1	GTS spec

Another unnecessary energy drain is when the coordinator disseminates the GTS allocation information – GTS descriptor. The GTS descriptor remains in the beacon frame for *aGTSDescPersistenceTime* (default: 4) superframes. When the device, which has requested a GTS slot, receives the first beacon containing the descriptor, the information in the beacons received afterwards become redundant. Note: other devices accessing the wireless channel in CAP do not need to know this information. Instead, it is sufficient for them to know the *Final CAP slot* through the superframe specification. Based on the analysis

above, we propose the mechanism as shown in Scheme 1 to make the dissemination of GTS descriptor more efficient.

Scheme 1 GTS information distribution

```
1: The coordinator allocates GTS slots for devices
2: The coordinator transmits a beacon including the GTS descriptors
3: Turn on the receiver and keep it on in all allocated GTS slots
4: The devices requesting the GTS slot receive the beacon.
5: if The GTS slot is of transmit type then
6:     if The device has data to send then
7:         Transmit data in the GTS slot allocated for this device
8:     else
9:         Transmit an ACK to the coordinator in the GTS slot allocated for this device. The seq
           number in ACK uses the seq of the beacon
10:    end if
11: else if The GTS slot is of receive type then
12:    The device transmits an ACK to the coordinator in the GTS slot allocated for this device.
       The seq number in ACK uses the seq of the beacon
13:    Turn receiver on
14: end if
15: if The coordinator receives the ACK/data in the dedicated GTS slot then
16:    It stops the dissemination of the GTS descriptor in beacons
17: else
18:    The coordinator repeats the above procedure from step 2
19: end if
```

This scheme requires the requesting devices to return an ACK either implicitly or explicitly upon the reception of GTS descriptors. For the GTSs of transmit type, requesting devices may utilize the allocated GTS for data transmission, which implicitly indicates that GTS descriptors have been received. If no data is available for transmission, requesting devices return acknowledgement frames to notify the coordinator. For the GTSs of receive type, requesting devices transmit acknowledgement frames in their allocated GTSs to the coordinator. Upon being notified that a GTS descriptor has been received in either way, the coordinator excludes it in coming superframe beacons. Thus, the number of times that GTS descriptors are included within beacons is reduced. The performance of the proposed scheme is evaluated next.

4.3 Performance evaluation

In this section, the proposed ACK-based scheme is evaluated. We acknowledge that the throughput in the GTSs is affected a bit because a small portion of it may be used for implicit ACK transmission. As a matter of fact, energy consumption is the primary and critical concern in WSNs. Therefore, the energy consumption is the only evaluation metric used in the following analysis.

For simplicity, we use the same ACK frame format as the 802.15.4 specification. Since each device that requests a GTS slot returns an ACK or implicit ACK in its own allocated slot, no collision occurs at the coordinator. During the GTS period, each device is expected to turn on its receiver or transmitter only at its designated slot for energy saving. We assume that the acknowledgment and data transmission in GTS slots are reliable due to the exclusive usage. In order to analyze the net energy saving of the proposed GTS allocation distribution algorithm, the following variables are defined. Let E_t and E_r be the energy for transmitting and receiving one byte respectively. N is the total number of devices in a PAN, and ρ is the percentage of devices waking-up. n_{ack} is the length of an ACK frame measured in terms of bytes, $n_{ack} = 11$. n_{bytes} is the bytes saved for a GTS descriptor, $n_{bytes} = 3$. Compared with the scheme defined in 802.15.4 specification, the extra energy cost by the proposed algorithm comes from the transmission and reception of ACKs to GTS descriptors. As indicated in the proposed algorithm shown in Scheme 1, the illustration and analysis for the energy cost are discussed as below.

Case 1: the GTS slot is of transmit type and the device has data to send at the current time. The device can utilize the allocated slot to transmit the data to the coordinator. The reception of data at the coordinator implicitly acknowledges the successful reception of the GTS descriptor from the specific device. Due to the fact that the data transmission is necessary regardless of whether the proposed algorithm is deployed or not, the extra energy cost is:

$$E_{cost} = 0 \tag{3}$$

Case 2: the GTS slot is of transmit type and the device has no data to send at the current time. In this case, the device transmits an ACK to the coordinator to notify its successful reception of the GTS descriptor. Note: the transmission takes place in the designated GTS slot, so no other devices in the PAN receive the packet. The extra costs are only at the specific device and the coordinator. The energy cost is:

$$\begin{aligned} E_{cost} &= E_{cost_{device}} + E_{cost_{coord}} \\ &= (E_r + E_t)n_{ack} \end{aligned} \tag{4}$$

Case 3: the GTS slot is of receive type. It is similar to the scenario of Case 2. And the cost is the same as that of the Case 2.

$$\begin{aligned} E_{cost} &= E_{cost_{device}} + E_{cost_{coord}} \\ &= (E_r + E_t)n_{ack} \end{aligned} \tag{5}$$

Assume there are k devices requesting GTS slots in a PAN, of which m devices fall into Case 1. $0 \leqslant m \leqslant k \leqslant 7$. Thus, the extra energy cost for the entire PAN is:

$$E_{cost} = (k - m)(E_r + E_t)n_{ack} \tag{6}$$

The energy savings of our proposed schemes are from reception and transmission of smaller size beacons at the devices and the coordinator respectively. The acknowledged GTS descriptors are excluded in the next beacons. For example, if 3 of the k devices acknowledged the receptions of their descriptors, only $k - 3$ descriptors will be included in the remaining beacons. The coordinator distributes GTS descriptors at most *aGTSDescPersistenceTime* (4) times. Table 3 shows the example of how to calculate the energy saving compared with the original scheme. The fourth column is the number of devices requesting GTS receiving their own descriptors in the beacon. If x devices receive their descriptors in the 1st beacon, their descriptors will not be included in the next three beacons. Note: in the PAN, not only the devices requesting for GTS track the beacons, all devices that are awake

Table 3. Example for calculation

Beacon Seq	**# descriptors** incIded(original)	**# descriptors** included(proposed)	**# devices recv their** descriptors in the beacon
1	k	k	x
2	k	k-x	y
3	k	k-x-y	z
4	k	k-x-y-z	k-x-y-z

are also tracking the beacons. Therefore, the energy saving is:

$$\begin{aligned} E_{saving} &= E_{devices} + E_{coord} \\ &= (3xE_r n_{bytes} + 2yE_r n_{bytes} + zE_r n_{bytes}) \times N\rho \\ &\quad +(3xE_t n_{bytes} + 2yE_t n_{bytes} + zE_t n_{bytes}) \times 1 \\ &= (3x + 2y + z)(E_r N\rho + E_t)n_{bytes} \end{aligned} \tag{7}$$

Alternatively, the above equation could be obtained in another way. Let P_1, P_2, P_3, and P_4 be the probabilities that the 1st, 2nd, 3rd and 4th beacon be the first successfully received one containing the descriptor by a device. The requesting devices can only receive one copy of their descriptors, either in the 1st beacon, or in the 2nd/3rd/4th beacon. Thus, the two conditions:

$$P_j = \begin{cases} 1 & \text{, a device receives its descriptor in } j\text{th beacon} \\ 0 & \text{, otherwise} \end{cases}$$

$$\sum_{j=1}^{4} P_j = 1 \tag{8}$$

are satisfied. The number of requesting devices that receive their descriptors in the 1st,

2nd, 3rd beacon are

$$\sum_{i=1}^{k} P_1^{(i)} = x \tag{9}$$

$$\sum_{i=1}^{k} P_2^{(i)} = y \tag{10}$$

$$\sum_{i=1}^{k} P_3^{(i)} = z \tag{11}$$

Therefore, Equation (7) can be rewritten in the following format.

$$\begin{aligned} E_{saving} &= E_{devices} + E_{coord} \\ &= (3\sum_{i=1}^{k} P_1^{(i)} E_r + 2\sum_{i=1}^{k} P_2^{(i)} E_r + \sum_{i=1}^{k} P_3^{(i)} E_r) n_{bytes} N\rho + (3\sum_{i=1}^{k} P_1^{(i)} E_t + 2\sum_{i=1}^{k} P_2^{(i)} E_t + 1\sum_{i=1}^{k} P_3^{(i)} E_t) n_{bytes} \\ &= (3\sum_{i=1}^{k} P_1^{(i)} + 2\sum_{i=1}^{k} P_2^{(i)} + \sum_{i=1}^{k} P_3^{(i)})(E_r N\rho + E_t) n_{bytes} \end{aligned} \tag{12}$$

So the condition for energy saving in the entire PAN is

$$E_{saving} > E_{cost} \tag{13}$$

Plugging Equation (6) and (12) into (13), we have

$$\begin{aligned} &(3\sum_{i=1}^{k} P_1^{(i)} + 2\sum_{i=1}^{k} P_2^{(i)} + \sum_{i=1}^{k} P_3^{(i)})(E_r N\rho + E_t) n_{bytes} \\ &\qquad > (k - m)(E_r + E_t) n_{ack} \end{aligned} \tag{14}$$

It is obvious that the PAN always benefits from the proposed scheme if all the GTS processing scenarios are of transmit type, and devices have data to send when they acknowledge the reception of descriptors. This case is very common in sensor networks where most data flows are from sensor devices to the sink (coordinator). The worst case is that all the requesting devices receive their descriptors in the fourth beacon. The proposed algorithm would result in a little energy waste due to the transmission of ACKs at the requesting devices and reception of ACKs at the coordinator.

The characteristics of CC2420 in Table 1 are used. It is easy to verify that if the requesting devices receive their GTS descriptors in the third beacon, the PAN could achieve energy savings in the case of 4 other devices listening. This condition is very easy to be satisfied in wireless sensor networks.

A number of variants of the scheme exist. For example, when multiple devices apply for GTS, the coordinator may exclude GTS descriptors that have been acknowledged through data transmission or ACKs, and continue to disseminate others. Another variance is to apply implicit ACK only, which could extend the lifetime of the device at the cost of energy waste at other devices. The proposed mechanism can also be applied to deallocate GTS initialized by the PAN coordinator.

The proposed mechanism has two merits. The first is power efficiency. A typical sensor network usually consists of a large number of devices with dense deployment. With our mechanism not only is the energy of the coordinator and the targeted device saved, the energy consumption of other devices that are tracking network beacons is also reduced. The second merit is that the proposed mechanism is compatible with the implementation of the existing specification without introducing extra complexity. If the MAC of the PAN coordinator incorporates the proposed scheme, and requesting devices adopt the standard MAC, the coordinator would transmit the GTS descriptors in four beacons in the case that requesting devices do not transmit data frames in their allocated GTSs. In the case that devices use their allocated GTSs of transmit type, the coordinator excludes their descriptors in the remaining beacon frames. This behavior does not affect devices to use their GTSs. If the coordinator adopts the standard MAC while requesting devices use the modified MAC, the coordinator distributes GTS descriptors in four beacons as defined in the standard, and it ignores any acknowledgement frames in the beginning of GTSs since it does not understand it. Thus, the proposed scheme has very good back-compatibility.

To investigate the performance of the proposed beacon structure and the innovated GTS

descriptor distribution, we perform some evaluations. In a typical dense microsensor network, a large number of nodes are deployed as redundant nodes. We consider a scenario where 1000 nodes are distributed around a sink. The sink acts as the coordinator, while the sensor nodes are PAN devices. We assume that all the nodes are within the communication range of the coordinator. In normal operation, among the 1000 nodes some of them are in sleep mode and others are awake, transmitting packets or waking up to check whether there are data pending at the coordinator for them. We focus on the energy consumed at devices when receiving beacons because devices are usually more power-constrained than the coordinator in a PAN.

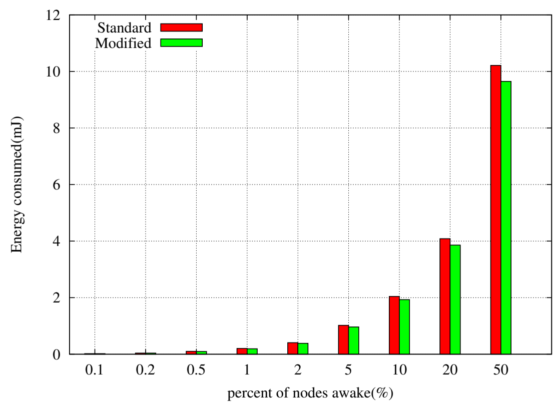


Figure 8. Energy for tracking a beacon at devices in the PAN

Figure 8 shows the energy for devices receiving a beacon in the PAN. With an increase in the number of nodes waking up to track *one* beacon, the overall energy consumed is increased. For one device, the energy used to receive 17 standard beacons is sufficient to receive 18 modified beacons. That is to say a device can save 5.6% energy in tracking beacons. Therefore adopting the modified beacon structure favors very low power devices.

In order to evaluate the performance of the innovated GTS allocation distribution algorithm, we assume only one device in the PAN requests GTS and it is approved. This is the simplest case. The coordinator disseminates the allocated GTS descriptor in beacons. The energy consumed by PAN devices to track the four beacons with the modified beacon structure and the GTS distribution mechanism simultaneously is shown in Figure 9.

The *fourth*, *third* and *second* represent the target device receiving the GTS descriptor in the fourth, third, and second beacon respectively, which indicates the previous beacons are missed due to various reasons such as sleeping. The *first* represents the target device receiving the GTS descriptor in the first beacon. From the figure, in the case of *fourth*, the PAN always consumes much more energy than in other cases. It is because the coordinator has to disseminate the GTS descriptor in four beacons due to no ACK from the target device. This is the worst case. If the device receives and acknowledges the first beacon, the coordinator will transmit the remaining three beacons with modified beacon structure. In this case, the PAN saves the largest amount of energy, which is about 17% saving compared with the standard protocol. This amount of energy saving is vital to significantly extend the lifetime of low power wireless sensor networks.

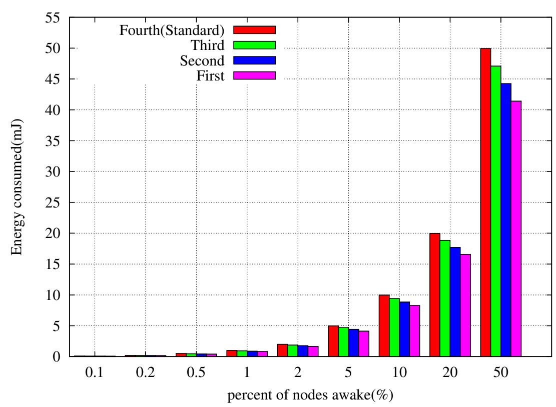


Figure 9. Energy for tracking the four beacons when applying our proposed schemes in the case of one GTS request

The scenario becomes complicated when multiple devices request GTS, and some of them may receive their GTS descriptors in the first beacon while others receive them in the remaining beacons. The energy consumption for tracking these beacons depend on the variances. There exist 120 combinations. Instead, a simplified case is considered here. Seven devices in the PAN request GTS. The requesting devices receive their descriptors either all in the first beacon, or all in the 2nd/3rd/4th beacon. Figure 10 shows the energy consumption by PAN devices to track the four beacons with the modified beacon structure

and the GTS distribution mechanism. Compared with the mechanism in 802.15.4 standard, our proposed algorithm can save up to 47.3% energy in tracking the four beacons if the requesting devices can receive their descriptors in the first beacon. When they receive their descriptors in the second and third beacon respectively, 31.5% and 15.8% energy saving could be achieved. Considering that the GTS is requested on demand and is deallocated when not used, a PAN can benefit significantly from the proposed algorithm and the modified beacon structure.

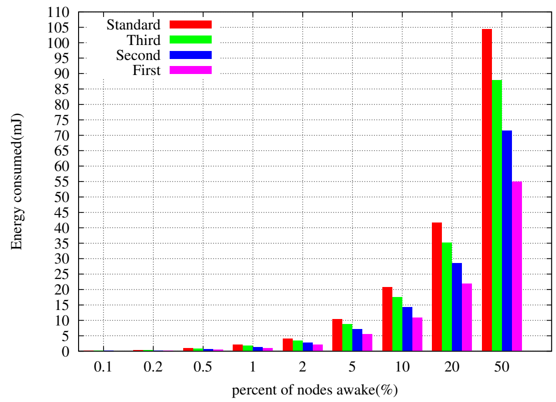


Figure 10. Energy for tracking the four beacons when applying our proposed schemes in the case of seven GTS requests

It is interesting to investigate the conditions for achieving energy saving in a PAN. Figure 11 shows the simplified cases, where all requesting devices receive their GTS descriptor in the first, second or third beacon. Note: this figure does not depend on the number of devices requesting for GTS. As shown in the figure, the PAN can achieve energy saving as long as there is another device tracking beacons if the requesting devices can receive their GTS descriptors in the first beacon. The PAN can always benefit from energy savings if 60% of the GTS requests fall into case 1 (GTSs are of transmit type and requesting devices have data to send). If all requesting devices receive their descriptors in the second beacon, two other waking-up devices are needed for net energy saving. In case of requesting devices receiving descriptors in the third beacon, there only needs four of devices to track the beacons in order to achieve net energy savings. The conditions are easy to be satisfied in

WSNs, where a large number of redundant nodes are deployed.

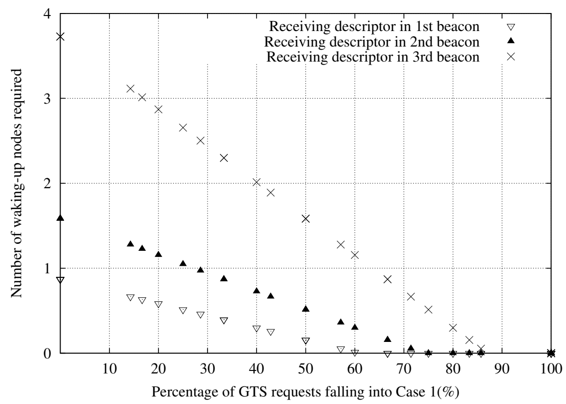


Figure 11. The number of waking-up nodes required for energy saving in a PAN

4.4 Conclusion

IEEE 802.15.4 standard targets a wide variety of applications, which require simple short-range wireless communications with limited power and relaxed throughput needs. It offers device-level wireless connectivity at low cost.

In this chapter, we investigate the performance of different data transmission methods in terms of power efficiency. Simulation results show that although superframe beacons from the PAN coordinator play the key role in channel access synchronization, they are also the overhead contributing to lower power efficiency. By examining the procedure of the three transmission methods closely, we streamline the size of network beacons to save the energy consumed by the devices which are tracking the beacons. In addition, a simple but efficient mechanism is proposed for a coordinator to disseminate GTS descriptors. The conditions for net energy saving are discussed. Evaluation results show that these solutions can reduce up to 47.3% energy consumption in tracking beacons at devices, and thus extend the lifetime of the entire PAN.

Our future work includes implementation of the proposed energy efficient mechanisms in 802.15.4 RF chips, and employing real experiments to test their performance.

CHAPTER 5

EFFICIENT CHANNEL RESERVATION FOR MULTICASTING GTS ALLOCATION AND PENDING ADDRESSES

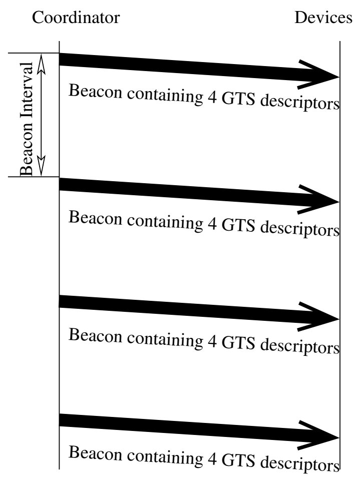
In this chapter, we present an energy efficient channel scheme to distribute GTS descriptors and pending addresses. In this scheme, the PAN coordinator multicasts such information within a special reserved time slot. Those devices that are not interested in these information can turn off transceivers to avoid overhearing, thus saving energy.

5.1 Introduction

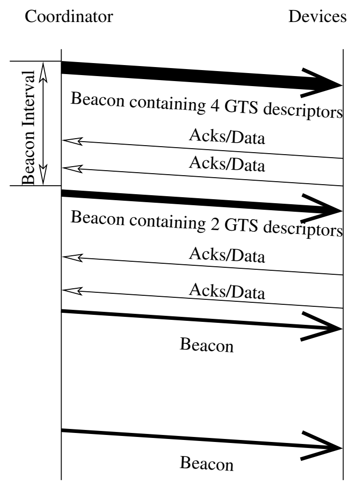
This chapter extends the work in Chapter 4. Although the proposed algorithm works well in reducing energy consumption, devices always overhear at least one beacon with GTS descriptors. Therefore, in this chapter a wireless channel reservation scheme is proposed for distributing GTS descriptors and pending addresses. Devices that are not interested in such information can turn off their radios for energy savings.

5.1.1 Motivation

In Chapter 4 and previous work [39], we explored methods to streamline the size of beacons for energy savings in order to prolong the lifetime of a PAN. Our work shows that the one byte GTS field – *GTS specification* in beacon frame structure can be included only when necessary. Also the new scheme was proposed to distribute GTS descriptors efficiently as shown in Fig 12(b). A requester sends to the coordinator either an acknowledgement (explicit ACK) or application data (implicit ACK) in the allocated GTS. The acknowledged GTS descriptors are excluded in remaining beacons. Compared with the standard approach to the dissemination of GTS descriptors as shown in Fig 12(a), the proposed method provides a reliability mechanism, and reduces the power consumption in a PAN. As defined in the 802.15.4 standard [3], only seven GTSs are allowed to be allocated in a PAN. Devices that require guaranteed time slots have to share the limited number of GTSs. During



(a) GTS descriptor distribution in the 802.15.4 standard



(b) Proposed method of distribution method

Figure 12. The comparison of GTS descriptor distribution

the share process, GTS descriptors appear in beacons in allocation and deallocation process. Therefore, the deployment of the proposed algorithms will save substantial amount of energy, especially in a WSN densely deployed with a large number of devices.

The proposed algorithm works well to reduce power consumption in devices that are tracking beacons. However, it does not completely solve the energy waste problem, because each descriptor is received at least once at each tracking devices although these GTS descriptors are not meaningful to them. This motivates us to reconsider the transmission method in 802.15.4.

Let us view the transmission as the follows. The communications between the coordinator and PAN devices can be classified into three groups: one-to-all, one-to-many, and one-to-one.

1. one-to-all: *all* stands for the set of waking-up devices. And *one* refers to the coordinator. This method has global impact on PAN devices. An example of this type is the transmission of beacons.

2. one-to-many: *many* stands for a group of devices in the set of waking-up devices.

And *one* refers to the coordinator. In the 802.15.4 standard, none of defined frames explicitly use this transmission method.

3. one-to-one: The first *one* refers to the coordinator, and the second *one* refers to a specific device. Most communications between the PAN coordinator and devices fall into this category. e.g. the transmission of data from a PAN device to the PAN coordinator.

We can use the term *broadcast, multicast,* and *unicast* to represent the three groups although they are usually concepts of network layer. If a packet that is supposed to transmit to a smaller group is transmitted to a larger group, it will cause unnecessary energy consumption at uninterested parties. Thus, if any information is supposed to distribute in multicast, but actually is distributed in broadcast, it certainly causes unnecessary energy waste at those devices that are not interested in such information. In other words, the PAN coordinator should minimize the use of broadcast. If the broadcast is necessary, the length of the packet should be minimized. So far, only the transmission of beacons falls into the category of broadcast. The beacon frame format in 802.15.4 is shown in Figure 13.

Octs:2	1	4/10	2	1	0/1	var	var	var	2
Frame control	Seq No	Address –ing fields	Superframe Spec	GTS Spec	GTS directions	GTS list	Pending address fields	Beacon payload	FCS

Figure 13. Beacon frame structure

The fields, *Frame control, Seq No, FCS*, are used to indicate the start and the end of a frame and identify the frame. The *Addressing field* tells where the beacon comes from. The *Superframe Spec* distributes the network parameters, such as superframe order, and beacon order, etc. All these information are necessary for every active device. The *Beacon Payload* is useful when a PAN coordinator needs to broadcast a network message, for example, to notify all devices that the PAN will be shutdown in 5 minutes. However, the fields, *GTS spec, GTS direction, GTS list, Pending address*, are only meaningful to a small set of

devices, which are either in need of guaranteed services or have transactions pending at the PAN coordinator. The receptions and processing of these fields cause energy waste at other devices, especially in a dense wireless PAN. The worst case would occur when the devices supposed to fetch pending data dies out of battery[1]. In this situation, their addresses appear in the *Pending Address* periodically within beacon frames until the transaction expires in the queue of the coordinator. Therefore, ideally these fields should be distributed in multicast, instead of broadcast, to avoid unnecessary energy costs.

5.1.2 Solution

How to solve this problem? To solve this problem, the information targeted to a small set of devices would be transmitted in a special time slot, instead of within beacon frames. The devices that most likely have transactions can listen in that special multicast slot if the bit in the frame control field of the current beacon indicates the existence of the slot. In practice, in home networking and industrial monitoring applications, the PANs powered by 802.15.4 typically consist of sensor devices and actuator devices. Those actuator devices can be configured to overhear the traffic in the multicast slot if the slot appears. Other devices may optionally listen to the reserved slot if they have found the existence of the slot in several consecutive superframes. If devices having pending transactions did not listen in this special slot, the transactions are still pended at the coordinator, and the coordinator will transmit the information in the reserved slot of the next superframe until either the transactions expire or the devices get notification finally and fetch the transactions. Now there are three questions remaining to answer:

1. What is the format of the new frame?
2. When should the coordinator transmit this frame?
3. What is the impact on 802.15.4 in terms of energy savings, throughput and delay bound? Any necessary changes?

[1] This problem can be solved through careful cross layer design

In the remainder of this section, we will present our solutions to the above problems.

5.2 Channel reservation scheme

GTS descriptors and pending addresses are separated from other information in beacons, and are transmitted within a special type frame to a small set of devices in a special slot. In this subsection, the format of the new frame and the modified beacon is presented, followed by the exploration of transmission time.

New Frame format and beacon format

We define a frame structure for carrying the GTS information and pending address. The frame structure is as shown in Figure 14, and we call this new frame *GTSPA*.

Octs:2	1	4/10	1	0/1	var	var	2
Frame control	Seq No	Addressing fields	GTS Spec	GTS directions	GTS list	Pending fields address	FCS

Figure 14. New frame structure

The corresponding change to beacon frame structure is as shown in Figure 15. Note: the field *GTS spec* is included when necessary as proposed in Chapter 4 [39].

Octs:2	1	4/10	2	var	2
Frame control	Seq No	Addressing fields	Superframe Spec	Beacon payload	FCS

Figure 15. New beacon structure

When to transmit this new frame?

GTS can provide contention-free guaranteed service due to the exclusive use. Thus in order to transmit the GTSPA frame, a special allocated time slot is reserved[2]. The existence of this slot is indicated by a bit in the current *frame control* field. Those devices who are interested in receiving such information can turn on their transceivers in this slot. The

[2]For simplicity, the period contains only one time slot

question is when to commit the transmission. For better control of channel access, there are four possible times to transmit the new frame as shown in Figure 16.

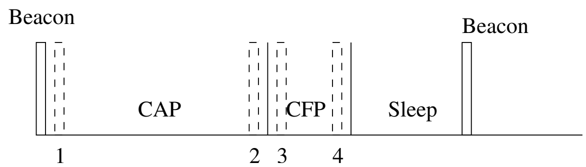


Figure 16. Possible times to transmit the new frame

Which one is the best? Intuitively the time 1 and 3 are better than 2 and 4 because it is easier to control channel access, especially for devices starting back-off. For the sake of energy saving, the earlier it sends, the better because devices can turn off their transceivers immediately after determining that there are no packets pending for them. However, the reservation at time 1 requires all devices to make changes in boundary of a back-off procedure although they do not need to listen to the special slot. Also devices have difficulties of determining the boundary of a backoff period. So for compatibility, the frame GTSPA should be transmitted at the end of CAP and before the start of CFP. The devices that are not interested in the information carried by a GTSPA frame are not aware of the existence of the slot and the frame transmission. Alternatively they can choose to overhear the traffic in the slot.

Note, without the transmission of the GTSPA frame in a superframe, the slot is deallocated and becomes a portion of CAP. This could be achieved through the change of the field "Final CAP slot" in the superframe field of a beacon frame.

5.3 Performance evaluation

Now we investigate the impact of the proposed mechanism on 802.15.4 using the following metrics: energy savings, throughput, and delay.

Energy consumption

The deployment of the proposed mechanism will have an affect on all devices, including

the PAN coordinator, the small number of devices among the set of active devices, and the rest of other active devices.

In order to analyze the energy saving of the proposed mechanism, the following variables are defined. Let E_{byte} be the energy to transmit one byte information. $N_{devices}$ is the total number of devices tracking beacons. We assume no beacon payload is included, and the short address of the PAN coordinator is used in mac header (MHR). No device applies for GTS. The PAN coordinator notifies $n + m$ devices of pending transactions, of which n pending address is of short type (2 bytes) and m pending address is of extended type (8 bytes). Other variables are:

l_{mhr}, the length of MAC header, 7 bytes

l_{mfr}, the length of MAC footer, 2 bytes

$l_{Superspec}$, the length of superframe specification, 2 bytes

$l_{GTSspec}$, the length of GTS specification, 1 byte

l_{pdr}, the length of PHY header, 6 bytes

$l_{PendAddrSpec}$, the length of pending address specification, 1 byte

l_{short}, the length of a short address, 2 bytes

$l_{extended}$, the length of an extended address, 8 bytes

$l_{payload}$, the length of payload in MAC layer.

$l_{overhead}$, the overhead of a frame, which is equal to $l_{mhr} + l_{mfr} + l_{pdr}$

The energy consumed to track such a standard beacon in an entire PAN is

$$\begin{aligned} E_{sbcost} &= (l_{mhr} + l_{mfr} + l_{Superspec} + l_{GTSspec} + l_{PendAddrSpec} \\ &\quad + l_{pdr} + l_{short} n + l_{extended} m) E_{byte} N_{devices} \\ &= (19 + 2n + 8m) E_{byte} N_{devices} \qquad (15) \end{aligned}$$

The new beacon structure contains only the necessary information for accessing channel in CAP. The energy for tracking a new beacon is

$$\begin{aligned} E_{nbcost} &= (l_{mhr} + l_{mfr} + l_{GTSspec} + l_{pdr})E_{byte}N_{devices} \\ &= 17E_{byte}N_{devices} \end{aligned} \quad (16)$$

In order to calculate the energy for overhearing an entire GTSPA frame, we need to find out how many frames are required to complete the transmission of a GTSPA frame. It is because in some cases one time slot is not enough for transmission of an entire GTSPA frame, the frame has to be segmented and transmitted in two or more consecutive superframes. Each frame contains the overhead of $l_{overhead}$ bytes. Thus the number of superframes required to complete the transmission of all information is

$$n_{frame} = \left\lceil \frac{l_{payload}}{T_s - l_{overhead} - \Delta(IFS)} \right\rceil \quad (17)$$

, where ΔIFS = LIFS (LIFS=160 bits [3]) bits if the frame is larger than *aMaxSIFSFrameSize* = 144 bits [3], otherwise, it is equal to SIFS (SIFS = 48) bits. The duration of one time slot is shown in Equation (18).

$$T_s = \frac{aBaseSuperframeDuration * 2^{so}}{16} \quad (18)$$

We use ρ to represent the percentage of devices overhearing a GTSPA frame in a superframe period. Therefore, the energy consumed to track GTSPA frames that contain the GTS and pending address that can be transmitted in a single beacon frame is:

$$\begin{aligned} E_{gpcost} &= ((l_{mhr} + l_{mfr} + l_{phr})n_{frame} + l_{short}n \\ &\quad + l_{extended}m)E_{byte}N_{device}\rho \\ &= (16n_{frame} + 2n + 8m)E_{bytes}N_{device}\rho \end{aligned} \quad (19)$$

The condition for net energy savings in a PAN is:

$$E_{sbcost} \geq E_{nbcost} + E_{gpcost} \tag{20}$$

, thus the parameter ρ should satisfy:

$$\rho \leq \frac{1 + n + 4m}{8n_{frame} + n + 4m} \tag{21}$$

Since only seven addresses are permitted to be included in a beacon, $n + m$ is less than or equal to seven. If $n + m = 0$, no GTSPA frame is transmitted and the special time slot becomes a portion of CAP. Figure 17 shows the boundary percentage of devices that listen to a GTSPA frame. The x-axis in this figure stands for the combination (0, 1), (0,2), ..., (0, 7), (1, 0), (1, 1), ..., (7, 0) from the left to the right. In the case of more than each percentage, the entire PAN consumes more energy than using the standard beacon frame structure.

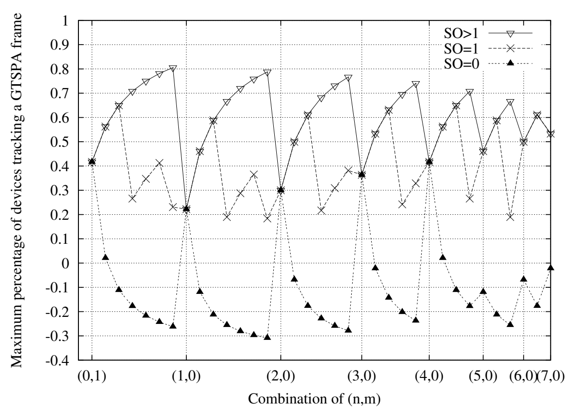


Figure 17. Boundary percentage for achieving net energy savings

In the case of SO > 1, the reserved slot is long enough to transit a whole GTSPA frame without segmentation. When there is only one short pending address to distribute, the length of pending address is the minimal. In this case, there should not exist more than 22% of devices that are tracking beacons to track the GTSPA frame in order to achieve net energy savings in a PAN. The overhead of MAC and PHY header in case of SO > 1 is not so obvious as other two cases. Generally the larger the GTSPA frame is, the higher the bottom

line for overhearing the frame is. In the case of SO = 1, one time slot is so short that it only allows to transmit a GTSPA frame containing a very limited number of addresses. The transmission of all addresses can only be completed in several beacon intervals, leading to a big overhead. Even though, at least more than 15% of devices that are tracking beacons are allowed to track the GTSPA frame. However, most combinations in the case of SO = 0 result in more energy consumption in a PAN with the use of proposed scheme than that with the standard way in 802.15.4. This is due to the overhead of MAC and PHY header. A possible solution is to dynamically reserve the number of slots according to needs instead of one time slot only. In that situation, no segmentation of a GTSPA frame is required. The cases of SO = 0, and 1 have the same boundary percentage with the case of SO>1 as shown in Figure 17.

Throughput

In this subsection, we discuss the throughput performance in the reserved time slot. Depending on the length of a GTSPA frame and the reserved time slot, the GTSPA frame may be required to be segmented and transmitted in several consecutive superframes. Hence we consider the average throughput. The normalized throughput in the reserved slot is:

$$\begin{aligned} Th &= \frac{l_{frame}}{T_s} \\ &= \frac{16 + \frac{2n+8m}{n_{frame}}}{T_s} \end{aligned} \tag{22}$$

, where l_{frame} stands for the length of a frame carrying the information of pending addresses and GTS allocation and T_s stands for the the length of the reserved slot expressed in terms of bytes in 2.4G band. Figure 18 shows the throughput of a reserved time slot for broadcasting pending addresses as a function of short and extended address included. The maximum throughput (0.8) is achieved in the case of SO = 0. The wasted bandwidth is mainly due to idle time and an interframe space. With the increase of SO, T_s becomes longer, which directly results in longer idle time in the reserved time slot. Therefore, the throughput performance degrades when SO increases. This is because of the fixed reservation of one

time slot. How to dynamically reserve a portion of long time slots according to needs is left for future work. Under the current proposed mechanism, SO = 2 produces better performance of both energy saving conditions and throughput as shown in Figure 17 and 18.

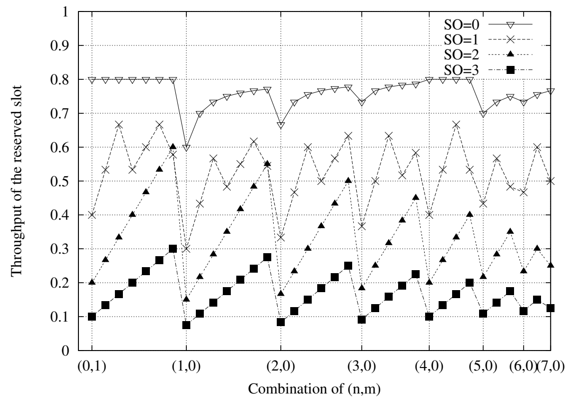


Figure 18. Throughput of a reserved time slot when used to broadcast pending addresses

If the coordinator decides to reserve the number of time slots based on the length of a GTSPA frame, the GTSPA frame is not segmented. In this situation, the throughput for the cases of SO = 0 and SO = 1 is different. The normalized throughput in the reserved slots is shown in Equation (23). Figure 19 shows the throughput difference between the use of one time slot and the use of dynamic reservation of the number of time slots. Generally, the throughput degrades with the use of multiple slots because there likely has more idle time except several cases. However, it reduces the overhead for transmission because all information are contained within a single GTSPA frame.

$$Th = \frac{16 + 2n + 8m}{\left\lceil \frac{16+2n+8m+\Delta(IFS)}{T_s} \right\rceil T_s} \tag{23}$$

Delay

Multicasting the pending address and the GTS allocation information in a reserved time slot rather than within a beacon have direct impact on when the devices receive such notifications. Figure 20 shows the delay D when the transmission of an entire frame could be completed within one time slot. For a specific SO and BO, the delay varies depending

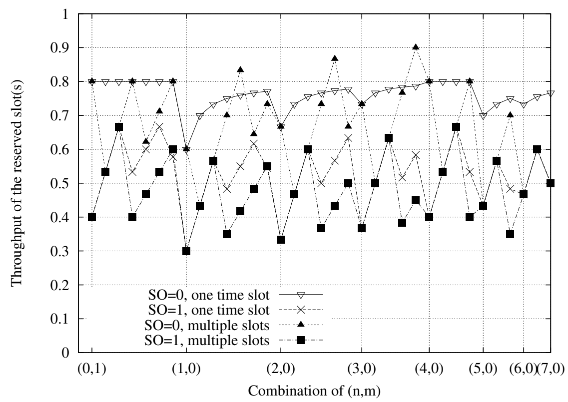


Figure 19. Throughput difference

on the length of the CFP. It is bounded by D_{max}:

$$\begin{aligned} D_{max} &= SuperframeDuration \\ &= aBaseSuperframeDuration * 2^{SO} \end{aligned} \tag{24}$$

In case of segmentation required, for example SO = 0, n = 0 and m = 7, the delay is bounded by Equation (25) because each time slot is possibly enough only for transmission of a pending address.

$$\begin{aligned} D_{max} &= 7 * SuperframeDuration \\ &= 7 * aBaseSuperframeDuration * 2^{SO} \end{aligned} \tag{25}$$

If the coordinator allocates more time slots when pending addresses and GTS information is not fit into one time slot, the delay is also bound by Equation (24).

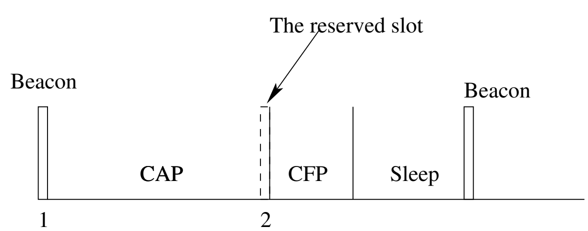


Figure 20. Delay of notifications with the use of multicast channel

5.4 Conclusion

The distribution of pending addresses and GTS information within beacon frames causes unnecessary energy drain in other devices that are not interested in such information. This in turn, reduces the lifetime of an entire PAN. We propose a channel reservation scheme for multicasting GTS allocation and pending addresses. This chapter presents details on the new GTSPA frame structure, the corresponding changes to the beacon structure, and the timing of transmitting the GTSPA frame. Deploying the scheme can achieve substantial energy savings when SO is larger than 0. We study the impact of the proposed scheme on the performance of 802.15.4 and the conditions for achieving net energy saving in a 802.15.4 PAN. Although devices experience delays in receiving notifications of pending transactions, the lifetime of an entire PAN gets extended because of net energy savings.

CHAPTER 6

DYNAMIC RESERVATION OF GTSS

In this chapter, we focus on the bandwidth underutilization problem in the GTSs. A new GTS allocation scheme is presented to address this problem. Our scheme uses the exactly same format of the GTS descriptor structure to convey allocation information while it shows significant improvement over the existing GTS allocation schemes.

6.1 Introduction

The research in previous chapters focus on the distribution of GTS descriptors from the prospect of energy efficiency. Several schemes have been proposed to reduce the energy consumption in the entire PAN when the coordinator distributes such GTS information. This chapter studies another GTS problem – bandwidth underutilization in GTSs.

6.1.1 Motivation

Under the current GTS allocation scheme in 802.15.4, the superframe duration is divided into 16 equally sized time slots. A device may be allocated a GTS consisting of multiple time slots. When the SO is larger, one time slot extends a longer period. However, in a typical WSN the traffic generated by various sensors may vary significantly. For example, in industrial monitoring applications, sensors work with the sendOnDelta concept [33]. Packets are generated only if changes occur. When the monitored process keeps changing, a large amount of data are generated and required to transmit to the PAN coordinator continuously. This leads to the phenomena that some pre-configured GTSs are not enough for some sensors, while most portions of other GTSs are wasted without servicing any traffic.

Even when the traffic from a sensor follows a well-defined behavior, the GTS allocated for it may still be partially used because the allocation of a GTS is based on the basic unit - a time slot. This happens when the amount of available guaranteed bandwidth is higher than the required bandwidth. The unused portion of each GTS is similar to the fragmentation

problem in memory allocation of an operating system [38]. The underutilization of the GTS(s) wastes the precious bandwidth resource.

In this chapter, we investigate how to improve the channel utilization in GTSs.

6.1.2 Related work

The earliest effort to investigate the GTS allocation is in [35]. It studied the limitation of the existing GTS allocation scheme defined in the 802.15.4 standard [3] and proposed a mechanism called i-GAME for GTS allocation.

In the i-GAME scheme, each node that requests the allocation of a GTS reports a traffic specification describing the generated flow. The specification is defined as a tuple: $F_{spec} = (b, r, D)$, where b is the maximum burst size, r is the average arrival rate and D denotes the delay requirement of the flow. An i-GAME management algorithm and an admission control function are run to check whether the GTS request can be satisfied. The outline of the algorithm is shown in Algorithm 2.

Algorithm 2 i-GAME management algorithm

```
A new flow spec arrives
if admissionCtrl(Flow) is false then
    if The limit for maximum number of GTSs is reached then
        Reject GTS request
    else
        Increase the length of the CFP
        Call i-GAME management algorithm
        Return
    end if
else
    Accept the GTS request
    Add the new flow
    Return
end if
```

The coordinator allows devices to share GTSs as long as it could satisfy the flow requirements and it has not reached the limit of the maximum number of GTSs. An illustrative example is shown in Figure 21.

The i-GAME scheme improves the bandwidth utilization by allowing several devices

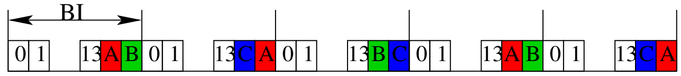


Figure 21. Two time slot allocation used by three nodes under round robin scheduling

to share the same set of GTSs. It also removes the limit of only seven devices that can have channel access within the CFP[1]. However, the scheme requires each requesting device to identify its flow characteristics. Also, how the coordinator enforces the order of devices to use the shared GTS remains unanswered, especially when the GTSs are of transmit type. These two drawbacks impose implementation difficulty for applications. In addition, within each GTS, some portions are still wasted because of fragmentation.

6.1.3 Solution

In applications, two requirements make the standard GTS allocation scheme bandwidth-unfriendly. One comes from the fact that the demand for bandwidth varies among the sensor nodes, namely in space domain. The other is that the demand for bandwidth from a sensor node varies from time to time, namely in time domain. Thus a static pre-configured GTS allocation scheme is not sufficient for the process with dynamical bandwidth requirements.

There are two kinds of approaches to handling the GTS requirements, called reactive approaches and active approaches. In a reactive approach, the PAN coordinator allocates GTS for a device according to the requirements defined in GTS request frame. It keeps monitoring the GTS usage by the device. If the usage is below an expected threshold, the coordinator reclaims the allocated GTS by actively transmitting GTS deallocation to the device. After that, the device does not own the GTS any more. The other is the active approach. The sensor node requests to deallocate GTS or requests to adjust the number of time slots in the GTS based on observation. The two approaches can improve the channel utilization by customizing to the needs from sensor nodes. The implementation of the two approaches can directly build a mechanism to adjust the number of GTSs through the interface of service requests at MAC sublayer.

[1] More than seven devices can be simultaneously in the state between GTS approval and GTS deallocation

The investigation of above mentioned approaches is left for future research. In this chapter, we try to solve the problem of bandwidth underutilization from another prospect. This is based on the observation that the GTS allocation scheme in the standard is rather coarse. At most times, allocated GTSs are more than the desired needs from the sensor nodes. In the next section, the detail of our revised GTS allocation scheme is presented.

6.2 Revised GTS allocation scheme

The standard GTS allocation scheme is based on the basic unit - time slot. As shown in Figure 22, the first GTS consists of two time slots and the second GTS consists of three time slots. Generally speaking, the guaranteed bandwidth available in the allocated GTS should be no less than the traffic generating rate of the requesting node. When the traffic rate is less than the available bandwidth, a portion of the GTS will be idle. This will become an unused hole in the CFP. Most probably, the sum of the total unused holes can satisfy a GTS needed by another node, but none of them alone is long enough. Thus, a portion of the CFP is wasted without servicing any traffic.

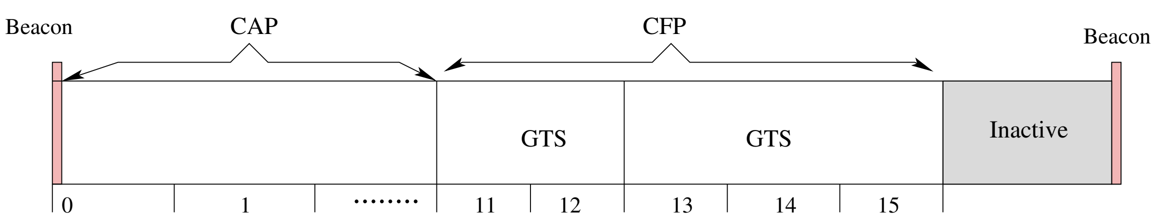


Figure 22. The standard GTS allocation scheme

Based on the observation of the source for bandwidth waste, the CFP can be divided into much smaller slots rather than using the same unit as the CAP. This is especially meaningful when the parameter *SO* is large. The smaller the basic slot is, the larger the allocated slots are addressed. For example, if the CFP is divided into 256 slots, eight bits are required to represent the starting slot, and another eight bits to represent the length of the allocated GTS. In this case, we would need to change the structure of the GTS descriptor. The standard structure of a GTS descriptor is shown in Figure 23. It uses four

bits to address the starting slot for a GTS and another four bits to represent the length of the GTS. Therefore, a GTS can be represented with the (startingSlot, length) pair. For example, the two GTSs in Figure 22 are $(1011, 0010)_b$ and $(1101, 0011)_b$ respectively. This is because the active period is divided into 16 time slots in total. However, if we look back the structure of beacon frame, it is easy to find the *last CAP slot* defined. The rest of the active period is reserved for CFP. Hence, there exists information redundancy. This provides us an opportunity to extend the division of the CFP into smaller slots through the redundant information.

Bits: 0–15	16–19	20–23
Device short address	GTS starting slot	GTS length

Figure 23. Format of the GTS descriptor

As mentioned before, division of the CFP into smaller time slots requires longer bit sequences to address the allocation information. However, we are in fact able to keep the format of the GTS descriptor intact. In this situation, we can divide the CFP into 16 equally sized slots. Figure 24 shows the new GTS allocation scheme. The merit of this scheme is that we do not need to change the packet structure. The only change required is that all PAN devices and the coordinator interpret the GTS allocation in another way.

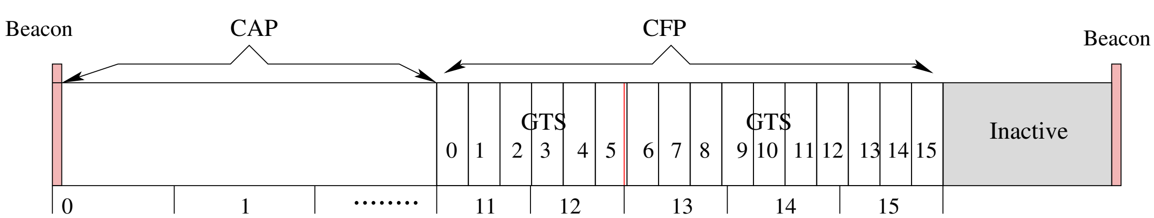


Figure 24. The proposed GTS allocation scheme

An intuitive example is shown in Figure 25. The colored areas indicate the GTS usage. In the standard GTS allocation scheme shown in Figure 25(a), the first GTS consists of slot 11, and slot 12; the second GTS consists of slot 13, slot 14, and slot 15. However, they are not fully utilized. The bandwidth waste is caused by the coarse time division.

In the new GTS allocation scheme in subfigure 25(b), the CFP is divided into 16 equally sized time slots. The first GTS consists of slot 0, 1, 2, and 3. The (startingSlot, length) pair in its GTS descriptor is $(0000, 0100)_b$. The second GTS consists of slot 4, 5, 6, 7, 8, 9, 10, 11. And the (startingSlot, length) pair is $(0100, 1000)_b$ in its GTS descriptor. The remaining of the CFP, the slots 12, 13, 14 and 15, can satisfy the need from another device with a comparable traffic generating rate of the first device. Thus the bandwidth usage is improved significantly. In addition, the accommodation of more GTSs in the same length of CFP increases the length of CAP. This will decrease the number of collisions implicitly, which results in the energy savings in the entire PAN. Alternatively, without the GTS request from another device, the PAN coordinator can specify 12 to the *last CAP* in the beacon frame, and let CFP start from slot 13.

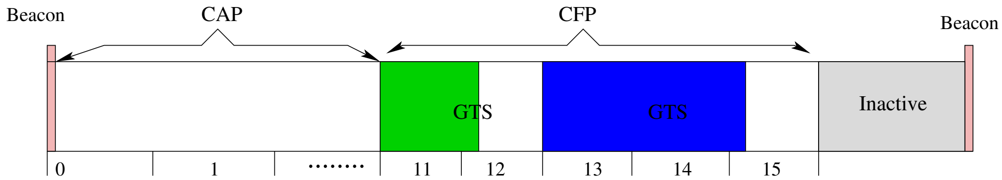
Our scheme is different from the work in [35] in that we try to accommodate more traffic flow within the CFP in a superframe duration, while they intend to share the GTS among multiple devices with a round-robin scheduler. The advantage of our scheme over [35] is that no delay occurs due to deploying the proposed scheme.

During the deployment of the proposed scheme, the allocation of GTSs may still leave a portion of CFP unused. Three cases exist depending on the duration of the leftover in the CFP:

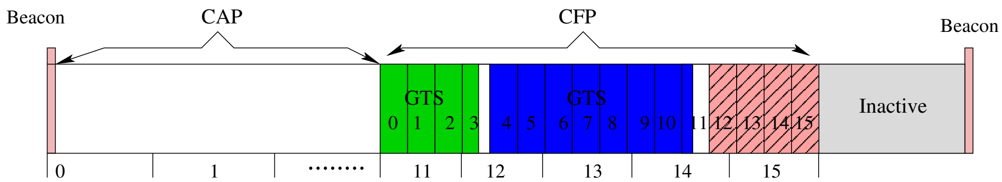
1. The remainder of the CFP fits the request of another device or some devices. It means that no single time slot is left in the CFP.

2. The remainder of the CFP is not larger than an integer times of T_s, and is not requested or not enough for other devices. T_s is the time duration of one standard time slot in a superframe period.

3. The remainder of the CFP is larger than an integer times of T_s, a portion of the remaining CFP is released to the CAP. The remaining is not requested or not enough for other devices. This case can be converted into Case 2 or Case 1. Practically, the



(a) the standard GTS allocation and their usage



(b) Proposed method of GTS allocation and their usage

Figure 25. The comparison of GTS allocation and usage

case can be avoided through requirement analysis. The discussion appears in Section 6.4.

To simplify the evaluation and the discussion, without losing generalization, we assume that the remainder of the CFP is not longer than the duration of a T_s.

6.3 Performance evaluation

In order to facilitate the discussion and evaluation, we define some variables. Let k_i be the number of time slots in a GTS allocated to the sensor node n_i. r_i denotes the average data generation at the node n_i. t_i denotes the time for the node n_i to transmit data in its GTS. m is the number of devices being allocated a GTS each. T_s is the time duration of one standard time slot in a superframe period. T'_s is the time duration of one new time slot in the CFP. The t_i can be expressed as follows:

$$t_i = \frac{r_i BI}{C} \tag{26}$$

, where BI is the time duration of a beacon interval and C is the data rate equal to 250 kbps. The duration of BI is shown in Equation (27).

$$BI = aBaseSuperframeDuration * 2^{BO} \tag{27}$$

And the t_i satisfies the following condition:

$$(k_i - 1)T_s < t_i \leq k_i T_s \tag{28}$$

The bandwidth utilization of this GTS allocation is defined as:

$$U_{k_i TS} = \frac{t_i}{k_i T_s} \tag{29}$$

The average bandwidth utilization of the CFP is defined as [2]:

$$\begin{aligned} U_{CFP} &= \frac{\sum_{i=1}^{m} t_i}{\sum_{i=1}^{m} k_i T_s} \\ &= \frac{\sum_{i=1}^{m} t_i}{T_s \sum_{i=1}^{m} k_i} \end{aligned} \tag{30}$$

Replacing the T_s, t_i, and BI as the Equation (18) (26)(27), we have

$$U_{CFP} = \frac{2^{BO-SO+4} \sum_{i=1}^{m} r_i}{C \sum_{i=1}^{m} k_i} \tag{31}$$

Note, under the standard GTS allocation scheme, it does not allow the accommodation of any other flows in the idle portion. Therefore, the average bandwidth utilization of the GTSs:

$$U_{GTSs} = U_{CFP} \tag{32}$$

In the new GTS allocation scheme, the T'_s can be expressed as:

$$T'_s = \frac{\sum_{i=1}^{m} k_i T_s}{16} \tag{33}$$

The number of new slots allocated in each GTS is

$$N'_i = \left\lceil \frac{t_i}{T'_s} \right\rceil \tag{34}$$

[2]Our definition is different from [35], where it is defined as $\frac{1}{m} \sum_1^m U_{k_i TS}$

And the total number of new slots allocated is

$$
\begin{aligned}
N' &= \sum_{i=1}^{m} N'_i \\
&= \sum_{i=1}^{m} \left\lceil \frac{t_i}{T'_s} \right\rceil
\end{aligned} \tag{35}
$$

And the time span of the remaining slots in the CFP is $(16 - N')T'_s$. If $(16 - N')T'_s$ is larger or equal to pT_s, the extra p time slot can be released to CAP for the common use.

$$
p = \left\lfloor \frac{(16 - N')T'_s}{T_s} \right\rfloor \tag{36}
$$

The bandwidth utilization of the allocated GTS for each device is

$$
U_{n_i} = \frac{t_i}{N'_i T'_s} \tag{37}
$$

, and the average of the bandwidth utilization in GTSs is

$$
U_{GTSs} = \frac{\sum_{i=1}^{m} t_i}{N' T'_s} \tag{38}
$$

Replacing the N', T'_s, t_i, T_s and BI as expressed in Equation (35) (33) (26) (18) and (27), we have the average bandwidth utilization:

$$
\begin{aligned}
U_{GTSs} &= \frac{16BI \sum_{i=1}^{m} r_i}{C \sum_{i=1}^{m} k_i T_s \sum_{i=1}^{m} \left\lceil \frac{16 r_i BI}{C \sum_{i=1}^{m} k_i T_s} \right\rceil} \\
&= \frac{2^{BO-SO+8} \sum_{i=1}^{m} r_i}{C \sum_{i=1}^{m} k_i \sum_{i=1}^{m} \left\lceil \frac{2^{BO-SO+8} r_i}{C \sum_{i=1}^{m} k_i} \right\rceil}
\end{aligned} \tag{39}
$$

In order to derive the average of the bandwidth utilization of the whole CFP, we assume p devices apply for GTSs in the new scheme, where $p >= m$. The total number of new slots allocated is

$$
N' = \sum_{i=1}^{p} \left\lceil \frac{t_i}{T'_s} \right\rceil \tag{40}
$$

And the condition $N' <= 16$ holds because there only exists 16 time slots.

The average of bandwidth utilization in the whole CFP is

$$U_{CFP} = \frac{\sum_{i=1}^{p} t_i}{\sum_{i=1}^{m} k_i T_s} \tag{41}$$

Replacing the t_i with Equation (26), we have the average bandwidth utilization:

$$U_{CFP} = \frac{2^{BO-SO+4} \sum_{i=1}^{p} r_i}{C \sum_{i=1}^{m} k_i} \tag{42}$$

In order to see the improvement of the proposed GTS allocation scheme, we assume SO = BO = 0, $r_1 = 18kbps$, and $r_2 = 36kbps$. The two devices will be allocated 2 and 3 GTS slots respectively in the standard allocation scheme because the condition (28) is satisfied. The bandwidth utilization depends on the traffic generating rate of the third device. The bandwidth utilization is shown in Figure 26. The channel utilization increases as the traffic from the third device grows because it fills the idle slots.

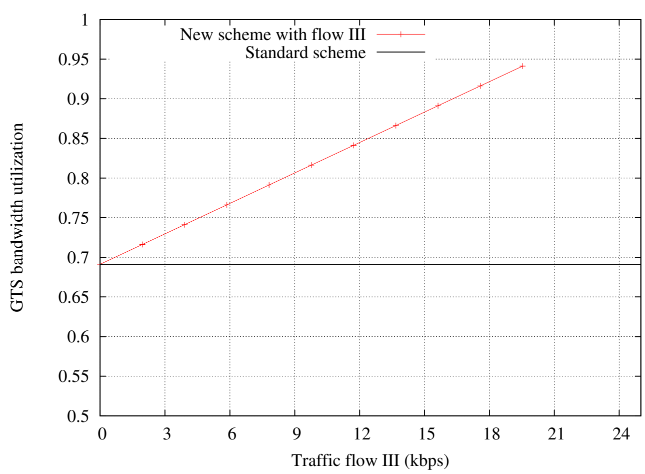


Figure 26. GTS bandwidth utilization

The comparison of the impact of two traffic flows on the bandwidth utilization in the allocated GTSs is shown in Figure 27. We assume SO = BO = 0. The traffic flow rate of r_1 is among (15.625, 31.25] kbps, which is required to transmit within 2 time slots in a GTS. The traffic flow rate of r_2 is among (31.25, 46.875] kbps and 3 time slots are needed in a GTS. The green and red grid represent the bandwidth utilization in the standard GTS allocation scheme and new allocation scheme respectively. From the figure, we can see that

the new proposed allocation scheme improves the bandwidth utilization in almost all cases. For example, when r_1 is 17.1875 kbps and r_2 is 32.8125 kbps, the bandwidth utilization in the proposed GTS allocation scheme is 93.1%, which is higher than 45% in the case of the standard GTS scheme. The exception happens when the traffic rate of each flows approach the bandwidth limit of the GTS. For example, in the case that r_1 is 15.625 kbps and r_2 is 46.875 kbps, the standard scheme still allocates GTSs with 2 and 3 slots for them respectively. However, in the new allocation scheme, due to the problem of aligning, the 5 standard time slots duration is not enough. This will result in the coordinator having to allocate one more standard slot in the CFP in order to accommodate the traffic needs. Thus, the bandwidth utilization in the proposed scheme is a bit lower than that in the standard scheme for this particular case.

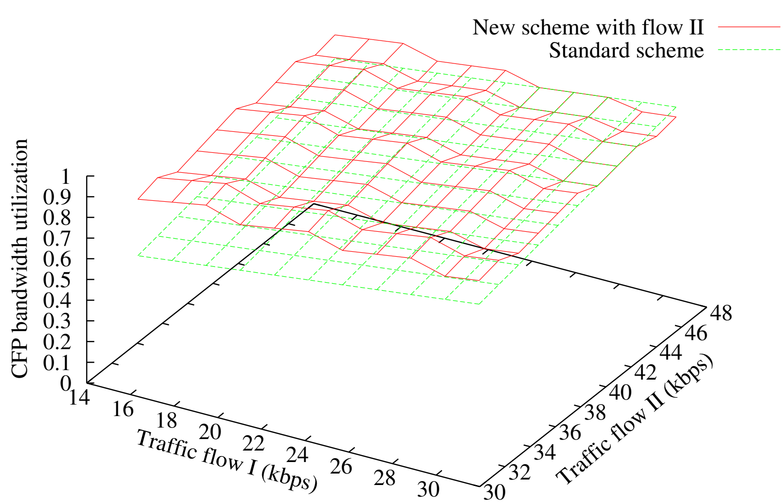


Figure 27. GTS bandwidth utilization

To see the impact of traffic flows on the bandwidth utilization cross the whole CFP, we use another simpler example because in previous case adding additional flow to the CFP resulting in a new dimension in the figure, which is difficult to see the results. We assume SO = BO = 0. And in the standard scheme only traffic I exists in the CFP. The flow rate r_1 of traffic I is among (15.625, 31.25] kbps, which requires 2 standard time slots

in a GTS. The flow rate r_2 of traffic II is among (0, 15.625] kbps and 1 standard time slot is needed in a GTS. The CFP consists of two standard time slots. The standard GTS scheme can only accept traffic I while the proposed scheme can accept both of them as long as N' is not larger than 16. The result is shown in Figure 28. We can see that the proposed new allocation scheme improves the bandwidth utilization in all cases where the reminder of CFP can accommodate traffic II. This improvement is due to that the proposed scheme enables the two flows to use the two standard time lots sized CFP, which is not possible using the standard GTS allocation mechanism. Moreover, with the granulated GTS allocation scheme, the length of the CFP is reduced significantly, thus increasing the CAP. This implicitly results in energy savings in an entire PAN because fewer transmission collision occurs in the CAP. Without traffic II, the bandwidth utilization is the same as that in the standard scheme.

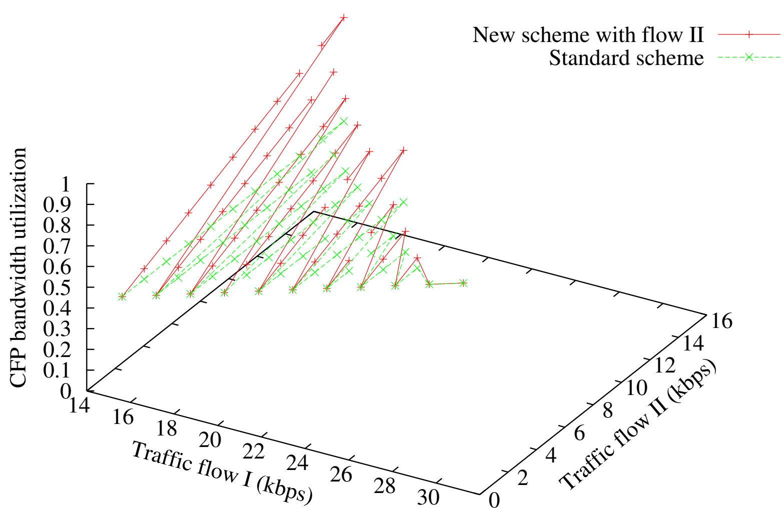


Figure 28. CFP bandwidth utilization

6.4 Implementation approach

This subsection presents some practical implementation considerations. From the description of the proposed GTS allocation scheme, we can see that the new scheme does not impose any changes to the existing frame structures and the communication protocol. Rather, the coordinator and requesting devices interpret the GTS descriptor in a new way. In order to achieve this, at the PAN device side, an algorithm is deployed to evaluate how many time slots are needed, and the requirements are enclosed in the frame *GTSReq*. The coordinator has to run an algorithm to check whether the remaining CFP is to satisfy the requirements of a device. The layering architecture incorporating the proposed scheme is shown in Figure 29. The new allocation scheme should be implemented at a higher layer above MAC. It computes the number of time slots required according to Equation (34), and fills the number and the rest of others in the *GTSCharacteristics*. The *GTSCharacteristics* is passed to the GTS request primitive *MLME_GTS* available in MLME_SAP.

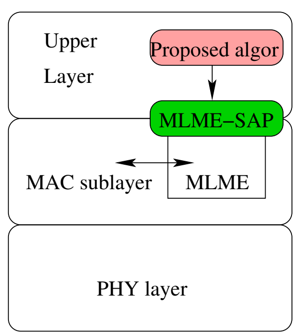


Figure 29. Laying architecture

The algorithm used by PAN devices is as Algorithm 3. The bandwidth requirement may be viewed as the outer bound of the traffic rate. As usual, the PAN devices have to periodically track the superframe beacon in order to synchronize to the PAN coordinator, and get the parameters such as *SO*, *BO*, and *Last CAP*, etc.

The PAN coordinator allocates GTS based on the requirements of a GTS request and the current available capacity/resources. The GTS may be revoked or changed at any time at the discretion of the PAN coordinator. The corresponding algorithm of the proposed

Algorithm 3 GTS allocation scheme running at the device

```
Requested to apply for a GTS by higher layer
if No CFP exists in superframe period,(determining this through Last CAP) then
    Requesting device sends a request, the requirement is based on the number of standard time
    slots.
else
    Requesting device sends a request based on the current new time lots.
end if
Receive a GTS descriptor and decode it
if Allocated more slots than requested then
    Interpret the numbers as the proposed slots in the CFP
else
    Interpret the numbers as standard slots in the CFP
end if
```

scheme is shown in Algorithm 4. Compared with the Algorithm 3, this is computation-intensive. As mentioned in the 802.15.4 standard [3], a PAN coordinator is a specific FFD with more powerful computation capability and resources. Thus, it is not an issue to deploy the proposed algorithm in the coordinator.

Algorithm 4 GTS allocation scheme running at the coordinator

```
Receive a GTS request
if No CFP exists in a superframe period then
    Allocates the time slots according to the requirements from devices based on the number of
    standard time slots.
else
    if The remaining CFP is enough to the requirement of the requesting devices then
        Allocate the time slots for the requesting devices
    else
        Increase the number of slots for the CFP
        Adjust the number of slots for the devices if needed,
        Allocate the time slots for the requesting devices.
        Update GTS descriptors through next beacon frame
    end if
end if
```

In order to improve the performance, The PAN coordinator may monitor the usage of each GTS. If a device always leaves a portion of GTS unused, but it does not meet the conditions of GTS expiration[3], the PAN coordinator may reallocate the number of time

[3]For a transmit GTS and a receive GTS, a data frame and an acknowledgement frame is not received from the device in the GTS at least every $2n$ superframes respectively. n is 2^{8-BO} for $0 \leq BO \leq 8$ and 1 otherwise.

slots in the GTS and make necessary rearrangements to other GTSs caused by the changes. Thus it may improve the GTS bandwidth utilization. How to implement this is out of the scope of this chapter.

6.5 Performance comparison study using Network Calculus

This section presents a performance comparison study of the existing three GTS allocation schemes using network calculus theory.

Network calculus is a theory of deterministic queueing systems found in computer networks [51]. It provides deep insights into the study and understanding of the flow problem in networks. With this theory, we perform delay bound analysis for the three GTS allocation schemes: the standard GTS allocation scheme, i-GAME, and ours.

There are two important concepts defined in network calculus theory. The first one is arrival curve, which models the traffic sent by the sources. The second is service curve, which models how fast the system handles the incoming traffic.

For integrated service networks (ATM or the integrated services internet), some limits exist on the arrival curve and the service curve in order to provide guaranteed services. For a given data flow with a cumulative arrival function $R(t)$, the assumption is as follows:

- There exists an arrival curve $\alpha(t)$ that upper bounds $R(t)$ such that $\forall s,\ 0 \le s \le t$, $R(t) - R(s) \le \alpha(t-s)$. This inequality means that the amount of traffic that arrives to receive service in any interval $[s, t]$ never exceeds $\alpha(t-s)$.

- There exists a minimum service curve $\beta(t)$ guaranteed to $R(t)$.

If we view the GTS as the service that the PAN coordinator provides to a set of devices, it is easy to see that network calculus can be applied to study the performance of GTS. Here, we show a simple example to demonstrate how network calculus is used to investigate the delay bound and system buffer. Consider a flow constrained by one leaky bucket, thus with an arrival curve of the form $\alpha = \gamma_{r,b}$, served in a node with the service curve guarantee $\beta_{R,T}$.

As shown in Figure 30, the delay bound refers to the maximum horizon distance between the arrival curve and the service curve. That is $T + b/R$ in this example. The maximum vertical distance between the two curves represents the buffer bound, which is $b + r * T$. This occurs only if $r \leq R$. Otherwise, the bounds become infinite.

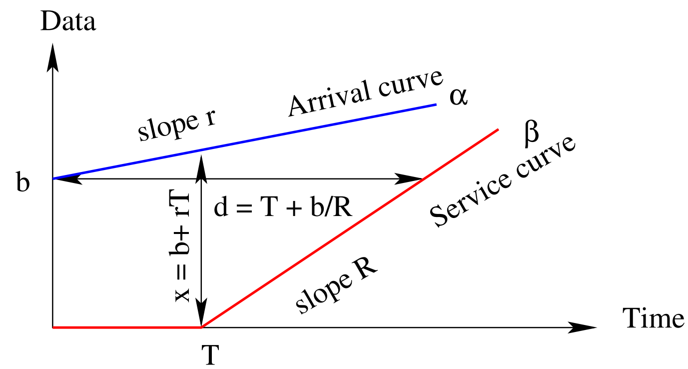


Figure 30. An example for network calculus

In the performance study, as in [34], the (b, r) model is used to describe the traffic from devices. It presents the data flow with a cumulative arrival function $R(t)$ upper bounded by the linear arrival curve $\alpha(t) = b + r * t$, where b is the burst tolerance and r is the rate. The work in [49] has proved that this model is valid in typical WSN applications.

Since each GTS provides service to a specific device in a discrete manner, separated by the CAP, the service curve of a GTS appears as a stair. As shown in Figure 31, the one-time slot GTS is partially used because of the required IFS between frames. T is the maximum latency that a burst may wait for a service. This latency occurs for a burst that arrives just after the end of the GTS. The delay bound is D_{max}^{stair}. However, the service curve is rather complicated in terms of expression. Therefore, it is approximated by a simple rate-latency curve as in [34]. In Figure 31, the approximation service curve is $\beta_{R,T} = R_{Ts}(t - T)^{+}$, where R_{Ts} is the available bandwidth of the GTS. The delay bound is approximated correspondingly to D_{max}.

Our comparison focuses on the following evaluation matrics: delay bound, bandwidth utilization, and implementation complexity. Without losing generalization, we assume the length of the CFP is one standard time slot. In order to facilitate the remaining discussions,

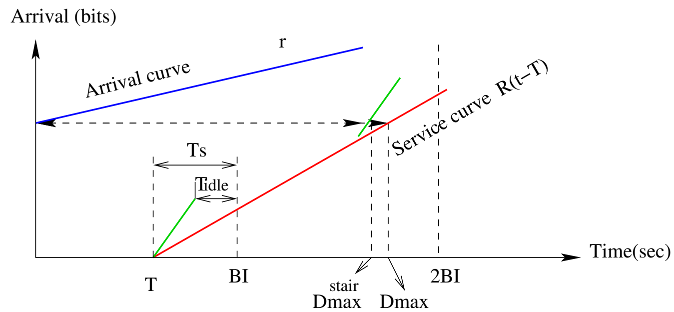


Figure 31. The GTS service curves

we assume there are N devices to share the one standard time slot GTS. And the arrival curve of each device is $\alpha(t) = b + r * t$. This assumption is for fair comparison purpose only because the i-GAME requires devices to share similar arrival rates. The limitation of the three allocation schemes will be discussed later.

Since our scheme divides the CFP into 16 slots, each device is allocated $\lceil \frac{16}{N} \rceil$ number of slots. Note, this is for analysis only. In our scheme, the number of slots allocated is based on the requirement of a particular device. The available bandwidth for each device is

$$R_i = \frac{\lceil \frac{16}{N} \rceil}{16} \cdot R_{Ts} \tag{43}$$

The delay bound of our allocation scheme is as follows:

$$\begin{aligned} D_{i,max} &= \frac{b}{R_i} + (BI - \frac{\lceil \frac{16}{N} \rceil \cdot T_s}{16}) \\ &= \frac{16b}{\lceil \frac{16}{N} \rceil \cdot R_{Ts}} + (BI - \frac{\lceil \frac{16}{N} \rceil \cdot T_s}{16}) \end{aligned} \tag{44}$$

The delay bounds of the standard scheme and i-GAME scheme were derived in [35]. The delay bound of the standard scheme is

$$D_{i,max} = \frac{b}{R_{Ts}} + (BI - T_s) \tag{45}$$

Note, the standard scheme does not allow sharing the GTS among multiple devices. The delay bound here is thus not related to the number of devices. The delay bound of the i-GAME scheme is

$$D_{i,max} = \frac{N \cdot b}{R_{Ts}} + (N \cdot BI - T_s) \tag{46}$$

The bandwidth utilization of our scheme is as shown in Equation (42). Since we assume the devices share the one-time standard time slot CFP, the bandwidth utilization is

$$U_{CFP} = \frac{2^{BO-SO+4} \sum_{i=1}^{N} r_i}{C} \tag{47}$$

, and the bandwidth utilization of i-GAME is the same as ours.

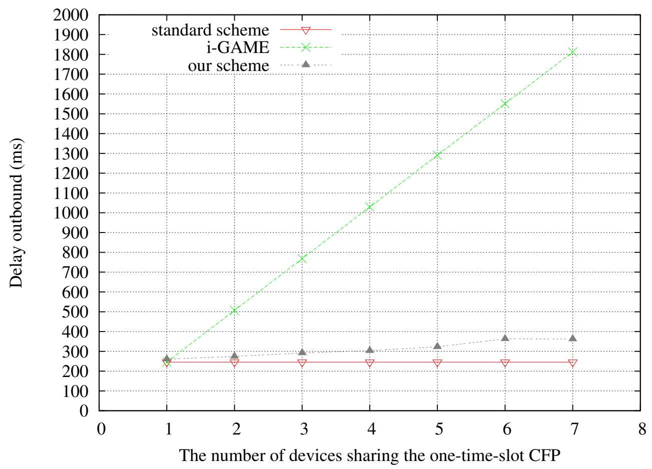


Figure 32. Delay bound of the three GTS allocation schemes

The bandwidth of the standard GTS allocation scheme is expressed as the average of GTS utilization among the devices.

$$U_{CFP} = \frac{1}{N} \sum_{i=1}^{N} \frac{2^{BO-SO+4} r_i}{C} \tag{48}$$

Since the flow rates for the devices are the same, the bandwidth utilization can be expressed as:

$$\begin{aligned} U_{CFP} &= \frac{2^{BO-SO+4} r_i}{C} \\ &= \frac{2^{BO-SO+4} r}{C} \end{aligned} \tag{49}$$

The burst size (b) for all flows is 200 bits, and the flow arrival rate (r) is 1.8 kps. We consider a PAN with the BO = SO = 4. We choose the $BO = SO > 3$ because in our scheme the one standard time slot CFP is impossible to accommodate the flows from up to seven devices due to protocol overheads occurred at physical layer and MAC sublayer. Figure

32 and 33 show the delay bound and bandwidth utilization of the three GTS allocation schemes. It is observed that our proposed scheme achieves the same improvement as the i-GAME scheme over the standard GTS allocation scheme. Additionally, the delay bound is much smaller than that of i-GAME.

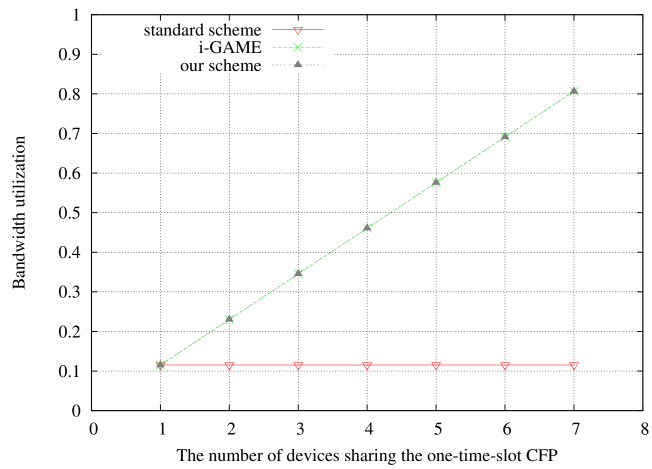


Figure 33. Bandwidth utilization of the three GTS allocation schemes

Table 4 shows the overall comparison of the existing three GTS allocation schemes in terms of several evaluation metrics. From the implementation prospect, the standard GTS allocation is the simplest, while it has the lowest bandwidth utilization. The difficulty of deploying i-GAME scheme is how to enforce the round-robin scheduling among the devices. This problem is not addressed in [35]. Although it may be solved by maintaining a counter at each requested devices, the i-GAME scheme is still error-prone subject to the loss of tracking a superframe beacon. The advantage of i-GAME is that it does not have the limitation of the number of devices to share a GTS. Theoretically, it allows as many devices to use a GTS if the requirements of bandwidth and delay bound are satisfied. However, how to distribute the GTS descriptor to these devices is not answered in [35]. In contrast, our scheme only permits up to seven devices to share the CFP because the GTS descriptor in the beacon frame remains intact. For a best performance, i-GAME requires the flows to share a similar pattern. Our proposed scheme does not have such a restriction. In fact, our scheme takes the advantage of the slot alignment and is more likely to accept more flows

in the same length of the CFP.

Table 4. Comparison of three GTS allocation schemes

Scheme	Delay bound	Bandwidth utilization	Implementation complexity	Sharing	Condition of sharing
Standard	small	low	easy	no	-
i-GAME	large	high	difficult	yes &unlimited	similar flow pattern
Ours	medium	high	easy	yes & limited to 7	no condition

6.6 Conclusion

We have presented a new GTS allocation scheme without introducing any changes in the frame format in order to improve the bandwidth utilization. The formulas for the number of new GTS slots and bandwidth utilization are derived. Evaluation results show that the proposed GTS scheme can improve bandwidth utilization significantly over the standard GTS scheme, thus increasing the length of CAP. Therefore, much energy savings can be achieved implicitly in an entire PAN. In addition, we have proposed the algorithms for deployment from the prospect of implementation. The performance comparison study shows that our proposed scheme has the merit of better performance in terms of delay bound, bandwidth utilization and easy implementation among the three GTS allocation schemes: the standard scheme, i-GAME, and ours.

Further work includes to design an add-on algorithm that allows the PAN coordinator to adjust the number of time slots in GTSs according to processes/applications with different dynamics. This will make the proposed GTS scheme work on-demand in the time domain. The problem of channel utilization can be further investigated with different traffic models.

CHAPTER 7

CONCLUSION

The objective of this research is to study the problem of energy efficiency in IEEE 802.15.4 MAC protocol, and design energy efficient schemes to improve its performance. More specifically, we examine the GTS scheme, varying from the GTS descriptor structure, the descriptor distribution scheme, to the GTS allocation mechanism. An energy efficient GTS descriptor distribution scheme is proposed, with redundant beacon information stripped. To further reduce the energy consumption caused by inappropriate transmission methods, we design a new channel reservation to transmit GTS allocation and pending addresses through multicast rather than broadcast. In addition, in order to improve the channel utilization in the CFP, the duration of the CFP is divided into 16 time slots rather than the standard time slot. The time slots can be addressed and allocated with the exactly same format of the standard GTS descriptor.

In this chapter, we summarize the research results.

7.1 802.15.4 simulation module in GTNetS

Currently there are two 802.15.4 simulators available in the public domain [42] [43]. Both simulators are based on widely accepted simulator – *Network Simulator 2* (NS2) [45]. However, they either implement only a subset of 802.15.4 features or do not provide an energy model for this power sensitive protocol. In addition, the choice of NS2 inherits some deficiencies that have been exposed over these years, such as a long learning curve, extensive memory requirements and limited scalability. Considering all these factors, we instead implemented a 802.15.4 simulation module in GTNetS as a part of this research. It is well known that modeling detail has impact on the accuracy or even correctness in wireless network simulations [44]. Thus, in the simulation module, 802.15.4 MAC and PHY are modeled elaborately so that the simulations conducted with the tool could be

as accurate as possible. Also, the data of the commercial 802.15.4 RF transceivers are incorporated in the tool.

The implementation of 802.15.4 MAC and physical layer functionality takes the advantages of *GTNetS* and inherits the layered design architecture. On the other hand, our implementation of 802.15.4 simulation module closely follows the specification and description language (SDL) description in Annex D of [3]. Meanwhile, for some discrepancy found between standard body and the SDL description, we chose to conform to the former. Some omissions in the SDL are also filled. The detail of this simulator is described in [36], and the simulator will be released for public use in coming months.

This part of work has resulted in the paper *"IEEE 802.15.4 Simulation Module in Network Simulator GTNetS"* appeared in proceedings of IEEE 63rd Vehicular Technology Conference (2006) [36].

7.2 Energy-efficient beacon structure and GTS descriptors distribution scheme

An energy-efficient beacon structure and GTS descriptors distribution scheme have been proposed. Its focus is on avoiding transfer of redundant information in beacons. Specifically, it includes the following aspects:

Modified beacon structure The standard beacon structure always contains at least 1 byte *GTS specification* all the time regardless of whether valid GTS information is included. This causes unnecessary energy drain at all devices in a PAN. Therefore, a reserved bit in superframe specification is used to designate whether the *GTS specification* is included. And the *GTS specification* field is an option and carried in beacons only when necessary.

GTS descriptors distribution The approval of a GTS request – GTS descriptor is transmitted within beacon frames. The 802.15.4 standard defines a PAN coordinator mechanically to send the descriptor within four superframe beacon. This is another source

of energy drain. To solve this problem, an ACK-based scheme is proposed. An acknowledged GTS descriptor would be excluded in remaining superframe beacons. For devices that request *receive* type GTS, they return an ACK frame at the beginning of their allocated slots. For devices that request *transmit* type GTS and have data to transmit, they transmit data in their allocated slots. This implicitly notifies the coordinator of the reception of GTS descriptors. Otherwise, they return an acknowledgement frame in their allocated slots. The proposed scheme is energy-efficient, and thus extends the lifetime of a PAN. It has significant impacts on a PAN when many devices are required to share the very limited number of GTSs. Also it could be used in GTS deallocation. Besides the merit of energy efficiency, it provides reliability scheme for transferring descriptors. If necessary, the PAN coordinator can distribute descriptors in more beacons.

This part of work has resulted in the paper *"Energy efficiency of different data transmission methods in IEEE 802.15.4: study and improvement"* appeared in the proceedings of International Symposium on Wireless Pervasive Computing (ISWPC 2007) [39].

7.3 Efficient channel reservation for multicasting GTS allocation and pending addresses

It is observed that each tracking device still receives a copy of each GTS descriptor although the above proposed mechanism is deployed. This problem is because of the method of distribution, broadcast. Since these information – GTS descriptors and pending addresses – are meaningful to only a small set of devices, the use of broadcast obviously results in extra energy drain. Therefore, a efficient channel reservation mechanism is proposed. It includes:

New GTSPA frame and modified beacon frame The information that is required to transmit through multicast is extracted from beacon structure. So the modified beacon frame contains only necessary information to access the CAP. The GTS descriptor and pending addresses are included in a new frame type, called *GTSPA* frame.

Allocate the slots at the end of CAP according to the requirements To avoid overhearing at other devices, the reserved slots are specified at the end of CAP and before the CFP. The existence of the slots is indicated by a bit in *frame control* field. The devices that request GTSs or potentially have pending transactions should turn on their receivers in these special reserved time slots. Other devices may turn off their transceivers during the slots. The number of slots reserved for multicasting depends on the packet size. In the case of no pending addresses and GTS descriptors, the slots are deallocated by the PAN coordinator and become a part of CAP. The detailed analysis is shown in [46].

This part of work has resulted in the paper *"Efficient channel reservation for multicasting GTS allocation and pending addresses in IEEE 802.15.4"* appeared in proceedings of International Conference on Wireless and Mobile Communications (ICWMC 2007) [46]. An enhanced version of this paper has been accepted by Journal of Communications (JCM).

7.4 GTS allocation revisited

The standard GTS allocation scheme is based on the basic unit - a standard time slot, which is equal to $1/16$ of a superframe duration. Requesting devices are allocated to a GTS consisting of an integer number of such a time slot. This results in bandwidth waste if the requirements from them are below the available bandwidth. Unlike the existing work in [35], where devices share a GTS by a round-robin scheduler, the CFP is equally divided into 16 smaller time slots. And the coordinator and requesting devices can allocate and use the bandwidth resource more closely to their needs. Hence, the same length of time duration can satisfy the requirements of more PAN devices. Bandwidth utilization is improved substantially. The merit of the proposed GTS allocation scheme is that it does not introduce any extra changes to the existing frame structure. The only change is that the PAN coordinator and devices interpret the GTS descriptor in a way different from the standard. This enables our approach easily implementable in off-the-shelf platforms.

This part of work has resulted in the following papers [52] [53] [54]:

- *"A new GTS allocation scheme for IEEE 802.15.4 networks with improved bandwidth utilization"*, to appear in 7th International Symposium on Communications and Information Technologies (ISCIT 2007).

- *"The GTS allocation scheme revisited"*, IET Electronics Letters, Vol.43, No. 18, pp.1005-1006, 2007.

- *"A performance comparison study of GTS allocation schemes in IEEE 802.15.4"*, to appear in IEEE 2007 International Workshop on Wireless Ad Hoc Mesh and Sensor Networks (Wamsnet-07).

Future work includes implementation of the proposed energy efficient mechanisms in 802.15.4 RF chips, and employing real experiments to validate the proposed algorithms and test their performance in terms of power consumption, throughput, delay, and implementation complexity. If the budget allows, we will buy some TelosB Motes for this purpose [41]. With the Motes' experiments, some other interesting issues will be investigated as well. For example, it is usually assumed GTS data transmission is reliable, with 100% delivery ratio. Is it true? Another issue is how a PAN coordinator deals with pending transactions when the owner of pending transactions dies out of battery.

Currently the completed research restricts the study in star topology like most of existing research works. In the star topology, devices in a PAN always can reach the coordinator in one hop distance. However, in a large WSN not all devices are able to communicate with the coordinator directly. Rather, they depend on some intermediate devices (routers) to forward messages. In this situation, several small PANs form into a large cluster tree network. Some interesting problems arise in this network topology. For example, what kind of shape of the cluster tree leads to the best network performance. What is the distance between the devices in order to achieve the best performance?

PUBLICATIONS

- Liang Cheng, Anu G. Bourgeois, Bo Yu, "Power Management in Wireless Ad Hoc Networks," *1st ACIS International Workshop on Self–Assembling Wireless Networks (SAWN),* May 2005.
- Liang Cheng, Xin Zhang, Anu G. Bourgeois, "IEEE 802.15.4 Simulation Module in Network Simulator GTNetS," *IEEE 63rd Vehicular Technology Conference (VTC Spring),* May 2006.
- Liang Cheng, Anu G. Bourgeois, "Energy efficiency of different data transmission methods in IEEE 802.15.4: study and improvement," *International Symposium on Wireless Pervasive Computing (ISWPC)*, 2007.
- Liang Cheng, Anu G. Bourgeois, "Efficient channel reservation for multicasting GTS allocation and pending addresses in IEEE 802.15.4," *International Conference on Wireless and Mobile Communications (ICWMC)*, 2007.
- Liang Cheng, Anu. G. Bourgeois, Xin Zhang, "A new GTS allocation scheme for IEEE 802.15.4 networks with improved bandwidth utilization", to appear in *IEEE 7th International Symposium on Communications and Information Technologies (ISCIT)*, 2007.
- Liang Cheng, Xin Zhang, Anu. G. Bourgeois, "The GTS allocation scheme revisited", IET Electronics Letters, Vol.43, No.18, pp. 1005-1006.
- Liang Cheng, Anu. G. Bourgeois, Xin Zhang, "A performance comparison study of GTS allocation schemes in IEEE 802.15.4", to appear in *IEEE International Workshop on Wireless Ad Hoc, Mesh and Sensor Networks (WAMSNet)*, 2007.

REFERENCES

[1] Feng Zhao and Leonidas Guibas, *Wireless Sensor Networks: An Information Processing Approach.* San Francisco, CA: Morgan Kaufmann Publishers, 2004.

[2] MAC Alphabet Soup, available on HTTP: https://apstwo.st.ewi.tudelft.nl/~koen/MACsoup.

[3] IEEE, IEEE 802.15.4 Standard Specification for Low-Rate Wireless Personal Area Networks (LR-WPANs), May 2003.

[4] Jose A. Gutierrez, Marco Naeve, Ed Callaway, Monique Bourgeois, Vinary Mitter and Bob Heile, "IEEE 802.15.4: a developing standard for low-power low-cost wireless personal area Networks", IEEE Network Magazine, vol.15 no.5, Sept/Oct 2001, pp.12-19.

[5] Ed Callaway, Venkat Bahl, Paul Gorday, Jose A. Gutierrez,Lance Hester, Marco Naeve and Robert Heile ,"Home networking with IEEE 802.15.4: a developing standard for low-rate wireless personal area networks", IEEE Communication Magazine, special issue on Home Networking, vol.40 no.8, Aug 2002, pp.70-77.

[6] Wei Ye, John Heidemann, and Deborah Estrin, "An energy efficient MAC protocol for wireless sensor networks", in *Proc. of IEEE Infocom*, vol.3, July 2002, pp.1567-1576.

[7] Koen Langendoen and Gertjan Halkes, *Energy efficient medium access control*, book chapter in *Embedded Systems Handbook*, available on HTTP: http://www.st.ewi.tudelft.nl/~koen/papers/MAC-chapter.pdf

[8] IEEE, IEEE 802.11 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999.

[9] IEEE, IEEE 802.16 Standard for Wireless Local and Metropolitan Area Networks, 2004.

[10] Jason Hill and David Culler, "Mica: A wireless platform for deeply embedded networks", in *Proc. of IEEE Micro*, Nov/Dec 2002, pp.12-24.

[11] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks", in *IEEE ICC*, vol.5, April 2002, pp.3418-3423.

[12] A. El-Hoiydi, J.-D. Decotignie, C. Enz and E. Le Roux, "WiseMAC: an ultra low power MAC protocol for the WiseNET wireless sensor network", Poster, in *ACM Sensys*, 2003, pp.302-303.

[13] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks", in *Proc. of ACM Sensys*, 2003, pp.171-180.

[14] Bluetooth SG, "Bluetooth specification version 1.1", available HTTP: http://www.bluetooth.org

[15] IEEE, 802.15.3 Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specs for High Rate Wireless Personal Area Networks, 2003.

[16] IEEE, 802.15.1 Wireless MAC and PHY Specifications for Wireless Personal Area Networks (WPAN)s , 2002.

[17] Gang Lu, Bhaskar Krishnamachari and Cauligi Raghavendra, "An adaptive energy efficient and low latency MAC for data gathering in sensor networks", In *Proc. of the 4th International Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks (WMAN)*, April 2004, pp.224-232.

[18] Jianliang Zheng, Myung J. Lee, "Will IEEE 802.15.4 make ubiquitous networking a reality?– a discussion on a potential low power, low bit rate standard", IEEE Communications Magazine, Vol.42, No.6, June 2004, pp. 140-146.

[19] Jose A. Gutierrez, Edgar H. Callaway, Raymond L. Barrett, *Low-rate wireless personal area networks – enabling wireless sensors with IEEE 802.15.4*, Standards Information Network IEEE Press, 2003.

[20] Jianliang Zheng and Myung J. Lee, *A comprehensive performance study of IEEE 802.15.4*, Sensor Network Operations, IEEE Press, Wiley Interscience, Chapter 4, pp. 218-237.

[21] Gang Lu, Bhaskar Krishnamachari and Cauligi Raghavendra, "Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks", in *Proc. of IEEE Workshop on Energy-Efficient Wireless Communications and Networks (EWCN)*, 2004, pp.701-706.

[22] Nicholas F. Timmons and William G. Scanlon, "Analysis of the performance of IEEE 802.15.4 for medical sensor body area networking", in *Proc. of 1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, 2004.

[23] Jin-Shyan Lee, "An experiment on performance study of IEEE 802.15.4 wireless networks", in *Proc. of IEEE International Conference on Emerging Technologies and Factory Automation*, Sept 2005, pp.451-458.

[24] Nada Golmie and David Cypher, *Performance analysis of low rate wireless technologies for medical applications*, Addison-Wesley, ISBN 0-321-24726-4.

[25] Jelena Misic, Shairmina Shafi and Vojislav B. Misic, "Performance of a beacon enabled IEEE 802.15.4 cluster with downlink and uplink traffic", in *IEEE Trans. on Parallel and Distributed Systems*, Vol. 17, No. 4, April 2006

[26] Jelena Misic, Shairmina Shafi and Vojislav B. Misic, "The impact of MAC parameters on the performance of 802.15.4 PAN, Ad Hoc Networks", Vol.3, No.5, Data Communication and Topology Control in Ad Hoc Networks, September 2005, pp. 509-528.

[27] Jelena Misic, Vojislav B. Misic and Shairmina Shafi, “Performance of IEEE 802.15.4 beacon enabled PAN with uplink transmissions in non-saturation mode - access delay for finite buffers”, in *Proc. of first International Conference on Broadband Networks*, Oct 2004, pp. 416-425.

[28] Ivan Howitt, Rogelo Neto, Jing Wang, and James M. Conrad, “Extended energy model for the low rate WPAN”, in *Proc. of Second IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, Nov 2005.

[29] Tae Rim Park, Tae Hyun Kim, Jae Young Choi, Sunghyun Choi, and Wook Hyun Kwon, “Throughput and energy consumption analysis of IEEE 802.15.4 slotted CSMA-CA”, in *IEE Electronics Letters*, Vol. 41, issue 18, Sept 2005.

[30] Ai-Chun Pang and Hsueh-Wen Tseng, “Dynamic backoff for wireless personal networks”, in *Proc. of IEEE Global Telecommunications Conference (Globecom)*, Nov/Dec 2004, pp.1580-1584.

[31] Bruno Bougard, Francky Catthoor, Denis C. Daly, Anantha Chandrakasan, and Wim Dehaene, “Energy efficiency of the IEEE 802.15.4 standard in dense wireless microsensor networks: modelling and improvement perspectives”, in *Proc. of the Conference on Design, Automation and Test in Europe*, 2005, pp.196-201.

[32] Mario Neugebauer, Jorn Plonnigs, and Klaus Kabitzsch, “A new beacon order adaptation algorithm for IEEE 802.15.4 networks”, in *Proc. of the 2nd European Workshop on Wireless Sensor Networks (EWSN)*, Jan 2005, pp.302-311.

[33] Mario Neugebauer and Klaus Kabitzsch, “A new protocol for a low power sensor network”, in *Proc. of the 23rd IEEE International Performance, Computing and Communicaions Conference (IPCCC)*, 2004, pp.393-399.

[34] Anis Koubaa, Mario Alves, and Eduardo Tovar, “GTS allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks”, in *Proc. of the 14th International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS)*, Apr. 2006.

[35] Anis Koubaa, Mario Alves, and Eduardo Tovar, “i-GAME: an implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks”, in *Proc. of 18th Euromicro Conference on Real-Time Systems*, Jul 2006.

[36] Liang Cheng, Xin Zhang, and Anu G. Bourgeois, “IEEE 802.15.4 simulation module in network simulator GTNetS”, in *Proc. of IEEE 63rd Vehicular Technology Conference (VTC)*, May 2006, pp.1308-1312.

[37] George F. Riley, “The Georgia Tech Network Simulator”, in *Proc. of ACM SIGCOMM Workshop on Models,Methods, and Tools for Reproducible Network Research*, 2003, pp.5-12.

[38] Abraham Silberschatz, Greg Gagne and Peter Baer Galvin, “Operating System Concepts”, 7 Edition, John Wiley & Sons. Inc, 2005.

[39] Liang Cheng and Anu G. Bourgeois, "Energy efficiency of different data transmission methods in IEEE 802.15.4: study and improvement", in *Proc. of International Symposium on Wireless Pervasive Computing (ISWPC)*, Feb 2007, pp.395-400.

[40] Data Sheet for CC2420 2.4GHz IEEE 802.15.4 RF transceiver, available HTTP: http://www.chipcon.com/files/CC2420_Data_Sheet_1_3.pdf

[41] Crossbow Technology Inc, TelosB Mote Research Platform, information online HTTP: http://www.xbow.com

[42] Jianliang Zheng and Myung J. Lee, Available HTTP: http://ees2cy.engr.ccny.cuny.edu /zheng/pub/

[43] Gang Lu, Bhaskar Krishnamachari and Cauligi Raghavendra, Available HTTP: http://ceng.usc.edu/ anrg/downloads/ns-pan.tar.gz

[44] John Heidemann, Nirupama Balusu, Jeremy Elson, Chalermek Intanagonwiwat, Kunchan Lan, Ya Xu, Wei Ye, Deborah Estrin, Ramesh Govindan, "Effects of detail in wireless network simulation", in *Proc. of the SCS Multiconference on Distributed Simulation*, Jan 2001, pp.3-11.

[45] Kevin Fall and Kannan Varadhan, "The *ns* manual", Available HTTP: http://www.isi.edu/nsnam/

[46] Liang Cheng and Anu G. Bourgeois, "Efficient channel reservation for multicasting GTS allocation and pending addresses in IEEE 802.15.4," in *Proc. of International Conference on Wireless and Mobile Communications (ICWMC)*, 2007, pp.46-51.

[47] Ivan Howitt and Jing Wang, "Energy efficient power control policies for the low rate WPAN", in *Proc. of Sensor and Ad Hoc Communications and Networks (SECON)*, Oct 2004, pp.527-536.

[48] Diba Mirza, Maryam Owrang and Curt Schurgers, "Energy-efficient wakeup scheduling for maximizing lifetime of IEEE 802.15.4 networks, in *Proc. of the First International Conference on Wireless Internet (WICON)*, 2005, pp.130-137.

[49] Anis Koubaa and Ye-Qiong SONG, "Evaluation and improvement of response time bounds for real-time applications under non-preemptive fixed priority scheduling", in *International Journal of Production and Research*, Vol.42, No.14, July 2004, pp.2899-2913.

[50] Jan Flora and Philippe Bonnet, "Never mind the standard: Here is the TinyOS 802.15.4 stack", available http://www.diku.dk /publikationer /tekniske.rapporter /rapporter/06-10.pdf

[51] Jean-Yves Le Boudec and Patrick Thiran, Network calculus - A Theory of Deterministic Queuing Systems for the Internet, available online at http://infoscience.epfl.ch/getfile.py?recid=282&mode=best

[52] Liang Cheng, Anu. G. Bourgeois, Xin Zhang, “The GTS allocation scheme revisited”, to appear in *IEEE 7th International Symposium on Communications and Information Technologies (ISCIT)*, 2007.

[53] Liang Cheng, Xin Zhang, Anu. G. Bourgeois, “The GTS allocation scheme revisited”, IET Electronics Letters, Vol.43, No. 18, pp. 1005-1006, 2007.

[54] Liang Cheng, Anu. G. Bourgeois, Xin Zhang, “A performance comparison study of GTS allocation schemes in IEEE 802.15.4”, to appear in *IEEE International Workshop on Wireless Ad Hoc, Mesh and Sensor Networks (WAMSNet)*, 2007.

[55] Marina Petrova, Janne Riihijarvi, Petri Mahonen and Saverio Labella, “Performance study of IEEE 802.15.4 using measurements and simulations”, in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC)*, April 2006, pp.487-492.

[56] Zhijia Chen, Chuang Lin, Hao Wen, Hao Yin, “An analytical model for evaluating IEEE 802.15.4 CSMA/CA protocol in low-rate wireless application,” in *Proc. of 21st International Conference on Advanced Information Networking and Application Workshops (AINAW)*, 2007, pp. 899-904.

[57] Raj Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*, Wiley- Interscience, New York, NY, April 1991