KTH Electrical Engineering

Modeling, Analysis, and Design of Wireless Sensor Network Protocols

PANGUN PARK

Doctoral Thesis
Stockholm, Sweden 2011

KTH School of Electrical Engineering
Automatic Control Lab
SE-100 44 Stockholm
SWEDEN

TRITA-EE 2011:001
ISSN 1653-5146
ISBN 978-91-7415-836-6

Akademisk avhandling som med tillstånd av Kungliga Tekniska högskolan framlägges till offentlig granskning för avläggande av teknologie doktorsexamen i telekommunikation tisdagen den 4 Mars 2011 klockan 10.15 i sal F3 Kungliga Tekniska högskolan, Lindstedtsvägen 26, Stockholm.

Tryck: Universitetsservice US AB

Abstract

Wireless sensor networks (WSNs) have a tremendous potential to improve the efficiency of many systems, for instance, in building automation and process control. Unfortunately, the current technology does not offer guaranteed energy efficiency and reliability for closed-loop stability. The main contribution of this thesis is to provide a modeling, analysis, and design framework for WSN protocols used in control applications. The protocols are designed to minimize the energy consumption of the network, while meeting reliability and delay requirements from the application layer. The design relies on the analytical modeling of the protocol behavior.

First, modeling of the slotted random access scheme of the IEEE 802.15.4 medium access control (MAC) is investigated. For this protocol, which is commonly employed in WSN applications, a Markov chain model is used to derive the analytical expressions of reliability, delay, and energy consumption. By using this model, an adaptive IEEE 802.15.4 MAC protocol is proposed. The protocol design is based on a constrained optimization problem where the objective function is the energy consumption of the network, subject to constraints on reliability and packet delay. The protocol is implemented and experimentally evaluated on a test-bed. Experimental results show that the proposed algorithm satisfies reliability and delay requirements while ensuring a longer lifetime of the network under both stationary and transient network conditions.

Second, modeling and analysis of a hybrid IEEE 802.15.4 MAC combining the advantages of a random access with contention with a time division multiple access (TDMA) without contention are presented. A Markov chain is used to model the stochastic behavior of random access and the deterministic behavior of TDMA. The model is validated by both theoretical analysis and Monte Carlo simulations. Using this new model, the network performance in terms of reliability, average packet delay, average queueing delay, and throughput is evaluated. It is shown that the probability density function of the number of received packets per superframe follows a Poisson distribution. Furthermore, it is determined under which conditions the time slot allocation mechanism of the IEEE 802.15.4 MAC is stable.

Third, a new protocol for control applications, denoted Breath, is proposed where sensor nodes transmit information via multi-hop routing to a sink node. The protocol is based on the modeling of randomized routing, MAC, and duty-cycling. Analytical and experimental results show that Breath meets reliability and delay requirements while exhibiting a nearly uniform distribution of the work load. The Breath protocol has been implemented and experimentally evaluated on a test-bed.

Finally, it is shown how the proposed WSN protocols can be used in control applications. A co-design between communication and control application layers is studied by considering a constrained optimization problem, for which the objective function is the energy consumption of the network and the constraints are the reliability and delay derived from the control cost. It is shown that the optimal traffic load when either the communication throughput or control cost are optimized is similar.

Acknowledgements

First of all I would like to thank my supervisor Professor Karl Henrik Johansson. I appreciate his guidance and support not only my research but also my life. After four years of his supervision, his impressive leadership becomes a big milestone in my life. I owe my gratitude to my co-supervisor Assistant Professor Carlo Fischione, who had many discussions and gave valuable comments on my research direction.

I am indebted to the coauthors of several papers included in this thesis. The coauthors are Jose Araujo, Dr. Yassine Ariba, Dr. Alvise Bonivento, Dr. Corentin Briat, Tekn. Lic. Piergiuseppe Di Marco, Assistant Professor Sinem Coleri Ergen, Professor Mikael Johansson, Assistant Professor Henrik Sandberg, Professor Alberto Sangiovanni-Vincentelli, Dr. Pablo Soldati, and Associate Professor Emmanuel Witrant. A special thanks to Dr. Adam Dunkels and Professor Mikael Skoglund for being my reference group. I am very pleased with their productive comments for my research. I am also particularly grateful to Dr. Jim Weimer, who read and commented the thesis. I would like to thank to our research engineers and Master students, Aitor Hernandez, Yian Qin, and David Andreu who struggled to reduce the gap between theory and practice.

I appreciate to all fellow Ph.D. students and professors at the Automatic Control Group, and to Karin Karlsson Eklund, for making the supportive work environment. I would like to take the opportunity to thank Piergiuseppe Di Marco for all the interesting discussions we had about research as well as our life in Lappis apartment. He is one of best people that I have ever met in my life since he is the most patient man even though I annoyed him in many times. Specially, he corrects my cooking time of the Italian pasta, 20 min. Now, I can survive. A special thanks to Pablo Soldati for being good counsellor of my life as well as good research colleague in front of white board. I would like to thank the energizer of our lab, Jose Araujo who is always enthusiastic and gives his energy to others.

Thanks also to Chitrupa, Phoebus, Andre, Haibo, Assad, and all other people in the Automatic Control Lab. I will never forget a funny subset, Burak, Euhanna, and Zhenhua. In particular, I thank Euhanna who seated beside me and threw bad jokes btw 9am-10pm every day.

Thanks to all the friends I met here in Sweden. I am grateful to Aram Anto for our jogging in Lappis even though that works only during the summer. I would like to remember my old friend, Ali Nazmi Özyagci with his ponytail hair. I must thank another old friend, Dae-Ho, wise advisor and good comedian even though he

is bit talkative. A special memory for being my friends, Hyun-Sil and Seung-Yun.

A great thank to my family in South Korea, for supporting me in all the time. Most of all I would like to thank my parents for their continuous presence, support and encouragement. I would like to thank H.J., who gave me third eye to look at other side of the world. I must express my friends, Chan-Woo, Sun-Wook, Jin-Ho, and Gi-Bum who gave me great pleasure in Korea.

The research described in this thesis is supported by the EU project FeedNetBack, Swedish Research Council, Swedish Strategic Research Foundation, and Swedish Governmental Agency for Innovation Systems.

Pangun Park
Stockholm, January 2011.

Contents

Chapter 1

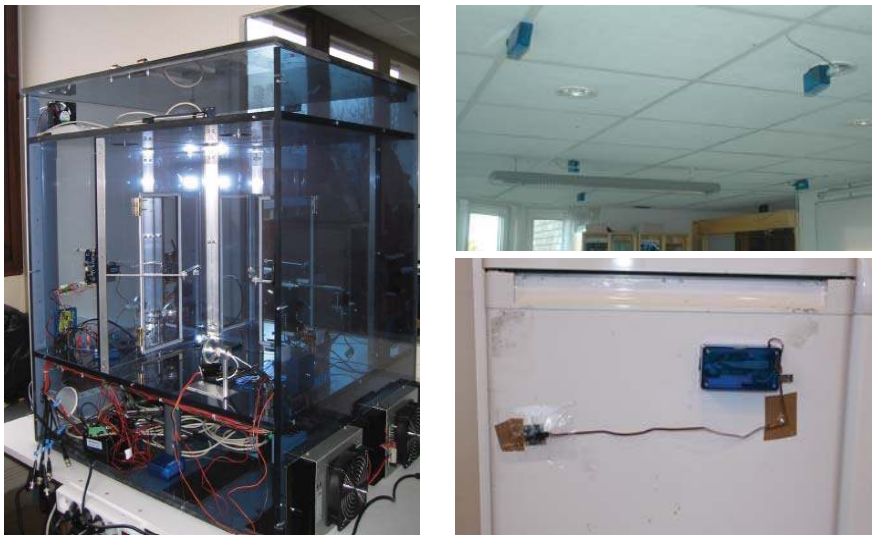
Introduction

Given the benefits offered by wireless sensor networks (WSNs) compared to wired networks, such as, simple deployment, low installation cost, lack of cabling, and high mobility, WSNs present an appealing technology as a smart infrastructure for building and factory automation, and process control applications [1, 2]. Emerson Process Management [3] estimates that WSNs enable cost savings of up to 90% compared to the deployment cost of wired field devices. Several market forecasts have recently predicted exponential growths in the sensor network market over the next few years, resulting in a multi-billion dollar market in the near future. ON World predicts that the emerging smart energy home market reaches 3 billion dollar in 2014 [4]. In particular, despite a challenging economy, ZigBee [5] annual unit sales have increased by 62% since 2007 and the market is on track to reach hundreds of millions of annual units within the next few years by over 350 global manufacturers [6]. Similarly, ABI research [7] predicts that in 2015 around 645 million 802.15.4 [8] chipsets will ship, compared to 10 million in 2009.

Although WSNs have a great potential for process, manufacturing and industrial applications, there is not yet a widespread use of WSNs. According to Gartner's Hype Cycles [9][1], WSNs are evolving very slowly into a mainstream adoption level. One of the fundamental reasons is that current technologies are not based on a design framework that is easy to use and applicable across several application domains. Today, each specific application development often requires expert knowledge over the stack: from the communication layer to application layer. This is evident for instance in the development of control systems based on WSNs. These systems are particularly challenging because they must support the right decision at the right moment despite any traffic condition, even in the presence of unexpected congestion, network failures or external manipulations of the environment. Furthermore, an energy efficient network operation is also a critical factor due to the limited battery lifetime of these sensors.

The main contribution of this thesis is to offer a framework for modeling, analysis,

[1] Gartner's Hype Cycles highlights the relative maturity of technologies across a wide range of IT domains, targeting different IT roles and responsibilities.



(a) UFAD test-bed [10]. (b) Smart home test-bed [11, 12].

Figure 1.1: Test-beds for building automation using WSNs.

and design of WSN protocols for control applications. The framework explicitly targets the need for a more efficient way to develop WSN applications. We especially focus on the minimization of the network energy consumption subject to constraints on reliability and delay. In addition, we propose how the communication protocol should adapt its variable parameters according to the traffic and channel conditions. The remainder of this chapter is organized as follows. In the next section, we motivate why WSNs are of interest through a couple of applications. In Section 1.2 we present challenges WSNs impose on control applications. Section 1.3 formulates the general mathematical problem used to design the protocols in this thesis. Finally, we present the contributions and an outline of the thesis. Symbols and acronyms used throughout the thesis are summarized in Appendix A.

1.1 Motivating Applications

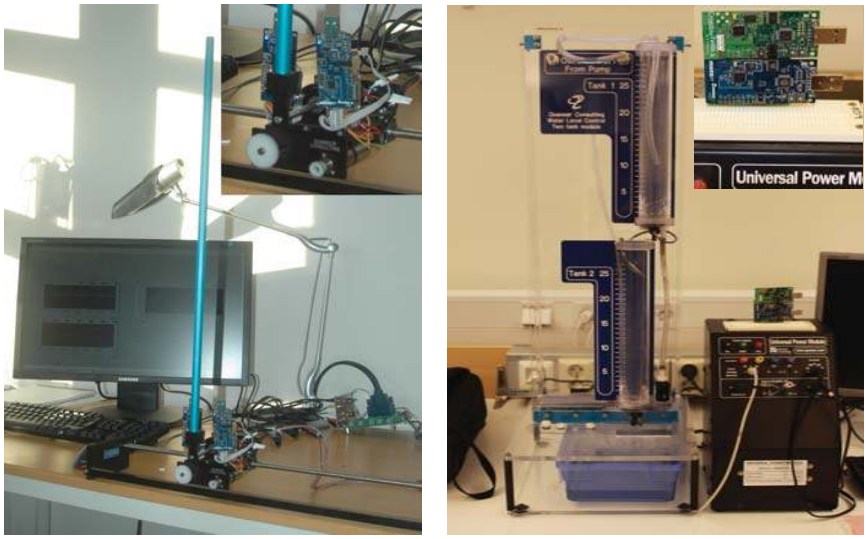
We consider here two scenarios where WSNs are used.

Building Automation

The European environment agency [13, 14] shows that the electricity and the water consumptions of buildings are about 30% and 43% of the total resource consumptions, respectively. The legislation in California (Title 24) [15], regarding energy efficiency of buildings, requires a certain amount of electricity demand management to be available. An ON World's survey [4] reports that 59% of 600 early adopters

Figure 1.2: Wireless control of froth flotation process at Boliden within the SOCRADES EU project (http://www.socrades.eu/).

in five continents are interested in new technologies that will help them better manage their energy, and 81% are willing to pay for energy management equipment if they could save up to 30% on their energy bill for smart energy home applications. In large scale contexts, the concept of intelligent green operation can be extended to urban districts, to form smart grids [16] as in the Stockholm Royal Seaport project [17]. Urban planners try to provide the solutions to minimize energy use and optimize waste management. The increase of energy efficiency of commercial buildings is one of the key drivers in the adoption of WSNs in building automation. Building automation covers all aspects of building system control including heating and air conditioning (HVAC), lighting control, and security systems. The low installation cost of mesh-based wireless systems allows the large retrofit market to be addressed as well as new constructions. An example of energy management systems using WSNs is the intelligent building ventilation control described in [10]. An underfloor air distribution (UFAD) indoor climate regulation process is set with the injection of a fresh airflow from the floor and an exhaust located at the ceiling level, as illustrated by the test-bed in Figure 1.1(a). The considered system is composed of ventilated rooms, fans, plenums, and a wireless network. It has been established that well-designed UFAD systems can reduce life-cycle building costs, improve thermal comfort, ventilation efficiency and indoor air quality, and conserve energy. Feedback regulation is a key element for an optimized system operation, achievable thanks to actuated diffusers and distributed measurements provided by the relatively low hardware and installation costs when using WSNs for communications in the ventilated area. Furthermore, the presence of a WSN in the building also permits run-time analysis of the performance and state of the UFAD units. Our smart home test-bed shown in Figure 1.1(b) monitors the electricity consumption of household devices, such as the microwave, dishwasher, and the coffee machine. The system also monitors the temperature change and provides early detection of



(a) Inverted pendulum control using WSNs [18].

(b) Coupled water tank control using WSNs [19].

Figure 1.3: Test-bed for process control using WSNs.

improperly functioning heating and cooling units. Infrared sensors count the number of people in each room. Information is fused and action is taken so that the heating can be lowered when many people enter a room, and lights can be switched off when there is no one in the room. Furthermore, additional energy is saved by catching inefficient unit operation early by monitoring the ventilation systems and water consumption using vibration sensors.

Process Control

Wireless communication can become a key technology in process control [20]. In comparison to traditional wired sensors, wireless sensors provide advantages in the manufacturing environment, such as an increased flexibility for locating and reconfiguring sensors, wire elimination in potentially hazardous locations, and easier network maintenance. Within the SOCRADES EU project, a wireless control system based on a IEEE 802.15.4 [8] network has been successfully developed for a froth flotation process at Boliden's plant in Sweden (see Figure 1.2).
To demonstrate and evaluate new wireless control solutions, we have developed a test-bed with several lab processes connected over a WSN. For example, we used an inverted pendulum (Figure 1.3(a)) and a coupled water tank (Figure 1.3(b)). For the inverted pendulum, the cart slides along a stainless steel shaft using linear bearings. The cart position is measured using a sensor coupled to the rack via an additional pinion. A pendulum mounted on the cart is free to fall along the

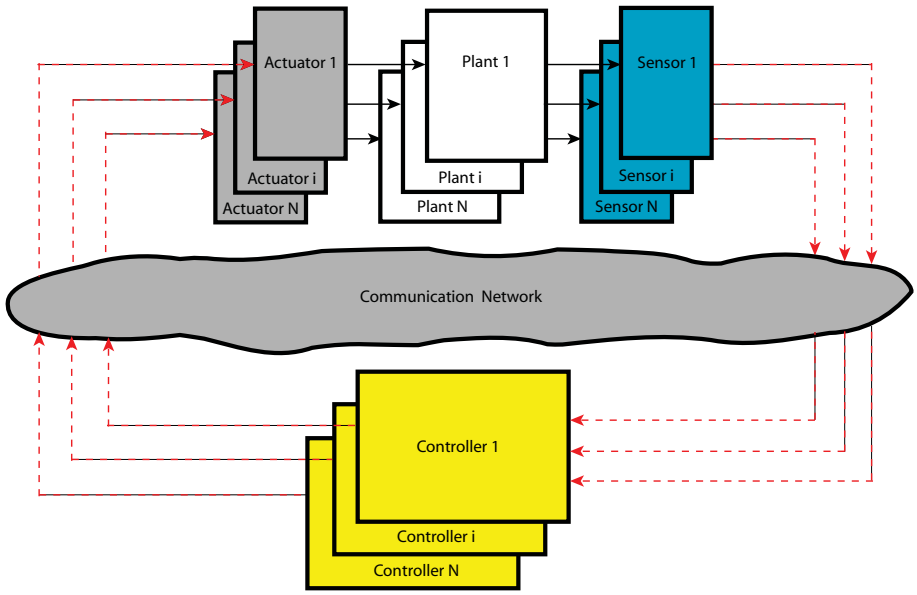


Figure 1.4: Overview of the networked control system. N plants are controlled by N controllers over a wireless network.

cart's axis of motion. The pendulum contends to transmit sensor measurements to the controller over a wireless network which induces packet losses and varying delays. The pendulum angle and cart position are measured using a potentiometer with wireless sensor nodes whose range is restricted by mechanical stops. Actuation commands are sent back to the cart motors over a WSN.
A coupled water tank system consists of a pump, a water basin and two tanks of uniform cross sections. The liquid in the lower tank flows to the water basin. A pump is responsible for pumping water from the water basin to the upper tank, which flows to the lower tank. The pressure sensors placed under each tank measure the water levels. The control loops regulate the coupled water tank systems where the tanks are co-located with the sensors and actuators and communicate wirelessly with a controller. One wireless sensor node interfaces the sensing channels with an ADC to sample the pressure sensor values for both tanks. The plant actuation is made through the DAC of the wireless sensor node to actuate the pump motor.

1.2 WSN Challenges in Control Applications

Figure 1.4 depicts the control architecture of networked closed-loop systems where multiple plants are controlled over a wireless network. Outputs of the plants are sampled at periodic or aperiodic intervals by the sensors and forwarded to the controller through a network. When the controller receives the measurements, a new

Safety	Class 0: Emergency action (always critical)
Control	Class 1: Closed loop regulatory control (often critical)
	Class 2: Closed loop supervisory control (usually non-critical)
	Class 3: Open loop control (human in the loop)
Monitoring	Class 4: Alerting Short-term operational consequence (e.g., event-based maintenance)
	Class 5: Logging & downloading/uploading No immediate operational consequence (e.g., history collection, SOE, preventive maintenance)

Table 1.5: ISA SP-100 defines application needs of industrial process by specifying usage class of WSN [20].

control command is computed. The control is forwarded to the actuator attached to the plant. The wireless network induces packet losses and varying delays. Hence, the network may cause stability problems for the closed-loop systems.
In Table 1.5, the industrial process are classified into three broad categories and six classes of WSN usage [20]. We remark that the importance of message timeliness increases as the class number decreases.
The protocol design for WSNs in control applications encounters more challenges than traditional WSN applications, namely:

- **Reliability**: Sensor readings must be sent to the sink of the network with a given probability of success, because missing sensor readings could prevent the correct execution of control actions or decisions. However, maximizing the reliability may increase the network energy consumption substantially [21]. Hence, the network designers need to consider the tradeoff between reliability and energy consumption.

- **Delay**: Sensor information must reach the sink within some deadline. Time delay is a very important QoS measurement since it influences performance and stability of control systems [22]. The delay jitter can be difficult to compensate for, especially if the delay variability is large. Hence, a probabilistic delay requirement must be considered instead of using average packet delay. Furthermore, the packet delay requirement is important since the retransmission of data packet to maximize the reliability may increase the delay. Outdated packets are generally not useful for control applications [23].

- **Energy Efficiency**: The lack of battery replacement, which is essential for affordable WSN deployment, requires energy-efficient operations. Since high reliability and low delay may require significant energy consumption, the re-

liability and delay must be flexible design parameters that still meet the requirements. Note that controllers can usually tolerate a certain degree of packet losses and delays [22]–[28]. Hence, the maximization of the reliability and minimization of the delay are not the optimal design strategies since these strategies will significantly decrease the network lifetime.

- **Sensor Traffic Patterns**: The type and amount of data to be transmitted is also important when considering control applications [22]. Control signals can be divided into two categories: real-time and event based. For real-time control, signals must be received within a specified deadline for correct operation of the system. In order to support real-time control, networks must be able to guarantee the delay of a signal within a specified time deadline. Hence, heavy traffic may be generated if sensors send data very frequently. Event-based control signals are used by the controller to make decisions but do not have a time deadline. The decision is taken if the system receives a signal or a timeout is reached. We remark here that some of the proposed protocol for environmental monitoring application, such as XMAC [29] and Fetch [30], operate in low traffic networks and can not handle the higher traffic loads of many control applications.

- **Adaptation**: The network operation should adapt to application requirement changes, time-varying wireless channels, and variations of the network topology. For instance, the set of application requirements may change dynamically and the communication protocol must adapt its parameters to satisfy the specific requests of the control actions. To support analytical model-based design instead of experience-based design, it is essential to have analytical models describing the relation between the protocol parameters and performance indicators (reliability, delay, energy consumption, etc).

- **Scalability**: Since the processing resources on WSN nodes are limited [31, 32], the calculations necessary to implement the protocol must be computationally light. These operations should be performed within the network, to avoid the burden of too much communication with a central coordinator. Therefore, the tradeoff between tractability and accuracy of the analytical model is very important. The protocol should also be able to adapt to variation in the network size, for example, size variations caused by the addition of new nodes.

As a consequence, the design of such networked control systems has to take into account a large number of factors that ensure correct implementation. Starting from these requirements, it is important to design an efficient communication protocol that satisfies the application requirements and optimizes the energy consumption of the network. Application requirements are a set of measurable service attributes imposed by the applications in terms of, for example, fairness, delay, jitter, available bandwidth, and packet loss. Figure 1.6 reports a typical example of the feasible control cost using the IEEE 802.15.4 protocol with respect to different sampling

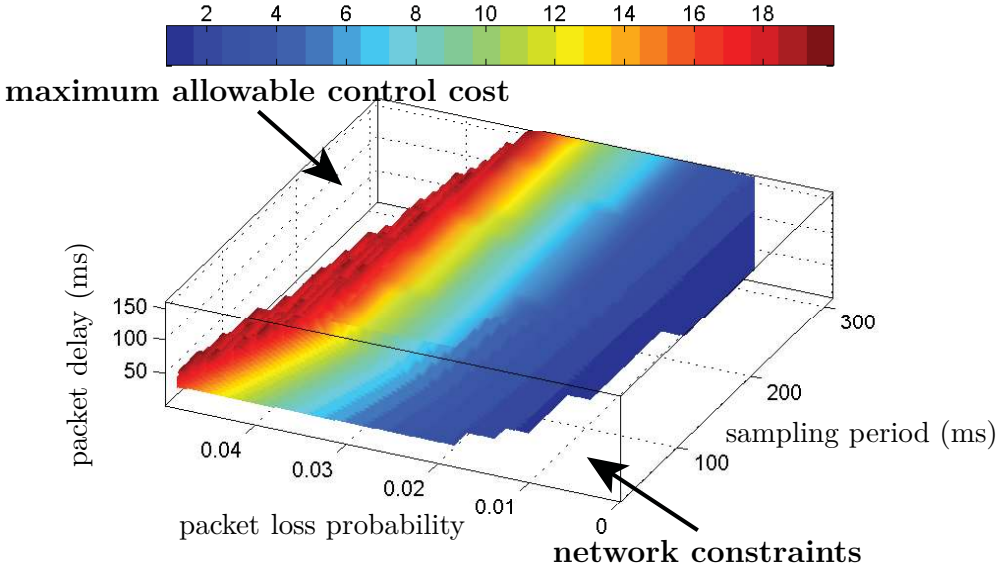


Figure 1.6: Achievable control cost over different sampling periods, packet loss probabilities, and packet delays of the IEEE 802.15.4 protocol. The colors indicate control cost.

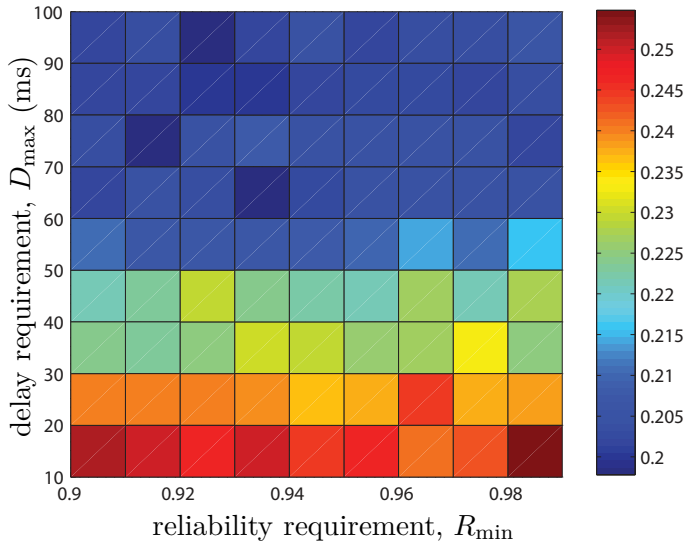


Figure 1.7: Power consumption of adaptive IEEE 802.15.4 with different reliability and average delay requirement.

periods, packet loss probabilities, and packet delays. The colors show the feasible control cost. A point is feasible if it satisfies a given maximum allowable control cost, packet loss probability, and delay for each sampling period. The feasible region is the set of all feasible points. In the figure, the transparent region denotes that the desired control cost is not feasible. It is natural that as the control requirement becomes more strict, the infeasible region increases. The performance of the wireless network affects the feasibility region of the control cost. Since short sampling periods increase the traffic load, the packet loss probability is closer to the critical value, above which the system is unstable. Hence, it is difficult to achieve a low packet loss probability when the sampling period is short. We remark that the infeasibility region due to the wireless network starts from the origin point where the continuous sampling, no packet loss, and no packet delay. The origin represents the most strict requirement for communication protocols. Therefore, no matter what communication protocol is used, the origin belongs to the infeasible region. The area and shape of the infeasibility region depends on the communication protocol. Additional details are discussed in Chapter 6.

Figure 1.7 reports a typical example of the power consumption of the network with various reliability and average delay requirements for adaptive IEEE 802.15.4 [33]. The colors indicates the average power consumption of the network. We clearly observe the tradeoff between the application requirements and power consumption of the network. Hence, the goal of the proposed design approach is to optimize the network behavior by considering the given constraints imposed by the application instead of just improving the reliability, delay, or energy efficiency without constraints. The objective function and requirements are used to solve a constrained optimization problem whose solution determines the policies and the parameters of the medium access control (MAC) and routing layer.

From the Figures 1.6 and 1.7, we remark that a tradeoff exists between control and communication performance. Traditional control design faces the problem of noisy feedback from the environment. Increasing the number of sensors may improve control performance, but at the risk of increasing network congestion and thus eventually leading to lossy and delayed control feedback. Similarly, decreasing the sampling period may not improve the control performance, but still increase the power consumption of the WSNs. Therefore, communication and control should be designed jointly. In this thesis, we offer a framework that embraces all the factors mentioned above.

1.3 Problem Formulation

The goal of this thesis is to model, analyze, and design WSN protocols. As part of this work, we will:

1. Model the important performance indicators, such as reliability, delay, energy consumption, using mathematical tools, and

2. Analyze the resulting performance of the protocol by means of the experiments and simulations.

By using the derive protocol model, we use a general constrained optimization problem for the designs. Our objective is to minimize the total energy consumption of each node or all nodes of the network, denoted by $E_{\text{tot}}(\mathbf{u})$ where $\mathbf{u}$ is a vector of decision variables. The application requirements impose constraints on the reliability and packet delay. Hence, the optimization problem is

$$\min_{\mathbf{u}} \quad E_{\text{tot}}(\mathbf{u}) \tag{1.1a}$$

$$\text{s.t.} \quad \mathbf{u} \in \mathcal{R} \cap \mathcal{D} \cap \mathcal{F}\,. \tag{1.1b}$$

The decision variables $\mathbf{u}$ are the protocol parameters of the physical layer (PHY), MAC, and routing layer. $\mathcal{R}$ and $\mathcal{D}$ are the feasible sets for the protocol parameters that meet the reliability and delay constraints, respectively. In addition, the feasible set $\mathcal{F}$ is due to physical layer properties of the hardware platform or limitations of the protocol standards. The derivation of analytical expressions of the energy consumption of the network, as well as reliability and delay for the packet delivery, is essential for the solution to the optimization problem. Therefore, the analytical modeling is a critical step to the protocol design in this thesis. Problem (1.1) is a mixed integer-real optimization problem, because $\mathbf{u}$ may take on both real and integer values. We model the components of Problem (1.1) and we derive a strategy to obtain its optimal solution, $\mathbf{u}^*$. As we will see later, the system complexity prevents us from deriving exact expressions for reliability, delay, and energy consumption. Approximations will be used to get tractable analytical models. Note that this constrained optimization problem can be local, in the sense that it is solved at a local node of the network using locally measurable information, or global, in the sense that includes information from the overall network and is solved centrally. Next, we give an example of a local optimization and an example of a global optimization, which are used in the thesis to design protocols.

Example 1

Chapter 3 presents a local optimization problem for IEEE 802.15.4 for reliable and timely communication. This protocol considers a star network topology with a personal area network coordinator, and N nodes with beacon-enabled slotted carrier sense multiple access/collision avoidance (CSMA/CA) and acknowledgements (ACKs). It minimizes the power consumption while meeting the reliability and delay constraints without any significant modifications of the IEEE 802.15.4 standard. Each node solves the optimization problem by estimating the channel condition, i.e., busy channel probability and channel accessing probability. The local constrained

optimization problem at node i is

$$\min_{\mathbf{u}_i} \quad E_{\text{tot},i}(\mathbf{u}) \tag{1.2a}$$

$$\text{s.t.} \quad \mathcal{R}_i = \{\mathbf{u}_i \,|\, R_i(\mathbf{u}) \geq R_{\min}\}\,, \tag{1.2b}$$

$$\mathcal{D}_i = \{\mathbf{u}_i \,|\, \Pr[D_i(\mathbf{u}) \leq D_{\max}] \geq \Omega\}\,, \tag{1.2c}$$

where $E_{\text{tot},i}$ is the energy consumption and $\mathcal{R}_i$, and $\mathcal{D}_i$ are the feasible sets for the protocol parameters that meet the reliability and delay constraints of node i, respectively. Note that the objective function and constraints are also functions of the decision variables of the other nodes in the network. The decision variables are the MAC parameters related to the backoff mechanism and the maximum number of retransmissions. A Markov chain model gives the analytical expressions of objective function and constraints of the local optimization problem. Each node updates its optimal protocol parameters by solving the local optimization problem. R_i is the reliability from node i to its receiver, and $R_{\min}$ is the minimum desired probability. D_i is a random variable describing the delay when transmitting a packet. $D_{\max}$ is the desired maximum delay, and Ω is the minimum probability with which such a maximum delay should be achieved. We remark that $D_{\max}$, Ω, and $R_{\min}$ are the application requirements, and $\mathbf{u}$ represents the protocol parameters. These parameters should be adapted to the traffic regime, wireless channel conditions, and application requirements for an efficient network.

Example 2

In Chapter 5, a global optimization problem is introduced to optimize the wake-up rate and the number of hops in the network. The cross-layer protocol solution, called Breath, is designed for industrial control applications where source nodes attached to the plant must transmit information via multi-hop routing to a sink. The protocol is based on randomized routing, MAC, and duty-cycling to minimize the energy consumption, while meeting reliability and packet delay constraints. The optimization problem is

$$\min_{\mathbf{u}} \quad E_{\text{tot}}(\mathbf{u}) \tag{1.3a}$$

$$\text{s.t.} \quad \mathcal{R} = \{\mathbf{u} \,|\, R(\mathbf{u}) \geq R_{\min}\}\,, \tag{1.3b}$$

$$\mathcal{D} = \{\mathbf{u} \,|\, \Pr[D(\mathbf{u}) \leq D_{\max}] \geq \Omega\}\,, \tag{1.3c}$$

where E_{tot} is the energy consumption, and $\mathcal{R}$ and $\mathcal{D}$ are the feasible sets for the protocol parameters that meet the reliability and delay constraints of the entire network, respectively. The decision variables are the wake-up rate and the number of hops, which are achieved by collaboration between the nodes in the network. The optimization problem is based on an analytical model for energy consumption, reliability, and delay of the network.

1.4 Thesis Outline and Contributions

In this section, we describe the outline and contribution of the thesis in more detail. The corresponding related works are presented in Chapter 2. The main contribution of the thesis is then given in four chapters. The material is organized as follows. Chapter 3 is on modeling and analysis of the random access scheme of the IEEE 802.15.4 protocol and applying adaptive protocol design. Chapter 4 is on modeling and analysis of the IEEE 802.15.4 hybrid protocol. Chapter 5 is on the cross-layer protocol solution, called Breath, by using an adaptive protocol design of WSNs. Chapter 6 is on control application using the proposed adaptive protocols. The outline of the thesis is as follows.

Chapter 3

This chapter presents an adaptive IEEE 802.15.4 protocol to support energy efficient, reliable and timely communications by tuning the MAC parameters of CSMA/CA algorithm. The protocol design scheme is grounded on a constrained optimization problem where the objective function is the power consumption of the network, subject to reliability and delay constraints on the packet delivery. A generalized Markov chain is proposed to model these relations by simple expressions without giving up the accuracy. The model is then used to derive an adaptive algorithm for minimizing the power consumption while guaranteeing reliability and delay constraints in the packet transmission. The algorithm does not require any modification of the IEEE 802.15.4 standard and can be easily implemented on network nodes. The protocol is experimentally implemented and evaluated on a test-bed with off-the-shelf wireless sensor nodes. Experimental results show that the analysis is accurate, that the proposed algorithm satisfies reliability and delay constraints, and that the approach ensures a longer lifetime of the network under both stationary and transient network conditions.
This chapter is based on the following publications:

- P. Park, P. Di Marco, C. Fischione, and K. H. Johansson , "Adaptive IEEE 802.15.4 Protocol for Reliable and Timely Communication", IEEE/ACM Transactions on Networking, 2010. Submitted.
- P. Park, C. Fischione, and K. H. Johansson, "Adaptive IEEE 802.15.4 protocol for energy efficient, reliable and timely communications", ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), Stockholm, Sweden, April, 2010.

Chapter 4

This chapter presents the novel modeling and analysis of the MAC protocol of IEEE 802.15.4 combining the advantages of a random access with contention with a time division multiple access (TDMA) without contention. The thesis focuses on

the IEEE 802.15.4 protocol, because it is becoming the most popular standard for low data rate and low power WSNs in many application domains. Understanding reliability, delay, and throughput is essential to characterize the fundamental limitations of this protocol and optimize its parameters. Nevertheless, there is not yet a clear understanding of the achievable performance of this hybrid MAC. The main challenge for an accurate analysis is the coexistence of the stochastic behavior of the random access and the deterministic behavior of the TDMA scheme. The Markov chains are used to model the contention access scheme and the behavior of the TDMA access scheme of the IEEE 802.15.4 protocol, which are validated by both theoretical analysis and Monte Carlo simulations. By using this new model, the network performance in terms of reliability, average packet delay, average queueing delay, and throughput is evaluated. It is also shown that the performance of the hybrid MAC differs significantly from what was reported previously in the literature. Furthermore, it is concluded that the tradeoff between throughput of the random access and the TDMA scheme is critical to maximize the throughput of the hybrid MAC.
The material presented in this chapter is based on the following publications:

- P. Park, C. Fischione, and K. H. Johansson , "Performance analysis of IEEE 802.15.4 Hybrid Medium Access Control Protocol", IEEE/ACM Transactions on Networking, 2010. Submitted.
- P. Park, C. Fischione, and K. H. Johansson, "Performance analysis of GTS allocation in Beacon enabled IEEE 802.15.4", IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), Rome, Italy, June, 2009.
- P. Park, P. Di Marco, P. Soldati, C. Fischione, and K. H. Johansson, "A Generalized Markov Chain Model For Effective Analysis of Slotted IEEE 802.15.4", IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Macau, P.R.C., October, 2009. Best Paper Award.

Chapter 5

In this chapter, a novel protocol Breath is proposed for control applications. Breath is designed for WSNs where nodes attached to plants must transmit information via multi-hop routing to a sink. Breath ensures a desired packet delivery and delay probabilities while minimizing the energy consumption of the network. The protocol is based on randomized routing, MAC, and duty-cycling jointly optimized for energy efficiency. The design approach relies on a constrained optimization problem, whereby the objective function is the energy consumption and the constraints are the packet reliability and delay. The challenging part is the modeling of the interactions among the layers by simple expressions of adequate accuracy, which are then used for the optimization by in-network processing. The optimal working point of the protocol is achieved by a simple algorithm, which adapts to traffic

variations and channel conditions with negligible overhead. The protocol has been implemented and experimentally evaluated on a test-bed with off-the-shelf wireless sensor nodes, and it has been compared with a standard IEEE 802.15.4 solution. Analytical and experimental results show that Breath is tunable and meets reliability and delay requirements. Breath exhibits a nearly uniform distribution of the working load, thus extending network lifetime.
This chapter is based on the following publications:

- P. Park, C. Fischione, A. Bonivento, K. H. Johansson, and A. Sangiovanni-Vincentelli, "Breath: a Self-Adapting Protocol for Reliable and Timely Data Transmission in Wireless Sensor Networks", IEEE Transactions on Mobile Computing, 2011. To appear.
- P. Park, C. Fischione, A. Bonivento, K. H. Johansson, and A. Sangiovanni Vincentelli, "Breath : a Self-Adapting Protocol for Wireless Sensor Networks in Control and Automation", IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), San Francisco, USA, June, 2008.

Chapter 6

In this chapter, we investigate how the design framework of WSNs applies to control applications. First, we show how the wireless network affects the performance of networked control systems by showing the feasible region of the control performance. It is shown that the optimal traffic load is similar when either the communication throughput or control cost are optimized. Second, a co-design between communication and control application layers is studied by considering a constrained optimization, for which the objective function is the energy consumption of the network and the constraints are the reliability and delay derived from the desired control cost. We illustrate the co-design through a numerical example.
This chapter is based on the following publication:

- P. Park, J. Araujo, and K. H. Johansson, "Wireless Networked Control System Co-Design", IEEE International Conference on Networking, Sensing and Control (ICNSC), 2011. To appear.

Chapter 7

We summarize the contributions of the thesis and discuss the possible future extensions.

Other Related Papers

The following publications, although not covered in this thesis, contain material that have influenced the thesis:

– **Investigations on IEEE 802.15.4:**

- P. Di Marco, P. Park, C. Fischione, and K. H. Johansson, "Analytical Modelling of Multi-hop IEEE 802.15.4 Networks", IEEE Transactions on Communications, 2010. Submitted.
- C. Fischione, P. Park, S. Coleri Ergen, K. H. Johansson, and A. Sangiovanni-Vincentelli, "Duty-cycling Analytical Modeling and Optimization in Unslotted IEEE 802.15.4 Wireless Sensor Networks", IEEE Transactions on Wireless Communications, 2010. Submitted.
- P. Di Marco, P. Park, C. Fischione, and K. H. Johansson, "Analytical Modelling of IEEE 802.15.4 for Multi-hop Networks with Heterogeneous Traffic and Hidden Terminals", IEEE Global Communications Conference (Globecom), Florida, USA, December, 2010.
- C. Fischione, S. Coleri Ergen, P. Park, K. H. Johansson, and A. Sangiovanni-Vincentelli, "Medium Access Control Analytical Modeling and Optimization in Unslotted IEEE 802.15.4 Wireless Sensor Networks", IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), Rome, Italy, June, 2009.

– **Cross-layer solutions:**

- C. Fischione, P. Park, P. Di Marco, and K. H. Johansson, "Design Principles of Wireless Sensor Networks Protocols for Control Applications", In S. K. Mazumder, editor, Wireless Networking Based Control, Springer, 2011.
- P. Di Marco, P. Park, C. Fischione, and K. H. Johansson, "TREnD: a timely, reliable, energy-efficient dynamic WSN protocol for control application", IEEE International Conference on Communications (ICC), Cape Town, South Africa, May, 2010.

– **Control applications using WSNs:**

- E. Witrant, P. Di Marco, P. Park, and C. Briat, "Limitations and Performances of Robust Control over WSN: UFAD Control in Intelligent Buildings", IMA Journal of Mathematical Control and Information, November, 2010.
- E. Witrant, P. Park, and M. Johansson, "Time-delay estimation and finite-spectrum assignment for control over multi-hop WSN", In S. K. Mazumder, editor, Wireless Networking Based Control, Springer, 2011.
- J. Araujo, Y. Ariba, P. Park, H. Sandberg, and K. H. Johansson, "Control Over a Hybrid MAC Wireless Network", IEEE International Conference on Smart Grid Communications (SmartGridComm), Maryland, USA, October 2010.

- E. Witrant, P. Park, and M. Johansson, C. Fischione, and K. H. Johansson, "Predictive control over wireless multi-hop networks", IEEE Conference on Control Applications (CCA), Singapore, October, 2007.

– **Transmit power control of WSN:**

- P. Park, C. Fischione, and K. H. Johansson "A simple power control algorithm for wireless ad-hoc networks", International Federation of Automatic Control (IFAC) world congress, Seoul, Korea, July, 2008.
- B. Zurita Ares, P. Park, C. Fischione, A. Speranzon, and K. H. Johansson, "On Power Control for Wireless Sensor Networks: System Model, Middleware Component and Experimental Evaluation", IFAC European Control Conference (ECC), Kos, Greece, July, 2007.

Contributions by the author

The thesis is partially based on papers written with co-authors. The author has actively contributed both to the development of the theory as well as the paper writing. The author order indicates the relative contribution for most papers.

Chapter 2

Related Work

This chapter presents the related existing literature of the thesis. It is organized as follows. First, we discuss the existing communication protocols of WSNs in terms of MAC and routing protocols. Second, we present the related existing studies for modeling and analysis of the IEEE 802.15.4 protocol. Third, the characteristics and challenges of networked control systems are presented.

2.1 MAC and Routing

During last years, many protocols for WSNs have been proposed for a variety of applications, such as area, environmental monitoring, and industrial network, both in academia (e.g., [21, 34]) and industry (e.g., [31]–[36]). In this section, we discuss the interesting protocols that have been developed in the recent years relevant for the category of applications we are concerned in this thesis. This section is organized as follows. We first discuss important MAC protocols for WSNs. Second, we study the related existing routing protocols of WSNs. In the third section, we introduce the most practical and promising standards and an existing commercial systems for the industrial communication community. In the Table 2.1, we summarize the characteristics of the relevant protocols. In the table, we have evidenced whether indications as energy **E**, reliability **R**, and delay **D** have been included in the protocol design and validation. We discuss these protocols in the following. Furthermore, Figure 2.2 presents the taxonomy of MAC protocols according to development time and technique being used. There are several surveys for both MAC protocols [37]–[39] and routing protocols [40]–[45] of WSNs. We classify the protocols not only according to the technique being used or the network structure but also remarking the main performance indications of different protocols. Since the protocol design of WSNs must take into account the QoS requirements of the application layer, it is essential to consider the main design objective of the different protocols. Furthermore, we also highlight the strengthes and performance issues of each protocol.

Table 2.1: Protocol comparison. The letters **E**, **R**, and **D** denote energy, reliability and communication delay. The circle with dot ⊙ denotes that a protocol is designed by considering the indication of the column, but it has not been validated experimentally in sensor nodes. The circle with plus ⊕ denotes that the protocol is designed by considering the indication and experimentally validated. The dot • denotes that the protocol design does not include indication and hence cannot control it, but simulation or experiment results include it. We remark that some existing protocols consider the analytical studies of **E**, **R**, and **D** for the design. However, most protocols investigate the upper boundary of these performance indicators based on strong assumptions such as no contention of the network. The term "Relay" of analysis column means that the protocol is designed by considering the relay region based on the location information.

Protocol	Class	Access Scheme	Analysis	**E**	**R**	**D**
SMACS [46]	MAC	Schedule		⊙		
PicoRadio [47]	MAC	Schedule		⊕		
PACT [48]	MAC	Schedule		⊙		
EAT [49]	MAC	Schedule		⊙	⊙	
TRAMA [50]	MAC	Schedule		⊙	•	•
LMAC [51]	MAC	Schedule		⊙		
PEDAMACS [52]	MAC	Schedule		⊙		⊙
MMSN [53]	MAC	Schedule		⊙	⊙	•
Fetch [30]	MAC	Schedule		•	⊕	•
Dozer [54]	MAC	Schedule		⊕	⊕	
PMMAC [55]	MAC	Schedule			⊕	
TSMP [56]	MAC	Schedule		⊕	⊕	•
SPAN [57]	MAC	Contention	**D**	⊙	•	•
LPL [58]	MAC	Contention		⊕		
STEM [59]	MAC	Contention	**E**, **D**	⊙		⊙
BMAC [60]	MAC	Contention	**E**	⊕	•	•
Cycled receiver [61]	MAC	Contention	**E**	⊙		
WiseMAC [62]	MAC	Contention	**E**, **D**	⊕		⊕
XMAC [29]	MAC	Contention	**E**, **D**	⊕	•	•
Koala [63]	MAC	Contention		⊕		
SMAC [64]	MAC	Hybrid	**D**	⊕	•	•
TMAC [65]	MAC	Hybrid		⊕		
DSMAC [66]	MAC	Hybrid	**D**	⊙		⊙
DMAC [67]	MAC	Hybrid		⊙	•	⊙
Funneling [68]	MAC	Hybrid		⊕	⊕	

Continued on next page

Table 2.1 Continued from previous page

Protocol	Class	Access Scheme	Analysis	**E**	**R**	**D**
SCP [69]	MAC	Hybrid	**E**	⊕	•	•
Crankshaft [70]	MAC	Hybrid		⊕	•	•
ZMAC [71]	MAC	Hybrid	**R**	⊕	⊕	•
CTP [72]	Routing	Topology		⊕	⊕	
BCP [73]	Routing	Topology		•	⊕	⊕
SPIN [74]	Routing	Data-centric		⊙	⊙	
EAR [75]	Routing	Data-centric		⊙		
Directed diffusion [76]	Routing	Data-centric	**E**	⊙	•	•
LEACH [77]	Routing	Hierarchical	**E**	⊙		
TEEN [78]	Routing	Hierarchical		⊙		
PEGASIS [79]	Routing	Hierarchical		⊙		
TTDD [80]	Routing	Hierarchical		⊙	•	•
HEED [81]	Routing	Hierarchical		⊙		
MECN [82]	Routing	Location	Relay	⊙		
GAF [83]	Routing	Location	**E**	⊙	•	•
GEAR [84]	Routing	Location		⊙	•	
GeRaF [85]	Routing	Location	Relay	⊙		⊙
MMSPEED [86]	Routing	Location			⊙	⊙
VCP [87]	Routing	Location			•	•
Breath [88]	Routing	Location	**E**, **R**, **D**	⊕	⊕	⊕

2.1.1 MAC Protocols

The MAC protocols are classified into three categories based on the medium access mechanism: *schedule*, *contention*, and *hybrid*-based access mechanism. In contrast to previous surveys [37]–[39] that categorize MAC protocol according to the used specific technique to save energy in WSNs, we consider the traditional way to classify multiple access protocols similar to [100].

Schedule-based MAC protocols

In schedule-based MAC, nodes only wake up and listen to the channel in assigned slots and then go back to sleep in other slots. This approach requires the knowledge of the network topology to establish a schedule that allows each node to access the

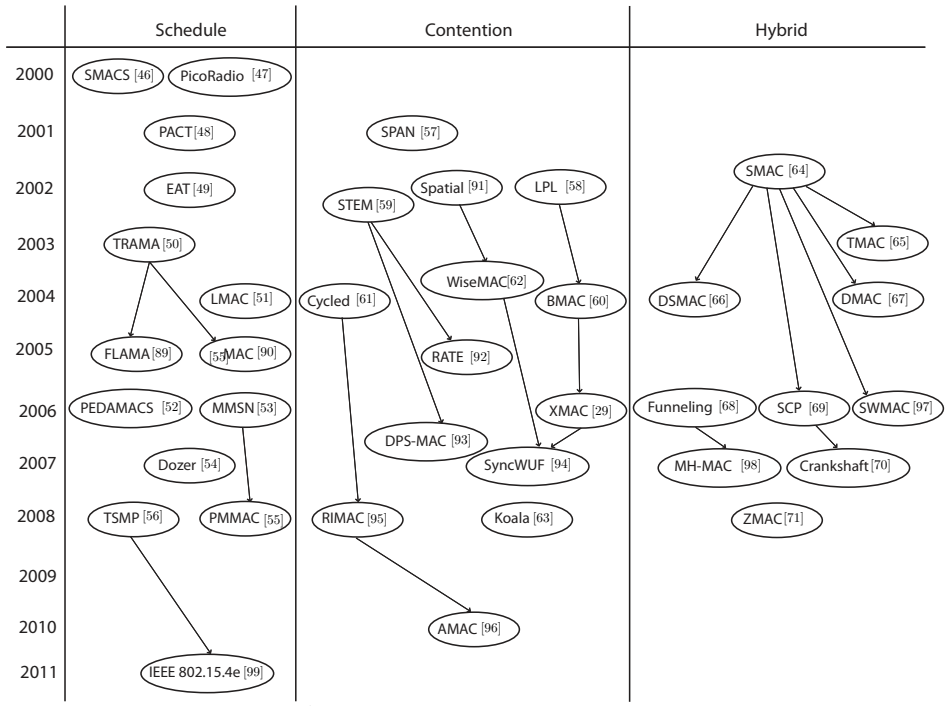


Figure 2.2: Taxonomy of MAC protocols. The arrows indicate inspirations for developing protocols.

channel and communicate with other nodes without having interference by another transmissions. The channel resources can be viewed, for this purpose, from a time, frequency, or mixed time-frequency standpoint. Hence, the channel can be divided by giving the entire frequency range (bandwidth) to a single user for a fraction of time as done in TDMA, or giving a fraction of the frequency range to every user all the time as done in frequency division multiple access (FDMA), or providing every user a portion of the bandwidth for a fraction of time as done in spread spectrum based systems such as code division multiple access (CDMA). Note that many schedule-based MAC protocols combine TDMA with FDMA where different time slots and frequency channels can be used by different nodes.

The schedule-based MAC protocols are attractive because once the schedule is set up, there are no collisions, no overhearing, and minimized idle listening. In addition, the schedule-based MAC protocol offers bounded latency, fairness and good throughput in high loaded traffic conditions with respect to the contention-based MAC. Remark that periodic and high-load traffic is most suitably scenarios for schedule-based MAC protocols.

However, the scheduled-based MAC protocols generates the problems or issues discussed as follows:

- **Complexity**: The lack of a central access point generates problems concerning an elevated complexity and high cost of slot assignment algorithms. While the distributed TDMA scheduling works well for medium sized networks, to determine a collision free schedule for large networks becomes quickly infeasible, which clearly impacts scalability.
- **Reduced flexibility**: When the traffic pattern or network topology changes, the global or local schedule needs to be updated. Hence, the scheduled-based MAC protocols may not be able to adapt to the highly dynamic topologies that occur in mobile environments since it may require too much communication overhead or too longer updating delay. Specifically, idle users of schedule-based protocols consume a portion of the channel resources. This portion becomes major when the number of potential users in the system is very large to the extent that conflict-free schemes are impractical. Moreover, adapting the slot assignment is not easy within a decentralized environment for traditional schedule-based MAC.
- **Overhead**: In general, the schedule-based MAC protocols require global or local time synchronization not only during the initial network setup phase but also during the runtime of the network to eliminate the clock drifting. Furthermore, to increase the flexibility for network topology changes, it may require heavy overhead of the network.
- **Memory limitation**: Collision free scheduling requires knowledge of the two-hop neighborhood topology, which uses a large memory footprint. Maintaining memory status consumes energy that scales with memory size.
- **Scalability**: The scalability of collision free slot assignment is a serious issue. Finding a collision-free schedule is a two-hop coloring problem.
- **Inefficient broadcast**: Unless the protocol is sender scheduled, the transmission of broadcast packets requires the repetition of the same packet several times which is clearly not energy efficient and gives longer delay.

In the following, we briefly describe a wide range of schedule-based MAC protocols.
• **SMACS** [46]: Selforganizing medium access control for sensor networks (SMACS) employs a distributed scheduling to establish the transmission and reception slots between neighboring nodes without having a master node of a centralized scheduling approach. This protocol uses a combination of TDMA and FDMA or CDMA for accessing the channel. In SMACS, a channel is assigned to a neighbor if discovered. Each link works on a different channel to reduce collisions. It consists of two phases: neighbor discovery and channel assignment. In neighbor discovery phase, a node wakes up and listens for a given time to receive invitation packets to find its neighbor. If it does not receive such a packet, it starts inviting others by sending an invitation packet. Nodes sleep and wake up randomly to save energy. When a link is formed between two nodes, they establish transmission-reception

slots in the channel assignment phase. These slots are used periodically to exchange data between nodes. However, nodes sleep to save energy outside these slots. The missing probability that two nodes never meet does not vanish. Furthermore, it is difficult to find the optimal routes and it is not energy efficient.

• **PicoRadio** [47]: In PicoRadio, each node listens to a common control channel and broadcasts a channel assignment packet to inform its neighbors about its channel. It keeps track of all of its one and two-hop neighbors' channels to avoid choosing an overlapping channel with them. During this channel set up period, nodes execute the following procedures. Each node gathers information of used channels of its neighborhoods when it wakes up. It then selects another unused channel from the channel pool and broadcasts its neighbors channel and its chosen channel. If a conflict occurs, the node that first detects the conflict switches to another unused channel. When implementing PicoRadio on IEEE 802.15.4 compatible hardware, 16 frequency channels are available.

• **PACT** [48]: In power aware clustered TDMA (PACT), the TDMA frame contains mini-slots for exchanging control information and the transmission slots of nodes according to some node assignment. During the control slot, each node declares upcoming transmissions so that nodes that will not be receivers go to sleep to save energy. Furthermore, to balance the working load, rotation is executed based on the residual energy of the nodes and traffic pattern which is similar to the one used by the protocol [77].

• **EAT** [49]: Energy aware TDMA based MAC (EAT) protocol employs a centralized scheduling at the sink. EAT assumes the formation of clusters in the network. The sink collects the information from the other sensor nodes within its cluster, performs the data fusion, communicates with the other sinks and finally sends the data to the control center. The sink assigns the time slots to the sensor nodes within its cluster and informs other nodes about the time slot when it should listen to other nodes and the time slot when it can transmit own data. It consists of four main phases: data transfer, refresh, event triggered-rerouting and refresh-based rerouting. The data transfer phase is for sending data in its allocated time slot. During refresh phase, nodes update their state to the sink. The sink requires this state information of nodes for performing rerouting during event triggered-rerouting. The refresh-based rerouting occurs periodically after the refresh phase. During both these rerouting phases, the sink executes the routing algorithms and sends new routes to the sensor nodes. Two approaches are introduced for slot assignment based on graph parsing strategy: breadth first search and depth first search. The breadth firth search technique assigns the time slot numbers starting from outermost sensor node giving them contiguous slots. While depth first search technique assigns contiguous time slots for the nodes on the route from outermost sensor node to the sink.

• **TRAMA** [50]: The traffic adaptive medium access (TRAMA) is a TDMA based protocol to increase the utilization in an energy-efficient manner. This protocol employs a contention period for two-hop topology construction and a contention free period for data exchange. The distributed election algorithm for each time slot is

used to select one transmitter within each two-hop neighborhood, which is similar to the node activation multiple access [101]. This election eliminates the hidden-terminal problem and hence ensures that all nodes in the one-hop neighborhood of the transmitter will receive data without any collision. TRAMA consists of three main parts:

- The neighbor protocol is for collecting the information about the neighboring nodes.
- The schedule exchange protocol is for exchanging the two-hop neighbor information and their schedule.
- The adaptive election algorithm decides the transmitting and receiving nodes for the current time slot using the neighborhood and schedule information.

The other nodes in the same time slot are switched to low power mode. TRAMA is shown to be more energy efficient and has higher throughput than the SMAC protocol. However, the latency of TRAMA is higher than the one of contention-based protocols, due to a higher percentage of sleep times. The flow-aware medium access protocol (FLAMA) [89] improves the TRAMA by avoiding periodic exchange of information between two-hop neighbors. In addition, μMAC [90] applies a similar idea of TRAMA by using external clock synchronization based on a beacon source.
• **LMAC** [51]: Lightweight MAC (LMAC) protocol is a self-organizing TDMA scheme that organizes time into frames containing the number of slots. Every node owns one slot in which it sends out a header to mark its occupancy, possibly followed by a data payload either addressed to a specific recipient or to all nodes in range. Therefore, a node must listen in all slots other than its own to check for incoming data. The header includes a list of all occupied slots in the owner's one-hop neighborhood to allow for collision-free transmissions and spatial re-use of slots. After merging the occupancy information of its neighbors, a new node joining the network selects a free transmission slot within its two-hop neighborhood. This distributed free slot selection mechanism allows the optional ACK messages. Multichannel LMAC [102] enhances the LMAC by adding multi-channel support.
• **PEDAMACS** [52]: Power efficient and delay aware medium access control protocol for sensor networks (PEDAMACS) is also based on centralized scheduling at the sink. The sink gathers information about traffic and topology during the setup phase. Then it calculates a global scheduling and sends it to the entire network. Note that the protocol assumes that the sink can reach all nodes when it transmits. The uplink communications follows a TDMA scheme established by the sink. The collision-free scheduling is based on coloring a conflict graph. During topology collection phase, each node sends information to the sink using typical CSMA. Afterwards, the sink starts flooding topology-learning packets. The main issue of this method is the traffic pattern is always convergecast. Convergecast is a common communication pattern across many WSN applications collecting information from many different source nodes to a single sink of the network. In addition, the assumption that the sink reaches all nodes is not always satisfied.

• **MMSN** [53]: The multi-frequency MAC for wireless sensor networks (MMSN) assigns evenly the frequency to the nodes of a 1-hop neighborhood. Each node needs to use the destination's frequency when transmitting, and its own frequency when receiving. During network run-time, nodes are synchronized and time is sliced up into slots. A backoff-based CSMA algorithm solves contention between nodes in a given frequency/time slot.
• **PMMAC** [55]: Practical multichannel MAC (PMMAC) proposes a multi-channel MAC protocol to cope with interference without assuming time synchronization of the network. The protocol dynamically assigns channel to nodes, and groups nodes sharing a channel into clusters. The nodes easily detect the high contention/interference of their channel by exchanging status messages measuring the loss ratio. Cluster heads then take the initiative to hop on the next available channel, followed by the other nodes in its cluster. Inter-cluster communication is done by temporarily changing to the destination's channel. Although nodes do not need to be synchronized, nodes need to broadcast status messages to their neighbors frequently.
• **TSMP** [56]: The time synchronized mesh protocol (TSMP) is a medium access and networking protocol based on the WirelessHART standard [103] for industrial automation. The main idea of TSMP is to use the benefits from synchronization of nodes in a multi-hop network, allowing scheduling of collision-free pair-wise and broadcast communication to meet the traffic needs of all nodes while cycling through all available channels. Note that TSMP employs in addition FDMA and frequency hopping to achieve a high robustness against interference and other channel impairments. Hence, TSMP combines the advantages of both the TDMA and FDMA mechanisms. In TSMP, the sink retrieves the list of nodes, their neighbors and their requirements in terms of traffic generation. From this information, it constructs a scheduling table in both time and frequency. Therefore, the duration of active periods of TSMP is flexible. Two simple rules to establish and manage the links are as follows: 1) never put two transmissions in the same time/frequency slot, 2) at a given time, a given node should not receive from two neighbors nor have to send to two neighbors. We remark that these rules are similar to the Chlamtac's algorithm [104]. Since TSMP is based on the scheduling of TDMA and FDMA, the latency is in general guaranteed to be bounded by a finite value which depends on the particular design of the time-frequency pattern. TSMP provides a simple solution for time synchronization of the network. Nodes maintain a precise sense of time and exchange only time offset information with neighbors to ensure alignment. These time offset values are exchanged during active periods together with the usual data and ACK with negligible overhead. Note that nodes compensate the clock drift based on these offset values.
Note that some schedule-based MAC protocols [105, 106] are able to cope with network dynamics and node mobility.

Contention-based MAC protocols

The contention-based MAC protocols differ in principle from schedule-based approaches since there is no centralized scheduler, hence, there is no guarantees for successful packet transmission. The key issue of the various contention-based protocols is how to resolve conflicts once they occur so all messages are transmitted successfully. In general, nodes compete for the use of the wireless medium and only the winner of this competition is allowed to access to the channel and transmit. One of the most important strengthes of contention-based MAC protocols is fairly simple mechanism compared to scheduled-based protocols, since it does not require either global synchronization or topology knowledge. Furthermore, idle users of the contention-based MAC protocols do not transmit and thus do not consume any portion of the channel resources with respect to the idle users of the scheduled-based protocols. CSMA is a representative schemes of contention-based approaches. A node having a packet to transmit first senses the channel before actually transmitting. If the node finds the channel clear, it starts transmitting. Otherwise, it postpones its transmission to avoid interfering with the ongoing transmission. CSMA does not rely on a central entity and is robust to node mobility, which makes it intuitively a good candidate for networks with mobility and dynamicity. However, traditional contention-based MAC protocols are not directly applicable for most of WSN applications due to poor energy efficiency.

One of the promising techniques to save energy is the preamble sampling where each node chooses its active schedule independently of other nodes around. Note that there are other terminologies that refer to a similar approach in the literature, e.g. low power listening (LPL) [58] and cycled receiver [61]. These protocols are collectively referred to as preamble sampling protocols in this thesis. In preamble sampling techniques, each node wakes up only for a short duration to check whether there is a transmission on the channel or not. In this way, each node spends most of the time in sleep mode.

To avoid deafness, each data frame is preceded by a preamble that is long enough to make sure that all potential receivers detect the preamble and then get the data frame. According to the duty-cycle parameter, nodes periodically switch their radios on to sample the channel. If a node finds that the channel is idle, it goes back to sleep immediately. However, if it detects a preamble transmission on the channel, then it keeps its radio on until it receives the subsequent data frame. Right after the reception of the data frame, the node sends an ACK frame, if needed, and goes back to sleep afterwards. To be effective, the duration of the preamble transmission needs to be at least as long as the check interval defined as the period between two consecutive instants of node wake-up. In this way, a node makes sure that all potential receivers are awake during its preamble transmission so that they get the subsequent data frame. This is highly beneficial for applications where traffic load of the network is very low such as surveillance applications.

Despite its successful usages, the contention-based MAC protocols cause some major problems:

- **High traffic load**: The contention-based MAC protocols significantly degrade the throughput when the traffic load increases due to the high contention with respect to the scheduled-based MAC protocols. Note that the distributed nature prevents them to achieve the same efficiency as ideal reservation-based protocols. In particular, the collisions become more critical problems for the preamble sampling protocol as the check interval and traffic load increase. Since the traffic patterns of many WSN applications are correlated [42], this is the critical issue of the preamble sampling protocol even though the average traffic load is small in some application scenarios. The high transmission cost counteracts the energy efficiency in situations with high collision rates.

- **Limited duty-cycle**: Lowering the duty-cycle implies putting nodes in sleep mode for larger periods, which means extending the check interval. While using a larger check interval reduces the cost of idle listening at the receiver, it increases the transmission cost as the transmitter uses a longer preamble. Hence, there is a tradeoff between the receiving cost of idle listening and transmission cost of longer preamble. There is an optimal value for the check interval beyond which nodes waste more energy in transmission than they save in reception. Finding the optimal check interval depends upon several parameters such as transmission power, reception power, traffic load and switching times of the radio chip. Therefore, preamble-sampling protocols have a limited duty-cycle that is determined by the optimal check interval value.

- **Optimal parameter**: As we discussed for the issues of the high traffic load and limited duty-cycle, determining the optimal parameters of check interval is not trivial since it is also function of traffic load, network topology, and hardware specifications. The check interval needs to consider the traffic load of the network, otherwise, the reliability and throughput significantly degrade due to the high contention when the check interval is expired. Note that as the check interval increases, the packet delay obviously increases. Hence, there is a tradeoff between energy consumption, throughput, and delay of the network.

The LPL [58] is one of the first preamble sampling protocols of contention-based MAC for WSNs. The idea proposed in LPL motivated the design of many preamble sampling protocols that follow a similar concept. In the following we will provide some details of these protocols.

• **LPL** [58]: It is one of well known techniques to save energy in contention-based MAC protocols. Each node independently repeats a sleep/active cycle without any coordination on their cycles and consequently wakes up independently of each other. When transmitter sends a data packet, it sends the longer preamble to cover one complete sleep interval, which assures that the receiver can detect a signal and eventually detect a start symbol, followed by the true message. This technique avoids the overheads and time synchronization, but leaves it up to the sending nodes to arrange a rendezvous with the intended receiver whenever it wakes up. BMAC [60] extends LPL technique by adding a user-controlled sleep interval. The

reverse approach [61, 95] to LPL is also possible by sending beacons at a regular interval from receiver to indicate that the node is ready to receive a data packet. This avoids sending a long wake-up preamble and shortens transmission times considerably. Furthermore, Energy-ware adaptive LPL (EA-ALPL) [107] suggests the adaptive listening mode according to traffic changes of BMAC.

• **STEM** [59]: The sparse topology and energy management (STEM) protocol uses the two channels: a wake-up channel and a data channel. The wake-up channel is to setup a meeting between the transmitter and the receiver to avoid deafness, whereas the data channel is only used for data exchange once the meeting occurs. Nodes follow a preamble sampling approach in the wake-up channel before sending the data on the data channel. However, it requires to implement the wake-up radios in the hardware. The dual preamble sampling MAC (DPS-MAC) [93] improves STEM by reducing the sampling duration of nodes. In DPS-MAC, when a node wakes up to sample the channel, it does not need to be awaken for a duration larger than the gap of inter-preamble packet.

• **Cycled receiver** [61]: The cycled receiver is the reverse approach to LPL [58], which shifts communication initiation from the transmitter side to the receiver side. When the receiver is ready to receive message, then it sends out beacons at a regular interval instead of listening periodically. To send a data frame, the transmitter stays awake and monitors the channel waiting for a beacon from the receiver. Once the transmitter receives the beacon, it transmits the data frame and waits for an ACK to end the session. This avoids sending a long wake-up preamble of LPL and shortens transmission times considerably. The cycled receiver achieves high energy savings for unicast and anycast communications. Receiver-Initiated (RI) MAC [95] and A-MAC [96] are the similar type of protocols. However, it cannot be used for broadcast and multicast communications, because it is receiver-initiated. Furthermore, the beacons from receivers interfere with ordinary traffic as well as with each other.

• **WiseMAC** [62]: The WiseMAC protocol is similar to spatial TDMA and CSMA with preamble sampling protocol [91] where all the sensor nodes have two communication channels. TDMA is used for accessing data channel and CSMA is used for accessing control channel. However, WiseMAC needs only one channel and uses non-persistent CSMA with preamble sampling technique to reduce power consumption during idle listening. The basic idea is to track the phase offsets of neighbors' schedules allowing senders to transmit a message just in time with a short-length preamble saving energy and bandwidth. All nodes in a network sample the medium with a common period, but their relative schedule offsets are independent. If a node finds the medium busy after it wakes up and samples the medium, it continues to listen until it receives a data packet or the medium becomes idle again. The size of the preamble is initially set to be equal to the sampling period. To reduce the power consumption incurred by the predetermined fixed-length preamble, WiseMAC offers a method to dynamically determine the length of the preamble. That method uses the knowledge of the sleep schedules of the transmitter node's neighbors. The nodes learn and refresh their neighbor's sleep schedule during ev-

ery data exchange as part of the ACK message. In this way, every node keeps a table of the sleep schedules of its neighbors. Based on the neighbors' sleep schedule tables, WiseMAC schedules transmissions so that the destination node's sampling time corresponds to the middle of the sender's preamble. To decrease the possibility of collisions caused by that specific start time of a wake-up preamble, a random wake-up preamble is advised. Furthermore, a lower bound for the preamble length is calculated based on potential clock drift between the source and the destination. The synchronized wake-up frame (SyncWUF) [94] combines the WiseMAC and the packet preamble technique [29]. The separate wake-up radio similar to STEM is applied to WiseMAC [92].

• **XMAC** [29]: The XMAC is a refinement of BMAC for packet-based radios. The transmitter sends a packet strobe instead of sending a long wake-up preamble of BMAC. Since the packets contain the address of the receiver, it allows overhearing nodes to switch off the radio after receiving a packet out of the strobe. Once the node receives a right packet strobe, it replies an ACK. Then, the actual message exchange takes place immediately. This ACK mechanism within short idle time reduces an average preamble transmission time with respect the length of the strobe preamble of BMAC. XMAC also includes a lookup table to adapt the duty-cycle of the nodes based on the traffic load. However, this is suboptimal solution when there are multiple transmitters and receivers in the network since XMAC only optimizes the energy consumption of the network with only one receiver. Furthermore, XMAC does not provide a functionality for broadcast communication.

Hybrid-based MAC protocols

The hybrid-based MAC protocols combine the advantages of both a random access with contention-based MAC and a deterministic access with schedule-based MAC. The idea of a hybrid MAC is not new. The IEEE 802.15.4 MAC has been inspired by the adaptive MAC protocol proposed in the late 70's by Kleinrock and Yemini [108] to maximize the throughput, which combined slotted ALOHA and TDMA. Many allocation schemes were designed to combine the advantages of both the ALOHA and the TDMA approaches. One of the extensions is to use a so called reservation scheme with contention, where users contend during a reservation period, and those who succeed in this contention transmit without experiencing interference. Such a scheme derives its efficiency from that reservation periods are several orders of magnitude shorter than transmission periods. The works proposed in [109]–[111] fall in this category of reservation schemes. Additional reservation protocols and their analysis can be found in [112]–[114]. For example, the demand assignment multiple access is successfully used for satellite and military communications [115] and for the IEEE 802.15.3c standard. The throughput of IEEE 802.15.3c is studied in [116]. The motivation of the hybrid-based MAC protocols for WSNs is to offer flexible QoS to several classes of applications rather than just maximizing the throughput of the network. The hybrid-based MAC protocols are classified into two categories based on how to combine the contention-based and schedule-based MAC: reservation-for-

contention and partition access mechanism.
In reservation-for-contention MAC, nodes define common active/sleep periods. The active periods are used for communication and the sleep ones for saving energy. This approach requires that nodes maintain a certain level of synchronization to keep active/sleep periods common to all nodes similar to schedule-based MAC. During the active periods, nodes contend for the channel using contention-based approaches similar to contention-based MAC. Hence, the contention-based MAC is located inside of the schedule-based MAC. However, the use of common active/sleep periods may not be suitable for applications with regular traffic or high traffic load. Note that the size of active period and the algorithm to resolve the contention at the beginning of active period are critical issues to improve the reliability and throughput of the network.
In partition MAC, the resource is divided into the contention-based MAC and schedule-based MAC by considering the traffic load of the network, time scale, geographic location of the nodes, etc. For instance, when the traffic load is low, contention-based approaches yield sufficient performance, however, when the traffic load is high, then scheduled protocols are a better choice.
Despite its attractive advantages to combine the advantages of both contention-based and schedule-based MAC, the hybrid-based MAC protocols makes some issues which are discussed as follows:

- **Optimal parameter**: Determining the optimal size of active periods requires to take into account the tradeoff between two parameters: idle listening and collisions. Short active periods reduce idle listening, but they increase contention and thus collision rates. Long active periods do the opposite, they reduce contention at the cost of increased idle listening. Hence, if the active period is fixed, then this makes rigid protocol, as nodes have no means to dynamically change their duty-cycle to meet time-varying or spatially non-uniform traffic loads. Note that variable workloads are expected in many WSN applications since they are embedded in the physical environments. For instance, nodes that are closer to a sink are most likely to relay more traffic than border nodes. The optimal portion between contention-based MAC and schedule-based MAC is also dependent on many parameters such as traffic load, network topology, and application requirements. Hence, it is difficult to find the optimal resource allocation for dynamical environments.

- **Sleep time**: Sleep periods are essential to save energy. However, they introduce extra end-to-end delay called sleep delay. Sleep delay significantly increases end-to-end latency in multi-hop networks, as intermediate nodes on a route do not necessarily share a common schedule. Hence, there is a tradeoff between sleep delay and optimal active periods.

- **Less understanding**: Understanding reliability, delay, and throughput is essential to characterize the fundamental limitations of this protocol and optimize its parameters. Nevertheless, there is not yet a clear understanding of the

achievable performance of these hybrid-based MAC with respect to the two approaches we have previously summarized. The main challenge for this analysis is the coexistence of the stochastic behavior of the contention-based MAC and the deterministic behavior of the schedule-based MAC. Furthermore, the analysis of the mutual effects of these two schemes is the fundamental step to understand the performance of the hybrid-based MAC protocols.

In the following, we first briefly describe some MAC protocols in the category of reservation-for-contention MAC by stating the essential behavior of the protocols. Then, we discuss some protocols of partition MAC category in details and highlight the key ideas.

• **SMAC** [64]: The basic idea of Sensor-MAC (SMAC) is to employ the periodic sleep–listen schedules based on local synchronization of the network. Neighboring nodes form virtual clusters to set up a common sleep schedule. If two neighboring nodes reside in two different virtual clusters, they wake up at the listen periods of both clusters. Hence, the nodes in this boundary consume more energy due to the idle listening and overhearing. During listen period, nodes are able to communicate with other nodes and send some control packets such as SYNC, Request to Send (RTS), Clear to Send (CTS) and ACK. By a SYNC packet exchange all neighboring nodes can synchronize together and using RTS/CTS exchange the two nodes can communicate with each other, which is similar to IEEE 802.11. Periodic sleep may result in high latency, especially for multi-hop routing algorithms, since all intermediate nodes have their own sleep schedules. The adaptive listening technique is proposed to improve the sleep delay and thus the overall latency. In that technique, the node that overhears its neighbor's transmissions wakes up for a short time at the end of the transmission. Hence, if the node is the next-hop node, its neighbor could pass data immediately. The end of the transmissions is known by the duration field of the RTS/CTS packets. However, this protocol is not energy efficient since the sensor will be awake even if there is no reception/transmission during listen period. The separate wake-up slots MAC (SWMAC) [97] divides the active period into slots to reduce the length of active period. Furthermore, the static sleep–listen periods of SMAC result in high latency and lower throughput. The coordinated adaptive sleeping [117] suggests the use of overhearing to reduce the sleep delay. The Mobility-aware SMAC (MSMAC) [118] extends a mechanism of SMAC by adapting the duty-cycle to improve connection setup times in mobile environments.

• **TMAC** [65]: The TMAC is an extension to S-MAC to handle traffic fluctuations in time and space of the network. TMAC uses an adaptive duty-cycle that the listen period ends when no activation event has occurred for a time threshold. The minimal amount of idle listening per frame is presented along with some solutions to the early sleeping problem. Although TMAC gives better results under variable loads, the synchronization of the listen periods within virtual clusters is broken. This is one of the reasons for the early sleeping problem. Furthermore, TMAC protocol has high latency as compared to the SMAC protocol. The energy efficient MAC

(E2MAC) [119] enhance the energy efficiency of TMAC by accumulating packets in a node until a certain buffer limit.

• **DSMAC** [66]: Dynamic Sensor MAC (DSMAC) adds a dynamic duty-cycle mechanism to SMAC to decrease the latency for delay-sensitive applications. At the initial phase, all nodes have the same duty-cycle. Furthermore, all nodes share their one-hop latency values within the SYNC period. When a receiver node notices that the average one-hop latency value is high, it decides to shorten its sleep time and announces it within the SYNC period. After a sender node receives this sleep-period decrement signal, it checks its queue for packets destined to that receiver node. If there is one, it decides to double its duty-cycle when its battery level is above a specified threshold. The duty-cycle is doubled so that the schedules of the neighbors will not be affected. The latency observed with DSMAC is better than that observed with SMAC. Moreover, the average power consumption per packet decreases.

• **DMAC** [67]: The principal aim of data–gathering MAC (DMAC) is to achieve low latency for convergecast communications, but still be energy efficient. The time is divided in small slots and runs CSMA with ACK within each slot to transmit/receive one packet. Hence, DMAC could be summarized as an improved slotted CSMA in which slots are assigned to the sets of nodes based on a data gathering tree. During the receive period of a node, all of its child nodes have transmit periods and contend for the medium. Low latency is achieved by assigning subsequent slots to the nodes that are successive in the data transmission path. Furthermore, DMAC includes an overflow mechanism to handle the problem when each single source node has low traffic rate but the aggregate rate at intermediate node is larger than the basic duty-cycle. In this mechanism the node will remain awake for one extra time slot after forwarding the packet.

• **SCP** [69]: The scheduled-channel-polling MAC (SCP) protocol combines LPL [58] with a global synchronized channel access. All nodes wake up at a regular interval and perform a synchronized carrier sense when there is no traffic. Note that since the polling time of all neighboring nodes is synchronized, the length of preamble strobe from a sender is significantly reduced compared to the preamble size of LPL. When a node has a packet to send, it waits for scheduled wake-up of the receiver and sends a short wake-up tone. Before sending the tone, it performs carrier sense within the first contention window to reduce chances of collision. If the node detects idle channel it will send the wake-up tone. Otherwise, it goes back to sleep and will perform regular channel polling. After a sender wakes up a receiver, it enters the second contention window. If the node still detects channel idle in the second contention phase, it starts sending data. This two-stage contention resolution policy achieves lower collision probability with shorter overall contention time.

• **Crankshaft** [70]: The Crankshaft is another type of a hybrid MAC which is similar to SCP [69]. It employs node synchronization and offset wake-up schedules to restraint the main cause of inefficiency in dense networks. Time is divided into frames consisting of broadcast and unicast slots. Every node is required to listen to all broadcast slots and to one of the unicast slots based on its MAC address.

A node that needs to transmit a packet to a particular node has to wait for this node's unicast slot and content for it using the contention resolution scheme. Note that the contention resolution is required since several nodes might want to send a packet in a particular slot. However, the throughput decreases at low traffic load due to idle slots.
In the following we discuss some protocols of partition MAC category in details and highlight some issues.
• **IEEE 802.15.4** [8]: The IEEE 802.15.4 standard specifies the physical layer and the MAC sublayer for low-rate wireless networks. The standard defines two channel access modalities: the beacon-enabled modality, which uses a slotted CSMA/-CA and the optional GTS allocation mechanism, and non-beacon modality, which is a simpler unslotted CSMA/CA without beacons. Most of the unique features of IEEE 802.15.4 are in the beacon-enabled mode combining the advantages in both contention-based and schedule-based MAC. Since the IEEE 802.15.4 standard is one of the most important protocols, we describe more details of this standard in Section 2.2.
• **Funneling-MAC** [68]: In convergecast communication, nodes close to the sink experience higher traffic loads so that higher packet loss happen in this region when a contention-based MAC protocol is used. Therefore, the main idea of Funneling-MAC is to use TDMA in regions close to the sink and CSMA elsewhere. The sink periodically sends beacons to dynamically drive the depth of the intensity region and to synchronize the nodes inside the intensity region to use TDMA. Nodes that receive a beacon consider themselves as inside the intensity region and become f-nodes. The f-nodes that are at the boundary of the intensity region become path-heads. Path-heads send their identities to the sink so that it knows their number and how many hops away they are from it. When the node receives the identities of path-heads and their relative depth, it runs a slot distribution algorithm. The sink node sends a schedule announcement packet in which it includes the path-heads and the number of slots assigned to each path-head. As the schedule is broadcast, all intermediate f-nodes can figure out their schedules. The multimode hybrid-MAC (MH-MAC) [98] is similar to Funneling MAC allowing nodes to switch from asynchronous mode to synchronous mode to adapt to varying traffic conditions.
• **ZMAC** [71]: The principal aim of Zebra MAC (ZMAC) is to increase the throughput in networks with variable traffic patterns. ZMAC defines a hybrid method that runs CSMA in low traffic and switches to TDMA in high traffic conditions. The distributed TDMA scheduling [120] is used to avoid hidden node collisions. The length of slots is large enough to allow transmission of multiple packets. Thus, there is no need for strong synchronization. Once the TDMA schedules are established, each node uses the assigned slot to it for transmission. If a node needs more than one slot, it attempts to utilize its neighbors' unused slots. To utilize the unused slots, a node starts a random backoff timer at the beginning of that slot. If the slot is not used by its owner by the end of random backoff timer, the node transmits data. Note that the random backoff is large enough to let the slot owner have access to its slots before the others. However, ZMAC needs to periodically run a distributed

TDMA scheduling [120] to compensate the clock drifts.

2.1.2 Routing Protocols

One of the main design goals of WSNs is to prolong the network lifetime while guaranteeing reliable data communication by employing energy management techniques. As we describe in Section 1.2, the design of routing protocols of WSNs is influenced by many challenging factors. By considering the underlying network structure, we classify the routing protocols as follows: *topology*, *data-centric*, *hierarchical*, and *location*-based routing similar to [40]. The underlying network structure can play a significant role in the operation of the routing protocol in WSNs. We discuss most of the well known routing protocols.

The topology-based routing protocols can be classified into two main classes: link-state and distance-vector protocols. In link-state protocols, each node first collects the neighborhood information and then floods the local topology information to the network (e.g., Open shortest path first (OSPF) [121] and optimized link state routing (OLSR) [122]). Hence, each node of the network has a global information of the network topology. Each node uses its information to compute a routing table to the next hop along the shortest path according to some metric. In distance-vector protocols, each node exchanges and has information on what routes its neighbors have rather than a map of the network topology e.g., (dynamic source routing (DSR) [123] and ad-hoc on-demand distance vector (AODV) [124]). Hence, each node knows the union of its neighbors routes and a route to itself. Note that each node does not have knowledge of the entire path to a destination. While link-state protocols typically present faster convergence and less overhead, distance-vector protocols are simpler, require smaller storage and less computational capability. Because nodes have very limited resources for storage and computation, distance-vector protocols are good alternative for WSNs.

Most of recently developed routing protocols are under the category of topology-based routing protocols. These protocols consider the effect of the MAC and routing layer to achieve the good performance in terms of energy efficiency, reliability, and delay depending on applications.

• **CTP** [72]: Collection tree protocol (CTP) is a best effort tree-based anycast protocol used to collect the data from sensor nodes. Note that CTP is mainly targeted for low traffic rates such as monitoring applications. CTP estimates first the quality of the link, then it decides the parent node merely based on the link quality. However, this approach may incur a load balancing problem because the node with good quality link will be selected as the preferred parent and consumes more energy. Furthermore, CTP uses a very aggressive retransmission policy, i.e., the default value of the maximum number of retransmissions is 32 times. Note that retransmission of old data to maximize the reliability may increase the delay and is generally not useful for control applications [23].

• **BCP** [73]: Backpressure collection protocol (BCP) considers the dynamic backpressure routing of wireless networks. In BCP, the routing and forwarding decision

is made on a per-packet basis by computing a backpressure weight of each outgoing link that is a function of localized queue and link state information. Therefore, the overhead due to the backpressure algorithm depends on all possible forwarding nodes of the next hop. Furthermore, the backpressure algorithm does not prevent loops of the routing and may incur in large delay.

In data-centric routing protocols, each node typically plays the same role and sensor nodes collaborate to perform a common task. Since many applications of WSNs have a large number of nodes, it is not feasible to assign a global identifier to each node. This consideration has led to data-centric routing. The basic idea of data centric routing is that routing, storage, and querying mechanisms can be performed more efficiently if communication is based on application data instead of the traditional IP global identifiers. This data centric approach to routing provides additional energy savings derived from the communication overhead of binding identifiers that are no longer needed, and facilitates in-network processing with additional savings derived from data aggregation and compression. Although a data-centric routing approach works well with static nodes, it is not able to handle complicated queries. Furthermore, it is not scalable to large sensor networks and data aggregation results in communication and computation overhead. Another issue is that the energy dissipation strongly depends on the traffic patterns, so the fairness of the network resources is issue. Two important early works of data centric routing protocols are sensor protocols for information via negotiation (SPIN) [74] and directed diffusion [76], where the energy is saved by eliminating the redundant data transmission through data negotiation between nodes. These two protocols have been an inspiration for many other protocols that follow a similar concept. In this section, we will describe these protocols in details and highlight the key ideas.

• **SPIN** [74]: The basic idea of SPIN is to name the data using high-level descriptors called meta-data instead of sending all the data. The sensor node operate more efficiently and conserve energy by eliminating the redundant data transmission. SPIN has three types of messages, that is, ADV, REQ, and DATA. Before sending a DATA message, the sensor node broadcasts an ADV message containing a descriptor of the DATA, which is the key feature of SPIN. If a neighbor is interested in the data, it sends a REQ message for the DATA and DATA is sent to this neighbor sensor node. By repeating this process, the sensor nodes of the network that are interested in the data will get a copy. SPIN solves the classic problems of flooding such as redundant information passing, overlapping of sensing areas and resource blindness by the meta-data negotiation. Hence, it achieves good energy efficiency as well as reduces the redundant data of the network. Furthermore, the topological changes are localized since each node needs to know only its single-hop neighbors. However, the data advertisement mechanism of SPIN does not guarantee reliable delivery of data. For instance, if the nodes that are interested in the data are far away from the source node and the relay nodes between source and destination are not interested in those data, then such data will not be delivered to the destination.

• **Directed diffusion** [76]: The main idea of directed diffusion is to use the attribute-value pairs for the data and to query the sensors based on demand by

using those pairs. The sink requests data by broadcasting interests through its neighbors. Each node receiving the interest can do caching for later use. The intermediate nodes are able to aggregate data based on data's name and attribute-value pairs. The interest entry also contains several gradient fields. Each sensor that receives the interest sets up a gradient toward the sensor nodes from which it receives the interest. The gradient includes the data rate, duration and expiration time. By using interest and gradients, paths are established between sink and sources. The sensed data are then returned in the reverse path of the interest propagation. Several paths can be established so that one of them is selected by reinforcement. When a path between a source and the sink fails, a new or alternative path should be identified. Note that directed diffusion differs from SPIN in terms of the on demand data querying mechanism. The communication of directed diffusion is neighbor-to-neighbor with no need for a node addressing mechanism. In addition, direct diffusion is energy efficient since it is on demand and there is no need for maintaining global network topology. Although this protocol achieves some energy saving by processing data in-network, the matching process for data and queries might require some extra overhead at the sensors. Since the naming schemes are dependent on applications, each time should be defined a priori. In addition, the query-driven data delivery model may not be feasible for some applications such as environmental monitoring.

In hierarchical routing protocols, nodes play different roles in the network. Sensor nodes are clustered and the one with the higher residual energy is usually chosen as the cluster head. These nodes as the leaders of their groups have some responsibilities like collecting and aggregating the data from their respective clusters and transmitting the aggregated data to the base station. Note that hierarchical routing lowers energy consumption within a cluster and by performing data aggregation and fusion to decrease the number of transmitted messages. The rest of the nodes perform the sensing in the propinquity of the target. Furthermore, it allows the system to cope with additional load and to cover a large area of interest without degrading the service. Therefore, hierarchical routing contributes to overall system scalability, lifetime, and energy efficiency of the network. Most of the literature studies the optimal policy to choose the cluster head or channel allocation mechanisms rather than the multi-hop network scenario for the routing. Although hierarchical routing introduces overhead due to the cluster configuration and their maintenance, earlier work has demonstrated that cluster-based protocols exhibit better energy consumption and performances when compared to flat network topologies for large-scale WSNs. However most of these protocols have problems such as network partitioning (i.e., the elected cluster head has no other cluster head in its communication range). Furthermore, they are not capable of handling node mobility and are hard to support time-critical applications due to the continuously cluster head evaluation procedure. In the following we will provide some details of these protocols.

• **LEACH** [77]: The low-energy adaptive clustering hierarchy (LEACH) [77] is one of the first hierarchical routing approaches for WSNs. The idea proposed in LEACH motivated the design of many hierarchical routing protocols that follow a similar

concept. The sensor nodes organize themselves into local clusters, with one node acting as the cluster head. LEACH utilizes the randomized rotation of cluster heads to balance the energy dissipation of nodes. The cluster heads not only collect data from their clusters, but also aggregate the collected data to reduce the amount of messages to send to the base station, which enhances the network lifetime. The sensor nodes elect themselves to be cluster heads at any given time with a certain probability at each interval. The election decision is made solely by each node independent of other nodes to minimize overhead in cluster head establishment. Each node first chooses a random number [0, 1]. The node becomes a cluster head for the current round if the number is less than the following threshold:

$$T(n) = \begin{cases} \frac{p}{1-p\left(r \bmod \frac{1}{p}\right)} & \text{if } n \in G, \\ 0 & \text{otherwise,} \end{cases}$$

where n is the given node, p is the desired percentage of cluster heads, r is the current round, and G is the set of nodes that have not been cluster heads in the last $1/p$ rounds. The new cluster head will broadcast its status to neighboring nodes. Note that the operation of LEACH is broken up into time steps, called rounds. These nodes will then determine the optimal cluster head and relay their desire to be in that particular cluster. The broadcast messages as well as cluster establishment messages are transmitted using CSMA. Following cluster establishment, cluster heads will create a transmission schedule and broadcast the schedule to all nodes in their respective cluster. The schedule consists of TDMA slots for each neighboring node. Note that this scheduling scheme allows for energy minimization as nodes similar to the schedule-based MAC. LEACH is completely distributed and requires no global knowledge of network. However, LEACH uses single-hop routing where each node can transmit directly to the cluster-head and the sink. Therefore, it is not applicable to networks deployed in large regions. Furthermore, the idea of dynamic clustering brings extra overhead, e.g. head changes, advertisements etc., which may reduce the gain in energy consumption.

• **TEEN** [78]: Threshold sensitive energy efficient sensor network protocol (TEEN) is a hybrid protocol of hierarchical clustering and data-centric approaches designed for time-critical applications. This hierarchical protocol responds to sudden changes in the sensed attributes. The network architecture is based on a hierarchical grouping where closer nodes form clusters and this process goes on the second level until base station is reached. After the clusters are formed, the cluster head broadcasts two thresholds to the nodes. These are hard and soft thresholds for sensed attributes. A node will report data only when the sensed value is beyond the hard thresholds or the change in the value is greater than the soft thresholds. Therefore, the number of packet transmissions is controlled by adjusting the hard and soft threshold values. However, TEEN cannot be applied for the applications where periodic reports are needed since the values of the attributes may not reach the threshold at all. Moreover, there is always a possibility that the sink may not be able to distinguish dead nodes from alive ones. In TEEN, the message propagation is accomplished by

cluster heads. If cluster heads are not in each other's transmission radius, the message will be lost. The adaptive threshold sensitive energy efficient sensor network protocol (APTEEN) [125] is an extension of TEEN and aims at both capturing periodic data collections and reacting to time-critical events. The architecture is same as in TEEN.

In location-based routing protocols, each node knows its own and its network neighbor's positions and the source of a message is informed about the position of the destination for the energy efficient routing paths. The location information is needed to calculate the distance between two particular nodes so that energy consumption can be estimated. For some location-based applications, the query can be diffused only to that particular region by using the location of sensors which will eliminate the number of transmission significantly. The location of nodes may be available using a small low-power GPS receiver [83]. The distance between neighboring nodes can be estimated on the basis of incoming signal strengths. One of the fundamental issues to use location-based routing is the availability of an accurate positioning system using GPS cards, which is inconceivable with current technology. Furthermore, many related works of location-based routing protocols define the communication cost as a function of geographic positions of the nodes, however, the information of geographic position is not sufficient to define the communication cost [99]. Note that it is almost impossible to model the wireless channel of the indoor office as the practical solution due to too many uncertainties we need to consider. Early works of location-based routing protocols are the minimum energy communication network (MECN) [82], geographic adaptive fidelity (GAF) [83], geographic and energy aware routing (GEAR) [84]. We remark that some of the protocols are originally designed for mobile ad hoc networks and consider the mobility of nodes during the design. Now, we will describe these protocols in details and highlight the key ideas.

• **GAF** [83]: GAF is an energy-aware location-based routing algorithm for mobile ad hoc networks. Each node uses its location using GPS equipment to associate itself with a point in the virtual grid. Nodes associated with the same point on the grid are considered equivalent in terms of the cost of packet routing. The main idea of GAF is the collaboration between the nodes to play different roles in each zone. There are three states defined in GAF: discovery for determining the neighbors in the grid, active reflecting participation in routing and sleep when the radio is turned off. The sleep time and related parameters are dependent on the applications and tuned during the routing process. The sleeping neighbors adjust their sleeping time by considering the routing fidelity and load balancing. Therefore, GAF substantially increases the network lifetime as the number of nodes increases. To handle the mobility, each node in the grid estimates its leaving time of grid and sends this to its neighbors. Note that before the leaving time of the active node expires, sleeping nodes wake up and one of them becomes active. Therefore, it keeps the network connected by maintaining a cluster header always in active mode for each region on its virtual grid. Simulation results show that GAF performs at least as well as a normal ad hoc routing protocol in terms of latency and packet loss and increases the lifetime of the network by saving energy. Furthermore, the cluster header does

not support aggregation or fusion as in the case of other hierarchical protocols.
• **GEAR** [84]: The main idea of GEAR is to use of geographic information while disseminating queries to appropriate regions since data queries often includes geographic attributes. Specifically, it restricts the number of interests in directed diffusion by only considering a certain region rather than sending the interests to the whole network. Each node keeps an estimated cost and a learning cost of reaching the destination through its neighbors. The estimated cost is a combination of residual energy and distance to destination. The learned cost is a refinement of the estimated cost that accounts for routing around holes in the network. A hole occurs when a node does not have any closer neighbor to the target region than itself. If there are no holes, the estimated cost is equal to the learned cost. The learned cost is propagated one hop back every time a packet reaches the destination so that route setup for next packet will be adjusted.
• **SPAN** [57]: SPAN selects some nodes as coordinators based on their positions. SPAN is similar to GAF [83] since it activates only a fraction of the nodes in a certain area at any given time. The coordinators form a network backbone used to forward messages. A node should become a coordinator if two neighbors of a non-coordinator node cannot reach each other directly or via one or two coordinators. The weakness of SPAN is that the energy consumption significantly increases as the number of nodes increases.
• **GeRaF** [85]: Geographic Random Forwarding (GeRaF) provides a complete solution combining the routing and CSMA/CA mechanism as MAC layer. It requires the location information of sensor nodes' and their neighbors'. Furthermore, the forwarding node is chosen among nodes that are awake at the time of the transmission request. Hence, the routing consumes more power and increases the latency of the network.

2.1.3 Standardizations

As we discussed in Sections 2.1.1 and 2.1.2, many MAC and routing protocols are developed for WSNs. However, most recent products of WSNs use standard-based networking and RF solutions. The recent release of standards from IEEE, Internet Engineering Task Force (IETF), and International Society of Automation (ISA) [20], brought the technology out of research labs and developed the numerous commercial products. In this section, we introduce the most practical and promising standards and discuss interesting protocols that are relevant to the standard works.
There have been many contributions to the standardized protocols for low-power devices such as ZigBee [5], IETF 6loWPAN [126], IETF routing over low power and lossy networks (ROLL) [127], WirelessHART [103], ISA SP-100 [20], IEEE 802.15.4 [8], and IEEE 802.11 [128]. In the following, we will describe these standards in details and highlight the key ideas.
• **IEEE 802.15.4** [8]: Many promising standards and an existing commercial system are based on the IEEE 802.15.4 standard. This standard is for a low data rate solution with efficient energy consumption and very low complexity. According to a

recent survey, this standard already represents more than 50% of the market [129]. IEEE 802.15.4 radio standard and ZigBee emerge as the prevalent choice for industrial and smart building applications. There are also many discussions of IEEE 802.15.4 for the routing over low power and lossy networks in the IETF working group [127]. Recently, many task groups launch IEEE 802.15.4 family for specific applications of WSNs. Summarizing, the task groups of IEEE 802.15.4:

- IEEE 802.15.4a
  It specifies two additional PHYs using Ultra-wideband (UWB) and Chirp Spread Spectrum (CSS) which is an amendment to IEEE 802.15.4 to provide communications and high precision ranging/location capability, high aggregate throughput, and ultra low power, scalability to data rates, longer range, and lower cost.

- IEEE 802.15.4e
  The intent of this amendment is to enhance and add functionality to the IEEE 802.15.4 MAC that are required to enable those application spaces: factory automation, process automation, asset tracking, general sensor control (industrial/commercial, including building automation), home medical health/monitor, telecom application, neighborhood area networks, and home audio. We remark that this task group will also include the multi-channel mechanism.

- IEEE 802.15.4f
  It define new wireless PHYs and enhancements to the IEEE 802.15.4 standard MAC layer which are required to support new PHY for active RFID system bi-directional and location determination applications.

- IEEE 802.15.4g
  It creates a PHY amendment to IEEE 802.15.4 to provide a global standard that facilitates very large scale process control applications such as the utility smart-grid [16] networks.

• **ZigBee** [5]: According to a recently published report by ON World [6], ZigBee is the winning protocol for wireless sensing and control by over 350 global manufacturers with combined annual revenues exceeding 1 trillion dollars. ZigBee market opportunities are expanding for smart energy, healthcare, retail, consumer electronics, building automation, and telecommunications through collaborations with other standards and industry groups. Furthermore, a Frost & Sullivan's survey [130] shows that ZigBee, Bluetooth [131], wireless local area network (WLAN) and other unlicensed technologies are widely adopted with respect to the standards like WirelessHART and ISA SP-100 for process industries in Europe, Middle East, and Africa. The ZigBee standard is prepared by the ZigBee alliance. ZigBee covers the networking layer and application layer of WSNs and is defined to work on top of a modified version of the IEEE 802.15.4 standard. ZigBee allows to create various kinds of networks: in star networks the ZigBee coordinator starts the network and

all the other network members are directly associated with the ZigBee coordinator. The ZigBee coordinator is co-located with the personal area network (PAN) coordinator of the underlying IEEE 802.15.4 network. In tree networks the ZigBee routers form a tree that is rooted at the ZigBee coordinator, whereas in mesh networks the network topology might be a general mesh involving ZigBee routers and the ZigBee coordinator.

• **WirelessHART** [103]: WirelessHART is a promising solution for the replacement of the wired HART protocol in industrial contexts. Power consumption is not a main concern in WirelessHART, whereas the data link layer is based on TDMA, which requires time synchronization and pre-scheduled fixed length time-slots by a centralized network manager. Such a manager should update the schedule frequently to consider reliability and delay requirements and dynamic changes of the network, which demands complex hardware equipments, and this is in contrast with the necessity of simple protocols able to work with limited energy and computing resources.

• **ISA SP-100** [20]: ISA SP-100 is currently working on a series of standards addressing the adoption of wireless technologies in different industries. ISA-SP100.11a addresses noncritical process applications that can tolerate delays up to 100 ms. Since it leverages the IEEE 802.15.4 standard, it inherits some of its properties: low-rates (up to 250 kbit/s) and low implementation complexity for simple end devices. In addition a data-link layer and an adaptation layer between MAC and data link layer are introduced. The data link layer controls the frequency hopping and adds a TDMA scheme. We remark that both standards [5, 20] target overlapping application areas and are based on the same underlying wireless technology.

• **IEEE 802.11** [128]: The IEEE 802.11 family of WLAN standards is composed of a number of specifications that primarily define the physical and MAC layers of the realm of WLAN technologies, and it has also been considered extensively in the context of wireless industrial communications, see [132, 133]. Similar to other standards from the IEEE 802.x series, the IEEE 802.11 MAC suggest the IEEE 802.2 logical link control (LLC) [134] as a standard interface to higher layers. Since IEEE 802.11 is a WLAN standard, its key intentions are to provide high throughput and a continuous network connection rather than energy efficiency of the network.

• **ROLL** [127]: ROLL is focused on routing issues for low power and lossy networks (LLNs). LLNs are made up of many embedded devices with limited power, memory, and processing resources. They are interconnected by a variety of links, such as IEEE 802.15.4, Bluetooth, low power WiFi, wired or other low power powerline communication (PLC) links. LLNs are transitioning to an end-to-end IP-based solution to avoid the problem of non-interoperable networks interconnected by protocol translation gateways and proxies. The working group focuses on routing solutions for a subset of these: industrial, connected home, building and urban sensor networks for which routing requirements have been specified. These application specific routing requirements will be used for protocol design. The framework will take into consideration various aspects including high reliability in the presence of time varying loss characteristics and connectivity while permitting low power operation with

very modest memory and CPU pressure in networks potentially comprising a very large number of nodes.

2.1.4 Discussion

In the previous section, we discussed many existing MAC and routing protocols. By considering the energy saving mechanism, we observe the ***explicit*** energy saving mechanism of MAC protocols and ***implicit*** energy saving mechanism of routing protocols. Recall that most routing protocols focus on how to reduce the traffic load of the network or the routing path rather than explicit sleep mechanism. In the following, we discuss some open issues and limitations on the protocol design of WSNs based on our protocol overview.

1. **Combination**
   A large number of MAC and routing protocols exist and will be developed. Therefore, the important question to answer is what combinations of the protocols give good performance for applications and what parameters must be shared and optimized among different layers to improve the performance. By considering the classifications of our protocol overview, it is natural that the contention-based MAC protocol supports the topology-based and data-centric routing protocols and some of location-based routing protocols. However, it is not clear how to combine other MAC and routing protocols under other different categories. One of the main reasons is that many MAC and routing protocols for WSNs cover more than the basic functionalities of its layer. Most schedule-based MAC protocols of WSNs support the packet forwarding mechanism for convergecast communication and even mesh communication [99]. For instance, the IEEE 802.15.4e task group suggests to use the routing protocol as the option when the packet forwarding mechanism fails. By a similar argument, many hierarchical-based routing protocol includes some mechanisms of MAC layer as well as routing mechanism. In particularly, the basic idea of the IEEE 802.15.4 standard [8] is similar to cluster-based routing protocols such as LEACH [77].

2. **Theoretical studies**
   Understanding the fundamental limitations of different protocols is important issue for the protocol designers of different layers. For instance, many contention-based MAC protocols show the very low duty-cycle of less than 1% without clarifying the traffic load and network setup. Careful exploitation of different layer interactions leads to more efficient network performance and hence better application performance. The cross-layer methodology [135] is one of the candidates since it considers the performance of communication and application jointly. It means formulating and solving an optimization problem involving a joint protocol stack design. However, the system dynamics representing the interactions among the protocols at the different layers is fairly

complex because of the existence of numerous parameters and the nonlinear nature of the protocol state machines at the different layers.

3. **Hardware factors**
   The hardware factors are essential issues to implement the protocols. In the following, we discuss some basic issues.

   Most existing protocols of WSN are designed for byte-stream or even bit-stream radios. However, the trend of state-of-the-art of platform shifts to packet-based radios such as CC2420 [136] used for the Tmote [31], MicaZ [32] and TelosB [137] node. With a packet-based radio, the transmission of a packet is done by radio chip because the micro-controller cannot afford the overhead of these operations. In these radios, the micro-controller has no control on each single transmitted bit over the air, it sends data to the radio that transforms it into a packet and sends it to the air interface. These radios perform the coding, physical preamble transmission, encryption, cyclic redundancy check (CRC), address filtering, automatic ACK transmission. Modulation schemes used by these radio are robust to noise but consume more power. Therefore, if the protocols are designed for byte-stream or even bit-stream radios, these protocols require a refinement for packet-based radios.

   The constraints on resources also involve memory and CPU. Typical sensors [31, 137] have in the order of 4KB of RAM. Typical TinyOS applications require $2-3$ KB of RAM out of these 4KB. 1KB can be allocated for packet buffering. It means a typical buffer can hold at most 10 packets. Hence, this memory limitation will affect the performance of MAC and routing protocols.

   Since the resources of the devices are strictly limited, there are many studies to apply the optimization tools for WSNs. However, the computation complexity and computing time to use the optimization tools are critical factors since the typical micro-controller [138] does not support well a heavy computing. Many of the cross-layer solutions proposed in the literature are hard to apply, because they require sophisticated processing resources, or instantaneous global network knowledge, which are out of reach of the capabilities of real nodes. As it was noted in [139], the complex interdependence of the decision variables (sleep disciplines, clustering, MAC, routing, power control, etc.) leads to difficult problems even in simple network topologies, where the analytical relations describing packet reception rate, delay and energy consumption may be highly nonlinear expressions.

   Recent radio chips are able to switch quickly between frequency channels. Many protocols based on multi-channel significantly increase the throughput and reliability, and decreases the latency of the network by reducing the contention [56, 53, 55].

4. **Heterogeneity**
   Prior researches tend to focus on specific protocol design for specific applications and platforms, leading to an explosion of the number of protocols.

However, many applications simultaneously need to share a common communication infrastructure. Therefore, the communication protocol needs to support not only heterogeneous platform but also heterogeneous applications. For instance, in building automation, the communication infrastructure covers all aspects of building system control including measurement and control message of HVAC and lighting control, and alarm message of security systems. The different application may set the different requirements to the performance of the communication infrastructure. As the heterogeneity increases, developing individual protocols will become exceedingly complicate and expensive.

5. **Correlated data**
   The physical phenomena monitored by densely deployed sensor nodes, e.g., environmental monitoring, home automation, usually yields the high degree of spatial and time correlation of the sensed data. This correlated data generates heavy traffic load to network, so performance such as reliability and latency significantly degrade. For instance, many contention-based protocols designed for monitoring applications achieve very low duty-cycle by assuming the low traffic load of the network. This assumption is reasonable by considering the average traffic load for general monitoring applications. However, this assumption does not capture the correlated data in the space-time dynamics. When the traffic load increases, the reliability of the preamble sampling based MAC protocols is not guaranteed anymore. In most cases, the correlated data are very critical events. With this in mind, researchers have recently designed a large number of protocols and algorithms supporting data fusion [42]. Therefore, it is essential to combine the preamble sampling based protocol which is common approaches for monitoring applications and data fusion techniques.

6. **Hybrid-based protocol**
   The hybrid-based protocol is an attractive solution since it exploits the advantages of both a contention-based and a schedule-based approach. This protocol may support different requirements of the different applications [8]. However, the critical question is how to combine the contention-based and schedule-based approach to achieve the good performance. As we discussed in Section 2.1.1, there are two main approaches: reservation-for-contention and partition approaches. We remark that the hybrid-based protocol requires very careful design, otherwise, the performance of the hybrid-based protocol is questionable due to the weakness of both the contention-based and schedule-based approaches.

2.2 Overview of the IEEE 802.15.4

As we discussed in Section 2.1.3, the IEEE 802.15.4 standard is appealing for many different applications and is the dominant protocol in the real market of WSNs [6]. In this section, we give an overview of the key points of IEEE 802.15.4 protocol.

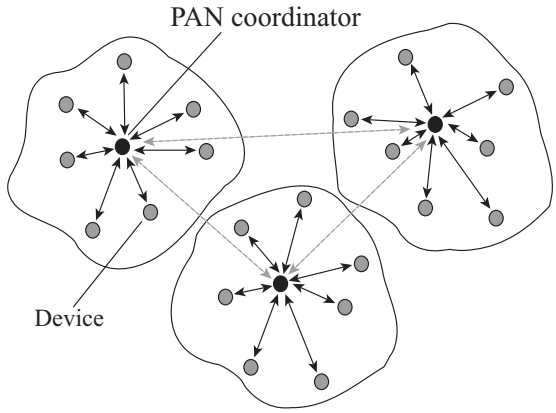


Figure 2.3: Typical star network topology of IEEE 802.15.4. The packets generated by the sensor nodes (grey circle) are transmitted toward the PAN coordinator (black circle) depicted in the middle of each cluster.

The standard specifies the physical layer and the MAC sublayer for low-rate wireless networks. The star network is a basic network topology presented in Figure 2.3, where all N nodes contend to send data to the PAN coordinator, which is the data sink. The standard defines two channel access modalities: the beacon-enabled modality, which uses a slotted CSMA/CA and the optional guaranteed time slot (GTS) allocation mechanism, and a simpler unslotted CSMA/CA without beacons. The communication is organized in temporal windows denoted superframes. Figure 2.4 shows the superframe structure of the beacon-enabled mode. In the following, we focus on the beacon-enabled modality.
The network coordinator periodically sends beacon frames in every beacon interval T_{BI} to identify its PAN and to synchronize nodes that communicate with it. The coordinator and nodes can communicate during the active period, called the superframe duration T_{SD}, and enter the low-power mode during the inactive period. The structure of the superframe is defined by two parameters, the beacon order (BO) and the superframe order (SO), which determine the length of the superframe and its active period, respectively, they are

$$T_{\mathrm{BI}} = \mathit{aBaseSuperframeDuration} \times 2^{\mathrm{BO}}\,, \tag{2.1}$$

$$T_{\mathrm{SD}} = \mathit{aBaseSuperframeDuration} \times 2^{\mathrm{SO}}\,, \tag{2.2}$$

where $0 \leq \mathrm{SO} \leq \mathrm{BO} \leq 14$ and *aBaseSuperframeDuration* is the number of symbols forming a superframe when SO is equal to 0. In addition, the superframe is divided into 16 equally sized superframe slots of length *aBaseSlotDuration*. Each active period can be further divided into a contention access period (CAP) and an optional contention free period (CFP), composed of GTSs. A slotted CSMA/CA mechanism is used to access the channel of non time-critical data frames and GTS requests during the CAP. In the CFP, the dedicated bandwidth is used for time-critical data

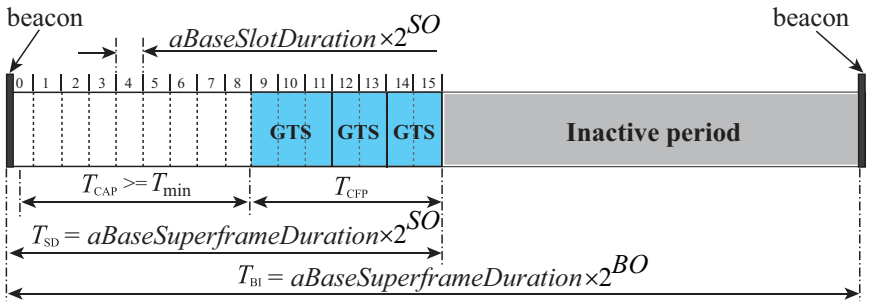
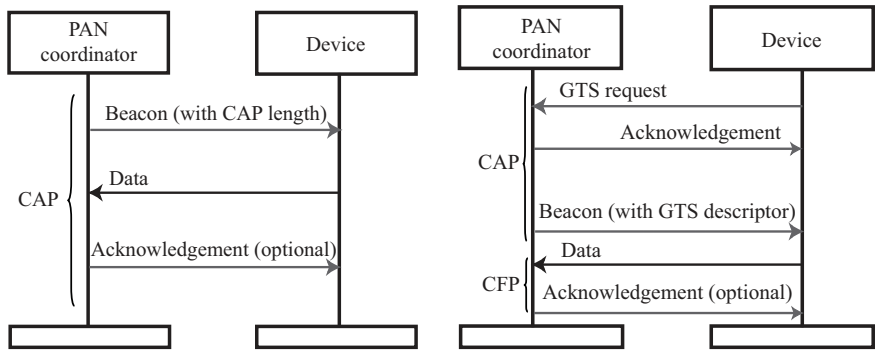


Figure 2.4: Superframe structure of IEEE 802.15.4.

frames. Figure 2.5 illustrates the date transfer mechanism of the beacon-enabled mode for the CAP and CFP. In the following section, we describe the data transmission mechanism for both CAP and CFP.

2.2.1 CSMA/CA mechanism of CAP

Consider a node trying to transmit a data packet during CAP. In slotted CSMA/CA of IEEE 802.15.4, first the MAC sublayer initializes four variables, i.e., the number of backoffs (NB=0), contention window (CW=2), backoff exponent (BE=*macMinBE*) and retransmission times (RT=0). Then the MAC sublayer delays for a random number of complete backoff periods in the range $[0, 2^{\mathrm{BE}} - 1]$ units. If the number of backoff periods is greater than the remaining number of backoff periods in the CAP, the MAC sublayer pauses the backoff countdown at the end of the CAP and resumes it at the start of the CAP in the next superframe. Otherwise the MAC sublayer counts its backoff delay. When the backoff period is zero, the node needs to perform the first clear channel assessment (CCA). The MAC sublayer proceeds if the remaining CSMA/CA algorithm steps (i.e., two CCAs), the frame transmission, and any ACK can be completed before the end of the CAP. If the MAC sublayer cannot proceed, it waits until the start of the CAP in the next superframe and apply a further random backoff delay in the range $[0, 2^{\mathrm{BE}} - 1]$ units before evaluating whether it can proceed again. Otherwise the MAC sublayer proceeds the CCA in the current superframe. If two consecutive CCAs are idle, then the node commences the packet transmission. If either of the CCA fails due to busy channel, the MAC sublayer increases the value of both NB and BE by one, up to a maximum value *macMaxCSMABackoffs* and *macMaxBE*, respectively. Hence, the values of NB and BE depend on the number of CCA failures of a packet. Once BE reaches *macMaxBE*, it remains at the value *macMaxBE* until it is reset. If NB exceeds *macMaxCSMABackoffs*, then the packet is discarded due to channel access failure. Otherwise, the CSMA/CA algorithm generates a random number of complete backoff periods and repeats the process. Here, the variable *macMaxCSMABackoffs* represents the maximum number of times the CSMA/CA algorithm is required to



(a) Non time-critical data packet or GTS request transmission.

(b) Time-critical data packet transmission.

Figure 2.5: Data transfers of beacon-enabled mode during the CAP and CFP.

backoff. If channel access is successful, the node starts transmitting packets and waits for an ACK. The reception of the corresponding ACK is interpreted as successful packet transmission. If the node fails to receive ACK due to collision or ACK timeout, the variable RT is increased by one up to *macMaxFrameRetries*. If RT is less than *macMaxFrameRetries*, the MAC sublayer initializes two variables CW=0, BE=*macMinBE* and follows the CSMA/CA mechanism to re-access the channel. Otherwise the packet is discarded due to the retry limit. Note that the default MAC parameters are $macMinBE = 3, macMaxBE = 5, macMaxCSMABackoffs = 4, macMaxFrameRetries = 3$. See [8] for further details.

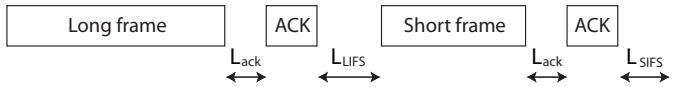
Figure 2.6 describes the date transmission with inter-frame spacing (IFS) period with and without ACKs. By knowing the duration of an ACK frame, ACK timeout, IFS, data packet length, and header duration, we define the successful packet transmission time L_s and the packet collision time L_c with ACK and the successful packet transmission time L_g without ACK as

$$L_s = L_p + L_{\mathrm{w,ack}} + L_{\mathrm{ack}} + L_{\mathrm{IFS}}\,, \tag{2.3}$$

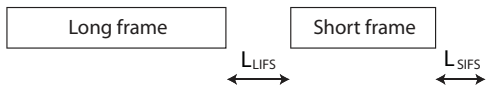
$$L_c = L_p + L_{\mathrm{m,ack}}\,, \tag{2.4}$$

$$L_g = L_p + L_{\mathrm{IFS}}\,, \tag{2.5}$$

where L_p is the total packet length including overhead and payload, $L_{\mathrm{w,ack}}$ is ACK waiting time, L_{ack} is the length of the ACK frame, L_{IFS} is the IFS time, and $L_{\mathrm{m,ack}}$ is the timeout of the ACK. To account for the data processing time required at the MAC sublayer, two successive frames transmitted from a node are separated by at least an IFS period; if the first transmission requires an ACK, the separation between the ACK frame and the second transmission is at least an IFS period. Note that the waiting time to receive ACK is in the range *aTurnaroundTime* (12 symbols) to *aTurnaroundTime* + *aUnitBackoffPeriod* (12 + 20 symbols). The IFS period depends on the length of the transmitted data frames.



(a) Acknowledged transmission



(b) Unacknowledged transmission

Figure 2.6: Data frame transmission mechanism with and without ACK.

2.2.2 GTS allocation of CFP

The coordinator is responsible for the GTS allocation and determines the length of the CFP in a superframe. To request the allocation of a new GTS, the node sends the GTS request command to the coordinator. The coordinator confirms its receipt by sending an ACK frame within CAP. Upon receiving a GTS allocation request, the coordinator checks whether there are sufficient resources and, if possible, allocates the requested GTS. We recall that Figure 2.5(b) illustrates the GTS allocation mechanism. The GTS capacity in a superframe satisfies the following requirements:

1. The maximum number of GTSs to be allocated to nodes is seven, provided there is sufficient capacity in the superframe.

2. The minimum length of a CAP $T_{\min}$ is *aMinCAPLength* .

Therefore the CFP length depends on the GTS requests and the current available capacity in the superframe. If there is sufficient bandwidth in the next superframe, the coordinator determines a node list for GTS allocation based on a first-come-first-served (FCFS) policy. Then, the coordinator includes the GTS descriptor which is the node list that obtains GTSs in the following beacon to announce the allocation information. The coordinator makes this decision within *aGTSDescPersistenceTime* superframes. Note that on receipt of the ACK to the GTS request command, the node continues to track beacons and waits for at most *aGTSDescPersistenceTime* superframes. A node uses the dedicated bandwidth to transmit the packet within the CFP. In addition, a transmitting node ensures that its transaction is complete one IFS period before the end of its GTS.

2.3 Networked Control Systems

Networked control systems (NCSs) are spatially distributed systems in which the sensors, actuators, and controllers connect through a communication network instead by traditional point-to-point connections, as shown in Figure 2.7. The significant advantages over traditional control architectures include reduced wiring and

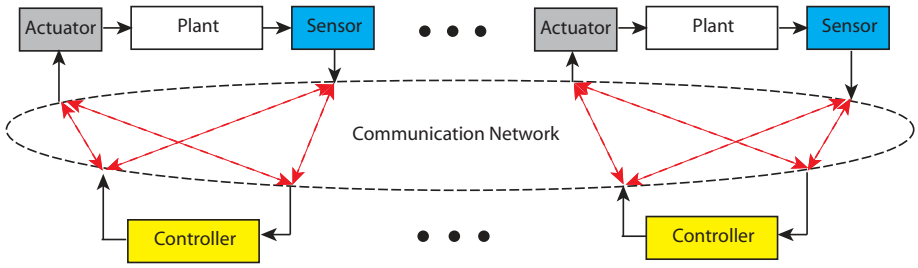


Figure 2.7: General networked control systems structure.

cost, increased modularity, easier maintenance, and high flexibility and reconfigurability [1, 21]. Networked control has become an enabling technology for many military, commercial and industrial applications such as mobile sensor networks [140], remote surgery [141], industrial automation [142]. Wireless communication is playing an increasingly important role in NCSs. Transmitting sensor measurements and control commands over wireless links allows rapid deployment, flexible installation, fully mobile operation and prevents cable problems in the control applications.
Figure 2.7 depicts the general structure of NCSs where a plant is remotely commissioned over a network. Outputs of the plant are sampled at periodic or aperiodic intervals by the sensor and forwarded to the controller through a network. When the controller receives the measurements, a new control command is computed. The control is forwarded to the actuator attached to the plant. Research on NCSs sometimes considers structures simpler than the general one depicted in Figure 2.7. For example, many practical NCSs have several sensing channels and the controllers are collocated with the actuators, as in heat, ventilation and air-conditioning control systems [143]. It is also common to consider single feedback loops closed over a network [144, 145].
Many papers have been written about networked control: extensive research on the impact on system performance and stability of the network and protocols can be found in [24, 146, 147]. Design methods on how to achieve high performance of control systems through a communication network have been recently proposed. The approaches can be grouped in two categories: design of the control algorithm and design of the communication protocol. There has been much research effort to design robust controllers and estimators that are adaptive and robust to the communication faults: packet dropout as a Bernoulli random process [25] or with deterministic rate [148], packet delay [149], and data rate limitation [150]. On the other side, communication protocols and their parameters are designed in order to achieve a given control performance. In [151], the authors present a scheduling policy to minimize a linear quadratic (LQ) cost under computational delays. In [152], the authors proposed an adaptive tuning scheme of the parameters of the link layer, MAC layer and sampling period through numerical results in order to minimize the LQ cost. In [153], the tournaments based MAC layer are presented as a way to give higher priorities for the critical systems in a distributed manner. However, these

approaches often consider only one aspect of the network faults: packet dropout [25],[148], packet delay [149, 151], or data rate limitation [150]. In [152], although the authors consider the simulation results of the wireless network, the framework has not been designed out of an analytical consideration of control performance. Furthermore, accurate modeling of communication networks requires heavy computation load and can still be hard. Regarding protocol design for communication networks, there has been much research on deterministic performance of control networks using token passing bus and controller area network (CAN) bus architectures. Comparatively, much less work on wireless NCSs has considered protocols for the recently developed standard such as IEEE 802.11 [128], 802.15.4 [8], and Bluetooth [131]. Even though many communication protocols are available in [38] and [40], these protocols are designed mainly to achieve high reliability and high energy efficiency for various applications of WSNs and not specifically for control applications. We summarize here the important network quality measures for NCSs.

- **Bandwidth**: Bandwidth is the information-carrying capacity of a communication channel. There is recent research on the problem of determining the minimum bit rate that is needed to stabilize a linear system [154, 155]. The data rate of a network must be considered together with the packet size and overhead since data are encapsulated into packets. Notice that the size of the headers depends on the protocol design of the communication network. In particular, WSNs have low data rate with respect to other wireless communications such as WLAN, Bluetooth, etc.

- **Sampling and Delay**: The time delay of data on the network is the total time between the data being available at the source node (e.g., sampled from sensors) and being available at the sink node (e.g., received at the controller). This process is significantly different from the usual periodic sampling in digital control system. The overall delay between sampling and receiving can be highly variable because both the network access delays (i.e., the time it takes for a shared network to accept data) and the transmission delays (i.e., the time during which data are in transit inside the network) depend on highly variable network conditions such as congestion and channel quality. In some NCSs, the data transmitted are time stamped, which means that the receiver may have an estimate of the delay duration and can take an appropriate corrective action. Many research results have attempted to characterize a maximum upper bound on the sampling interval for which stability can be guaranteed. These results implicitly attempt to minimize the packet rate that is needed to stabilize a system through feedback. Furthermore, the delay jitter needs to be considered since it can be much more difficult to compensate for, especially if the variability is large.

- **Packet Dropout**: Another significant factor is the reliability of the network in NCSs compared to standard digital control system, e.g., packet loss of the wireless channel. Packet dropouts result from communication errors in

the physical layer of the wireless link or from buffer overflows due to congestion of the network. Longer forward delays result in packet reordering, which essentially amounts to a packet dropout if the receiver discards "outdated" arrivals which is also critical factor for time-bounded traffic such as audio and video [156]. Reliable transmission protocols, such as MMSPEED [86] and RMST [157] of WSNs, guarantee the eventual delivery of packets by using an acknowledgement mechanism. However, these protocols may not be appropriate for NCSs since repeated retransmissions of old data is generally not useful for control applications. Maximizing the reliability may increase substantially the network energy consumption [21]. Furthermore, in general, there is tradeoff between the reliability of a network and the delay of packet delivery [33]. We remark here that controllers can usually tolerate a certain degree of packet losses and delay [22, 24, 25, 28].

Chapter 3

Adaptive IEEE 802.15.4 Protocol for Energy Efficient, Reliable and Timely Communications

The IEEE 802.15.4 MAC protocol for wireless sensor networks can support energy efficient, reliable, and timely packet transmission by tuning the medium access control parameters *macMinBE*, *macMaxCSMABackoffs*, and *macMaxFrameRetries*. Such a tuning is difficult, because simple and accurate models of the influence of these parameters on the probability of successful packet transmission, packet delay, and energy consumption are not available. Moreover, it is not clear how to adapt the parameters to the changes of the network and traffic regimes by algorithms that can run on resource-constrained nodes. In this chapter, a generalized Markov chain is proposed to model these relations by simple expressions without giving up the accuracy. In contrast to previous work, the presence of limited number of retransmissions, acknowledgments, unsaturated traffic, and packet size is accounted for. The model is then used to derive an adaptive algorithm for minimizing the power consumption while guaranteeing a given successful packet reception probability and delay constraints in the packet transmission. The algorithm does not require any modification of the IEEE 802.15.4 MAC and can be easily implemented on network nodes. The protocol has been experimentally implemented and evaluated on a test-bed with off-the-shelf wireless sensor nodes. Experimental results show that the analysis is accurate, that the proposed algorithm satisfies reliability and delay constraints, and that the approach ensures a longer lifetime of the network under both stationary and transient network conditions.

Recall that in Section 2.2, we give a brief overview of the CSMA/CA algorithm of IEEE 802.15.4 MAC. In this chapter, we focus on the slotted CSMA/CA mechanism of beacon-enabled modality. The remainder of this chapter is as follows. Section 3.2 summarizes existing work of analytical modeling and adaptive tuning of the slotted CSMA/CA mechanism of the CAP. Section 3.3 lists the main contributions of the chapter and their relation to the literature. In Section 3.4, we propose a generalized

Markov chain model of CSMA/CA with retry limits and unsaturated traffic regime. In Section 3.5, the optimization problem to adapt the MAC parameters is investigated. In addition, implementation issues are also discussed. Numerical results achieved during stationary and transitionary conditions are reported in Section 3.6. Finally, Section 3.7 summarizes the chapter.

3.1 Motivation

It is known that IEEE 802.15.4 may have poor performance in terms of power consumption, reliability and delay [158], unless the MAC parameters are properly selected. It follows that (a) it is essential to characterize the protocol performance limitations, and (b) to develop methods to tune the IEEE 802.15.4 parameters to enhance the network lifetime and improve the quality of the service experienced by the applications running on top of the network.
This chapter focuses on the modeling and optimization of the performance metrics (reliability, delay, power consumption) for IEEE 802.15.4 WSNs. This problem is specially appealing for many control and industrial applications [1, 22]. We show that existing analytical studies of IEEE 802.15.4 are not adequate to capture the real-world protocol behavior, when there are retry limits to send packets, ACKs, and unsaturated traffic. We derive and use a new model to pose an optimization problem where the objective function is the power consumption of the nodes, the constraints are the reliability and delay of the packet delivery. In particular, the analytical model describes the relation between the protocol parameters and performance indicators in terms of reliability, delay, and power consumption. The main idea of the proposed analysis is to tradeoff the power consumption of the network with the application requirements in terms of reliability and delay. In other words, our goal is to optimize the network behavior by considering the given constraints imposed by the application instead of just improving the reliability, delay or energy efficiency separately.

3.2 Related Work

The modeling of IEEE 802.15.4 is related to IEEE 802.11 [128]. We first discuss the literature concerning the analysis of the slotted CSMA/CA algorithm of IEEE 802.11 and 802.15.4, then we review previous work about adaptive MAC mechanisms for these protocols.

3.2.1 Analytical Model of MAC

The basic functionalities of the IEEE 802.11 MAC has been modelled by Bianchi with a Markov chain under saturated traffic and ideal channel conditions [159]. Extensions of this model have been used to analyze the reliability [160], the delay [161], the MAC layer service time [162], and throughput [163] of IEEE 802.11.

The analysis of the packet delay, throughput, and power consumption of the IEEE 802.15.4 MAC has been the focus of several simulation-based studies, e.g., [164, 165], and some recent theoretical works, e.g., [166]–[171]. Inspired by Bianchi's work, a Markov model for IEEE 802.15.4 and an extension with ACK mechanism have been proposed in [166] and [158]. A modified Markov model including retransmissions with finite retry limits has been studied in [167] as an attempt to model the CSMA/CA mechanism. However, simulations show that this analysis gives inaccurate results in terms of power consumption and throughput under unsaturated traffic with finite retry limits. In [168], the authors consider finite retry limits with saturated traffic condition, which is not a realistic scenario for WSN applications.
While the previous works [166]–[168] do not consider the active and inactive periods of the IEEE 802.15.4 MAC, this aspect is investigated in [169]–[171]. In [169], the authors assume unsaturated traffic and source nodes with a finite buffer, but only uplink communications. In [170], instead, downlink communications are also taken into account, but nodes are assumed to have infinite size buffers. Furthermore, in [170], the power consumption, reliability, and delay performance are not investigated. In [171], a throughput analysis has been performed by an extension of the Markov chain proposed in [170], where the superframe structure, ACK, and retransmissions are considered. However, in [171] it is also shown that the models in [169] and [170], although very detailed, fail to agree with simulation results. Moreover, the Markov chain proposed in [171] does not model the length of data and ACK packets, which is crucial to analyze the performance metrics for IEEE 802.15.4 networks with low data rate. In [172], the authors approximate the CAP as the simple nonpersistent CSMA, which is a similar approach to [173]. However, the authors assume that the entire superframe duration is active, that is, SO = BO without considering the inactive period, which is not realistic. A coexistence of IEEE 802.11 and IEEE 802.15.4 standards is considered in [174] using a similar type of Markov chain model of [171].
A multihop network scenario is considered in [175]–[177]. In [175], the authors extend the framework proposed in [172] to two-hop network scenarios, where sensors communicate with the coordinator through an intermediate relay node, which simply forwards the data packets. In [176] and [177] the authors propose the use of a relay for interconnecting two IEEE 802.15.4 clusters and analyze the performance based on queuing theory.

3.2.2 Adaptive Tuning of MAC

Several algorithms to tune the MAC of IEEE 802.11 and IEEE 802.15.4 protocols have been proposed to improve the performance of these protocols in terms of reliability, throughput, or delay. The algorithms can be grouped in those based on the use of physical layer measurements, and those based on the use of link-layer information.
An adaptive tuning based on physical layer measurements has been investigated in [173]–[179], where a p-persistent IEEE 802.11 protocol has been considered to

optimize the average backoff window size. The channel access probability p that maximizes the throughput or minimize the power consumption is derived. This algorithm and its scalability to the network size have been studied also for IEEE 802.15.4 [178]. However, that study was less successful, because the channel sensing mechanism, the optional ACK, and retransmission mechanisms are hard to be approximated by a p-persistent MAC. Furthermore, in [178] and [179] a saturated traffic regime is assumed, which is a scenario of reduced interest for typical WSN applications.

Link-based optimizations for IEEE 802.11 and 802.15.4 have been investigated in [180]–[184], where simple window adjustment mechanisms that are based on ACK transmissions have been considered. In these papers, the algorithms adapt the contention window size depending on the successful packet transmission, packet collision and channel sensing state, but the algorithms are not grounded on an analytical study. In [180], different backoff algorithms are presented to improve the channel throughput and the fairness of channel usage for IEEE 802.11. A fair backoff algorithm is also studied in [181] and [182]. A link-based algorithm of the IEEE 802.15.4 random backoff mechanism to maximize the throughput has been presented in [183]. In [184], a dynamic tuning algorithm of the contention window size is evaluated on goodput, reliability, and average delay. An IEEE 802.15.4 enhancement based on the use of link-layer information has some drawback. First of all, it requires a modification of the standard. Although link-based mechanisms are simple to implement, the ACK mechanism may be costly since it introduces large overhead for small size of packets. For instance, alarm messages in industrial control application are a single byte whereas the ACK has a size of 11 byte. In addition, the ACK mechanism requires extra waiting time. Moreover, link-based algorithms adapt the MAC parameters for each received ACK, which mean a slow and inefficient adaptation to the dynamical changes of the network such as traffic, channel variations, and network topology.

3.3 Original Contribution

We consider a star network with a PAN coordinator, and N nodes transmitting toward the PAN. These nodes use the beacon-enabled slotted CSMA/CA and ACK. The parameters of the CSMA/CA algorithm that influence reliability, delay and energy consumption are the minimum value of the backoff exponent *macMinBE*, the maximum number of backoffs before declaring a channel access failure *macMaxCSMABackoffs*, and the maximum number of retries allowed after a transmission failure *macMaxFrameRetries* that each node can select.

In this chapter, we propose a novel modeling and adaptive tuning of the IEEE 802.15.4 MAC for reliable and timely communication while minimizing the energy consumption. The protocol is adjusted dynamically by a constrained optimization problem that each node of the network solves. The objective function, denoted by $\widetilde{E}_{\text{tot}}$, is the total energy consumption for transmitting and receiving packets of a

node. The constraints are given by the reliability and average delay. The constrained optimization problem for a generic transmitting node in the network is

$$\min_{\mathbf{V}} \quad \widetilde{E}_{\text{tot}}(\mathbf{V}) \tag{3.1a}$$

$$\text{s.t.} \quad \widetilde{R}(\mathbf{V}) \geq R_{\min}\,, \tag{3.1b}$$

$$\mathbb{E}[\widetilde{D}(\mathbf{V})] \leq D_{\max}\,, \tag{3.1c}$$

$$\mathbf{V}_0 \leq \mathbf{V} \leq \mathbf{V}_m\,. \tag{3.1d}$$

The decision variables of the node $\mathbf{V} = (m_0, m, n)$ are

$$m_0 \triangleq \mathit{macMinBE}\,,$$
$$m \triangleq \mathit{macMaxCSMABackoffs}\,,$$
$$n \triangleq \mathit{macMaxFrameRetries}\,.$$

$\widetilde{R}(\mathbf{V})$ is the reliability, and $R_{\min}$ is the minimum desired probability for successful packet delivery. $\mathbb{E}[\widetilde{D}(\mathbf{V})]$ is the average delay for a successfully received packet, and $D_{\max}$ is the desired maximum average delay. The constraint $\mathbf{V}_0 \leq \mathbf{V} \leq \mathbf{V}_m$ captures the limited range of the MAC parameters. In the problem, we used the symbol $\widetilde{\ }$ to evidence that the energy, reliability, and delay expression are approximations. We will show later that we use approximations of high accuracy and reduced computational complexity so that nodes can solve the problem.

Main contributions of the chapter are the following: (a) the modeling of the relation between the MAC parameters of IEEE 802.15.4 and the selected performance metrics, (b) the derivation of simple relations to characterize the operations of the MAC by computationally affordable algorithms, (c) formulation and solution of a novel optimization problem for the MAC parameters, (d) the practical implementation of the optimization by an adaptive algorithm on a test-bed using TelosB sensors [137]. and (e) performance evaluations of the algorithm by experiments and simulations of both stationary and transient network conditions.

Unlike previous work, we propose a generalized Markov model of the exponential backoff process including retry limits, acknowledgements and unsaturated traffic regime. However, the numerical evaluation of these performance metrics asks in general for heavy computations. This is a drawback when using them to optimize the IEEE 802.15.4 MAC parameters by in-network processing [185], because a complex computation is out of reach for resource limited sensing devices. Therefore, we devise a simplified and effective method that reduces drastically the computational complexity while ensuring a satisfactory accuracy.

Based on our novel modeling, we propose an adaptive tuning of MAC parameters that uses the physical layer measurement of channel sensing. This adaptive IEEE 802.15.4 is furnished with two distinctive features: it does not require any modification of the existing standard, and it makes an optimization of all the MAC parameters of IEEE 802.15.4. Specifically, in contrast to link-based adaptation [180]–[184], our algorithm does not require ACK mechanism or RTS/CTS handshakes

(or related standard modification). In contrast to [173]–[179], we do not use the (inaccurate) p-persistent approximation and the modification of the standard therein proposed, and we do not require any hardware modification to make an estimate of the signal-to-noise ratio. Our adaptive tuning optimizes the considered MAC parameters, all at once, and not only some of them, as proposed in [173]–[184].
The proposed adaptive IEEE 802.15.4 improves the power efficiency substantially while guaranteeing reliability and delay constraints. The adaptation is achieved by distributed asynchronous iterations that only require channel condition information, the number of nodes of the network, and the traffic load. We show that the convergence is fast and robust to errors in the estimation of the channel condition, number of nodes, and traffic load. Moreover, a fairness indicator is used to show that the sensor nodes receive a fair share of the communication resources, namely a good fairness is achieved.

3.4 Analytical Modeling

In a star network, all N nodes contend to send data to the PAN coordinator, which is the data sink. We remark that a star network is an important network topology, as considered in many of the papers from the literature mentioned above, a number of important standardization groups [99, 186], and commercial products [187, 188] for applications such as asset tracking, process control, and building automation. Throughout this chapter we consider applications where nodes asynchronously generate packets with probability η when a node sends a packet successfully or discard a packet or the sampling interval is expired. Otherwise a node stays for hT_b s without generating packets with probability $1-\eta$, where h is an integer and T_b is the time unit *aUnitBackoffPeriod* (corresponding to 20 symbols). The data packet transmission is successful if an ACK packet is received.
In such a scenario, we propose an effective analytical model of the slotted CSMA/CA by a Markov chain. The chain gives us the objective function, energy (3.1a), and constraints on reliability (3.1b) and delay (3.1c) of the optimization problem. Experimental results validate the proposed model.

3.4.1 Markov Chain Model

In this section, we develop a generalized Markov chain model of the slotted CSMA/-CA mechanism of beacon-enabled IEEE 802.15.4. Compared to previous results, e.g., [158], [166]–[171], the novelty of this chain consists in the modeling of the retry limits for each packet transmission, ACK, the inclusion of unsaturated traffic regimes, and packet size.
Let $s(t), c(t)$ and $r(t)$ be the stochastic processes representing the backoff stage, the state of the backoff counter and the state of retransmission counter at time t experienced by a node to transmit a packet. By assuming independent probability that nodes start sensing, the stationary probability τ that a node attempts a first carrier sensing in a randomly chosen slot time is constant and independent of

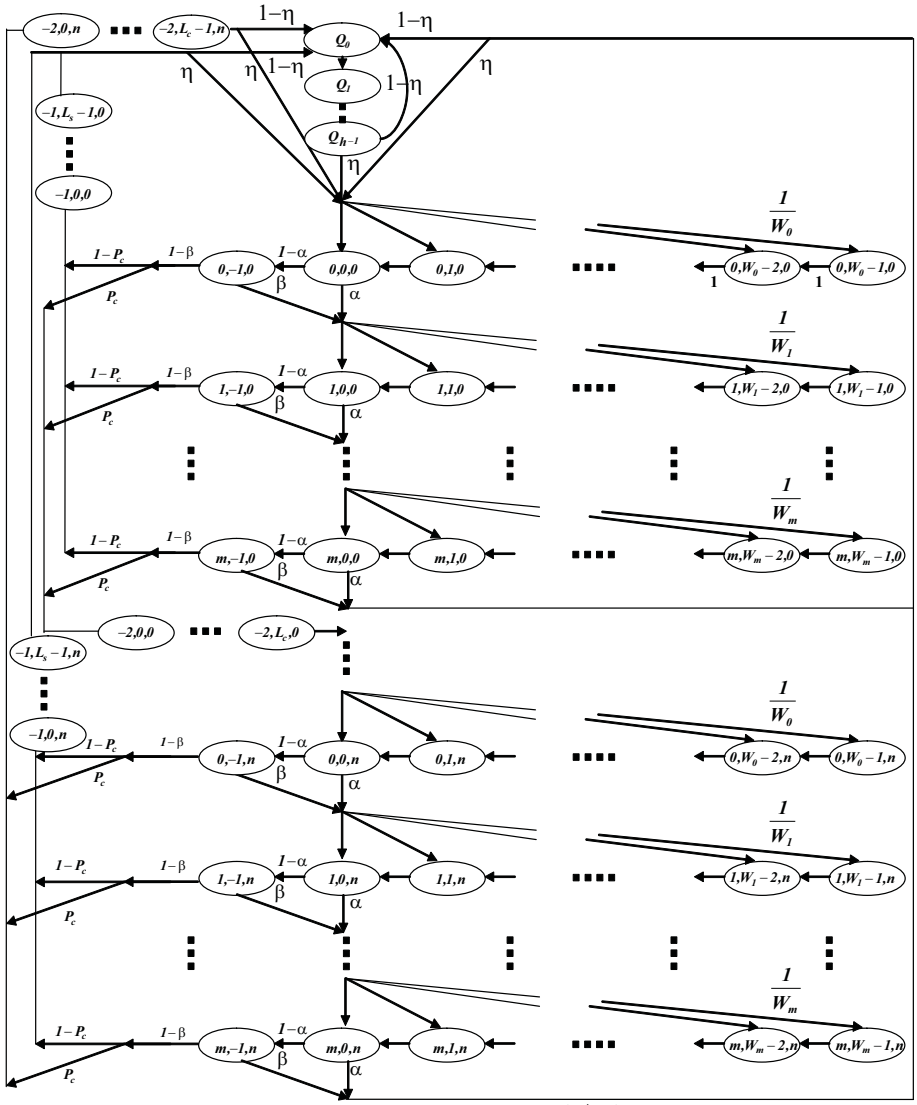


Figure 3.1: Markov chain model for CSMA/CA of IEEE 802.15.4.

$$b_{0,0,0} = \begin{cases} \left[\frac{1}{2}\left(\frac{1-(2x)^{m+1}}{1-2x}W_0 + \frac{1-x^{m+1}}{1-x}\right)\frac{1-y^{n+1}}{1-y} + (1-\alpha)\frac{1-x^{m+1}}{1-x}\,\frac{1-y^{n+1}}{1-y} + \right. \\ (L_s(1-P_c)+L_cP_c)(1-x^{m+1})\frac{1-y^{n+1}}{1-y} + h\frac{1-\eta}{\eta}\left(\frac{x^{m+1}(1-y^{n+1})}{1-y}\right. \\ \left. \left. +P_c(1-x^{m+1})y^n + (1-P_c)\frac{(1-x^{m+1})(1-y^{n+1})}{1-y}\right)\right]^{-1} \\ \qquad\qquad \text{if} \quad m \le m_b - m_0 \\ \left[\frac{1}{2}\left(\frac{1-(2x)^{m_b-m_0+1}}{1-2x}W_0 + \frac{1-x^{m_b-m_0+1}}{1-x} + (2^{m_b}+1)x^{m_b-m_0+1}\frac{1-x^{m-m_b+m_0}}{1-x}\right)\right. \\ \times\,\frac{1-y^{n+1}}{1-y} + (1-\alpha)\frac{1-x^{m+1}}{1-x}\,\frac{1-y^{n+1}}{1-y} + (L_s(1-P_c)+L_cP_c)(1-x^{m+1}) \\ \times\frac{1-y^{n+1}}{1-y} + h\frac{1-\eta}{\eta}\left(\frac{x^{m+1}(1-y^{n+1})}{1-y} + P_c(1-x^{m+1})y^n\right. \\ \left. \left. +(1-P_c)\frac{(1-x^{m+1})(1-y^{n+1})}{1-y}\right)\right]^{-1} \qquad \text{otherwise} \end{cases} \tag{3.2}$$

other nodes. The triple $(s(t), c(t), r(t))$ is the three-dimensional Markov chain in Figure 3.1, where we use (i, k, j) to denote a particular state. We denote the MAC parameters by $\mathbf{V} = (m_0, m, n), m_b \triangleq macMaxBE, W_0 \triangleq 2^{m_0}, W_m \triangleq 2^{\min(m_0+m, m_b)}$. The Markov chain consists of four main parts corresponding to the idle-queue states, backoff states, CCA states, and packet transmission states. The states $(Q_0, \ldots, Q_{h-1})$ correspond to the idle-queue states when the packet queue is empty and the node is waiting for the next packet generation time. Hence, the node sets its radio to sleep mode during the idle-queue states. Note that the idle-queue states $(Q_0, \ldots, Q_{h-1})$ take into account the sampling interval. The states from $(i, W_m - 1, j)$ to $(i, W_0 - 1, j)$ represent the backoff states. The radio circuits of the node is set in idle mode or in sleep mode during the backoff period. The states $(i, 0, j)$ and $(i, -1, j)$ represent first CCA (CCA_1) and second CCA (CCA_2), respectively. Let α be the probability that CCA_1 is busy, and β the probability that CCA_2 is busy. The states $(-1, k, j)$ and $(-2, k, j)$ correspond to the successful transmission and packet collision, respectively. Recall that we define the successful packet transmission time L_s and the packet collision time L_c with ACK in Eqs (2.3) and (2.4), respectively.
We have the following results:

Lemma 1. *Let the stationary probability of the Markov chain in Figure 3.1 be*

$$b_{i,k,j} = \lim_{t\to\infty} P(s(t) = i, c(t) = k, r(t) = j),$$

where $i \in (-2, m), k \in (-1, \max(W_i - 1, L_s - 1, L_c - 1)), j \in (0, n)$. *Then, for* $0 \le i \le m$

$$b_{i,k,j} = \frac{W_i - k}{W_i} b_{i,0,j}\,, \quad 0 \le k \le W_i - 1\,, \tag{3.3}$$

where

$$W_i = \begin{cases} 2^i W_0, & i \leq m_b - m_0 \,, \\ 2^{m_b}, & i > m_b - m_0 \,, \end{cases}$$

and

$$b_{i,0,j} = \left[(1-\alpha)(1-\beta) P_c \sum_{i=0}^{m} (\alpha + (1-\alpha)\beta)^i \right]^j (\alpha + (1-\alpha)\beta)^i b_{0,0,0} \,, \tag{3.4}$$

where $b_{0,0,0}$ *is given in Eq.* (3.2)*,* $x = \alpha + (1-\alpha)\beta$*,* $y = P_c(1 - x^{m+1})$*, and* P_c *is the collision probability. Moreover,*

$$b_{-1,k,j} = (1 - P_c)(1 - x) \sum_{i=1}^{m} b_{i,0,j} \,, \quad 0 \leq k \leq L_s - 1 \,, \tag{3.5}$$

and

$$b_{-2,k,j} = P_c (1 - x) \sum_{i=1}^{m} b_{i,0,j} \,, \quad 0 \leq k \leq L_c - 1 \,. \tag{3.6}$$

Proof. See Section B.1. □

We remark here that the term $b_{0,0,0}$, which plays a key role in the analysis, is different from the corresponding term given in [166]–[171] due to our accurate modeling of the retransmissions, ACK, unsaturated traffic, and packet size. In the next section, we demonstrate the validity of the Markov chain model by experiments.
Now, starting from Lemma 1, we derive the channel sensing probability τ and the busy channel probabilities α and β. The probability τ that a node attempts CCA_1 in a randomly chosen time slot is

$$\tau = \sum_{i=0}^{m} \sum_{j=0}^{n} b_{i,0,j} = \frac{1 - x^{m+1}}{1 - x} \frac{1 - y^{n+1}}{1 - y} b_{0,0,0}. \tag{3.7}$$

This probability depends on the probability P_c that a transmitted packet encounters a collision, and the probabilities α and β. These probabilities are developed in the following.
The term P_c is the probability that at least one of the $N - 1$ remaining nodes transmit in the same time slot. If all nodes transmit with probability τ, P_c is

$$P_c = 1 - (1 - \tau)^{N-1} \,,$$

where N is the number of nodes. Similarly to [158], we derive the busy channel probabilities α and β as follows:

$$\alpha = \alpha_1 + \alpha_2 \,, \tag{3.8}$$

where α_1 is the probability of finding channel busy during CCA_1 due to data transmission, namely,

$$\alpha_1 = L_p(1-(1-\tau)^{N-1})(1-\alpha)(1-\beta)\,,$$

and α_2 is the probability of finding the channel busy during CCA_1 due to ACK transmission, which is

$$\alpha_2 = L_{\mathrm{ack}}\frac{N\tau(1-\tau)^{N-1}}{1-(1-\tau)^N}(1-(1-\tau)^{N-1})(1-\alpha)(1-\beta)\,,$$

where L_{ack} is the length of the ACK. By a similar argument, the probability of finding the channel busy during CCA_2 is

$$\beta = \frac{1-(1-\tau)^{N-1}+N\tau(1-\tau)^{N-1}}{2-(1-\tau)^N+N\tau(1-\tau)^{N-1}}\,. \tag{3.9}$$

Now, we are in the position to derive the carrier sensing probability τ and the busy channel probabilities α and β by solving the system of non-linear equations (3.7), (3.8), and (3.9) for these probabilities, see details in [189]. From these probabilities then one could derive the expressions of the reliability, delay for successful packet delivery, and power consumption that are needed in (3.1). The drawback of such an approach is that there is no closed form expression for these probabilities, and the system of equations that gives τ, α and β must be solved by numerical methods. This may be computationally demanding and therefore inadequate for use in simple sensor devices. In the following, we instead present a simple analytical model of the reliability, delay for successful packet delivery, and power consumption. The key idea is that sensor nodes can estimate the busy channel probabilities α and β and the channel sensing probability τ. Therefore, nodes exploit local measurements to evaluate the performance metrics, rather than solving nonlinear equations. Details follow in the sequel, where we derive these approximations for Eqs. (3.1a)–(3.1c).

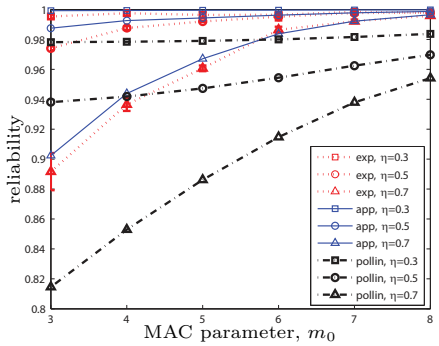
3.4.2 Reliability

The main contributions of this section are the derivation of both precise and approximated expression of the reliability (3.1b) of the optimization problem (3.1), where we recall the reliability is the probability of successful packet reception.

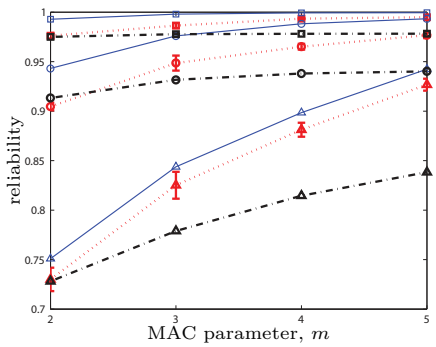
Proposition 1. *The reliability is*

$$R(\boldsymbol{V}) = 1 - \frac{x^{m+1}(1-y^{n+1})}{1-y} - y^{n+1}\,. \tag{3.10}$$

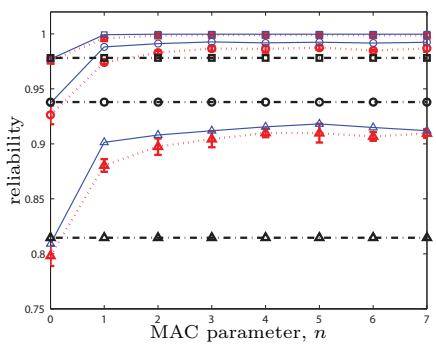
Proof. In slotted CSMA/CA, packets are unsuccessfully received due to two reasons: channel access failure and retry limits. Channel access failure happens when a packet fails to obtain idle channel in two consecutive CCAs within $m+1$ backoffs.



(a) $m_0 = 3, \ldots, 8$, $m_b = 8$, $m = 4$, $n = 1$



(b) $m = 2, \ldots, 5$, $m_0 = 3$, $m_b = 8$, $n = 1$



(c) $n = 0, \ldots, 7$, $m_0 = 3$, $m_b = 8$, $m = 4$

Figure 3.2: Reliability as a function of the traffic regimes $\eta = 0.3, 0.5, 0.7$, and MAC parameters $m_0 = 3, \ldots, 8$, $m_b = 8$, $m = 2, \ldots, 5$, $n = 0, \ldots, 7$ as obtained by our proposed analysis, experimental implementation, and with Pollin's Markov chain model [158]. The length of the packet is $L_p = 5$ and the number of nodes is $N = 10$. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 2×10^5 time slots each. The percentage error of the analytical model for the reliability is 0.993%.

Furthermore, a packet is discarded if the transmission fails due to repeated collisions after $n+1$ attempts. Following the Markov model presented in Figure 3.1, the probability that the packet is discarded due to channel access failure is

$$P_{dc} = x^{m+1} \sum_{j=0}^{n} y^j = \frac{x^{m+1}(1-y^{n+1})}{1-y} \,. \tag{3.11}$$

The probability of a packet being discarded due to retry limits is

$$P_{dr} = y^{n+1} \,. \tag{3.12}$$

The reliability is given by

$$R(\mathbf{V}) = 1 - P_{dc} - P_{dr} \,,$$

from which the proposition follows. □

Approximation 1. *An approximation of the reliability is*

$$\widetilde{R}(\boldsymbol{V}) = 1 - x^{m+1}(1+\widetilde{y}) - \widetilde{y}^{n+1} \tag{3.13}$$

where

$$\begin{aligned} \widetilde{y} =& (1-(1-(1+x)(1+\hat{y})\widetilde{b}_{0,0,0})^{N-1})(1-x^2) \,, \\ \widetilde{b}_{0,0,0} =& 2/(W_0(1+2x)(1+\hat{y}) + 2L_s(1-x^2)(1+\hat{y}) \\ & + h(1-\eta)/\eta(1+\hat{y}^2+\hat{y}^{n+1})) \,, \end{aligned}$$

and $\hat{y} = (1-(1-\tau)^{N-1})(1-x^2)$.

Derivation: The expression of the state probability $b_{0,0,0}$ is the main responsible for the non-linear equations that give α, β and τ. Therefore, we approximate $b_{0,0,0}$. Let the approximation be $\widetilde{b}_{0,0,0}$. Given $z \geq 0$, we use that

$$\frac{1-z^{m+1}}{1-z} \approx 1+z \,, \qquad \text{if} \quad z \ll 1 \,. \tag{3.14}$$

By using this approximation, Eq. (B.9) is approximated by

$$\sum_{i=0}^{m} \sum_{k=0}^{W_i-1} \sum_{j=0}^{n} b_{i,k,j} \approx \frac{b_{0,0,0}}{2} \left[(1+2x)W_0 + 1 + x\right](1+y) \,. \tag{3.15}$$

Similarly, Eq. (B.10) is approximated by

$$\sum_{i=0}^{m} \sum_{j=0}^{n} b_{i,-1,j} \approx b_{0,0,0}(1-\alpha)(1+x)(1+y) \approx 0 \,. \tag{3.16}$$

Eq. (B.11) is approximated by

$$\sum_{j=0}^{n}\left(\sum_{k=0}^{L_s-1} b_{-1,k,j} + \sum_{k=0}^{L_c-1} b_{-2,k,j}\right) \approx b_{0,0,0} L_s (1-x^{m+1})(1+y), \tag{3.17}$$

where we assume that the packet collision time is approximated to the packet successful transmission time, namely $L_s \approx L_c$. Finally, using $K_0 = h(1-\eta)/\eta$, the approximated idle-queue stages of Eq. (B.12) is

$$\sum_{l=0}^{h-1} Q_l \approx b_{0,0,0} K_0 \left[1 + y + P_c(1-x^{m+1})(y^n - y - 1)\right]. \tag{3.18}$$

By summing together Eqs. (3.15)–(3.18) and applying the approximation of Eq. (3.14), the state probability is

$$\widetilde{b}_{0,0,0} \approx \frac{2}{W_0 r_1 + 2 r_2} \tag{3.19}$$

where

$$\begin{aligned} r_1 &= (1+2x)(1+\hat{y}), \\ r_2 &= L_s(1-x^2)(1+\hat{y}) + K_0(1-\hat{y}^2+\hat{y}^{n+1}), \\ \hat{y} &= (1-(1-\tau)^{N-1})(1-x^2). \end{aligned}$$

Now, we put $\widetilde{b}_{0,0,0}$ into Eq. (3.10) to obtain the approximated reliability:

$$\widetilde{R}(\mathbf{V}) = 1 - x^{m+1}(1+\widetilde{y}) - \widetilde{y}^{n+1},$$

where $\widetilde{y} = (1-(1-\widetilde{\tau})^{N-1})(1-x^2)$ and $\widetilde{\tau}$ is the approximated carrier sensing probability $\widetilde{\tau} = (1+x)(1+\hat{y})\widetilde{b}_{0,0,0}$. □

We remark that $\widetilde{R}(\mathbf{V})$ is a function of the measurable busy channel probabilities α and β, the channel access probability τ and the MAC parameters m_0, m_b, m, n. The approximation is based on estimated values of x and τ.

We have implemented the IEEE 802.15.4 protocol to assess our analysis of the approximated model of the reliability by experimental results, which are reported in Figure 3.2. The IEEE 802.15.4 protocol was implemented on a test-bed using TelosB sensors [137] running the Contiki operating system [190] based on the specifications of the IEEE 802.15.4 [8]. The implementation is available for download [191]. Figure 3.2 compares the reliability given by Eq. (3.13), the analytical model in [158], and experimental results as a function of the traffic regimes $\eta = 0.3, 0.5, 0.7$ with $N = 10$ nodes and different MAC parameters m_0, m, n. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 2×10^5 time slots each. In the figure, note that "Pollin" refers to the reliability model derived in [158].

Our analytical expression matches quite well the experimental results. Note that the percentage error of the reliability given by Eq. (3.13) is 0.993%. The expression is closer to experimental results under low traffic regime $\eta = 0.3, 0.5$ than high traffic regime $\eta = 0.7$ because the approximation given by Eq. (3.14) holds if $x = \alpha + (1-\alpha)\beta \ll 1$, but x increases as the traffic and the number of nodes increases. The reliability approaches 1 under very low traffic regime $\eta = 0.3$. In Figure 3.2(a), 3.2(b), the reliability increases as MAC parameters m_0, m increase, respectively. In Figure 3.2(c), we observe that the improvement of reliability is small as the retry limits n increases for $n \geq 3$. Notice that the reliability saturates to 0.92 for traffic regime $\eta = 0.7$ for $n \geq 3$. Hence, the retransmissions are necessary but not sufficient to obtain high reliability under high traffic regimes.

3.4.3 Delay

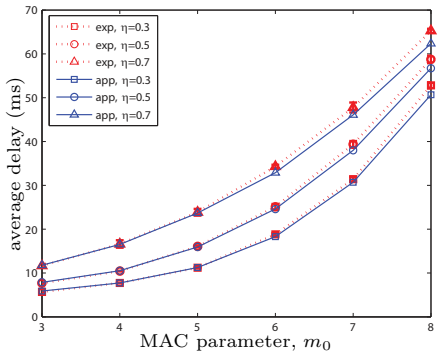
In this section, we derive the constraint of average delay (3.1c) of the optimization problem (3.1). The average delay for a successfully received packet is defined as the time interval from the instant the packet is at the head of its MAC queue and ready to be transmitted, until the transmission is successful and the ACK is received. In this section, we develop an approximation for such an average delay, which is given by Approximation 2. To this aim, we need some intermediate technical steps. In particular, we characterize (a) the expression of the delay for a successful transmission at time $j+1$ after jth events of unsuccessful transmission due to collision and (b) the expected value of the approximated backoff delay due to busy channel. We address these issues in the following.
Let D_j be the random time associated to the successful transmission of a packet at the jth backoff stage. Denote with $\mathcal{A}_j$ the event of a successful transmission at time $j+1$ after jth events of unsuccessful transmission. Let $\mathcal{A}_t$ be the event of successful transmission within the total attempts n. Then, the delay for a successful transmission after j unsuccessful attempts is

$$D = \sum_{j=0}^{n} \mathbb{1}_{\mathcal{A}_j|\mathcal{A}_t} D_j \,,$$

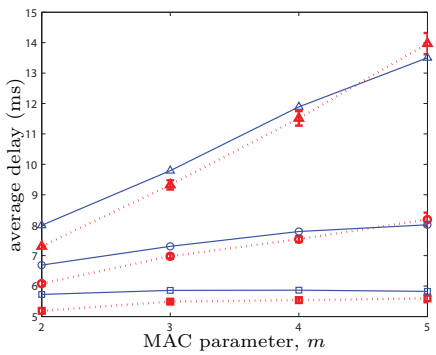
where $\mathbb{1}_{\mathcal{A}_j|\mathcal{A}_t}$ is 1 if $\mathcal{A}_j|\mathcal{A}_t$ holds, and 0 otherwise and $D_j = L_s + j\,L_c + \sum_{h=0}^{j} T_h$, with T_h being the backoff stage delay, L_s is the packet successful transmission time, and L_c is the packet collision transmission time as defined in Eqs (2.3) and (2.4), respectively

Lemma 2. *The probability of successful transmission at time $j+1$ after j events of unsuccessful transmission due to collision is*

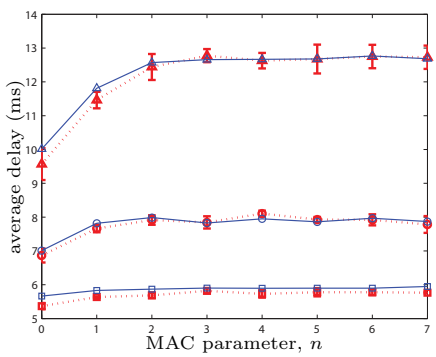
$$\Pr(\mathcal{A}_j|\mathcal{A}_t) = \frac{(1-y)\,y^j}{1-y^{n+1}} \,. \tag{3.20}$$



(a) $m_0 = 3, \ldots, 8$, $m_b = 8$, $m = 4$, $n = 1$



(b) $m = 2, \ldots, 5$, $m_0 = 3$, $m_b = 8$, $n = 1$



(c) $n = 0, \ldots, 7$, $m_0 = 3$, $m_b = 8$, $m = 4$

Figure 3.3: Average delay as a function of the traffic regimes $\eta = 0.3, 0.5, 0.7$ and MAC parameters $m_0 = 3, \ldots, 8$, $m_b = 8$, $m = 2, \ldots, 5$, $n = 0, \ldots, 7$ as obtained by our proposed analysis and experimental implementation. The length of the packet is $L_p = 5$ and the number of nodes is $N = 10$. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 2×10^5 time slots each. The percentage error of the analytical model for the average delay is 3.155%.

Proof. A transmission may be successful with probability $1 - P_c$, or collide with probability P_c. Then, the probability of the event $\mathcal{A}_j|\mathcal{A}_t$ is

$$\Pr(\mathcal{A}_j|\mathcal{A}_t) = \frac{P_c^j(1-x^{m+1})^j}{\sum_{k=0}^{n}\left(P_c(1-x^{m+1})\right)^k}$$

where the normalization comes by considering all the possible events of successful attempts $\mathcal{A}_t$. Note that $(1 - x^{m+1})$ is the probability of successful channel access within the maximum number of m backoff stages. □

In the following, we give the total backoff delay T_h. Let $T_{h,i}$ be the random time needed to obtain two successful CCAs from the selected backoff counter value in backoff level i. Recall that a node transmits the packet when the backoff counter is 0 and two successful CCAs are detected [8]. Denote with $\mathcal{B}_i$ the event occurring when the channel is busy for i times, and then idle at the time $i+1$. Let $\mathcal{B}_t$ be the event of having a successful sensing within the total number of m sensing attempts. If the node accesses an idle channel after its i th busy CCA, then

$$T_h = \sum_{i=0}^{m} \mathbb{1}_{\mathcal{B}_i|\mathcal{B}_t}\, T_{h,i}\,,$$

where

$$T_{h,i} = 2\,T_{sc} + \sum_{k=1}^{i} T_{h,k}^{sc} + \sum_{k=0}^{i} T_{h,k}^{b}\,, \tag{3.21}$$

and where $2T_{sc}$ is the successful sensing time, $\sum_{k=1}^{i} T_{h,k}^{sc}$ is the unsuccessful sensing time due to busy channel during CCA, and $\sum_{k=0}^{i} T_{h,k}^{b}$ is the backoff time.

Lemma 3. *The expected value of the approximated backoff delay is*

$$\begin{aligned}\mathbb{E}[\widetilde{T}_h] =& 2T_b\left(1+\frac{1}{4}\left(\frac{1-\gamma}{1-\gamma^{m+1}}\left(2W_0\frac{1-(2\gamma)^{m+1}}{1-2\gamma}-\frac{3(m+1)\gamma^{m+1}}{1-\gamma}\right)\right.\right.\\ &\left.\left.+\frac{3\gamma}{1-\gamma}-(W_0+1)\right)\right)\,,\end{aligned} \tag{3.22}$$

where $\gamma = \max(\alpha, (1-\alpha)\beta)$.

Proof. By considering the busy channel during two CCAs, the probability of the event $\mathcal{B}_i|\mathcal{B}_t$ is approximated by

$$\widetilde{\Pr}(\mathcal{B}_i|\mathcal{B}_t) = \frac{\gamma^i}{\sum_{k=0}^{m}\gamma^k}\,, \tag{3.23}$$

$$\begin{aligned}\widetilde{E}_{\text{tot,i}}(\mathbf{V}) =& \frac{P_i\tau}{2}\left[\frac{(1-x)(1-(2x)^{m+1})}{(1-2x)(1-x^{m+1})}W_0 - 1\right] + P_{sc}(2-\alpha)\tau + (1-\alpha)(1-\beta)\tau \\ & \times \left(P_t L_p + P_i + L_{\text{ack}}\left(P_r(1-P_c) + P_i P_c\right)\right) + P_w(1-\eta)\left(x^{m+1}(1+y)\right. \\ & \left. + P_c(1-x^2)y^n + (1-P_c)(1-x^2)(1+y)\right)\widetilde{b}_{0,0,0} \qquad (3.26)\end{aligned}$$

$$\begin{aligned}\widetilde{E}_{\text{tot,s}}(\mathbf{V}) =& P_{sc}(2-\alpha)\tau + (1-\alpha)(1-\beta)\tau\left(P_t L_p + P_i + L_{\text{ack}}\left(P_r(1-P_c) + P_i P_c\right)\right) \\ & + P_w\left(\tau - \frac{\widetilde{b}_{0,0,0}(1-(0.5x)^{m+1})}{W_0(1-0.5x)}\frac{1-y^{n+1}}{1-y}\right) \qquad (3.27)\end{aligned}$$

where $\gamma = \max(\alpha, (1-\alpha)\beta)$. Note that the approximation is based on the worst case of the busy channel probabilities. For a more accurate model, see [192]. The approximation of the average backoff period is

$$\begin{aligned}\mathbb{E}[\widetilde{T}_h] =& \sum_{i=0}^{m}\widetilde{\Pr}(\mathcal{B}_i|\mathcal{B}_t)\,\mathbb{E}[\widetilde{T}_{h,i}] \qquad (3.24)\\ =& 2T_{sc} + \sum_{i=0}^{m}\widetilde{\Pr}(\mathcal{B}_i|\mathcal{B}_t)\sum_{k=0}^{i}\left(\frac{2^k W_0 - 1}{2}T_b + 2T_{sc}\,k\right)\end{aligned}$$

where the approximated sensing time $\mathbb{E}[\widetilde{T}_{h,i}]$ considers the worst case, i.e., a failure of the second sensing (CCA_2), which implies that $T_{sc} = T_b$ and that each sensing failure takes $2T_{sc}$ in Eq. (3.21). □

Now, we are in the position to derive an approximation of the average delay for successfully received packets.

Approximation 2. *The expected value of the approximated delay is*

$$\mathbb{E}[\widetilde{D}(\boldsymbol{V})] = T_s + \mathbb{E}[\widetilde{T}_h] + \left(\frac{y}{1-y} - \frac{(n+1)\,y^{n+1}}{1-y^{n+1}}\right)(T_c + \mathbb{E}[\widetilde{T}_h])\,. \qquad (3.25)$$

Derivation: By considering the Lemma 2, we derive

$$\mathbb{E}[\widetilde{D}(\mathbf{V})] = \sum_{j=0}^{n}\Pr(\mathcal{A}_j|\mathcal{A}_t)\,\mathbb{E}[\widetilde{D}_j]$$

where $\mathbb{E}[\widetilde{D}_j] = T_s + j\,T_c + \sum_{h=0}^{j}\mathbb{E}[\widetilde{T}_h]$ and $\mathbb{E}[\widetilde{T}_h]$ is given in Lemma 3. □

Figure 3.3 shows the average delay as obtained by Eq. (3.25) as a function of different traffic regimes $\eta = 0.3, 0.5, 0.7$ with a given number of nodes $N = 10$ and different MAC parameters m_0, m, n. The vertical bars indicate the standard

deviation as obtained out of 5 experimental runs of 2×10^5 time slots each. The analytical model predicts well the experimental results. Note that the percentage error of the average delay given by Eq. (3.25) is 3.155%. The accuracy is reduced under high traffic regime $\eta = 0.7$ due to the approximation given by Eq. (3.14). Observe that the average delay increases as traffic regime increases due to high busy channel probability and collision probability. Figure 3.3(a) shows that the average delay increases exponentially as m_0 increases. Hence, we conclude that m_0 is the key parameter affecting the average delay when compared to m, n.

3.4.4 Power Consumption

Here, we derive the objective function, power consumption of the node (3.1a) of the optimization problem (3.1). We propose two models for the average power consumption, depending on the radio state during the backoff mechanism specified by the IEEE 802.15.4 standard. Let us denote by *I-mode* and *S-mode* the situation when the radio is set in idle mode or in sleep mode during backoff period, respectively. The node sets its radio to sleep mode during the idle-queue states for hT_b s. Note that since we consider the uplink communication from the nodes to the coordinator, the energy consumption of CSMA/CA mechanism and traffic load are critical factors for the energy consumption of the nodes.

Approximation 3. *The energy consumption of the I-mode* $\widetilde{E}_{\mathrm{tot,i}}(\boldsymbol{V})$ *is given by Eq.* (3.26) *and of the S-mode* $\widetilde{E}_{\mathrm{tot,s}}(\boldsymbol{V})$ *is given by Eq.* (3.27)*, where state probability* $\widetilde{b}_{0,0,0}$ *is given by Eq.* (3.19)*,* $P_i, P_{sc}, P_{sp}, P_w, P_t$ *and* P_r *are the average power consumption in idle-listen, channel sensing, sleep states, wake-up state, transmit and receiving states, respectively.*

Derivation: By considering the Markov chain given in Figure 3.1, we see that the average power consumption of *I-mode* $\widetilde{E}_{\mathrm{tot,i}}(\mathbf{V})$ is

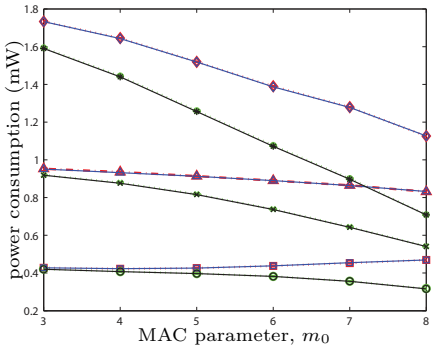
$$\widetilde{E}_{\mathrm{tot,i}}(\mathbf{V}) = E_{b,i} + E_{sc} + E_t + E_q + E_{w,i}\,.$$

In the following, we derive these terms.
The idle backoff power consumption is

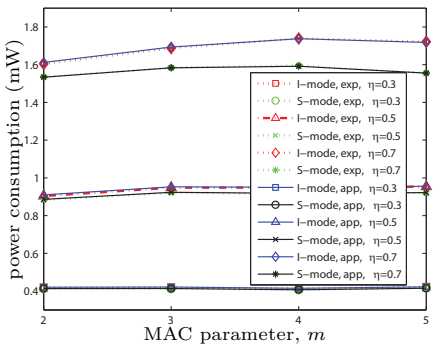
$$E_{b,i} = P_i \sum_{i=0}^{m} \sum_{k=1}^{W_i-1} \sum_{j=0}^{n} b_{i,k,j} = \frac{P_i \tau}{2} \left[\frac{(1-x)(1-(2x)^{m+1})}{(1-2x)(1-x^{m+1})} W_0 + 1 \right], \tag{3.28}$$

where the carrier sensing probability τ is measured by each node and P_i is the average power consumption in idle-listen.
By putting together Eqs. (B.9), (B.10) and (3.7), the average power consumption of the sensing state is

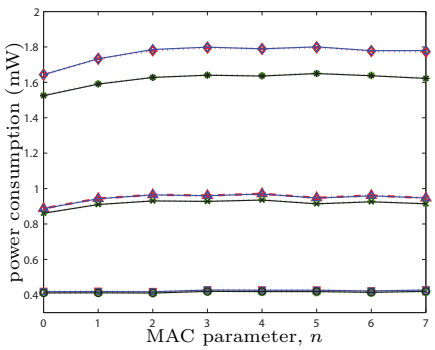
$$E_{sc} = P_{sc} \sum_{i=0}^{m} \sum_{j=0}^{n} (b_{i,0,j} + b_{i,-1,j}) = P_{sc}(2-\alpha)\tau\,, \tag{3.29}$$



(a) $m_0 = 3, \ldots, 8$, $m_b = 8$, $m = 4$, $n = 1$



(b) $m = 2, \ldots, 5$, $m_0 = 3$, $m_b = 8$, $n = 1$



(c) $n = 0, \ldots, 7$, $m_0 = 3$, $m_b = 8$, $m = 4$

Figure 3.4: Average power consumption of *I-mode* and *S-mode* as a function of the traffic regimes $\eta = 0.3, 0.5, 0.7$ and MAC parameters $m_0 = 3, \ldots, 8$, $m_b = 8$, $m = 2, \ldots, 5$, $n = 0, \ldots, 7$ as obtained by our analysis and experimental implementation. The length of the packet is $L_p = 5$ and the number of nodes is $N = 10$. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 2×10^5 time slots each. The percentage errors of the analytical model for the average power consumption are 0.193% and 0.175% for *I-mode* and *S-mode*, respectively.

where P_{sc} is the average power consumption in channel sensing. Similarly, by substituting Eq. (B.11) and Eq. (3.7), the average power consumption for packet transmission including both successful transmission and packet collision E_t is

$$\begin{aligned} E_t =& P_t \sum_{i=-2}^{-1} \sum_{k=0}^{L_p-1} \sum_{j=0}^{n} b_{i,k,j} + P_i \sum_{i=-2}^{-1} \sum_{j=0}^{n} b_{i,L_p,j} \\ &+ \sum_{j=0}^{n} \sum_{k=L_p+1}^{L_p+L_{\text{ack}}+1} \left(P_r\, b_{-1,k,j} + P_i\, b_{-2,k,j}\right) \\ =& (1-x)\tau \left(P_t L_p + P_i + L_{\text{ack}} \left(P_r(1-P_c) + P_i P_c\right)\right), \end{aligned} \tag{3.30}$$

where P_t and P_r are the average power consumption in transmit and receiving states, respectively. Analogously, E_q is the power consumption of idle stage without packet generation:

$$E_q = P_{sp} \sum_{l=0}^{h-1} Q_l \approx 0\,, \tag{3.31}$$

where P_{sp} is the average power consumption in sleep states, which we assume negligible. Since a node wakes up only after generating packet, the wake-up power consumption $E_{w,i}$ is

$$\begin{aligned} E_{w,i} =& P_w(\eta) Q_{h-1} \\ =& P_w(1-\eta) \left(x^{m+1}(1+y) + P_c(1-x^2)y^n + (1-P_c)(1-x^2)(1+y)\right) \widetilde{b}_{0,0,0}\,, \end{aligned} \tag{3.32}$$

where P_w is the average power consumption in wake-up state and the state probability $\widetilde{b}_{0,0,0}$ is given in Eq. (3.19). By summing Eqs. (3.28)–(3.32), we obtain the average power consumption of *I-mode* in closed form.
The average power consumption of *S-mode* $\widetilde{E}_{\text{tot,s}}(\mathbf{V})$ during backoff states can be derived by following an approach similar to the *I-mode*:

$$\widetilde{E}_{\text{tot,s}}(\mathbf{V}) = E_{b,s} + E_{sc} + E_t + E_q + E_{w,s}\,,$$

where the sleep backoff power consumption is

$$E_{b,s} = P_{sp} \sum_{i=0}^{m} \sum_{k=1}^{W_i-1} \sum_{j=0}^{n} b_{i,k,j} \approx 0\,,$$

the wake-up power consumption is

$$E_{w,s} = P_w \sum_{i=0}^{m} \sum_{j=0}^{n} b_{i,1,j} \approx P_w \left(\tau - \frac{\widetilde{b}_{0,0,0}}{W_0} \frac{1-(0.5x)^{m+1}}{1-0.5x} \frac{1-y^{n+1}}{1-y}\right), \tag{3.33}$$

and E_{sc}, E_t, E_q is given in Eqs. (3.29), (3.30), (3.31), respectively. Note that since the radio is set in sleep mode during backoff period, node wakes up for each CCA_1 state. □

Figure 3.4 compares our proposed analytical model and experimental results for the power consumption for both *I-mode* and *S-mode* as a function of different traffic regimes $\eta = 0.3, 0.5, 0.7$ with a number of nodes $N = 10$ and different MAC parameters m_0, m, n. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 2×10^5 time slots each. The percentage errors of the average power consumption for *I-mode* given by Eq. (3.26) and *S-mode* given by Eq. (3.27) are 0.193% and 0.175%, respectively. The power consumption of *S-mode* and *I-mode* decreases as m_0 increases because of sleep mode and idle mode during the backoff time for high traffic regime $\eta = 0.5, 0.7$. The main component of the average power consumption is the transmit or receiving power rather than power consumption during backoff time for high traffic regime, i.e., $P_t > P_i > P_{sp}$ and $P_r > P_i > P_{sp}$. However, the power consumption of *I-mode* increases as the MAC parameters (m_0, m, n) increase under low traffic regime $\eta = 0.3$. Since the node needs to stay more time in idle sleep stage without packet generation, the main component of average power consumption is the idle backoff time rather than transmit or receiving power consumption under low traffic regime $\eta = 0.3$. It is interesting to observe that the power consumption has a weaker dependence on m and n than on m_0.

3.5 Optimization

In the previous sections we developed the expressions of the performance metrics. Here, we present a novel approach where each node locally solves the optimization problem. Consider the reliability, delay and power consumption as investigated in Section 3.4. The optimization problem (3.1) can be written by using Eq. (3.13) given by Approximation 1 for reliability constraint, Eq. (3.25) given by Approximation 2 for delay constraint and Eq. (3.26) or (3.27) given by Approximation 3 for the power consumption. Note that the power consumption is given by Eq. (3.26) if the *I-mode* is selected, and it is given by Eq. (3.27), if the *S-mode* is selected. The solution of the optimization problem gives the optimal MAC parameter (m_0, m, n) that each node uses to minimize its energy expenditure, subject to reliability and delay constraints. Notice that the problem is combinatorial because the decision variables take on discrete values.

A vector of decision variables $\mathbf{V}$ is feasible if the reliability and delay constraints are satisfied. The optimal solution may be obtained by checking every combination of the elements of $\mathbf{V}$ that gives feasibility, and then checking the combination that gives the minimum objective function. Clearly, this approach may have a high computational complexity, since there are $6 \times 4 \times 8 = 192$ combinations of MAC parameters to check [8]. Therefore, in the following we propose an algorithm that

gives the optimal solution by checking just a reduced number of combinations.
From Figures 3.2, 3.3 and 3.4, we remark here that the reliability and power consumption of both *I-mode* and *S-mode* are increasing function as the parameter n increases. This property is quite useful to solve (3.1) by a simple algorithm with reduced computational complexity, as we see next.
The search of optimal MAC parameters uses an iterative procedure according to the component-based method [193]. In particular, the probabilities α, β, and τ are estimated periodically by each node. If a node detects a change of these probabilities, then the node solves the local optimization problem (3.1) using these estimated values. The solution is achieved by finding the value of n that minimizes the energy consumption given a pair of values for m_0 and m. Since the power consumption is increasing with n, it follows that the minimum is attained at the lowest value of n that satisfies the constraints. Given that the reliability is increasing with n, simple algebraic passages give that such a value is $n = f(m_0, m)$, with

$$f(m_0, m) = \left\lceil \frac{\ln(1 - x^{m+1}(1 + \widetilde{y}) - R_{\min})}{\ln(\widetilde{y})} - 1 \right\rceil , \tag{3.34}$$

where $\widetilde{y} = (1 - (1 - \widetilde{\tau})^{N-1})(1 - x^2)$ and

$$\widetilde{\tau} = \frac{2r_3}{2^{m_0} r_1 + 2r_2} ,$$

with

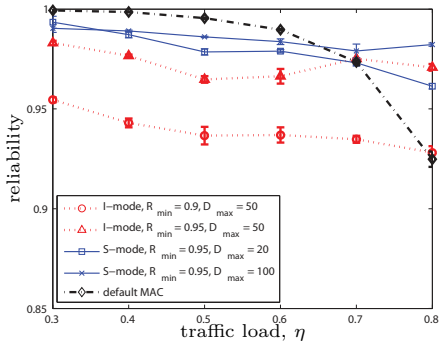
$$\begin{aligned} r_1 &= (1 + 2x)(1 + \hat{y}) , \\ r_2 &= L_s(1 - x^2)(1 + \hat{y}) + \frac{h(1 - \eta)(1 + \hat{y}^2 + \hat{y}^{n+1})}{\eta} , \\ r_3 &= (1 + x)(1 + \hat{y}) , \end{aligned}$$

and $\hat{y} = (1 - (1 - \tau)^{N-1})(1 - x^2)$. Eq. (3.34) returns the optimal retry limits given a pair m_0, m. Notice that x and $\hat{y}$ are measurable since node estimates α, β, and τ. By using this simple algorithm, a node checks just $6 \times 4 = 24$ combinations of the MAC parameters m_0, m instead of $6 \times 4 \times 8 = 192$ combinations that would be required by an exhaustive search.
In addition, we propose an alternative approach having a low computation cost. It is particularly suitable for devices with very limited computing capabilities. The optimal solution (m_0, m, n) of problem (3.1) can be computed off-line and stored in a look-up table as function of the busy channel probabilities α and β and the channel access probability τ. The table can be thought of as a matrix with the set of values of α, β, τ. Each node can estimate α, β, τ, and read from the look-up table the entries of the solutions at location α, β, τ closer to the estimated values. Then, the node uses (m_0, m, n) corresponding to the entry of the look-up table. Note that the size of the matrix can be significantly reduced by using a lossless data compression [194].

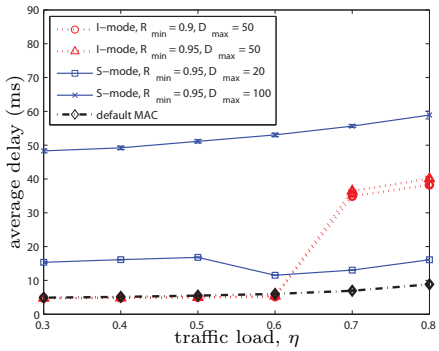
We have seen by the Approximations 1, 2 and 3 that the performance metrics are function of the busy channel probabilities α and β and the channel access probability τ. Once these probabilities are known at a node, the optimal MAC parameters of that node can be readily computed by the simple algorithm. In the algorithm, the number of nodes and packet generation rates are assumed to be known, whereas the busy channel probability and channel access probability are periodically estimated in each node during the sensing states of the MAC layer, and they do not require an ACK mechanism, as we describe the details in the following. In addition, the robustness of the algorithm to possible errors in the estimation of the number of nodes and traffic load is then investigated in Section 3.6.3.
The average busy channel probabilities α and β are estimated at each node while sending a data packet to the coordinator. These probabilities are initialized at the beginning of the node's operation. The estimations of the busy channel probabilities and the channel access probability use a sliding window. When the node senses the channel at CCA_1 or CCA_2, these probabilities are updated by $\alpha = \delta_b\alpha + (1-\delta_b)\hat{\alpha}, \beta = \delta_b\beta + (1-\delta_b)\hat{\beta}$ for some $\delta_b \in (0,1)$, respectively. Note that $\hat{\alpha}$ and $\hat{\beta}$ are the busy channel probability of CCA_1 and CCA_2 of the current sliding window, respectively. Therefore, a node does not require any extra communication and sensing state to estimate these probabilities compared to the IEEE 802.15.4 standard. By contrast, the estimation algorithms for IEEE 802.11 proposed in [173] and [195] are not energy efficient since a node needs to sense the channel state during the backoff stage. This allows one to estimate the average length of idle period. Hence, these schemes are implementable only in *I-mode*. By contrast, our scheme is applied in both *I-mode* and *S-mode* and does not require any computation load during the backoff stage.
During an initialization phase of the algorithm, a node communicates with the initial MAC parameters $m_0 = 3, m_b = 8, m = 4, n = 1$. Then, the busy channel probabilities α and β and the channel access probability τ are estimated in each node during the channel sensing state of IEEE 802.15.4 without any extra state. The application requirements are communicated by the coordinator to the node if there are changes. It is also possible that each node makes a decision of application requirements depending on the data type e.g., strict delay requirement for alarm message.

3.6 Numerical Results

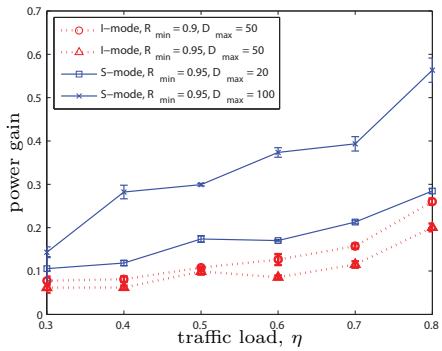
In the following, we present an extensive set of real-world experiments and Monte Carlo simulations to analyze the performance of the new adaptive algorithm for tuning the MAC parameters, under both stationary and transient conditions. The IEEE 802.15.4 MAC protocol is implemented on a real test-bed using TelosB sensors [137] and Contiki OS [190]. The analytical modeling that we have proposed in Section 3.4 is based on the Markov chain modeling that has been validated experimentally. Therefore, the Monte Carlo simulations that we use here are repre-



(a) Reliability



(b) Average delay



(c) Power gain

Figure 3.5: Stationary condition: reliability, average delay and power gain of the *I-mode*, *S-mode* of proposed scheme and IEEE 802.15.4 with default parameter ($macMinBE = 3, macMaxBE = 5, macMaxCSMABackoffs = 4, macMaxFrameRetries = 3$) as a function of the traffic load $\eta = 0.3, \ldots, 0.8$, the reliability requirement $R_{\min} = 0.9, 0.95$ and delay requirement $D_{\max} = 20, 50, 100$ ms for the length of the packet $L_p = 5$ and $N = 10$ nodes. Note that "default MAC" refers to IEEE 802.15.4 with default MAC parameters. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 5×10^5 time slots each.

sentative of the real-world behavior of the network.
In the stationary conditions, the application requirements and network scenario are constant, whereas in transient condition there are variations. The simulations are based on the specifications of the IEEE 802.15.4 and the practical implementation aspects described in Section 3.5. In the experiments and simulations, the network considers the *I-mode* and *S-mode* of the node to compare the performance on the reliability, average packet delay and power consumption. Furthermore, we investigate the fairness of resource allocation, robustness to network changes and sensitivity to inaccurate parameter estimations. Note that it is not possible to compare our algorithm to other algorithms from the literature as the link-based ones [180]–[184], because they modify the IEEE 802.15.4 standard and are focused on different performance metrics (e.g., throughput). However, it is possible to show that our algorithm outperform significantly the results in [180]–[184]. This is due to that these results use the ACK feedback, which has a low update frequency with respect to the channel and network variations, whereas our algorithm reacts much faster. Details follow in the sequel.

3.6.1 Protocol Behavior in Stationary Conditions

In this subsection, we are interested to the improvement of performance metrics of the proposed scheme at stationary conditions of the network, namely without changing application requirements and network scenarios. We also present a fairness analysis of the adaptive protocol.
Figure 3.5 compares the experimental results for the reliability, average delay, and power gain values of the IEEE 802.15.4 MAC protocol as obtained by our algorithm and with default MAC parameters. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 5×10^5 time slots each. Both the *I-mode* and *S-mode* for various traffic configurations and requirements are considered. The requirements for both the *I-mode* and *S-mode* are $R_{\min} = 0.9, 0.95, D_{\max} = 50$ and $R_{\min} = 0.95, D_{\max} = 20, 100\,\text{ms}$, respectively. Figure 3.5(a) shows that both *I-mode* and *S-mode* satisfy the reliability constraint for different traffic regime. We observe a strong dependence of the reliability of default MAC on different traffic regime. At the high traffic regime $\eta = 0.8$, the reliability of default MAC is 0.89. In Figure 3.5(b), the delay constraint is fulfilled both *I-mode* and *S-mode*. Observe that average delay of *I-mode* decreases when traffic regime is low $\eta \leq 0.6$. This is due to that the optimal MAC parameters at higher traffic regime increase more than the ones at lower traffic regime to satisfy the reliability constraint.
Recall that the target of our proposed adaptive algorithm is to use the tradeoff between application constraints and energy consumption instead of just maximization of reliability or minimization of delay. Therefore, to characterize quantitatively the power consumption, we define the power gain as

$$\rho = \frac{E_{\text{def}} - E_{\text{tot}}(\mathbf{V})}{E_{\text{def}}},$$

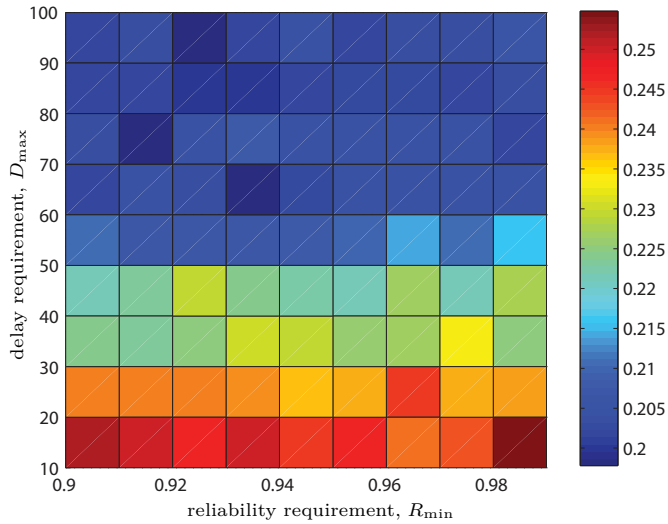


Figure 3.6: Stationary network condition: power consumption of *S-mode* as a function of reliability constraint $R_{\min} = 0.9, \ldots, 0.99$ and delay requirement $D_{\max} = 10, \ldots, 100$ ms for the traffic load $\eta = 0.5$, the length of packet $L_p = 3$ and $N = 10$ nodes.

where E_{def} and $E_{\text{tot}}(\mathbf{V})$ are the average power consumption of *I-mode* or *S-mode* for default MAC and proposed scheme, respectively. The closer ρ to 1, the better the power efficiency. Figure 3.5(c) shows that the power gain increases as traffic regime increases. This improvement is higher for *S-mode* than *I-mode*, e.g., $\rho \approx 0.57$ for *S-mode* with $R_{\min} = 0.95, D_{\max} = 100$. Although there is a strong dependence of the power gain on the traffic regime, our proposed algorithm gives a better energy efficiency than the default MAC. Therefore, the experimental results show clearly the effectiveness of our adaptive IEEE 802.15.4 protocol while guaranteeing the constraints.

Next, we observe the tradeoff between the power consumption, reliability and delay constraints by using the Monte Carlo simulations. Figure 3.6 shows the dependence of the power consumption in *S-mode* with reliability and delay constraints for a given traffic load, length of packets, and number of nodes. Observe that as the delay constraint becomes strict the power consumption increases. In other words, the reliability constraint of *S-mode* is less critical than delay constraint, see more results in [192].

The fairness of resource management is one of the most important concerns when implementing the tuning algorithm of the MAC parameters. We use Jain's fairness index [196] to study the fairness of our proposed scheme for both *I-mode* and *S-mode* based on the experimental results of Figure 3.5. We compute the fairness index of 10 nodes in a stable network. The closer fairness index to 1, the better the achieved fairness. Figure 3.7 shows the fairness index of the reliability for the different requirements and traffic configurations with a given length of the packet and

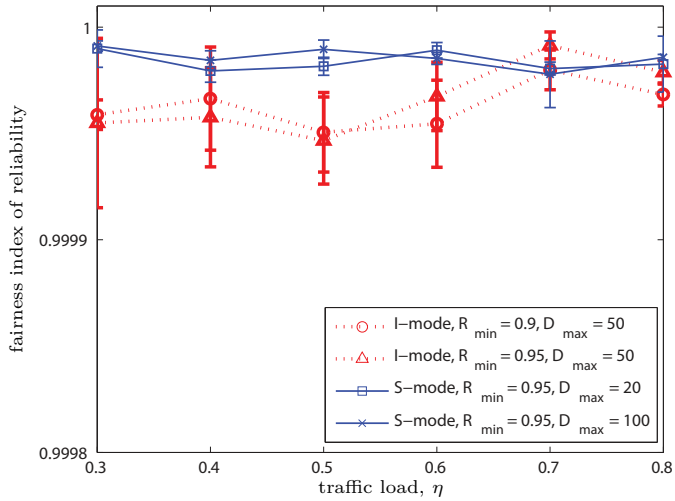


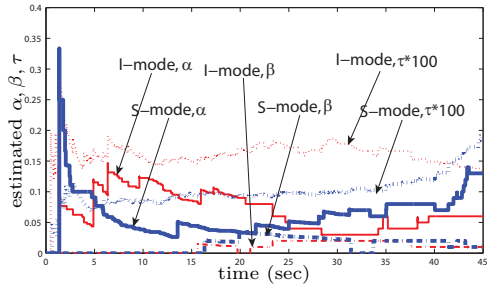
Figure 3.7: Fairness index of the reliability as a function of the traffic load $\eta = 0.3, \ldots, 0.8$, reliability requirement $R_{\min} = 0.9, 0.95$ and delay requirement $D_{\max} = 20, 50, 100$ ms for the length of the packet $L_p = 5$ and $N = 10$ nodes.

number of nodes. Figure 3.7 reports a very high fairness achievement on reliability greater than 0.9999. A similar behavior is found for delay and power consumption. In other words, the MAC parameters of each node converge to the optimal MAC parameter values. Therefore we conclude that most of the nodes can share equally the common medium.

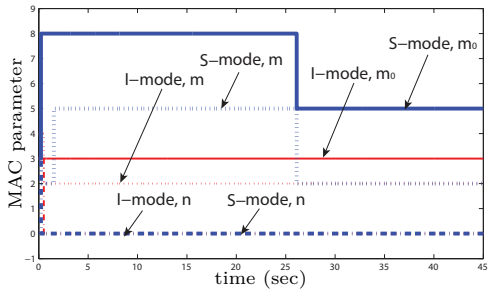
3.6.2 Protocol Behavior in Transient Conditions

The adaptive IEEE 802.15.4 protocol is based on the estimation of the busy channel probabilities α and β and the channel access probability τ. In this section, we investigate the convergence time of the optimal MAC parameters obtained by our adaptive algorithm when the delay constraint changes.

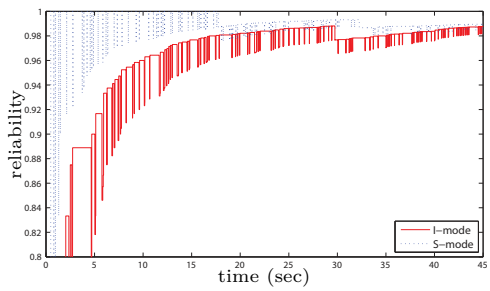
Figures 3.8(a), 3.8(b), 3.8(c), 3.8(d) show the behavior of channel state, MAC parameters, reliability and packet delay of the Monte Carlo simulations when the delay requirement changes for both *I-mode* and *S-mode* with a given traffic load, length of packets, and number of nodes, respectively. Figure 3.8(a) reports the busy channel probabilities α and β and channel access probability τ over time. In Section 3.5, we noticed that the update frequency of α, β, τ is different. τ is updated in each *aUnitBackoffPeriod* and α and β are updated when a node stay in CCA_1 and CCA_2, respectively. Hence, the update frequency order of α, β, and τ is τ first, then α, and finally β. We remark here that the update frequency of link-based adaptation is lower than the update frequency of β of our algorithm since link-based adaptation requires an ACK transmission [180]–[184]. The update frequency



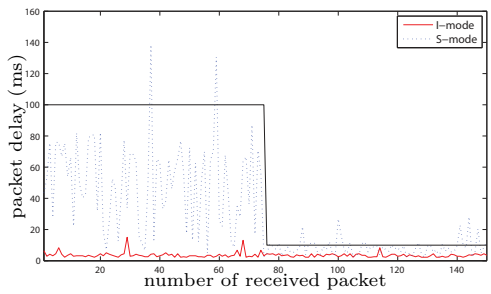
(a) α, β, τ behavior



(b) MAC parameter (m_0, m, n) behavior



(c) Reliability behavior



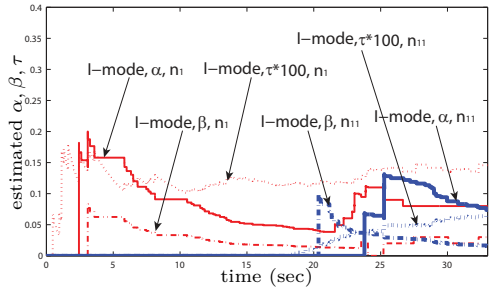
(d) Packet delay behavior

Figure 3.8: Transient condition: busy channel probabilities, channel access probability, MAC parameters, reliability and delay of *I-mode* and *S-mode* for the traffic load $\eta = 0.4$, length of the packet $L_p = 3$ and $N = 10$ nodes when the delay requirement changes from $D_{\max} = 100\,\text{ms}$ to $D_{\max} = 10\,\text{ms}$ at 26 s.

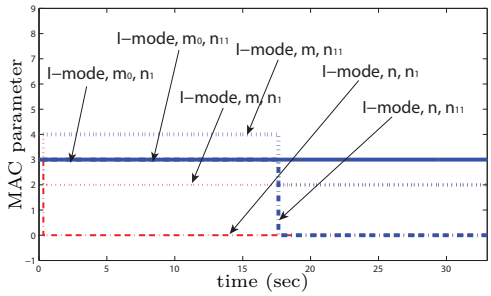
of channel estimation is a critical issue where the traffic regime is low such as in monitoring applications.
Figure 3.8(b) shows the adaptation of the MAC parameters. The optimal (m_0, m, n) of *I-mode* and *S-mode* adapts to $(3, 2, 0)$ and $(8, 5, 0)$ before the requirement changes, respectively. Observe that the algorithm returns different parameters for *I-mode* and *S-mode* due to the different power consumption model, see details in Section 3.4. After the requirement changes at time 26 s, the MAC parameters (m_0, m, n) of *S-mode* adapt from $(8, 5, 0)$ to $(5, 2, 0)$. We observe that the convergence of the MAC parameters of proposed scheme is very fast since our algorithm is based on analytical model instead of heuristic considerations as in link-based adaptation, where the algorithms adapt the contention window size by the ACK transmission [180]–[184]. In addition, recall that our adaptive IEEE 802.15.4 is based on the physical sensing information before transmitting packets.
Figure 3.8(c) shows the cumulative packet reception rate of *I-mode* and *S-mode*. Note that the oscillation of reliability is due to packet loss. In Figure 3.8(c), the reliability of *S-mode* is larger than *I-mode* since the MAC parameters m_0 and m are larger than the ones of *I-mode* before the requirement changes. By the same argument, we observe that the packet delay of *S-mode* is about six times the one measured of *I-mode* in Figure 3.8(d). In addition, the packet delay is much more variable in *S-mode* than the one in *I-mode*. Specifically, with *I-mode*, we have a reduction in the average MAC delay and a shorter tail for the MAC delay distribution with respect to the *S-mode*. After the requirement changes, the packet delay converges to around 10 ms. In addition, the reliability decreases due to the decreasing of the parameters m_0 and m in Figure 3.8(c).

3.6.3 Robustness and Sensitivity Analysis

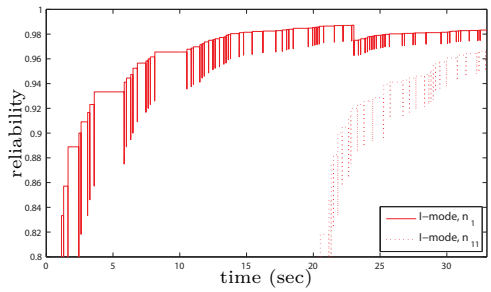
The performance analysis carried out so far assumed that the number of nodes and traffic configuration are fixed. This assumption has allowed us to verify the effectiveness of our adaptive algorithm for IEEE 802.15.4 in steady state conditions. However, one of the critical issues in the design of wireless networks is time varying condition. Therefore, in the following analysis, we will investigate our algorithm to react to changes in the number of nodes and traffic load when each node has an erroneous estimation of these parameters by using the Monte Carlo simulations.
Figures 3.9 show the dynamical behavior of the *I-mode* node when the number of nodes changes from $N = 10$ to $N = 20$ with an erroneous estimation of the number of nodes. At time 17.6 s, the number of nodes sharply increases to 20, when it was estimated to be 10. We assume that the wrong estimation happens due to some errors in the estimation phase or a biasing induced by the hidden-node phenomenon. This causes a significant increase of the contention level. Note that n_1 is one of existing nodes before the network change and n_{11} is one of the new nodes that enters the network at time 17.6 s using its initial MAC parameters. In Figure 3.9(a), we observe that the busy channel and channel access probabilities of node n_{11} become stable after the network changes by updating the MAC parameters. Figure 3.9(b)



(a) α, β, τ behavior

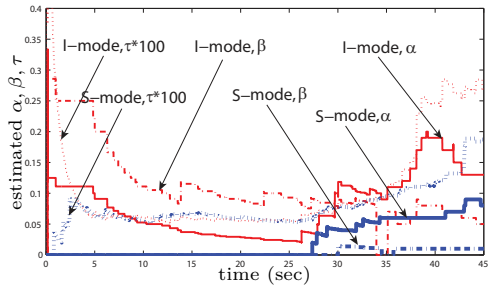


(b) MAC parameter (m_0, m, n) behavior

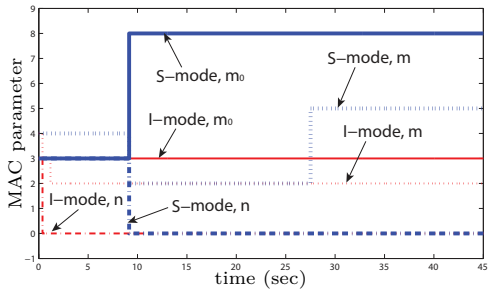


(c) Reliability behavior

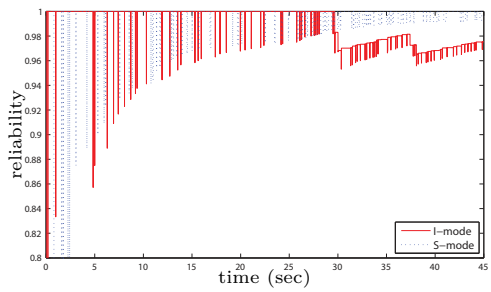
Figure 3.9: Robustness when the number of nodes changes: busy channel probabilities, channel access probability, MAC parameters and reliability behavior of *I-mode* when the number of nodes changes sharply from $N = 10$ to $N = 20$ at time 17.6 s. Note that n_1 and n_{11} represent the behavior of one of $N = 10$ nodes plus new nodes after time 17.6 s. Traffic load is $\eta = 0.4$, length of the packet is $L_p = 3$, the reliability and delay constraint are $R_{\min} = 0.95$ and $D_{\max} = 100$ ms, respectively.



(a) α, β, τ behavior



(b) MAC parameter (m_0, m, n) behavior



(c) Reliability behavior

Figure 3.10: Robustness when the traffic load changes: busy channel probabilities, channel access probability, MAC parameters, reliability and delay behavior of *I-mode* and *S-mode* when the traffic load changes sharply from $\eta = 0.2$ to $\eta = 0.5$ at time 25.6 s. The length of the packet is $L_p = 3$, the reliability and delay constraint are $R_{\min} = 0.95$ and $D_{\max} = 100$ ms, respectively.

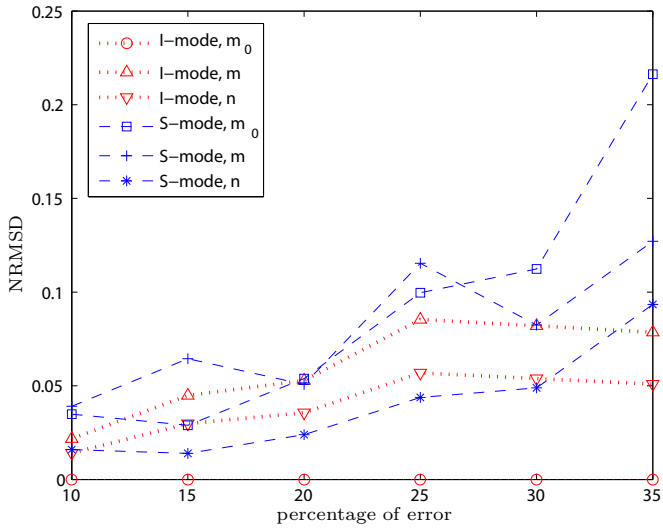


Figure 3.11: Sensitivity: NRMSD of *I-mode* and *S-mode* when the traffic load $\eta = 0.4$, length of the packet $L_p = 3$, reliability requirement $R_{\min} = 0.95$ and delay requirement $D_{\max} = 100\,\text{ms}$, and $N = 10$ nodes with different percentage error in busy channel probabilities α and β and channel access probability τ.

shows that the MAC parameters (m_0, m, n) converge to $(3, 2, 0)$ of node n_1 and n_{11}. The figures indicate that the system reacts correctly to the erroneous estimation of the number of nodes after a few seconds. In Figure 3.9(c), the reliability fulfills the requirement $R_{\min} = 0.95$ for both the existing and new nodes. Similar behaviors are observed for *S-mode*, see further details in [192].

Figures 3.10 present the behavior of the node when the traffic load changes sharply from $\eta = 0.2$ to $\eta = 0.5$ at time 25.6 s. Nodes use a wrong estimation of the traffic load, which is estimated to be $\eta = 0.2$, after the traffic load changes. The results indicate that our algorithm is quite effective for the traffic configuration change. In Figure 3.10(a), the busy channel and channel access probability increase as a result of higher traffic regime $\eta = 0.5$ for both *I-mode* and *S-mode*. Figure 3.10(b) shows that the parameter m of *S-mode* updates from 2 to 5 due to the increasing busy channel probability after the traffic load changes at time 28 s. The figure indicates that the system reacts correctly to the erroneous estimation of traffic configuration and, in few seconds, the estimation of α, β and τ allow to reach the optimal MAC parameters. In Figure 3.10(c), the reliability requirement $R_{\min} = 0.95$ is fulfilled for both *I-mode* and *S-mode*. The reliability of *I-mode* is greater than 0.95 with some fluctuations after traffic load increases.

Figure 3.11 illustrates the sensitivity of adaptive IEEE 802.15.4 with respect to the estimation errors to the busy channel probabilities α and β and the channel access probability τ. The normalized root mean squared deviation (NRMSD) between the optimal MAC parameters with exact estimation and the ones with erroneous

estimation is used as the indicator of sensitivity. The normalization is taken over the range of MAC parameters (m_0, m, n). The NRMSD is approximately below 10% if the percentage of error is smaller than 20% for α, β, τ. It is interesting to observe that m_0 of *I-mode* is very robust to errors. This is due to the power consumption model, i.e., to the dominant factor m_0 of power consumption in *I-mode*. The robustness of MAC parameter is $m_0 > n > m$ and $n > m > m_0$ for *I-mode* and *S-mode*, respectively. We can show that errors below 20% in the estimation of α, β, τ give a performance degradation below 3% in terms of reliability, packet delay and energy gain for low traffic load.

3.7 Summary

In this chapter we developed an analysis based on a generalized Markov chain model of IEEE 802.15.4, including retry limits, acknowledgements and unsaturated traffic regime. Then, we presented an adaptive MAC algorithm for minimizing the power consumption while guaranteeing reliability and delay constraints of the IEEE 802.15.4 protocol. The algorithm does not require any modifications of the standard. The adaptive algorithm is grounded on an optimization problem where the objective function is the total power consumption, subject to reliability and delay constraints on the packet delivery and the decision variables are the MAC parameters (*macMinBE*, *macMaxCSMABackoffs*, *macMaxFrameRetries*) of the standard. The proposed adaptive MAC algorithm is easily implementable on sensor nodes by estimating the busy channel and channel access probability.
We provided a test-bed implementation of the protocol, building a WSN with TelosB sensors and Contiki OS. Furthermore, we investigated the performance of our algorithm under both stationary and transient conditions by experiments and Monte Carlo simulations. Numerical results showed that the proposed scheme is efficient and ensures a longer lifetime of the network. In addition, we showed that, even if the number of active nodes, traffic configuration and application constrains change sharply, our algorithm allow the system to recover quickly and operate at its optimal parameter by estimating just the busy channel and channel access probabilities. We also studied the robustness of the protocol to possible errors during the estimation process on number of nodes and traffic load. Results indicated that the protocol reacts promptly to erroneous estimations.

Chapter 4

Performance Analysis of the IEEE 802.15.4 Hybrid Medium Access Control Protocol

The IEEE 802.15.4 communication standard is becoming the most popular protocol stack for low data rate and low power WSNs in many application domains, such as industrial control, home automation, health care, and smart grids [8, 1]. To offer flexible quality of service to several classes of applications, the MAC protocol of IEEE 802.15.4 WSNs combines the advantages of a random access with contention with a TDMA without contention. Understanding reliability, delay, and throughput is essential to characterize the fundamental limitations of this MAC and optimize its parameters. Nevertheless, there is not yet a clear investigation of the achievable performance of this hybrid MAC. In this chapter, a new analytical framework for modeling the behavior of the hybrid MAC protocol of the IEEE 802.15.4 standard is proposed. The main challenge for an accurate analysis is the coexistence of the stochastic behavior of the random access and the deterministic behavior of the TDMA scheme. The analysis is done in three steps. First, the contention access scheme of the IEEE 802.15.4 exponential backoff process is modelled through a new Markov chain that takes into account retry limits, acknowledgements, unsaturated traffic and superframe period. Second, the behavior of the TDMA access scheme is modeled by another Markov chain. Finally, the two chains are coupled to obtain a model of the hybrid MAC, which is validated by both theoretical analysis and Monte Carlo simulations. By using this new model, the network performance in terms of reliability, average packet delay, average queueing delay, and throughput is evaluated. It is established that the probability density function of the number of received packets per superframe follows a Poisson distribution. Furthermore, it is determined under which conditions the guaranteed time slot allocation mechanism of IEEE 802.15.4 is stable. It is shown that the performance of the hybrid MAC differs significantly from what was reported previously in the literature. It is concluded that the tradeoff between throughput of the random access and the TDMA scheme for a fixed-length superframe is critical to maximize the throughput of the hybrid MAC.

Recall that in Section 2.2, we give a brief overview of the IEEE 802.15.4 MAC, where we summarize the characteristic of the CAP and the TDMA period, (or CFP). In this chapter, we focus on the beacon-enabled modality because it features the hybrid MAC. The outline of the chapter is as follows. In Section 4.2, we summarize previous work for the GTS allocation mechanism of the CFP for the IEEE 802.15.4 MAC. In Section 4.3, we present the system model and assumption to analyze the performance of the IEEE 802.15.4 MAC. Then, we follow three steps to analyze the performance of the hybrid MAC. First, we propose a new Markov chain model of CSMA/CA with retry limits, ACKs, unsaturated traffic and superframe period of the CAP in Section 4.4. Then, we analyze the performance of the GTS allocation of the CFP based on a Markov chain model in Section 4.5. Third, we present a new model for the hybrid MAC by connecting these two chains in Section 4.6. In Section 4.7, we validate our analysis by Monte Carlo simulations, and show the performance of the hybrid MAC in terms of reliability, packet delay, queueing delay, and throughput. In Section 4.8, we summarize the chapter.

4.1 Background

One of the important aspects of IEEE 802.15.4 is the combination of the contention access and the TDMA MAC [8, 99]. It enables the protocol to serve a variety of applications. In Section 2.1.1, we discuss the reservation scheme of hybrid MAC protocol. The essential difference between these reservation schemes and the IEEE 802.15.4 standard is that the contention access period supports not only the reservation scheme but also the data packet communication. Therefore, the IEEE 802.15.4 hybrid MAC is a more general protocol than the simpler hybrid MAC based on reservation schemes. In this chapter, we give an analysis of the hybrid MAC of IEEE 802.15.4. To our knowledge, this is the first study considering simultaneously random access and TDMA of the IEEE 802.15.4 protocol.

4.2 Related Work

In this section, we discuss the related existing literatures of the GTS allocation mechanism of the IEEE 802.15.4 protocol. We recall that in Section 3.2.1, we present the related work of CSMA/CA algorithm of the CAP. Most of the literature does not consider satisfactorily the CFP, where the GTS mechanism operates. Simulation studies in [197]–[199] consider the CAP and CFP. An interesting theoretical performance evaluation of the GTS allocation has been proposed by Koubaa et al. [200] by using network calculus. These papers focus on the impact of the IEEE 802.15.4 MAC parameters (SO, BO), the delay, throughput, and energy consumption of the GTS allocation. In [201], a round-robin scheduler is proposed to improve the bandwidth utilization based also on a network calculus approach. Hence, network calculus assumes a continuous flow model, whereas communication happens through

low data rate packets in reality. Network calculus gives the worst-case traffic flows, which leads to severe under-utilization of time slots in actual implementations.
Some interesting algorithms have been proposed to improve the performance of the GTS allocation mechanism. To maximize the bandwidth utilization, a smaller slot size and an offline message scheduling algorithm are proposed in [202] and [203], respectively. In [204], the delay constraint and bandwidth utilization are considered for the design of a GTS scheduling algorithm. In [205], authors propose an adaptive GTS allocation scheme by considering low delay and fairness. We remark that, despite the related existing studies, there is no explicit consideration of theoretical study of both CAP and CFP of the hybrid MAC of IEEE 802.15.4.

4.3 System Model

We consider a star network with a coordinator and N nodes (see Table A.1 for main symbols used in this chapter). Every node contends to send data packet[1] to the coordinator. The coordinator acts as a data sink and we assume it does not experience the hidden node problem. Throughout this chapter we consider applications where nodes asynchronously generate packets for transmission. We consider the underlying minimum time unit corresponding to *aUnitBackoffPeriod*, as defined in the IEEE 802.15.4 standard and we denote it T_b. In the standard, T_b corresponds to 20 symbols in the physical layer (i.e., $320\,\mu s$ for 2.45 GHz). When a node just has sent a packet successfully or just discarded a packet, we assume that a new packet is generated with probability η_t. If a new packet is not generated, then the node tries to generate a new packet after $h\,T_b$ s, where h is a positive integer. This packet is generated with probability η_p. We consider two different types of data packets: non time-critical data packets to be transmitted during the CAP, and time-critical data packets to be transmitted during the CFP using the GTS allocation mechanism.
When a node decides to generate a data packet, it generates a non time-critical data packet with probability η_d and time-critical data packet with probability $1 - \eta_d$ in our model. A node uses a beacon-enabled slotted CSMA/CA algorithm to send a non time-critical data packet and a GTS request to the coordinator during the CAP. Note that the packet transmission is successful if an ACK packet is received. For a time-critical data packet, the node informs the need of GTS resources by sending the request during the CAP. The coordinator allocates a number of GTSs by considering the received GTS requests. Each node may need to send a multiple number of time-critical packets wherein each packet has a fixed length due to the maximum length of a packet defined in the standard. The requests are stored in a queue of the coordinator, and wait to be served in the next superframes, where the related GTS may be allocated. If too many requests arrive with respect to the coordinator queue size, then we have a queue overflow. We assume an ideal channel

[1]Throughout Chapters 4 and 3, we use the term "packet" to denote a protocol data unit or frame at the MAC layer.

condition of physical layer and perfect channel channel sensing capability of nodes. Furthermore, we make the natural assumption that each node forwards a non time-critical packet or a GTS request within $2T_{\mathrm{BI}}$ i.e., the maximum packet delay of the CAP is $2T_{\mathrm{BI}}$.
Based on the introduced model and assumptions, we propose an analytical model of the slotted CSMA/CA algorithm of the CAP and the GTS allocation of the CFP based on two Markov chain models in Sections 4.4 and 4.5. Then, in Section 4.6, we connect these to have a model that allows us to investigate the performance of the hybrid MAC of IEEE 802.15.4 in terms of the reliability of the CAP, the average delay of the CAP, the queueing delay of the CFP, and the throughput of the network.

4.4 Performance Analysis of CAP

In this section, we analyze the performance of the CSMA/CA algorithm of the CAP. In Section 4.4.1, we propose the model of such an algorithm to send a non time-critical data packet and GTS requests. The core contribution of the section is the proposal of a new Markov chain model. Then, in Section 4.4.2, we derive the reliability and average packet delay of CSMA/CA mechanism of the CAP based on this model. In contrast to existing literature, all the key characteristics of IEEE 802.15.4 are considered, such as the limited number of retransmissions, ACKs, unsaturated traffic, packet size, and superframe structure.

4.4.1 Modeling of CAP

Here, we develop a generalized Markov chain model of the slotted CSMA/CA algorithm of the beacon-enabled IEEE 802.15.4 MAC. The core contribution of the analysis is the derivation of the stationary probability distribution of the chain, which is summarized by Proposition 2. Compared to previous results, e.g., [158]–[172], the novelty of this chain consists in the modeling of the retry limits for each packet transmission, the ACK, the inclusion of unsaturated traffic regimes, packet size, and superframe structure. We will also discuss the strength of the proposed Markov chain model with respect to previous studies [169]–[171], which do not take into account the superframe structure accurately.
Let $b(t), c(t), e(t)$ and $f(t)$ be the stochastic processes representing the backoff stage, the state of the backoff counter, the state of retransmission counter, and the state of deferred transmission at time t experienced by a node. The binary variable $f(t)$ indicates if a transmission has been deferred ($f(t) = 1$) or not ($f(t) = 0$), which is due to the limited size of superframe duration to transmit a packet. By making the natural assumption that nodes start sensing independently, the stationary probability τ that a node attempts a first carrier sensing in a randomly chosen time slot is constant and independent of other nodes. The quadruple $(b(t), c(t), e(t), f(t))$ is the state evolution of the Markov chain. We use (i, j, k, l) to denote a particular state. We assume the following notation for the MAC parameters: $m_0 \triangleq macMinBE, m \triangleq$

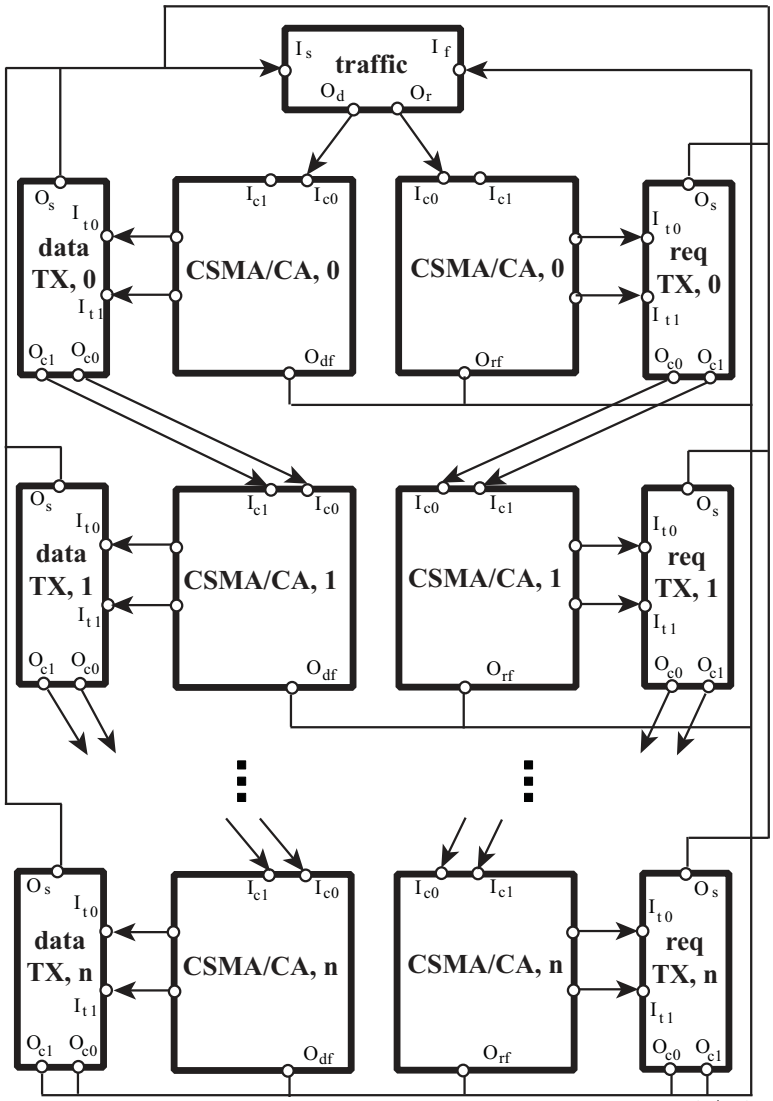
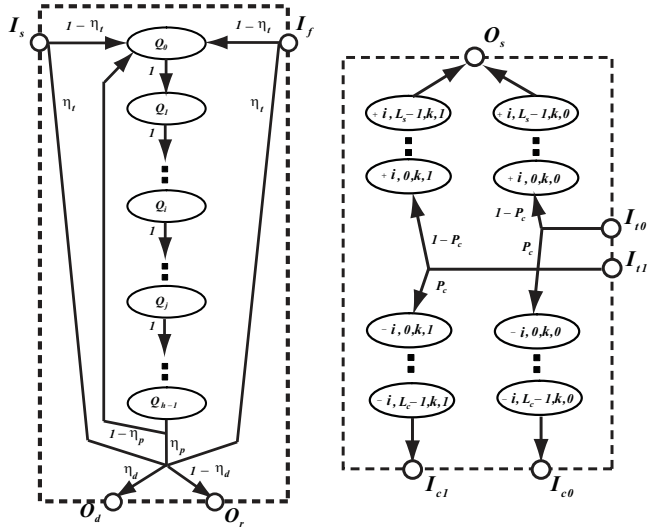
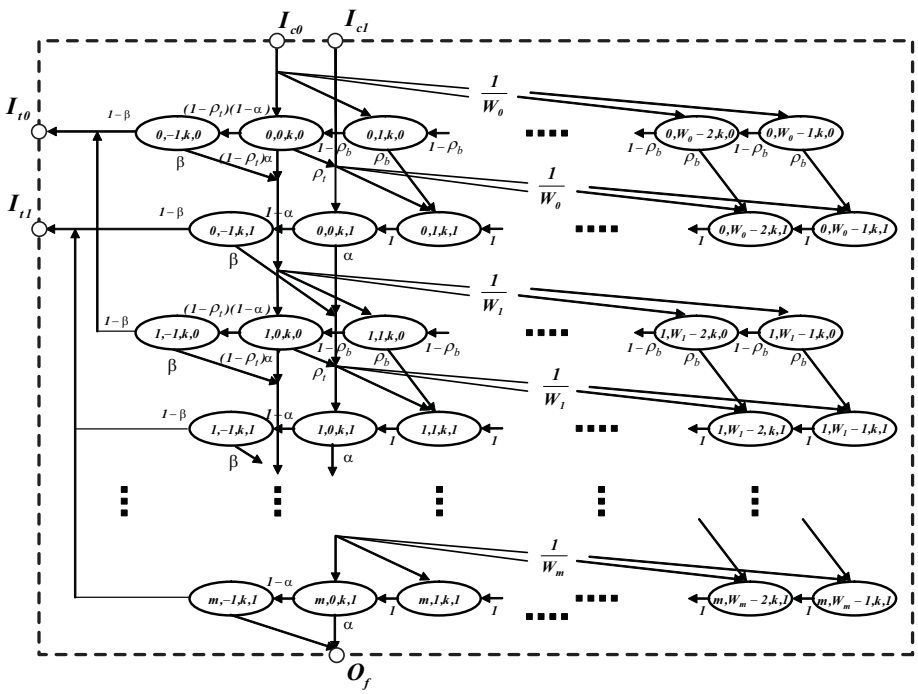


Figure 4.1: Generalized Markov chain building blocks modeling the CSMA/CA algorithm of the IEEE 802.15.4 MAC for a single node.



(a) Traffic generation block. (b) Packet transmission block.



(c) CSMA/CA algorithm block of the k-th retransmission state.

Figure 4.2: Detailed description of the functionalities of every block of Figure 4.1.

macMaxCSMABackoffs, $n \triangleq$ *macMaxFrameRetries*, $m_b \triangleq$ *macMaxBE*, $W_0 \triangleq 2^{m_0}$, $W_m \triangleq 2^{\min(m_0+m,m_b)}$, where *macMinBE* is the minimum value of the backoff exponent, *macMaxCSMABackoffs* is the maximum number of backoffs allowed, *macMaxFrameRetries* is the maximum number of retries allowed, and *macMaxBE* is the maximum value of the backoff exponent in the CSMA/CA algorithm.

The Markov chain consists of three parts corresponding to the traffic generation block, CSMA/CA algorithm blocks, and packet transmission blocks in Figure 4.1. The states $Q_0, \ldots, Q_{h-1}$ of Figure 4.2(a) correspond to the idle-queue states when the packet queue is empty and the node is waiting for the next packet generation time. Note that the idle-queue states $Q_0, \ldots, Q_{h-1}$ take into account the sampling interval of the node. Each node generates a non time-critical data packet with probability η_d and a time-critical packet with probability $1 - \eta_d$. Then, the node performs the CSMA/CA algorithm to check a clean channel to send the non time-critical data packet or the GTS request for the time-critical packet. The Markov chain of the CSMA/CA algorithm consists of two parts corresponding to the backoff states and CCA states reported in Figure 4.2(c). The states from $(i, W_m - 1, k, l)$ to $(i, W_0 - 1, k, l)$ represent the backoff states. The states $(i, 0, k, l)$ and $(i, -1, k, l)$ represent the first CCA (CCA_1) and second CCA (CCA_2), respectively. Let α be the probability that CCA_1 is busy, and β the probability that CCA_2 is busy. If a node fails to obtain a clear channel due to repeated busy channel, then the packet is discarded. If the channel sensing is successful, then the node goes to the packet transmission block. In Figure 4.2(b), the states $(+i, j, k, l)$ and $(-i, j, k, l)$ correspond to successful transmission and collision, respectively. Note that the states $i = 1$ and $i = 2$ denote the non time-critical data packet and the GTS request of time-critical data packet, respectively.

Before deriving the stationary probability of the Markov chain of Figure 4.1, we need some definitions. Let P_c be the collision probability. If the packet collides, then the node repeats the CSMA/CA algorithm until a maximum number of retransmissions n. If the packet transmission is successful, then it goes back to the traffic generation block. Recall that we define the successful packet transmission time L_s and the packet collision time L_c with ACK and the successful packet transmission time L_g without ACK in Eqs (2.3)–(2.5). Note that the successful packet transmission time $L_{s,d}$ and the packet collision time $L_{c,d}$ of a non time-critical data packet with packet length $L_{p,d}$ are obtained by substituting $L_{p,d}$ with L_p in Eqs. (2.3) and (2.4), respectively. Similarly, we derive the successful GTS request transmission time $L_{s,r}$ and the GTS request collision time $L_{c,r}$ for a length of GTS requests $L_{p,r}$.

To compute the stationary probability of the Markov chain, we first derive the probability ρ_t that the transmission is deferred due to the lack of the remaining time slots in a CAP as follows. The total number of time slots that are needed for a single transmission is $2L_{\mathrm{sc}} + L_p + L_{\mathrm{w,ack}} + L_{\mathrm{ack}}$ where two slots $2L_{\mathrm{sc}}$ are included due to the number of time slots for performing two CCAs and other components take into account the packet transmission with an ACK frame in Eqs. (2.3) and (2.4).

Hence, ρ_t is approximated by

$$\rho_t = \frac{L_{\text{tx}}}{T_{\text{CAP}}} = \frac{2L_{\text{sc}} + L_p + L_{\text{w,ack}} + L_{\text{ack}}}{T_{\text{CAP}}}\,, \tag{4.1}$$

where T_{CAP} is the total number of time slots in a CAP. Similarly, the probabilities $\rho_{p,d}$ and $\rho_{p,r}$ of the events of deferred attempts due to the lack of remaining slot times for a non time-critical data packet and for a GTS request are obtained by replacing L_p with $L_{p,d}$ and $L_{p,r}$ in Eq. (4.1), respectively. We remark here that the previous literature [169]–[171] does not consider the extra backoff mechanism for the event of deferred attempts of the transmission which is explicitly described in the standard [8]. Next, ρ_b is the probability that the MAC sublayer pauses the backoff countdown at the end of the CAP due to the limited length. Analogously, the approximated probability is

$$\rho_b = \frac{1}{T_{\text{CAP}}}\,. \tag{4.2}$$

Note that the previous literature [169]–[171] does not take into account this event although it is a very critical aspect for delay analysis, as we show in Section 4.4.2. We have the following results:

Proposition 2. *Let the stationary probability of the Markov chain in Figure 4.1 be*

$$S_{i,j,k,l} = \lim_{t\to\infty} \Pr\left[b(t) = i, c(t) = j, e(t) = k, f(t) = l\right],$$

where $i \in (+2,-2,+1,-1) \cup (0,m), j \in (-1, \max(W_i - 1, L_s - 1, L_c - 1)), k \in (0,n), l \in (0,1)$. *Then,*

$$S_{i,j,k,0} = \varpi_{i,j} g^k S_{0,0,0,0}\,,$$

with $\varpi_{i,j} = \left[1-(1-\rho_b)^{W_i - j}\right]\left[1-(1-\rho_b)^{W_i}\right]^{-1}$, *and*

$$g = P_c(1-\rho_t)(1-x)\sum_{i=0}^{m}\xi_i \sum_{j=0}^{W_0-1}\frac{(1-\rho_b)^j}{W_0}\,,$$

where $\xi_i = \prod_{r=1}^{i}(1-\rho_t)x/W_r \sum_{j=0}^{W_r-1}(1-\rho_b)^j$, $x = \alpha + (1-\alpha)\beta$ *and* $\xi_0 = 1$. *Moreover,*

$$\begin{aligned} S_{i,j,k,1} = \Bigg[& \sum_{r=j}^{W_i-1}\rho_b \varpi_{i,r}\xi_i g^k + \rho_t \frac{W_i - j}{W_i}\xi_i g^k + x\frac{W_i - j}{W_i} v_{i-1,k} u(i-1) \\ & + P_c(1-x)\frac{W_0 - j}{W_0}\sum_{i=0}^{m} v_{i,k-1}\delta(i)u(k-1)\Bigg] S_{0,0,0,0}\,, \end{aligned}$$

where $u(i)$ is the unit step function, $\delta(i)$ is the unit discrete delta function and

$$
\begin{aligned}
v_{i,k} &= x^i c + x^i \sum_{r=1}^{i} \frac{\rho_t + \rho_b \sum_{j=1}^{W_r-1} \varpi_{r,j}}{x^r} g^k + x^i a_k \,, \\
a_k &= P_c(1-x) \sum_{i=0}^{m} v_{i,k-1}, \qquad k \geq 1 \,, \\
v_{0,0} &= \rho_t + \rho_b \, c \sum_{j=1}^{W_0-1} \varpi_{0,j} \,,
\end{aligned}
$$

and $c = \rho_t + \rho_b \sum_{j=1}^{W_0-1} \varpi_{0,j}$ *and* $a_0 = 0$.

Proof. See Section C.1. □

We remark here that the probability $S_{0,0,0,0}$, which plays a key role in the analysis, is different from the corresponding term given in [158]–[171] due to our new modeling of retry limits, ACK, unsaturated traffic, packet size, and superframe period. In Section 4.7, we demonstrate the validity of this Markov chain model by Monte Carlo simulations, and show how the performance analysis is affected by the new more accurate model we derive in this chapter.

4.4.2 Performance Indicators of CAP

We now use the Markov chain model developed in the previous section to derive the performance indicators of the CAP in terms of reliability and average delay in Section 4.4.2 and 4.4.2, respectively. The main contributions of this section are given by Propositions 3 and 4 below.

Reliability

The main contributions of this section is the derivation of the probability of successful packet reception, or reliability. With this goal in mind, we derive first the probability that a node attempts CCA_1 in a randomly chosen time slot:

$$
\begin{aligned}
\tau &= \sum_{i=0}^{m} \sum_{k=0}^{n} \sum_{l=0}^{1} S_{i,0,k,l} \\
&= \left(\frac{1-g^{n+1}}{1-g} \sum_{i=0}^{m} \xi_i + \sum_{i=0}^{m} \sum_{k=0}^{n} v_{i,k} \right) S_{0,0,0,0} \,. \qquad (4.3)
\end{aligned}
$$

This probability depends on the probability that a transmitted packet encounters a collision P_c, the probability that CCA_1 is busy α, and the probability that CCA_2 is also busy β. These probabilities are developed in the following.

Recall that the term P_c is the probability that at least one of the $N-1$ remaining nodes transmits in the same time slot. If all nodes transmit with probability τ, then

$$P_c = 1 - (1-\tau)^{N-1},$$

where recall that N is the number of total nodes present in the network.
Similarly to [158], we derive the busy channel probabilities α and β as follows. The busy channel probability of CCA_1 is

$$\alpha = \alpha_1 + \alpha_2, \tag{4.4}$$

where α_1 is the probability of finding channel busy during CCA_1 due to data transmission, namely

$$\alpha_1 = \overline{L}_p(1-(1-\tau)^{N-1})(1-\alpha)(1-\beta),$$

where the average length of packet is $\overline{L}_p = \eta_d L_{p,d} + (1-\eta_d)L_{p,r}$, and α_2 is the probability of finding the channel busy during CCA_1 due to ACK transmission, which is

$$\alpha_2 = L_{\mathrm{ack}}\frac{N\tau(1-\tau)^{N-1}}{1-(1-\tau)^N}(1-(1-\tau)^{N-1})(1-\alpha)(1-\beta),$$

where L_{ack} is the length of the ACK. The busy channel probability of CCA_2 is

$$\beta = \frac{1-(1-\tau)^{N-1}+N\tau(1-\tau)^{N-1}}{2-(1-\tau)^N+N\tau(1-\tau)^{N-1}}, \tag{4.5}$$

see details in [166]. The expressions of the carrier sensing probability τ and the busy channel probabilities α and β form a system of non-linear equations that can be solved via numerical methods.

Proposition 3. *Consider the definitions given in Proposition 2. Then, the reliability is*

$$\begin{aligned} R =& 1-(1-\rho_t)x\xi_m\frac{1-g^{n+1}}{1-g} - x\sum_{k=0}^{n} v_{m,k} \\ & - P_c(1-x)\left((1-\rho_t)\sum_{i=0}^{m}\xi_i g^n + \sum_{i=0}^{m} v_{i,n}\right). \end{aligned} \tag{4.6}$$

Proof. In slotted CSMA/CA, packets are discarded due to two reasons: (i) channel access failure and (ii) retry limits. Channel access failure happens when a packet fails to obtain idle channel in two consecutive CCAs within $m+1$ backoffs. Furthermore, a packet is discarded if the transmission fails due to repeated collisions after $n+1$ attempts. Following the Markov model presented in Figure 4.1, the probability that

the packet is discarded due to channel access failure is

$$P_{dc} = \sum_{k=0}^{n} ((1-\rho_t) x S_{m,0,k,0} + x S_{m,0,k,1}) S^{-1}_{0,0,0,0}$$
$$= (1-\rho_t) x \xi_m \frac{1-g^{n+1}}{1-g} + x \sum_{k=0}^{n} v_{m,k} \,. \tag{4.7}$$

The probability of a packet being discarded due to retry limits is

$$P_{dr} = \sum_{i=0}^{m} \sum_{l=0}^{1} P_c (1-\beta) S_{i,-1,n,l} S^{-1}_{0,0,0,0}$$
$$= P_c (1-x) \left((1-\rho_t) \sum_{i=0}^{m} \xi_i g^n + \sum_{i=0}^{m} v_{i,n} \right) . \tag{4.8}$$

Therefore, by considering Eqs. (4.7) and (4.8), the reliability is

$$R = 1 - P_{dc} - P_{dr} \,. \tag{4.9}$$

□

Delay

In this section, we derive the analytical expression of the average delay for a successfully received packet. The average delay for a successfully received packet is defined as the time interval from the instant the packet is at the head of its MAC queue and ready to be transmitted, until the transmission is successful and the ACK is received.

Proposition 4. *The average delay to transmit a packet and successfully receive an ACK is*

$$\begin{aligned} \mathbb{E}[D] = & \sum_{k=0}^{n} (1-P_c) P_c^k \left[\Pr[\mathscr{B}_0]^k \left(\Pr[\mathscr{B}_0](L_s + kL_c \right. \right. \\ & + (k+1)\mu_{\mathscr{B}_0}) + \Pr[\mathscr{T}](L_s + kL_c + k\mu_{\mathscr{B}_0}) + \mu_{\mathscr{T}}) \\ & + \sum_{r=0}^{k-1} \Pr[\mathscr{B}_0]^{k-1-r} \Pr[\mathscr{B}_1]^{r+1} (\Pr[\mathscr{T}](L_s + kL_c \\ & \left. \left. + (k-1-r)\mu_{\mathscr{B}_0} + (r+1)\mu_{\mathscr{B}_1}) + \mu_{\mathscr{T}}) u(k-1) \right] P_{tot}^{-1} \,, \end{aligned} \tag{4.10}$$

where

$$\begin{aligned} \Pr[\mathscr{T}] = & \Pr[\mathscr{C}_0] + \Pr[\mathscr{C}_1] + \Pr[\mathscr{F}_0] + \Pr[\mathscr{F}_1] \,, \\ \mu_{\mathscr{T}} = & \Pr[\mathscr{C}_0]\mu_{\mathscr{C}_0} + \Pr[\mathscr{C}_1]\mu_{\mathscr{C}_1} + \Pr[\mathscr{F}_0]\mu_{\mathscr{F}_0} + \Pr[\mathscr{F}_1]\mu_{\mathscr{F}_1} \,, \\ P_{tot} = & \sum_{k=0}^{n} (1-P_c) P_c^k \left[\Pr[\mathscr{B}_0]^k \left(\Pr[\mathscr{B}_0] + \Pr[\mathscr{T}] \right) \right. \\ & \left. + \sum_{r=0}^{k-1} \Pr[\mathscr{B}_0]^{k-1-r} \Pr[\mathscr{B}_1]^{r+1} \Pr[\mathscr{T}] u(k-1) \right] \,, \end{aligned}$$

where $\mu_* = \sum_{i=0}^{m} \rho_{*,i} / \Pr[*]$ *, and where* $*$ *is one of the events* $\{\mathscr{B}_0, \mathscr{B}_1, \mathscr{C}_0, \mathscr{C}_1, \mathscr{F}_0, \mathscr{F}_1\}$*, with*

$$\begin{aligned}
\rho_{\mathscr{B}_{0,i}} =& \varepsilon_i \sum_{k=0}^{i} (1-\rho_t)\Psi_i(1-x)\mu_{D_{k,i}} \,, \\
\rho_{\mathscr{B}_{1,i}} =& \sum_{k=0}^{i} \alpha^{i-k}((1-\alpha)\beta)^k(1-x)\mu_{D_{k,i}} \,, \\
\rho_{\mathscr{C}_{0,i}} =& \varepsilon_i \sum_{k=0}^{i} \rho_t \Psi_i(1-x)\left(\mu_{D_{k,i}} + T_{SP} + L_{tx} + \mu_{\Phi_i}\right) \,, \\
\rho_{\mathscr{C}_{1,i}} =& \varepsilon_i \sum_{k=0}^{i} (1-\Psi_i)(1-x)\left(\mu_{D_{k,i}} + T_{SP} + \mu_{\Phi_i}\right) \,, \\
\rho_{\mathscr{F}_{0,i}} =& \rho_t x \sum_{f=0}^{i-2} \sum_{h=0}^{i-f-1} \varepsilon_f \sum_{k=0}^{f} \Psi_f \alpha^{i-f-h-1}((1-\alpha)\beta)^h \\
& \times (1-x)\left(\mu_{D_{k,i}} + T_{SP} + L_{tx} + \mu_{\Phi_f}\right) \,, \\
\rho_{\mathscr{F}_{1,i}} =& x \sum_{f=0}^{i-2} \sum_{h=0}^{i-f-1} \varepsilon_f \sum_{k=0}^{f} (1-\Psi_f)\,\alpha^{i-f-h-1} \\
& \times ((1-\alpha)\beta)^h(1-x)\left(\mu_{D_{k,i}} + T_{SP} + \mu_{\Phi_f}\right) \,,
\end{aligned}$$

with

$$\begin{aligned}
\Pr[\mathscr{B}_0] =& \sum_{i=0}^{m} \varepsilon_i \sum_{k=0}^{i} (1-\rho_t)\Psi_i(1-x) \,, \\
\Pr[\mathscr{B}_1] =& \sum_{i=0}^{m} \sum_{k=0}^{i} \alpha^{i-k}((1-\alpha)\beta)^k(1-x) \,, \\
\Pr[\mathscr{C}_0] =& \sum_{i=0}^{m} \varepsilon_i \sum_{k=0}^{i} \rho_t \Psi_i(1-x) \,, \\
\Pr[\mathscr{C}_1] =& \sum_{i=0}^{m} \varepsilon_i \sum_{k=0}^{i} (1-\Psi_i)(1-x) \,, \\
\Pr[\mathscr{F}_0] =& \rho_t x \sum_{i=0}^{m} \sum_{f=0}^{i-2} \sum_{h=0}^{i-f-1} \varepsilon_f \sum_{k=0}^{f} \Psi_f \alpha^{i-f-h-1}((1-\alpha)\beta)^h(1-x)u(i-2) \,, \\
\Pr[\mathscr{F}_1] =& x \sum_{i=0}^{m} \sum_{f=0}^{i-2} \sum_{h=0}^{i-f-1} \varepsilon_f \sum_{k=0}^{f} (1-\Psi_f)\,\alpha^{i-f-h-1}((1-\alpha)\beta)^h(1-x)u(i-2) \,,
\end{aligned}$$

and $\mu_{\Phi_k} = \sum_{l=0}^{W_k-1} T_b\, l / W_k$, $\Psi_i = \sum_{j=0}^{W_i-1} (1-\rho_b)^j / W_i$, $\mu_{D_{k,i}} = \mu_{\Phi_k} + (i+1)2L_{sc}$,

$$\epsilon_r = (1-\rho_t)\Psi_r \, \max(\alpha\,, (1-\alpha)\beta) \,,$$

$$\varepsilon_i = \begin{cases} \prod_{r=0}^{i-1} \epsilon_r & \text{if } i \geq 1\,, \\ 1 & \text{otherwise}\,, \end{cases}$$

T_{SP} is the inactive period.

Proof. See Section C.2. □

We remark that $\mathbb{E}[D]$ is a function of the MAC parameters W_i, m, n, L_{sc} of CSMA/CA mechanism as well as the MAC parameters $T_{\text{SP}}, \rho_t, \rho_b$ related to superframe period. Furthermore, the busy channel probabilities α and β, and collision probability P_c of the network affect the average delay.

4.5 Performance Analysis of CFP

We now turn our attention from the CAP to the CFP. We present the modeling of the GTS allocation mechanism based on a new Markov chain model in Section 4.5.1. Then, in Section 4.5.2 we build on this modeling to characterize the average queueing delay of the GTS allocation.

4.5.1 Modeling of CFP

The modeling of the GTS allocation is given in two steps. First, we derive the constraints on the number of time slots to allocate from the IEEE 802.15.4 specification. Then, we model the behavior of the GTS allocation by a Markov chain.
First, we derive the number of GTSs Δ_u that can be allocated as a function of the IEEE 802.15.4 MAC parameters BO, SO and the number of time-critical data packets φ_n for each GTS request that can be served. Recall that during the CFP each packet has a fixed length L_g given by Eq. (2.5).
We assume that all GTS transmissions are successful and each node is allocated at most one GTS. Furthermore, we assume that there is no reallocation of the GTS. Let the duration of a superframe slot be T_{SS}, then

$$T_{\text{SS}} = \frac{T_{\text{SD}}}{N_{\text{SS}}} = T_0 \times 2^{\text{SO}-4}\,,$$

where T_{SD} is the number of symbols forming a superframe, N_{SS} is the number of slots contained in a superframe, and T_0 is the number of symbols forming a superframe when $\text{SO} = 0$. Consider that a single GTS may extend over a number of superframe slots θ_n. Since a given GTS needs to be larger than the total forward delay $\varphi_n L_g$, it follows that

$$\theta_n \geq \underline{\theta} = \left\lceil \frac{\varphi_n L_g}{T_{\text{SS}}} \right\rceil, \tag{4.11}$$

where $\underline{\theta}$ is the minimum number of superframe slots for a single GTS to serve the data frames $\varphi_n L_g$. Because the minimum CAP length, $T_{\text{CAP}} \geq T_{\min}$, the constraint

of the maximum number of GTSs that the coordinator can allocate in a superframe is

$$T_{\mathrm{SD}} - T_{\mathrm{CFP}} = T_{\mathrm{SD}} - T_{\mathrm{SS}}\, k\, \underline{\theta} \geq T_{\min}\,,$$

where k is the number of waiting requests and T_{CFP} is the number of symbols forming a CFP. The maximum number of GTSs to be allocated to nodes is

$$\Delta_u = \min\left(\left\lfloor \frac{N_{\mathrm{SS}}\left(1-\frac{T_{\min}}{T_{\mathrm{SD}}}\right)}{\underline{\theta}} \right\rfloor,\, N_{\mathrm{GTS}}\right), \tag{4.12}$$

where N_{GTS} is the maximum number of GTS descriptors limited to seven in the IEEE 802.15.4 [8]. Note that Δ_u is a function of the parameters BO, SO and application constraints, i.e., the number of time-critical data packets φ_n wherein each packet has a fixed length L_g.

Now, we develop a Markov model for the GTS allocation mechanism. Let $r(t)$ be the stochastic processes representing the number of waiting requests of the coordinator at the beginning of the superframe t. This stochastic process defines a one-dimensional Markov chain. If a number of received requests are over the limited queue size q after the new arrivals, then some of the new requests will be dropped on the base of a FCFS policy. We denote by $\bar{q}$ the state of the Markov chain indicating that the new arrived requests are dropped due to the limited queue size. Let λ_i be the probability of i successful requests during the CAP, given that the maximum number of requests is $\bar{\lambda}$. If the arriving requests observe that the queue size is over a threshold q, then these arriving requests are dropped because of a time limitation requirement N_{DPT}. Note that q is fixed and it is given by $\Delta_u\,(N_{\mathrm{DPT}}+1)$ where Δ_u is the maximum number of GTSs for each superframe and $N_{\mathrm{DPT}} = aGTSDescPersistenceTime$. After N_{DPT} superframes, the GTS description of the beacon is removed. When there are k requests, $\Delta_u \leq k \leq q$, then some requests $k - \Delta_u$ will be delayed to obtain GTS in the next superframe. The transition probabilities of the chain are

$$P(j|i) = \lambda_j,\ \text{for } 0 \leq i < \Delta_u, j \neq \bar{q}\,, \tag{4.13}$$

$$P(j|i) = \lambda_{j+\Delta_u-i},\ \text{for } i \geq \Delta_u, j \geq i - \Delta_u\,, i \neq \bar{q} \tag{4.14}$$

$$P(j|i) = 0,\ \text{for } i \geq \Delta_u, j < i - \Delta_u, j \neq \bar{q}\,, \tag{4.15}$$

$$P(\bar{q}|i) = \lambda_q^+,\ \text{for } 0 \leq i < \Delta_u\,, \tag{4.16}$$

$$P(\bar{q}|i) = \lambda^+_{q+\Delta_u-i},\ \text{for } i \geq \Delta_u\,, \tag{4.17}$$

where $\lambda_k^+ = 1 - \sum_{i=0}^{k} \lambda_i = \sum_{i=k+1}^{\bar{\lambda}} \lambda_i$. Eq. (4.13) gives the probability that j requests arrive when there are less than Δ_u requests, the maximum number of GTSs. Note that GTSs will be allocated to previously arrived requests before the CAP of the superframe, i.e., all waiting requests at the beginning of the current superframe arrive during the CAP of the previous superframe. Eq. (4.14) gives the probability

that j requests remain at the current superframe when there are some requests $i \geq \Delta_u$ at the previous superframe. The coordinator allocates a maximum number of GTS Δ_u to serve the requests $i \geq \Delta_u$. Furthermore, since the number of waiting requests is greater than Δ_u, some requests will fail to obtain GTS at the superframe. Hence, there are two different groups of requests at the current superframe, one group of requests arriving before the current superframe and another group of requests arriving at the current superframe. Let us call the first group "*old requests*" and the second group "*new requests*" throughout this chapter. Eq. (4.15) shows the probability that there are less requests in the current superframe than $i-\Delta_u$ if there are $i \geq \Delta_u$ requests at previous superframe. Note that the transition probability of Eq. (4.15) is equal to 0 due to the maximum number of GTSs Δ_u. Eqs. (4.16) and (4.17) show the transition probabilities of the drop state $\bar{q}$ for the requests at the current superframe.

Let us denote the probability that the process described by the Markov chain is in state k at time t by $\pi_k^t = \Pr[r(t) = k]$. Let the state probability vector at time t be $\boldsymbol{\pi}^t = (\pi_0^t, \pi_1^t, \ldots, \pi_{\bar{q}}^t)$, for $t \in \{1, \cdots\}$ and $k \in \{0, \cdots, q, \bar{q}\}$. Then, we obtain $\boldsymbol{\pi}^t = \boldsymbol{\pi}^1 \mathbf{P}^t$ where $\mathbf{P}^t$ is the t-step transition probability matrix

$$\mathbf{P} = \begin{pmatrix} \lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u+1} & \cdots & \lambda_q & \lambda_q^+ \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u+1} & \cdots & \lambda_q & \lambda_q^+ \\ 0 & \lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u} & \cdots & \lambda_{q-1} & \lambda_{q-1}^+ \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u+1} & \lambda_{\Delta_u+1}^+ \\ 0 & 0 & \cdots & 0 & \lambda_0 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u}^+ \\ 0 & 0 & \cdots & 0 & \lambda_0 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u}^+ \end{pmatrix}, \tag{4.18}$$

The stationary distribution is

$$\boldsymbol{\pi} = \lim_{t \to \infty} \boldsymbol{\pi}^t = \mathbf{1} \left(\mathbf{P} - \mathbf{I} + \mathbf{1}^T \mathbf{1} \right)^{-1}, \tag{4.19}$$

where $\boldsymbol{\pi} \in \mathbb{R}^{1 \times (q+2)}$, $\mathbf{P}$ is the transition probability matrix given in Eq. (4.18), $\mathbf{I} \in \mathbb{R}^{(q+2) \times (q+2)}$ is the identity matrix, $\mathbf{1} \in \mathbb{R}^{1 \times (q+2)}$ is the vector with all elements equal to one, and $\mathbf{1}^T \mathbf{1} \in \mathbb{R}^{(q+2) \times (q+2)}$ is the matrix with all the elements equal to one [206].

4.5.2 Performance Indicators of CFP

In this section, we analyze the expected queueing delay of the GTS allocation, namely the average delay between the arrival time of a GTS request to a coordinator and the actual transmission time of a time-critical packet after its allocation in some of the next superframes. The core contribution of this section is given by Proposition 5.

The coordinator determines a node list of the GTS allocation of the next superframe based on a FCFS policy. When *new requests* are received during the CAP, then the

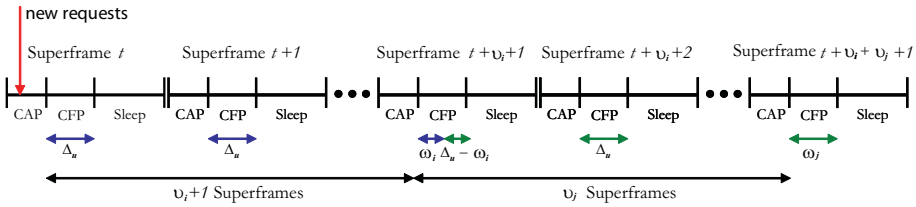


Figure 4.3: Waiting time of j *new requests* that observe i *old requests* at superframe t.

delay of the GTS allocation can be estimated by observing the queue size of the waiting requests. Note that a coordinator makes a preliminary decision whether it is able to serve a request or not. If there are no sufficient GTS resources for the request in the current superframe, the GTS allocation will be delayed to the next superframe. Assume that the arrival process of requests is uniformly distributed during the CAP. The average delay between the arrival time of the CAP and the end of the CAP at a superframe is the half of the CAP.

Suppose that j requests arrive at superframe t and that there are i requests already waiting in the queue. We can calculate the number of successful requests that obtain a GTS and delayed requests that fail to obtain a GTS at the next superframe, as explained in the following. Based on the FCFS policy, j *new requests* are able to obtain GTS after i *old requests* arriving before the current superframe t. In Figure 4.3, we first compute the number of superframes to allocate GTSs for *old requests*. i *old requests* require υ_i+1 superframes, plus ω_i remainders that obtain the GTSs later, after υ_i+1 superframes have been served. Hence, i *old requests* require υ_i+2 superframes to obtain GTS if there are ω_i remainders. These remainders will share the CFP along with the *new requests* at superframe $t+\upsilon_i+1$, see Figure 4.3. It follows that

$$\upsilon_i = \left\lfloor \frac{i-\Delta_u}{\Delta_u} \right\rfloor , \qquad \omega_i = \text{rem}\left(\frac{i-\Delta_u}{\Delta_u}\right) ,$$

where υ_i is the integer quotient and the function **rem** returns an integer remainder. The nominator $i-\Delta_u$ takes into account the current GTS allocation mechanism of superframe t. Note that the *new requests* are not able to get GTS at the current superframe t since the beacon includes information of the GTS allocation at each superframe.

By considering the waiting time of *old requests*, it is possible to calculate the delay of the j *new requests*. The quotient υ_i of *old requests* can be considered as a delay offset in the computation of the delay of the *new requests*. We consider ω_i remainders by summing it with j *new requests* due to the shared CFP with the ω_i remained requests. In a similar way, the quotient and remainder of the *new requests* are given by

$$\upsilon_j = \left\lfloor \frac{j+\omega_i}{\Delta_u} \right\rfloor , \qquad \omega_j = \text{rem}\left(\frac{j+\omega_i}{\Delta_u}\right) .$$

The *new requests* require $\upsilon_j + 1$ superframes to allocate all GTS requests. In Figure 4.3, ω_j waiting requests will remain waiting to obtain GTS at superframe $t + \upsilon_i + \upsilon_j + 1$.
To analyze the delay, we define the beacon interval as

$$T_{\mathrm{BI}}^{t,t+1} = T_{\mathrm{CFP}}^{t} + T_{\mathrm{SP}}^{t} + T_{\mathrm{CAP}}^{t+1},$$

where T_{CFP}^{t} is a CFP, T_{SP}^{t} is an inactive period, and T_{CAP}^{t} is a CAP at superframe t. Let $D_{i,j,t}$ denotes the expected delay of j *new requests* that observe a queue size of i waiting requests at superframe t. We distinguish the number of successful requests that obtain GTS out of j *new requests* for a different superframe. Note that j *new requests* of the superframe t will obtain GTSs from the superframe $t + \upsilon_i + 1$ to the superframe $t + \upsilon_i + \upsilon_j + 1$. The delay of allocating GTS is the sum of three components:

- Arrival delay of the CAP: We consider the arrival delay based on the FCFS policy of queue management. Note that that the arrival time of *new requests* is uniformly distributed in the CAP.
- Offset delay of *old* and *new requests*: The *new requests* need to wait a number of superframes before obtaining GTS since there are *old* and *new requests* arriving before the current request.
- Service delay of the CFP: If the *new requests* are able to obtain GTS at the current superframe, then the superframe slot size T_{SS} and the minimum number of superframe slots $\underline{\theta}$ need to be accounted for.

Hence, the following proposition holds:

Proposition 5. *Let q be the queue size, then the expected delay of j new requests arriving at superframe t is*

$$\mathbb{E}[D_q(t)] = \sum_{i=0}^{\Delta_u - 1} G(i,q) + \sum_{i=\Delta_u}^{q} G(i, q - i + \Delta_u) + G(q, \Delta_u)\frac{\pi_{\bar{q}}^{t}}{\pi_{q}^{t}}, \tag{4.20}$$

where

$$G(a,b) = \sum_{j=1}^{\bar{\lambda}} P_{a,j}^{t} \left(D_{a,j,t}\, u(b-j) + D_{a,b,t}\, u(j-(b+1)) \right),$$

with $P_{i,j}^{t} = \pi_i^t \lambda_j / \sum_{k=1}^{\bar{\lambda}} \lambda_k$, and $\bar{\lambda}$ is the maximum number of requests.

Proof. See Section C.3. □

We remark here that the average queueing delay mainly depends on the traffic pattern λ_i of the number of GTS requests and protocol parameters BO, SO of Δ_u.

4.6 Hybrid Markov Chain Model

We are now in the position to give the core contribution of the overall chapter. We propose a performance analysis of the hybrid MAC by connecting the two Markov chain models of CAP and CFP we have developed in Section 4.4 and Section 4.5. Then, we characterize the throughput of hybrid MAC by taking into account both the non time-critical data packet transmission of the CAP and the time-critical data packet transmission of the CFP.

4.6.1 Connection between CAP and CFP

In this section, we connect the two Markov chain models of the CAP of Section 4.4.1 and the CFP of Section 4.5.1. Recall that the number of GTS requests depends on the number of time-critical data packets that the nodes want to send to the coordinator. With contention-based transmissions during the CAP, there will be loss and delay of GTS requests due to the busy channel and collisions. The number of GTS requests affects the CAP length, i.e., increasing the number of GTS requests decreases the CAP length for a fixed-length superframe. Hence, the performance of the GTS allocation of the CFP depends significantly on the number of successfully received GTS requests by the coordinator during the CAP. The coupling between the two chains is given by the probability density function (PDF) of the number of GTS requests per superframe, which abstracts the CSMA/CA mechanism of the CAP. Such a PDF is the input to analyze the GTS allocation mechanism of the CFP. We first derive the average number of successfully received GTS requests for a given CAP length T_{CAP}. Then, we propose the approximated PDF of the number of GTS requests per superframe based on the analysis of the CAP.
Suppose that the arrival process of GTS requests is independent with the reliability of GTS requests R of the CAP given in Eq. (4.6). The average service time for a successfully received packet including the average packet generation time and packet delay is

$$\mathbb{E}[\psi_s] = \frac{1-\eta_t}{\eta_p} h + \mathbb{E}[D_s]\,,$$

where $\mathbb{E}[D_s]$ is the average delay for successfully received packets given in Eq. (4.10). By the same argument, the average service time for discarded packets due to channel access failure and retry limits are described by the following equations: $\mathbb{E}[\psi_{dc}] = (1-\eta_t)\,h/\eta_p + \mathbb{E}[D_{dc}]\,, \mathbb{E}[\psi_{dr}] = (1-\eta_t)\,h/\eta_p + \mathbb{E}[D_{dr}]\,$, where $\mathbb{E}[D_{dc}]$ and $\mathbb{E}[D_{dr}]$ are the average delay for discarded packets due to channel access failure and retry limits, respectively. To consider the limited CAP length T_{CAP}, we derive the feasible set of $(\mathbb{E}[\psi_s], \mathbb{E}[\psi_{dc}], \mathbb{E}[\psi_{dr}])$ which should satisfy the following constraint:

$$\mathscr{A} = \{\mathbf{a} | T_{\mathrm{CAP}} - \min(\boldsymbol{\psi}_c) < \boldsymbol{\psi}_c^T \mathbf{a} \leq T_{\mathrm{CAP}}, \mathbf{a} \in \mathbb{Z}_+^3\}\,,$$

where $\boldsymbol{\psi}_c = (\,\mathbb{E}[\psi_s],\, \mathbb{E}[\psi_{dc}],\, \mathbb{E}[\psi_{dr}]\,)^T$, $\mathbf{a} = (\,a_1,\, a_2,\, a_3\,)^T$, and $\mathbb{Z}_+^3$ is triples of non-negative integers. Let us assume that the feasible set is a sequence of independent,

identically distributed random variables which consists of V_{tot} trials, and each taking three possible outcomes, A_1, A_2, A_3. Notice that each possible outcome can occur with probability R given by Eq. (4.6), P_{dr} given by Eq. (4.7), and P_{dc} given by Eq. (4.8). Therefore, the probability mass function that A_1 occurs a_1 times, A_2 occurs a_2 times, and A_3 occurs a_3 times follows a multinomial distribution:

$$\Pr\left[\bigcap_{i=1}^{3} A_i = a_i\right] = \frac{\frac{V_{\text{tot}}!}{a_1!a_2!a_3!} R^{a_1} P_{dc}^{a_2} P_{dr}^{a_3}}{\sum\limits_{\mathbf{a}\in\mathscr{A}} \frac{V_{\text{tot}}!}{a_1!a_2!a_3!} R^{a_1} P_{dc}^{a_2} P_{dr}^{a_3}},$$

where $V_{\text{tot}} = \sum\limits_{i=1}^{3} a_i$. From the multinomial distribution, the probability that a successfully received GTS request occurs a_1 times is

$$\Pr[A_1 = a_1] = \frac{\sum\limits_{a_2,a_3\in\mathscr{A}} \frac{V_{\text{tot}}!}{a_1!a_2!a_3!} R^{a_1} P_{dc}^{a_2} P_{dr}^{a_3}}{\sum\limits_{\mathbf{a}\in\mathscr{A}} \frac{V_{\text{tot}}!}{a_1!a_2!a_3!} R^{a_1} P_{dc}^{a_2} P_{dr}^{a_3}}.$$

Note that if $\mathbb{E}[\psi_s] = \mathbb{E}[\psi_{dc}] = \mathbb{E}[\psi_{dr}]$, then it follows that the distribution is binomial. Therefore, the expected number of GTS requests for a given CAP length T_{CAP} is

$$\mathbb{E}[A_1] = \sum_{a_1=0}^{\left\lfloor \frac{T_{\text{CAP},k}}{\mathbb{E}[\psi_s]} \right\rfloor} a_1 \Pr[A_1 = a_1]. \tag{4.21}$$

Next, we derive the average number of successfully received non time-critical packet and GTS requests for a given CAP length for a hybrid MAC. Assume that the average number of successfully received packet is proportional to the superframe length. Let us denote the average number of successfully received packets N_{SD} including the non time-critical packet and GTS requests for a given superframe length T_{SD}. According to the standard [8], the CAP and CFP lengthes are updated via the following equations:

$$\begin{aligned} T_{\text{CFP}}^{t+1} &= \min\left(\frac{(1-\eta_d)N_{\text{SD}}}{T_{\text{SD}}} T_{\text{CAP}}^{t},\ \Delta_u\right) T_{\text{SS}}, \\ T_{\text{CAP}}^{t+1} &= T_{\text{SD}} - T_{\text{CFP}}^{t+1}, \end{aligned} \tag{4.22}$$

where $T_{\text{CFP}}^{0} = 0$ and $T_{\text{CAP}}^{0} = T_{\text{SD}}$, and where the index t denotes the discrete time in the superframe unit. Recall that the probability to generate a time-critical packet is $1 - \eta_d$. Then, an instantaneous average number of successfully received non time-critical packets N_{CAP}^{t} and GTS requests N_{CFP}^{t} for a given superframe length T_{SD}

are given by

$$N_{\mathrm{CAP}}^{t} = \frac{\eta_d N_{\mathrm{SD}} T_{\mathrm{CAP}}^{t}}{T_{\mathrm{SD}}}\,,$$
$$N_{\mathrm{CFP}}^{t} = \frac{(1-\eta_d) N_{\mathrm{SD}} T_{\mathrm{CAP}}^{t}}{T_{\mathrm{SD}}}\,. \tag{4.23}$$

Lemma 4. *The iteration* (4.22) *has the fixed point*

$$T_{CAP}^{\star} = \begin{cases} \frac{T_{SD}^2}{T_{SD}+(1-\eta_d)N_{SD}T_{SS}} & N_{SD} < \frac{N_{SS}}{1-\eta_d}\,, \\ T_{SD} - \Delta_u T_{SS} & N_{SD} \geq \frac{N_{SS}\Delta_u}{(1-\eta_d)(N_{SS}-\Delta_u)}\,, \end{cases}$$

and depending on N_{SD}, the portion of CAP and CFP lengthes follows one of three cases:

$$\begin{array}{ll} \text{convergence} & \text{if } N_{\mathrm{SD}} < \frac{N_{\mathrm{SS}}}{1-\eta_d}\,, \\ \text{saturation} & \text{if } N_{\mathrm{SD}} \geq \frac{N_{\mathrm{SS}}\Delta_u}{(1-\eta_d)(N_{\mathrm{SS}}-\Delta_u)}\,, \\ \text{oscillation} & \text{if } \frac{N_{\mathrm{SS}}}{1-\eta_d} \leq N_{\mathrm{SD}} < \frac{N_{\mathrm{SS}}\Delta_u}{(1-\eta_d)(N_{\mathrm{SS}}-\Delta_u)}\,. \end{array}$$

Proof. See Section C.4. □

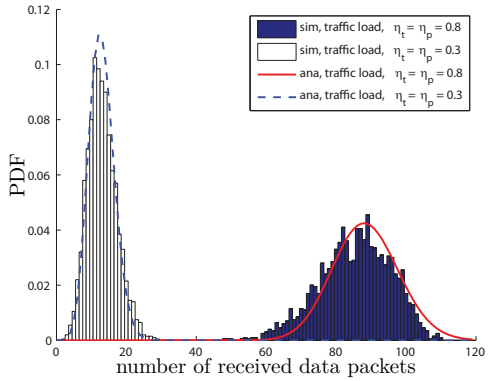
Proposition 6. *The average number of successfully received non time-critical packets N_{CAP} and GTS requests N_{CFP} for a given superframe length T_{SD} are given by*

$$N_{CAP} = \frac{\eta_d N_{SD} T_{CAP}^{\star}}{T_{SD}}\,, \tag{4.24}$$
$$N_{CFP} = \frac{(1-\eta_d) N_{SD} T_{CAP}^{\star}}{T_{SD}}\,. \tag{4.25}$$

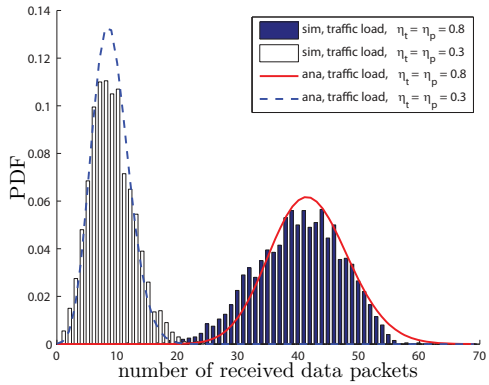
Proof. By following Lemma 4 and Eq. (4.23), we compute the average number of successfully received non time-critical packets and GTS requests. Note that the oscillatory mode predicted by Lemma 4 does not exist if $2\Delta_u \leq N_{\mathrm{SS}}$. The IEEE 802.15.4 MAC fulfills the convergence condition since $\Delta_u \leq N_{\mathrm{GTS}}$ and $N_{\mathrm{SS}} = 16$. □

4.6.2 PDF of the Number of Received Packets

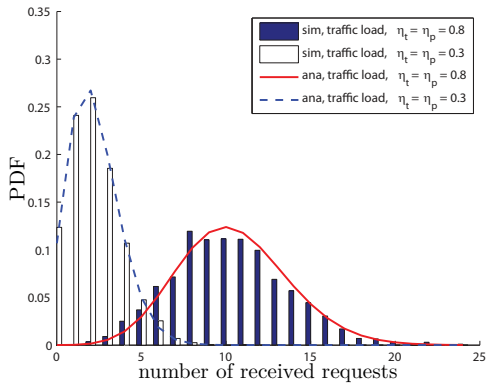
We are now in the position to propose the PDF of the number of GTS requests per superframe, which has the input vector to analyze the GTS allocation mechanism of the CFP in Section 4.5. The challenge for the derivation of the analytical model is the stochastic behavior of all nodes of the network, which makes it quite complicated and requires heavy computations to characterize the exact PDF of the number of GTS requests per superframe. In our earlier work [189], it is shown that



(a) SO $= 5, \eta_d = 1$



(b) SO $= 5, \eta_d = 0.8$



(c) SO $= 5, \eta_d = 0.8$

Figure 4.4: Probability density function of the number of received data packets and requests per superframe as a function of the different probabilities for generating a non time-critical data packet $\eta_d = 0.8, 1$, and traffic load $\eta_t = \eta_p = 0.3, 0.8$ with a given superframe order SO $= 5$, MAC parameters $m_0 = 3, m_b = 8, m = 4, n = 1$, the number of time-critical data packets $\varphi_n = 2$ for each GTS request, and the packet length $L_p = 7$.

the exponential distribution is a good approximation of the packet delay distribution. Since the interval of packet transmission is constant, the inter-arrival time of packet transmission can be approximated as exponentially distributed. Therefore, the Poisson distribution with the mean value given in Eq. (4.21) approximates the arrival process of GTS requests in the CAP.

We validate the PDF of the number of received packets per superframe by using the Poisson approximation we have proposed. Figure 4.4 shows the PDF of the number of received data packets and requests per superframe as obtained by Monte Carlo simulations and the Poisson distribution with the mean value given in Eqs. (4.24) and (4.25) as a function of the probabilities for generating non time-critical data packets with $\eta_d = 0.8, 1$, and traffic load with $\eta_t = \eta_p = 0.3, 0.8$ with the superframe order SO = 5, MAC parameters $m_0 = 3, m_b = 8, m = 4, n = 1$, number of time-critical data packets $\varphi_n = 2$ for each GTS request, and the packet length $L_p = 7$. Figure 4.4(a) reports the PDF of the number of received data packets for the CAP without having a GTS allocation mechanism. We observe that a Poisson distribution predicts well the PDF of the number of received data packets. The higher traffic load $\eta_t = \eta_p = 0.8$ gives a greater number of received data packets than lower traffic load $\eta_t = \eta_p = 0.3$. Figures 4.4(b) and 4.4(c) show the PDF of the number of data packets and requests per superframe for the probability of generating a non time-critical data packet $\eta_d = 0.8$. Note that the mean number of data packets for the traffic load $\eta_t = \eta_p = 0.3, 0.8$ is closer than the case of the probability for generating a non time-critical data packet $\eta_d = 1$. This is due to the decreasing CAP length caused by the GTS allocation based on the number of received requests. If the GTS allocation mechanism is activated with $\eta_d < 1$, then the CAP length decreases since the GTS allocation takes a portion of the superframe resource.

4.6.3 Throughput of Hybrid MAC

Here we characterize the throughput of the hybrid MAC, namely, the average amount of both non time-critical packets and time-critical packets that can be transmitted during the beacon interval of length T_{BI}. The normalized system throughput is defined as

$$\Theta = \frac{N_{\mathrm{CAP}} L_{p,d} + \min(N_{\mathrm{CFP}}, \Delta_u)\varphi_n L_{p,g}}{T_{\mathrm{BI}}}, \tag{4.26}$$

where an average number of successfully received non time-critical packets N_{CAP} and GTS requests N_{CFP} are given in Eqs. (4.24) and (4.25), respectively. The first and second terms of the nominator in Eq. (4.26) gives the average number of successful non time-critical packets of the CAP and time-critical packets of the CFP in a slot unit, respectively. The normalized system throughput Θ depends on the traffic pattern since an average number of successfully received non time-critical packets N_{CAP} and GTS requests N_{CAP} are related to the number of data packets φ_n, the frame size L_p, the mean and variance of requests.

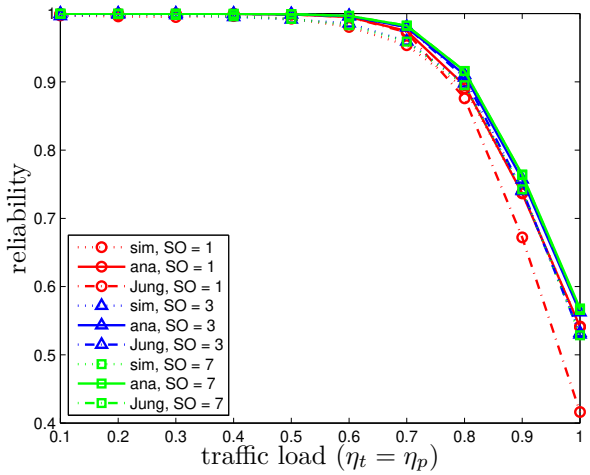
4.7 Numerical Results

Here we present extensive Monte Carlo simulations of the hybrid IEEE 802.15.4 MAC to validate our analytical expressions in terms of the reliability given in Eq. (4.6), average packet delay given in Eq. (4.10), average queueing delay given in Eq. (4.20), and throughput given in Eq. (4.26). The simulations are based on the specifications of the IEEE 802.15.4 [8], and includes several values of the traffic load, the probability for generating a non time-critical packet, and superframe length. Note that we fix the beacon order BO = 10 and MAC parameters $m_0 = 3, m_b = 8, m = 4, n = 1$ in the simulations.

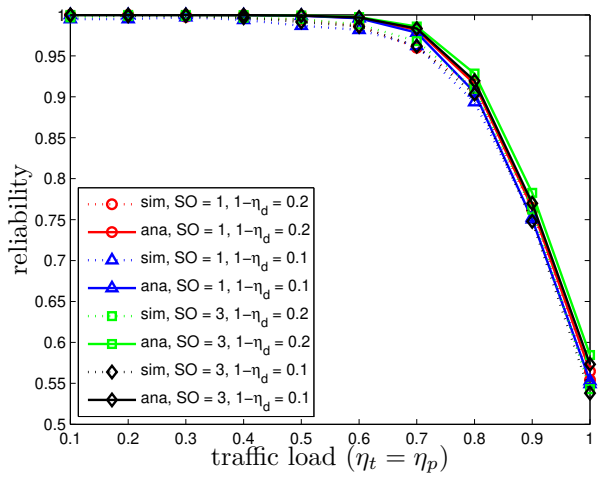
4.7.1 Reliability

Figure 4.5(a) compares the reliability of the CAP without having GTS allocations $\eta_d = 1$ of our proposed model, the analytical model derived from [171], and Monte Carlo simulations as a function of traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$ and superframe order SO $= 1, 3, 7$ with the packet length $L_p = 7$. In the figure, note that "Jung" refers to the reliability model derived from the Markov chain model in [171]. The analytical model of the reliability as obtained by Eq. (4.6) follows well the simulation results. We observe that the smaller SO the worse is the reliability. The reason is that a small SO increases the event of deferred attempts due to the lack of remaining slot of the CAP, and then it increases the contention at the beginning of superframe. We remark that the effect of SO on the reliability of the CAP is negligible due to the extra backoff mechanism for the event of deferred attempts of the standard. The reliability model in [171] is not accurate when the traffic load increases and SO decreases. In [169]–[171], authors observe a high contention and high packet loss at the beginning of superframes since they do not consider the extra backoff mechanism for the event of deferred attempts which is explicitly described in the standard [8]. We remark that the number of deferred attempts increases as the traffic load increases and SO decreases. Therefore, the extra backoff mechanism for the event of deferred attempts at the beginning of superframe significantly improves the reliability of the CAP.

Figure 4.5(b) shows the reliability of hybrid MAC with the CFP $\eta_d < 1$ as a function of traffic load, the probabilities for generating a time-critical data packet $1 - \eta_d = 0.1, 0.2$, and superframe order SO $= 1, 3$ with the number of time-critical data packets $\varphi_n = 2$ for each GTS request and the packet length $L_p = 7$. The analytical model predicts well the simulation results. Since the CFP length increases as the probability for generating time-critical packet η_d increases, the CAP length decreases for a fixed superframe order. As we observe in Figure 4.5(a), the reliability decreases as the CAP length decreases due to the deferred attempts of packet transmission. Although the CAP length is even more decreasing as the probability for generating a time-critical packet $1 - \eta_d$ increases, the effect of different probabilities $1 - \eta_d$ is negligible.



(a) Superframe order SO $= 1, 3, 7$ with a given probability for generating a non time-critical data packet $\eta_d = 1$ and the packet length $L_p = 7$.



(b) Superframe order SO $= 1, 3$ and the probabilities for generating a time critical data packet $1 - \eta_d = 0.1, 0.2$, with the number of time-critical data packets $\varphi_n = 2$ for each GTS request and the packet length $L_p = 7$.

Figure 4.5: Reliability as a function of traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$. Note that "Jung" refers to the reliability expression derived from Markov chain model [171].

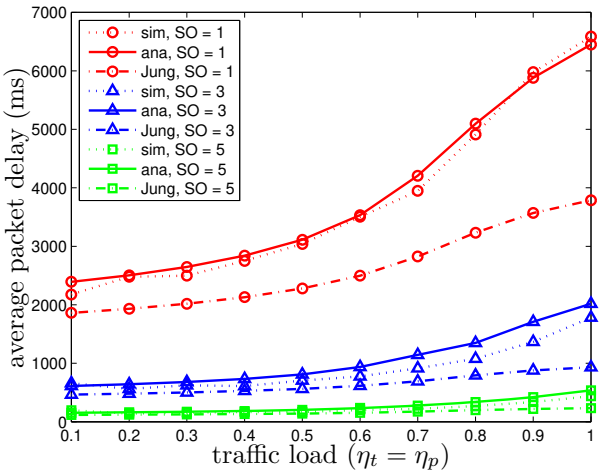
4.7.2 Delay

In this section, we study the average packet delay of the CAP and the average queueing delay of the GTS allocation mechanism of the CFP.

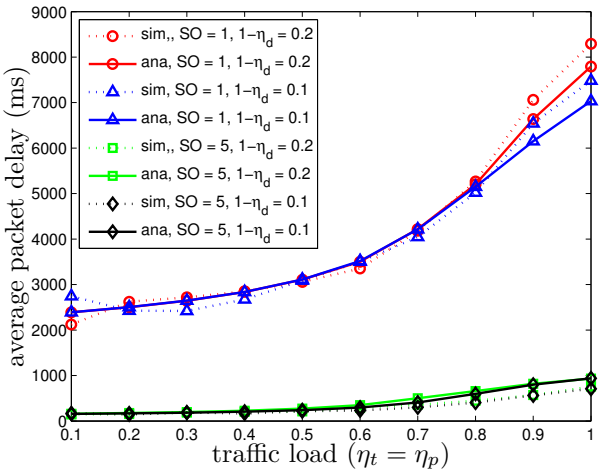
Figure 4.6(a) compares the average packet delay of data packets of the pure CAP without having GTS allocations $\eta_d = 1$ given in Eq. (4.10), the analytical model in [171], and the Monte Carlo simulations as a function of traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$ and superframe order SO $= 1, 3, 5$ with the packet length $L_p = 7$. Since in [169]–[171] there is not direct investigation of the average packet delay with superframe structure, we derive it from the Markov chain model in [171]. Our analytical model of the average packet delay follows well the simulations results. The average packet delay increases as the traffic load increases due to the high contention. Furthermore, the average packet delay increases as the parameter SO decreases. This is due to that as the parameter SO decreases, the inactive period of the fixed-length superframe $BO = 10$ increases. Note that if BO = SO, then the behavior of the hybrid MAC is very similar to CSMA/CA algorithm of the CAP, see details in [189]. The average packet delay derived from [171] does not predict well the simulation results. The main reason is that [169]–[171] do not explicitly consider the deferred attempt of the backoff state. We remark that the deferred attempt of the backoff state becomes a critical factor as decreasing the parameter SO because more packets are delayed for the inactive period before actual packet transmission. Furthermore, the backoff mechanism for the deferred attempt is not considered accurately as we discussed in Section 4.7.1.

Figure 4.6(b) shows the average packet delay of the hybrid MAC with the CFP as a function of traffic load, the probabilities for generating a time-critical data packet $1 - \eta_d = 0.1, 0.2$, and superframe order SO $= 1, 5$ with the number of time-critical data packets $\varphi_n = 2$ for each GTS request, and the packet length $L_p = 7$. We observe a similar behavior of the average packet delay of the hybrid MAC as the delay of the pure CAP reported in Figure 4.6(a). The average packet delay increases as the probability for generating a non time-critical data packet increases. This is due to that as the GTS requests increases, the portion of the CFP increases for the fixed-length superframe. Therefore, the deferred packet delay increases since a node waits more time for the inactive period and the CFP. By a similar argument, we see that the average packet delay of hybrid MAC in Figure 4.6(b) is higher than the delay of the pure CAP in Figure 4.6(a). Note the scale of the axis.

Next, we discuss the queueing delay of the GTS allocation mechanism. Figure 4.7 shows the average queueing delay of the CFP as a function of traffic load, the probabilities for generating a time-critical data packet $1 - \eta_d = 0.1, 0.2$, and superframe order SO $= 5, 7, 9$ with the number of time-critical data packets $\varphi_n = 2$ for each GTS request and the packet length $L_p = 7$. The analytical model of the queueing delay given by Eq. (4.20) predicts well the simulation results. As the traffic load $\eta_t = \eta_p$ and the probability for generating a time-critical data packet $1 - \eta_d$ increase, the queueing delay increases due to a higher number of received GTS requests at the coordinator. By a similar argument, as the parameter SO increases, the queueing



(a) Superframe order SO $= 1, 3, 5$ with a given probability for generating a non time-critical data packet $\eta_d = 1$ and the packet length $L_p = 7$.



(b) Superframe order SO $= 1, 5$ and the probabilities for generating a time-critical data packet $1 - \eta_d = 0.1, 0.2$, with the number of time-critical data packets $\varphi_n = 2$ for each GTS request and the packet length $L_p = 7$.

Figure 4.6: Average packet delay as a function of traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$.

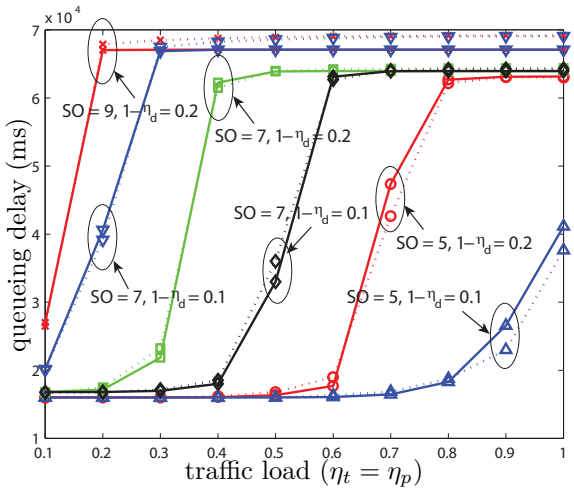


Figure 4.7: Average queueing delay of GTS request as a function of traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$, the probabilities for generating a time-critical data packet $1 - \eta_d = 0.1, 0.2$, and superframe order SO $= 5, 7, 9$ with the number of time-critical data packets $\varphi_n = 2$ for each GTS request and a packet length $L_p = 7$.

delay increases due to a higher number of received GTS requests. Observe that the queueing delay is saturated when the number of received GTS requests increases for higher traffic load and higher superframe order because of the maximum number of GTSs to be allocated to nodes, see details in Section 4.5.1.

4.7.3 Throughput

Figure 4.8 compares the throughput of the pure CAP without having GTS allocation given in Eq. (4.26), the throughput given in [171], and Monte Carlo simulations as a function of traffic load and superframe order SO $= 3, 5, 7, 9$ with the packet length $L_p = 7$. The analytical model of throughput predicts well the simulation results. The throughput of the CAP increases as the traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$ increases. Furthermore, as the parameter SO increases, the throughput of the CAP increases due to the longer CAP length. As the traffic load decreases and SO increases, the throughput in [171] is very close to the throughput given in Eq. (4.26) because the number of deferred attempts decreases.
Figure 4.9 shows the throughput of hybrid MAC as a function of traffic load, the probabilities for generating a time-critical data packet $1 - \eta_d = 0.1, 0.2$, and superframe order SO $= 3, 5, 7$ with the number of time-critical data packets $\varphi_n = 2$ for each GTS request and a packet length $L_p = 7$. Figure 4.9(a) presents the hybrid throughput, namely the sum of throughput of the CAP in Figure 4.9(b) and the CFP in Figure 4.9(c). We observe a similar trend to the throughput of the CAP as in Figure 4.8. The throughput of the CAP and CFP increases as the traffic

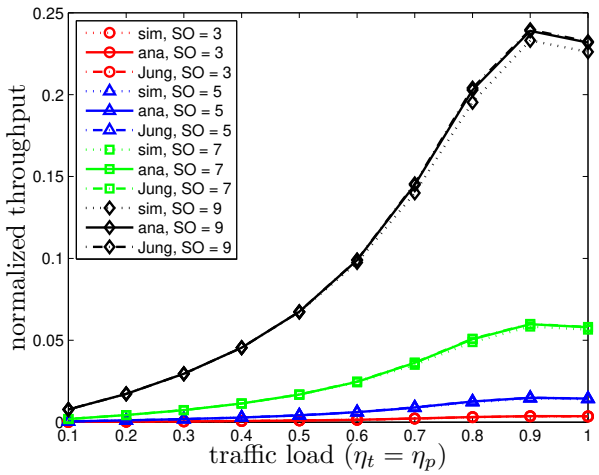
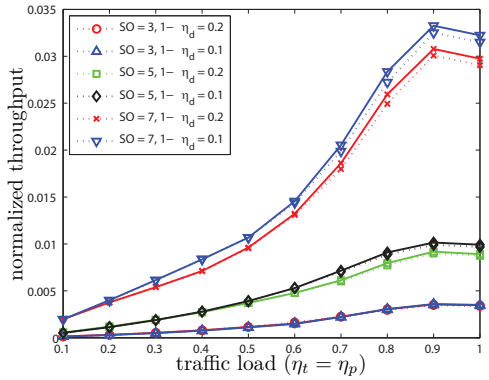


Figure 4.8: Throughput as a function of traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$ and superframe order SO $= 3, 5, 7, 9$ with a probability for generating a non time-critical data packet $\eta_d = 1$ and the packet length $L_p = 7$.

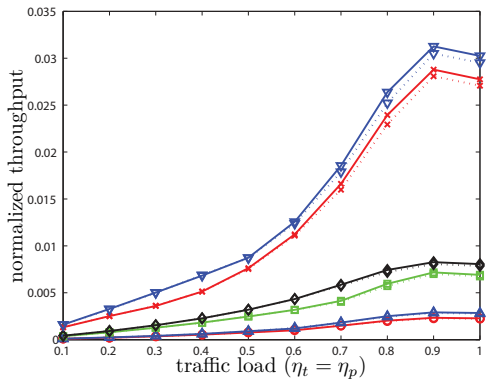
load $\eta_t = \eta_p$ and SO increase. It is interesting to observe that as the probability for generating a time-critical data packet $1 - \eta_d$ increases, the throughput of CFP increases, however the throughput of the CAP decreases. Hence, a tradeoff exists between throughput of the CAP and CFP for a fixed-length superframe. In our setup, the throughput of the CAP is more dominant for the hybrid throughput of the network. Main reason is that the throughput of the CFP is strictly limited by the maximum number of GTSs. For the lower parameter SO $= 3, 5$, the hybrid throughput of different probabilities for generating a time-critical data packets is closer with respect to the throughput of the CAP because the throughput of the CFP compensates the lower throughput of the CAP. Therefore, the throughput of the hybrid MAC depends on the probability for generating time-critical packets.

4.8 Summary

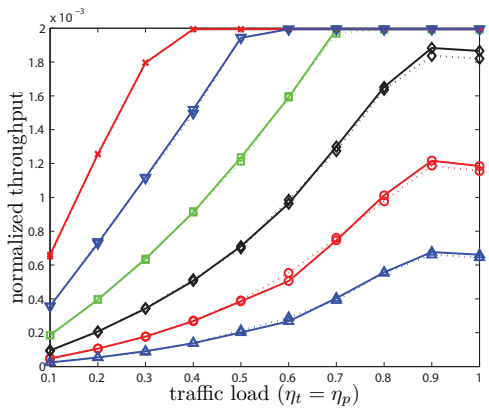
In this chapter, we developed an analytical framework for modeling the behavior of the MAC protocol of the IEEE 802.15.4 standard in the hybrid modality. Since the hybrid MAC provides an access scheme for the CAP and an access scheme for the CFP in each superframe, the analysis of the mutual effects of these two schemes is the fundamental step to understand the performance of the IEEE 802.15.4 MAC. We first model the contention access scheme of the IEEE 802.15.4 exponential backoff process through a Markov chain that included retry limits, ACKs, unsaturated traffic, and superframe period. Our model explicitly considers the deferred attempt of the backoff state and the extra backoff mechanism for the event of deferred at-



(a) Hybrid throughput



(b) CAP throughput



(c) CFP throughput

Figure 4.9: Throughput as a function of traffic load $\eta_t = \eta_p = 0.1, \ldots, 1$, the probabilities for generating a time-critical data packet $1 - \eta_d = 0.1, 0.2$, and superframe order SO $= 3, 5, 7$ with the number of time-critical data packets $\varphi_n = 2$ for each GTS request and the packet length $L_p = 7$.

tempts with respect to the previous literature, which are critical for the reliability and delay analysis. Then, we analyzed the behavior of contention-free scheme using a second Markov chain. To connect the two access modalities, the probability distribution function of the number of received GTS requests per superframe was used as the input parameter to analyze the performance of the CFP.
We evaluated the network performance in terms of reliability of the CAP, average packet delay of the CAP, queueing delay of the CFP, and throughput of the hybrid MAC. In addition, we showed that a Poisson distribution approximates well the number of successfully received packets for a given CAP length. Monte Carlo simulations confirmed that the proposed analysis offers a satisfactory accuracy.

Chapter 5

Breath: an Adaptive Protocol for Industrial Control Applications using Wireless Sensor Networks

An energy-efficient, reliable and timely data transmission is essential for WSNs employed in scenarios where plant information must be available for control applications. To reach a maximum efficiency, cross-layer interaction is a major design paradigm to exploit the complex interaction among the layers of the protocol stack. This is challenging because latency, reliability, and energy are at odds, and resource constrained nodes support only simple algorithms. In this chapter, the novel protocol Breath[1] is proposed for control applications. Breath is designed for WSNs where nodes attached to plants must transmit information via multi-hop routing to a sink. Breath ensures a desired packet delivery and delay probabilities while minimizing the energy consumption of the network. The protocol is based on randomized routing, medium access control, and duty-cycling jointly optimized for energy efficiency. The design approach relies on a constrained optimization problem, whereby the objective function is the energy consumption and the constraints are the packet reliability and delay. The challenging part is the modeling of the interactions among the layers by simple expressions of adequate accuracy, which are then used for the optimization by in-network processing. The optimal working point of the protocol is achieved by a simple algorithm, which adapts to traffic variations and channel conditions with negligible overhead. The protocol has been implemented and experimentally evaluated on a test-bed with off-the-shelf wireless sensor nodes, and it has been compared with a standard IEEE 802.15.4 solution. Analytical and experimental results show that Breath is tunable and meets reliability and delay requirements. Breath exhibits a nearly uniform distribution of the working load, thus extending network lifetime. Therefore, Breath is a good candidate for efficient, reliable, and

[1]Since the protocol adapts to the network variations by enlarging or shrinking next-hop distance, sleep time of the nodes, and transmit radio power, we think that it behaves like a breathing organism.

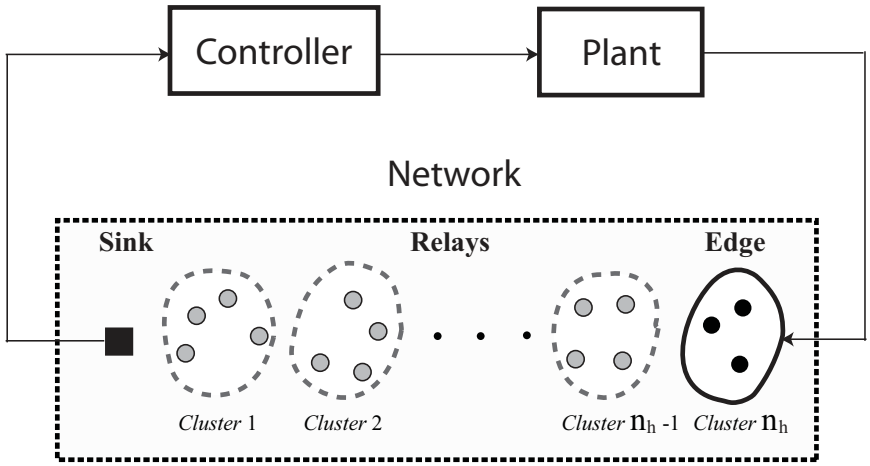


Figure 5.1: Wireless control loop. An wireless network closes the loop from sensors to controller. The network includes nodes (black dots) attached to the plant, $n_h - 1$ relay clusters (grey dots), and a sink (black rectangular) attached to the controller.

timely data gathering for control applications.
The rest of the chapter is organized as follows: In Section 5.1 we define the system scenario. In Section 5.2, we introduce Breath in detail. In Section 5.3 an optimization problem is posed to optimize the protocol, whereas in Section 5.4 the constraints and cost function of the protocol are modelled. In Section 5.5, we derive the optimal solution and in Section 5.6 we present an adaptive algorithm to obtain the working point of the protocol. The fundamental working limits of Breath are given in Section 5.7. An experimental implementation of the protocol is presented in Section 5.8. Finally, in Section 5.9 summarizing remarks and future perspectives are given.

5.1 System Scenario

We consider the scenario depicted in Figure 5.1, where a plant is remotely controlled over a WSN [23, 25]. Outputs of the plant are sampled at periodic intervals by the sensors with total packet generation rate of η_s pckt/s (see Table A.1 for main symbols used in this chapter). We assume that packets associated to the state of the plant are transmitted to a sink, which is connected to the controller, over a multi-hop network of uniformly and randomly distributed relaying nodes. No direct communication is possible between the plant and the sink. Relay nodes forward incoming packets. When the controller receives the measurements, they are used in a control algorithm to compensate the control output. The control law induces constraints on the communication delay and the packet loss probability. Packets must reach the sink within some minimum reliability and maximum delay. These

boundaries are denoted as *application requirements* throughout this chapter. The application requirements are chosen by the control algorithm designers. Since they can change from one control algorithm to another, or a control algorithm can ask to change the application requirements from time to time, we allow them to vary. We assume that nodes of the network cannot be recharged, so the operations must conserve energy. The system scenario is quite general, because it applies to any interconnection of a plant by a multi-hop WSN to a controller tolerating a certain degree of data loss and delay [25, 24].
A typical example of the scenario described above is an industrial control application. In particular, a WSN with nodes uniformly distributed in the walls or in the ceiling can be deployed as the network infrastructure that support the control of the state of the robots in a manufacturing cell. Typically, a cell is a stage of an automation line. Its physical dimensions range around 10 or 20 meters on each side. Several robots cooperate in the cell to manipulate and transform the same production piece. The typical way of monitoring the state of a robot is to observe its vibration pattern of the different parts of a robot. If the values of these vibrations are above a given threshold, a controller send the control message to these robots. Hence, each node senses vibrations and has to report the data to the controller within a delay. The decision making algorithm runs on the controller, which is usually a processor placed outside the cell. Multi-hop communication is needed to overcome the deep attenuations of the wireless channel due to moving metal objects and save energy consumption.

5.2 The Breath Protocol

The Breath protocol groups all N nodes between the cluster of nodes attached to the plant and the sink with $n_h - 1$ relay clusters. Data packets can be transmitted only from a cluster to the next cluster closer to the sink. Clustered network topology is supported in networks that require energy efficiency, since transmitting data through relays consumes less energy than routing directly to the sink [77]. In [207], a dynamic clustering method adapts the network parameters. In [77] and [81], a cluster header is selected based on the residual energy levels for clustered environments. However, the periodic selection of clustering may not be energy-efficient, and does not ensure the flexibility of the network to a time-varying wireless channel environment. A simpler geographic clustering is instead used in Breath. Nodes in the forwarding region send short beacon messages when they are available to receive data packets. Beacon messages are exploited to carry information related to the control parameters of the protocol. When a node receives a beacon message with the updated number of clusters $n_h - 1$, then the node adapts to its cluster based on a rough knowledge of its location.
In the following sections, we will describe the protocol stack and state machine of Breath in Section 5.2.1 and 5.2.2, respectively.

5.2.1 The Breath Protocol Stack

Breath uses a randomized routing, a CSMA/CA mechanism at the MAC, radio power control at the physical layer, and sleeping disciplines. We give details in the following.

In many industrial environments, the wireless conditions vary heavily because of moving metal obstacles and other radio disturbances. In such situations, routing schemes that use fixed routing tables are not able to provide the flexibility over mobile equipments, physical design limitations and reconfiguration typical of an industrial control application. Fixed routing is inefficient in WSNs due to the cost of building and maintaining routing tables. To overcome this limitation, routing through a random sequence of hops has been introduced in [85]. The Breath protocol is built on an optimized random routing, where next hop route is efficiently selected at random. Randomized routing allows us to reduce overhead because no node coordination or routing state needs to be maintained by the network. Robustness to node failures is also considerably increased by randomized routing. Therefore, nodes route data packets to next-hop nodes randomly selected in a forwarding region.

Each node, either transmitter or receiver, does not stay in an active state all time, but goes to sleep for a random amount of time, which depends on the traffic and channel conditions. Since traffic, wireless channel, and network topology may be time-varying, the Breath protocol uses a randomized duty-cycling algorithm. Sleep disciplines turn off a node whenever its presence is not required for the correct operation of the network. SPAN [57], SMAC [64], and GAF [83] focus on controlling the effective network topology by selecting a connected set of nodes to be active and turning off the rest of the nodes. These approaches require extra communication, since nodes maintain partial knowledge of the state of their individual neighbors. In Breath, each node goes to sleep for an amount of time that is a random variable dependent on traffic and network conditions. Let μ_c be the cumulative wake-up rate of each cluster, i.e., the sum of the wake-up rates that a node sees from all nodes of the next cluster. The cumulative wake-up rate of each cluster must be the same for each cluster to avoid congestions and bottlenecks.

The MAC of Breath is based on a CSMA/CA mechanism similar to IEEE 802.15.4. Both data packets and beacon packets are transmitted using the same MAC. Specifically, the CSMA/CA checks the channel activity by performing CCA before the transmission can commence. Each node maintains a variable NB for each transmission attempt, which is initialized to 0 and counts the number of additional backoffs the algorithm does while attempting the current transmission of a packet. Each backoff unit has duration T_b ms. Before performing CCAs a node takes a backoff of random$(0, W-1)$ backoff units i.e., a random number of backoffs with uniformly distributed over $0, 1, \ldots, W-1$. If the CCA fails, i.e., the channel is busy, NB is increased by one and the transmission is delayed of random$(0, W-1)$ backoff periods. This operation is repeated at most m times, after which a packet is discarded.

The Breath protocol assumes that each node has a rough knowledge of its loca-

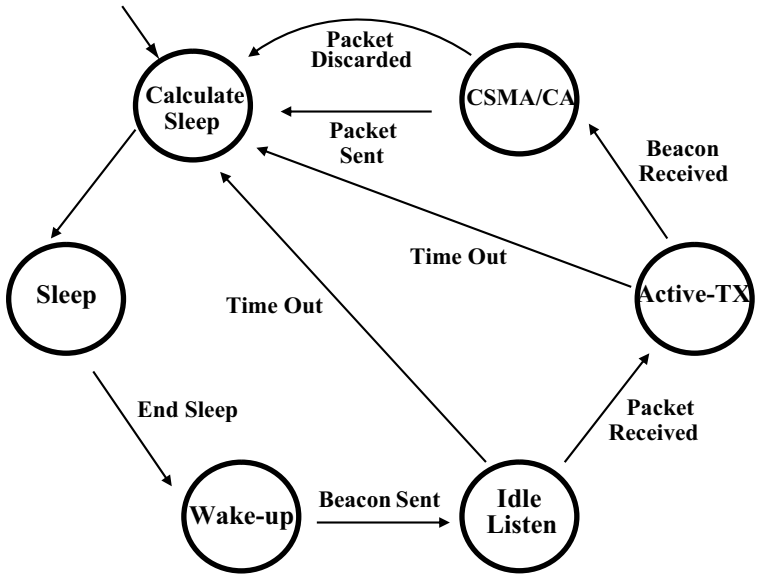


Figure 5.2: State machine description of a relay node executing the Breath protocol.

tion. This information, which is commonly required for the applications we are targeting [21], can be obtained running a coarse positioning algorithm, or using the received signal strength indicator (RSSI), which is typically provided by off-the-shelf sensor nodes [136]. Some radio chips already provide a location engine based on RSSI [208]. Location information is needed for tuning the transmit radio power and to change the number of hops, as we will see later. The energy spent for radio transmission plays an important role in the energy budget and for the interference in the network. Breath, therefore, includes an effective radio power control algorithm.

5.2.2 State Machine Description

Breath distinguishes between three node classes: edge nodes, relays, and the sink. The edge nodes wake-up as soon as they sense packets generated by the plant to be controlled. Before sending packets, the edge node waits for a beacon message from the cluster of nodes closer to the edge. Upon the reception of a beacon, the node sends the packet.
Consider a relay node k. Its detailed behavior is illustrated by the state machine of Figure 5.2, as we describe in the following:

- **Calculate Sleep State**: the node calculates the parameter μ_k for the next sleeping time and generates an exponentially distributed random variable having average $1/\mu_k$. After this the node goes back to the Sleep State. μ_k is computed such that the cumulative wake-up rate of the cluster μ_c is ensured.

- **Sleep State**: the node turns off its radio and starts a timer whose duration is an exponentially distributed random variable with average $1/\mu_k$. When the timer expires, the node goes to the Wake-up State.

- **Wake-up State**: the node turns its beacon channel on, and broadcasts a beacon indicating its location. Then, it switches to listen to the data channel, and it goes to the Idle Listen State.

- **Idle Listen State**: the node starts a timer of a fixed duration that must be long enough to receive a packet. If a data packet is received, the timer is discarded, the node goes to the Active-TX State, and its radio is switched from the data channel to the beacon channel. If the timer expires before any data packet is received, the node goes to the Calculate Sleep State.

- **Active-TX State**: the node starts a waiting timer of a fixed duration. If the node receives the first beacon coming from a node in the forwarding region within the waiting time, it retrieves the node ID and goes to the CSMA/CA State. Otherwise if the waiting timer is expired before receiving a beacon, the node goes to the Calculate Sleep State.

- **CSMA/CA State**: the node switches its radio to hear the data channel, and it tries to send a data packet to a node in the next cluster by the CSMA/CA MAC. If the channel is not clean within the maximum number of tries, the node discards the data packet and goes to the Calculate Sleep State. If the channel is clear within the maximum number of attempts, the node transmits the data packet using an appropriate level of radio power and goes to the Calculate Sleep State.

The sink node sends periodically beacon messages to the last cluster of the network to receive data packets. Such a node estimates periodically the traffic rate and the wireless channel conditions. By using this information, the sink runs an algorithm to optimize the protocol parameters, as we describe in Section 5.3. Once the results of the optimization are achieved, they are communicated to the nodes by beacons. According to the protocol given above, the packet delivery depends on the traffic rate, the channel conditions, number of forwarding regions, and the cumulative wake-up time. In the next sections, we show how to model and optimize online these parameters.

5.3 Protocol Optimization

The protocol is optimized dynamically by a constrained optimization problem. The objective function, denoted by $E_{\mathrm{n,tot}}(n_h, \mu_c)$, is the total energy consumption for transmitting and receiving packets from the edge cluster to the sink. The constraint

are given by the end-to-end packet reception probability and end-to-end delay probability. The optimization problem is

$$\min_{n_h,\mu_c} \quad E_{\mathrm{n,tot}}(n_h,\mu_c) \tag{5.1a}$$

$$\text{s.t.} \quad R(n_h,\mu_c) \geq R_{\min}\,, \tag{5.1b}$$

$$\Pr[D(n_h,\mu_c) \leq D_{\max}] \geq \Omega\,, \tag{5.1c}$$

$$n_h \geq 2\,, \tag{5.1d}$$

$$\mu_{\min} \leq \mu_c \leq \mu_{\max}\,. \tag{5.1e}$$

The decision variables are the cumulative wake-up rate μ_c of each cluster and the number of relay clusters, $n_h - 1$. $R(n_h,\mu_c)$ is the reliability from the edge cluster to the sink, and $R_{\min}$ is the minimum desired probability. $D(n_h,\mu_c)$ is a random variable describing the delay to transmit a packet from the edge cluster to the sink. $D_{\max}$ is the desired maximum delay, and Ω is the minimum probability with which such a maximum delay should be achieved. Constraint (5.1d) is due to that there is at least two hops from the edge cluster to the sink. Constraint (5.1e) is due to that the wake-up rate cannot be less than a minimum value $\mu_{\min}$, and larger than a maximum value $\mu_{\max}$ due to hardware reasons. Note that Problem (5.1) is a mixed integer-real optimization problem, because μ_c is real and n_h is integer. We need to have Ω and $R_{\min}$ close to one. We let $\Omega \geq 0.95$ and $R_{\min} \geq 0.9$, namely we assume that the delay $D_{\max}$ must be achieved at least with a probability of 95%, and the reliability must be larger than 90%. We remark that $D_{\max}$, Ω, and $R_{\min}$ are application requirements, and n_h, μ_c and nodes' radio transmit power are *protocol parameters* that must be adapted to the traffic rate η_s, the wireless channel conditions, and the application requirements for an efficient network operation.
In the following we shall propose an approach to model the quantities of Problem (5.1), along with a strategy to achieve the optimal solution, namely the values of n_h^* and μ_c^* that minimize the cost function and satisfy the application requirements. As we will see later, the system complexity prevents us to derive the exact expressions for the analytical relations of the optimization problem. An approximation of the requirements and an upper bound of the energy consumption will be used.

5.4 Modeling of the Protocol

In this section, we model the reliability, packet delay distribution and total energy consumption of the network.

5.4.1 Reliability Constraint

In this subsection, we provide an analytical expression for the reliability constraint (5.1b) in Problem (5.1).

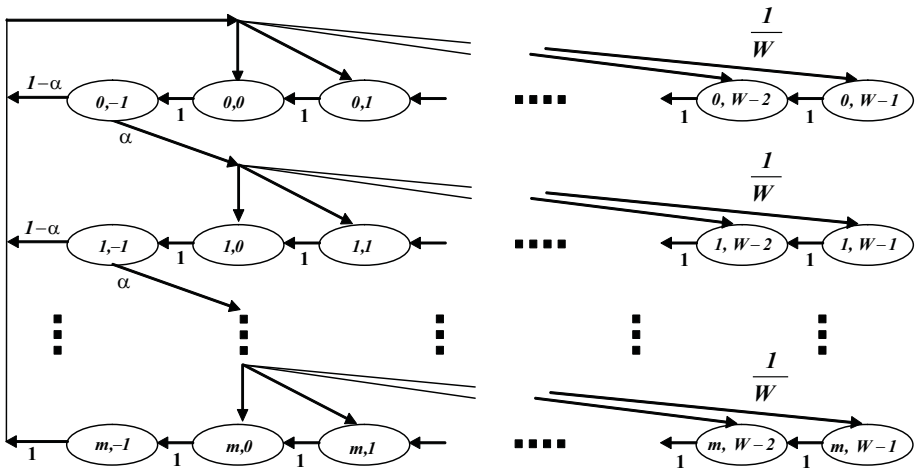


Figure 5.3: Markov chain model for CSMA/CA state evolution of Breath.

A data packet can be lost at a hop because of a bad wireless channel or packet collisions. The collision probability is determined by the CSMA/CA MAC. Therefore, to analyze such a behavior, we use a Markov chain. The approach is similar to the one proposed in [159] and [209] (see also [210, 211, 170]). Let m be the maximum backoff stage, and W be the maximum backoff time of CSMA/CA. Let $s(t) \in (0, \ldots, m)$ and $b(t) \in (0, \ldots, W-1)$ be the stochastic processes representing the backoff stage and the backoff time counter, respectively. The delay spent before a node senses the channel idle is modelled by the Markov chain depicted in Figure 5.3. The Markov chain state is $(s(t), b(t))$, where $b(t) = -1$ refers to the assessment of the channel state during CCA. Denote the Markov chain's steady-state probabilities by $b_{i,k} = \Pr((s(t), b(t)) = (i, k))$. They allow us to compute the probability of successful transmission in CSMA/CA as the probability that exactly one node transmits and $N_i - 1$ are silent:

$$\psi_{\mathrm{sc}}(N_i) = \frac{\tau(1-\tau)^{N_i-1}}{1-(1-\tau)^{N_i}}\,,$$

where

$$\tau = \sum_{i=0}^{m} b_{i,0} = \frac{2}{W+3}\,.$$

From the Markov chain, we derive also the busy channel probability $\alpha(N_i)$, which is

$$\alpha(N_i) = \frac{1-(1-\tau)^{N_i-1}}{2-(1-\tau)^{N_i-1}}\,. \tag{5.2}$$

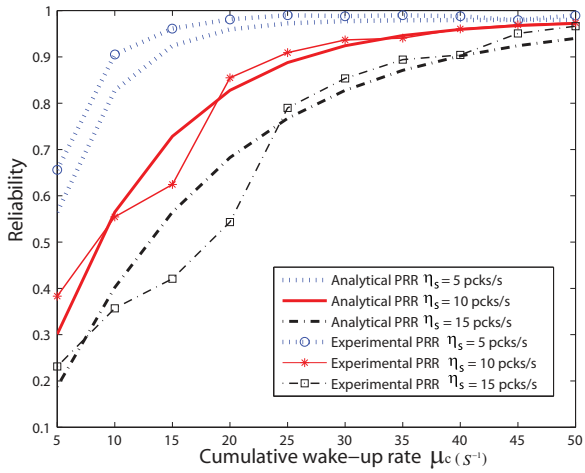


Figure 5.4: Reliability as obtained by Eq. (5.4) and experimental results as a function of μ_c. Curves refer to traffic rates $\eta_s = 5, 10, 15$ pckt/s for $n_h = 2$ hops and $N = 15$ nodes.

We will use this probability in Section 5.4.3.
The probability of successful transmission in CSMA/CA $\psi_{\rm sc}(N_i)$ depends on the number of nodes N_i that are contending to transmit packets. We need therefore to compute the probability $\psi_{\rm sb}(\mu_c, N_i)$ that a generic number N_i of contending nodes compete within a period $1/\eta_s$ to transmit a data packet. By recalling that the cumulative wake-up rate is exponentially distributed random variable with intensity μ_c, and noting that $e^{-\mu_c N_i/\eta_s}$ is the probability to have more than N_i contending nodes, we conclude that

$$\psi_{\rm sb}(\mu_c, N_i) = e^{-\mu_c(N_i-1)/\eta_s} - e^{-\mu_c N_i/\eta_s} . \tag{5.3}$$

Hence, the reliability with $\psi_{\rm sb}(\mu_c, j)$ and $\psi_{\rm sc}(j)$ is

$$R(n_h, \mu_c) = \prod_{i=1}^{n_h} p_i \sum_{j=1}^{\infty} \psi_{\rm sb}(\mu_c, j)\psi_{\rm sc}(j) , \tag{5.4}$$

where p_i denotes the probability of successful packet reception during a single-hop transmission from cluster i to cluster $i-1$.
Since the components of the sum in (5.4) with $N_i \geq 2$ give a small contribution, we set $N_i = 2$ and validate (5.4) by experimental results. Figure 5.4 reports the reliability vs μ_c, as obtained by Eq. (5.4) with $N_i = 2$ and experiments for a two-hop network. We see that (5.4) provides a good approximation of the experimental results because it is always around 5% of the experiments for reliability values of practical interest (larger than 0.7). The same behavior is found for n_h up to 4.

We can rewrite the reliability constraint $R(n_h, \mu_c) \geq R_{\min}$ by using (5.4) with $N_i = 2$, thus obtaining

$$\mu_c \geq f_r(n_h, R_{\min}) \triangleq \eta_s \ln(2C_r) - \eta_s \ln \left[C_r - 1 + \sqrt{(C_r - 1)^2 - 4C_r \left(R_{\min}^{1/n_h} / p_{\min} - 1 \right)} \right], \tag{5.5}$$

where $C_r = \tau(1-\tau)/(1-(1-\tau)^2)$, and $p_{\min} = \min(p_1, \ldots, p_{n_h})$. Note that we used the worst channel condition of the network $p_{\min}$, which is acceptable for optimization purpose because in doing so we consider the minimum of (5.4). Since the argument of the square root in (5.5) must be positive, an additional constraint is introduced:

$$n_h \leq n_{h,r} \triangleq \frac{\ln(R_{\min})}{\ln(p_{\min})}. \tag{5.6}$$

This constraint is a function of the minimum desired reliability requirement $R_{\min}$ and the worst channel condition $p_{\min}$. As the reliability requirement becomes stricter or the worst channel condition $p_{\min}$ decreases, the constraint $n_{h,r}$ decreases. Note that the logarithm function gives a negative numerical value when $0 \leq R_{\min} \leq 1$ and $0 \leq p_{\min} \leq 1$. We will use (5.5) and (5.6) in Section 5.5 to find the solution of Problem (5.1). Now, we turn our attention to the delay constraint.

5.4.2 Delay Constraint

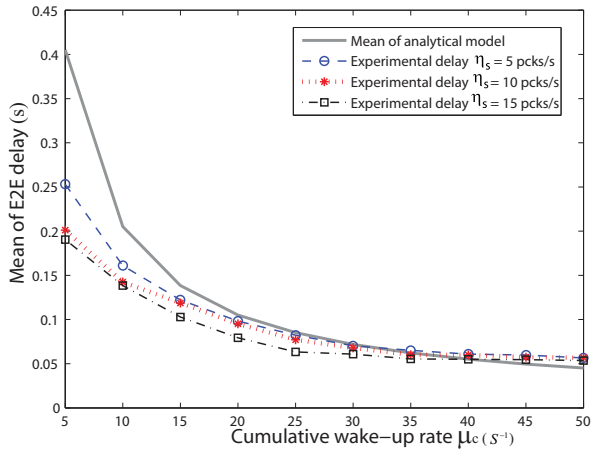
The delay $D(n_h, \mu_c)$ between edge to sink is given by the sum of the delays experienced by a packet at each hop. There are two sources of delay:

- Time to wait before the first wake-up of a node in the next cluster: Let such a time be denoted with ϵ_i for cluster i.
- Time to wait for clean channel: Since the Breath protocol uses CSMA/CA, a node spends a random time before sensing idle channel. Denote with ε_i such a time for cluster i.

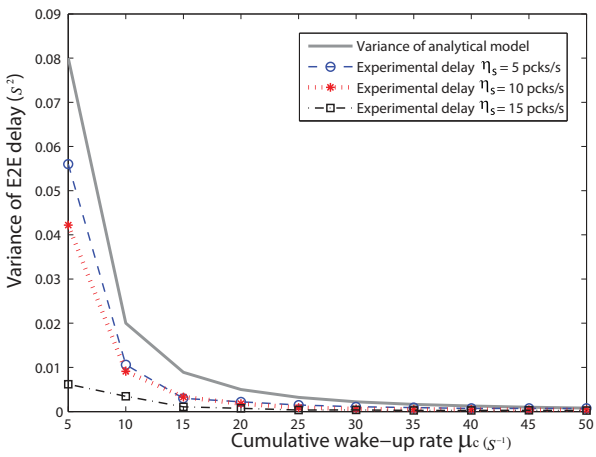
By summing these delays per each hop, we obtain the delay model

$$D(n_h, \mu_c) = \sum_{i=1}^{n_h} (\epsilon_i + \varepsilon_i). \tag{5.7}$$

In this equation, ϵ_i is an exponentially distributed random variable whose intensity μ_c is the sum of the wake-up intensities of the nodes in the next cluster. Characterization of ε_i is more difficult, owing to the backoff mechanism of the CSMA/CA algorithm. However, we assume that the backoff time can be approximated by a



(a) average delay



(b) variance of delay

Figure 5.5: Validation of average and variance of delay given by Eq. (5.8), (5.9) by experimental results, respectively. The traffic rates $\eta_s = 5, 10, 15$ pckt/s are considered w.r.t. wake-up rates μ_c from 5 to 50, $n_h = 2$ hops and $N = 15$ nodes.

Gaussian distribution whose average is matched with the average and standard deviation of a uniformly distributed random variable between 0 and $(m+1)(W-1)$. Namely,

$$\varepsilon_i \in \mathcal{N}\left(\frac{(m+1)(W-1)T_b}{2}, \frac{(m+1)(W^2-1)T_b^2}{12}\right).$$

We remark that this approximation significantly reduces the computation complexity when compared to an accurate analysis of the probability density function of the packet delay, which we have proposed in [212]. According to such an assumption, the delay $D(n_h, \mu_c)$ is approximated by a Gaussian random variable $\mathcal{N}(\mu_D, \sigma_D^2)$, where

$$\mu_D = \frac{n_h}{\mu_c} + \frac{n_h(m+1)(W-1)T_b}{2}, \tag{5.8}$$

$$\sigma_D^2 = \frac{n_h}{\mu_c^2} + \frac{n_h(m+1)(W^2-1)T_b^2}{12}. \tag{5.9}$$

We validated these approximations by comparing the analysis to experimental results. Figures 5.5(a), 5.5(b) show the mean and variance of the delay given by Eq. (5.8), (5.9) and the experimental results, respectively. The analytical model describes well the experimental data because it gives an upper bound for wake-up rates up to $35\,\mathrm{s}^{-1}$, and then the model underestimates the experimental result for less than 5%. These properties are quite useful for optimization purposes. Same dependence is found on n_h up to 4.

We are now in the position to express the delay constraint in Problem (5.1) by using Eqs. (5.8) and (5.9) that we just derived:

$$\Pr[D \le D_{\max}] \approx 1 - Q\left(\frac{D_{\max} - \mu_D}{\sigma_D}\right) \ge \Omega, \tag{5.10}$$

where $Q(x) = 1/\sqrt{2\pi}\int_x^\infty e^{-t^2/2}dt$ is the complementary standard Gaussian distribution. After some manipulations, it follows that (5.10) can be rewritten as

$$\mu_c \ge \frac{12\,C_{d1}\,n_h + 2\sqrt{3\,C_{d3}\,n_h\,\left[12\,C_{d1}^2 + C_{d2}\,(n_h - C_{d3})\right]}}{12\,C_{d1}^2 - C_{d2}\,C_{d3}},$$

where

$$C_{d1} = D_{\max} - \frac{n_h(m+1)(W-1)T_b}{2},$$
$$C_{d2} = n_h(m+1)(W^2-1)T_b^2,$$
$$C_{d3} = \left(Q^{-1}(1-\Omega)\right)^2.$$

Since $T_b^2 = 0.1024 \times 10^{-6}$ [31], and n_h, m, W are positive integers, it follows that $T_b^2 \ll n_h(m+1)(W^2-1)$. Then $C_{d2} \ll C_{d1}$ and (5.10) is approximated by

$$\mu_c \ge f_d(n_h, D_{\max}, \Omega) \triangleq \frac{2\left[n_h + Q^{-1}(1-\Omega)\sqrt{n_h}\right]}{2D_{\max} - n_h(m+1)(W-1)T_b}. \tag{5.11}$$

Inequality (5.11) has been derived under the additional constraint

$$n_h \leq n_{h,d} \triangleq \frac{2D_{\max}}{(m+1)(W-1)T_b}\,. \tag{5.12}$$

This constraint is a function of the maximum allowable delay requirement $D_{\max}$ and the parameters of CSMA/CA algorithm such as the maximum number of CSMA/CA transmission tries m, the maximum number of random backoff W and the unit of backoff period T_b. As the delay requirement $D_{\max}$ becomes stricter or the parameters of CSMA/CA algorithm m, W, T_b increase, the constraint $n_{h,d}$ decreases. We will use (5.11) and (5.12) in Section 5.5 to find the solution of the optimization problem (5.1). Now, we investigate the total energy consumption.

5.4.3 Energy Consumption

The total energy consumption is

$$E_{\mathrm{n,tot}}(n_h, \mu_c) = E_{\mathrm{pck}}(n_h, \mu_c) + E_{wu}(n_h, \mu_c)\,, \tag{5.13}$$

where $E_{\mathrm{pck}}(n_h, \mu_c)$ is the total energy for transmission and reception of data packets and $E_{wu}(n_h, \mu_c)$ is the energy consumption for wake-up, listening and beaconing during a time T, which we characterize in Section 5.4.3, 5.4.3, respectively.

Data Packet Communication Energy

Assuming n_h hops, and recalling that edge node emits η_s pckt/s,

$$E_{\mathrm{pck}}(n_h, \mu_c) = T\eta_s \sum_{i=1}^{n_h} \left[E_m(d_i) + \frac{P_r}{\mu_c} + E_{\mathrm{ca}}(\mu_c) + E_r \right], \tag{5.14}$$

where E_r accounts for the fixed cost of the RF circuit for the reception of a data packet. The term $E_m(d_i)$ is the energy consumption for radio transmission, where d_i is the transmission distance to which a data packet has to be sent. The term $E_{\mathrm{ca}}(\mu_c)$ is the energy spent during the CSMA/CA state.
The energy model given by Eq. (5.14) is derived under the assumption that all packets generated at the edge nodes reach the sink. Obviously, some packet may be lost before reaching the sink, therefore (5.14) gives an upper bound on the energy consumption. This is reasonable, since our goal is the minimization of the cost function.
The energy spent for radio transmission is a function of the radio power used to transmit packets:

$$E_m(d_i) = V\, I(P_t(d_i))\, t_m\,, \tag{5.15}$$

where V is the voltage consumption of the RF circuit at the node, t_m is the transmission time of a data packet, $I(P_t(d_i))$ is the current consumption of the electronic

circuit needed to transmit packets at radio power $P_t(d_i)$, and d_i is the distance from the transmitter which a packet must reach to with some desired probability. The relation between the current consumption and radio power depends on the hardware platform. For Tmote sensors, it holds that [213]

$$I(P_t(d_i)) \approx -19P_t(d_i)^4 + 53P_t(d_i)^3 - 53P_t(d_i)^2 + 29P_t(d_i) + 8.7\,.$$

Given this approximation, minimization of $E_m(d_i)$ is achieved by minimizing $P_t(d_i)$. $P_t(d_i)$ can be minimized by computing the minimum radio power that ensures packets to reach a given distance with a given probability, as we see next.
The optimal transmit power is derived by considering the distribution of the signal to interference plus noise ratio (SINR). By imposing a requirement p_{con} on the probability of successful packet reception at a distance d_k from node k, we can translate the requirement on the average SINR, thus obtaining a bound $\overline{\gamma}_c$ on such an average SINR. From this we can then derive the transmit radio power necessary to successfully receive packets at a distance d_k with probability p_{con}. It follows that the minimum transmit power is [214]

$$P_t(d_k)_{\text{ dB}} = \overline{\gamma}_{c\text{ dB}} + \text{PL}(d_0)_{\text{ dB}} + 10\,\beta_k\,\log_{10}\frac{d_k}{d_0} + P_{n\text{ dB}} - \frac{\ln 10}{20}\sigma^2_{\gamma_k\text{ dB}}\,,$$

where $\text{PL}(d_0)_{\text{ dB}}$ is the path loss at a reference distance d_0, β_k is the path loss decay constant, P_n is the noise floor, and σ_{γ_k} is the variance of the SINR (see [214] for details). We remark that the power $P_t(d_k)_{\text{ dB}}$ minimizes the energy spent for radio transmission in Eq. (5.15). Notice that the actual packet reception probability p_i may fluctuate around p_{con} due to the delay of this power control and the limited maximum transmit power.
In the following, we characterize E_{ca}. Consider the energy spent for transmission of a data packet in the i th cluster. Let E_{ca} is the energy spent by a node to check the channel status by the CSMA/CA algorithm upon the reception of a beacon. This energy, which is due to CCA, is dependent on the maximum number of tries m. We have two situations: the number of contending nodes N_i attempting to transmit a data packet is less then m, or the number of contending nodes is larger than m. If $N_i < m+1$, all nodes will succeed to sense a clean channel with the energy $E_{\text{ca1}}(N_i)$, otherwise we need to consider the transmission success and failure probabilities to perform CCA with the energy $E_{\text{ca2}}(N_i)$, which is the function of the busy channel probability $\alpha(k)$ conditioned on k contending nodes defined by Eq. (5.2), see the details in [214]
By summing two the energy components $E_{\text{ca1}}(i), E_{\text{ca2}}(i)$, the average energy consumption spent by the CSMA/CA is

$$E_{\text{ca}}(\mu_c) = \sum_{i=1}^{\infty} \psi_{\text{sb}}(\mu_c, i)\,[E_{\text{ca1}}(i)\,u(m-i) + E_{\text{ca2}}(i)\,u(i-m-1)]\,, \qquad (5.16)$$

where $\psi_{\text{sb}}(\mu_c, i)$ is the probability to have i contending nodes in a cluster given by Eq. (5.3) and $u(x) = 1$ if $x \geq 0$, whereas $u(x) = 0$ otherwise.

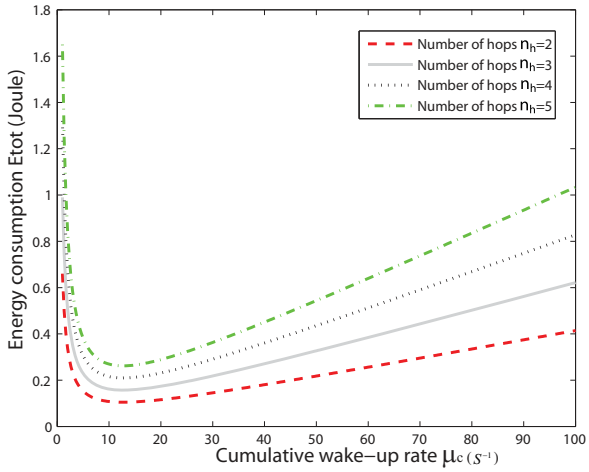


Figure 5.6: Total energy consumption given by Eq. (5.18) for a different number of hops ($n_h = 2, 3, 4, 5$) over wake-up rates μ_c from 1 to 100 in traffic rate ($\eta_s = 5$ pcks/s) and $N = 15$ nodes.

Control Signalling Energy

A node randomly cycles between an awake state and a sleep state. Each time a node wakes up, it spends an energy given by the power needed to wake-up P_w during the wake-up time T_w, plus the energy to listen for the reception of a data packet within a maximum time T_{ac}. After a node wakes up, it transmits a beacon to the next cluster. Let the wireless channel loss probability be $1 - p_i$ of cluster i, then nodes of cluster $i-1$ have to wake-up on average $1/p_i$ times to create the effect of a single wake-up so that a transmitter node successfully receives a beacon. Recalling that there are n_h hops and a cumulative wake-up rate per cluster μ_c, the total cost in a time T for wake-ups and beaconing is

$$E_{wu}(n_h, \mu_c) = T\mu_c \sum_{i=1}^{n_h} \frac{1}{p_i} \left[E_b(d_i) + P_w T_w + P_r(T_{ac} - T_w)\right] , \tag{5.17}$$

where $E_b(d_i)$ is the expected energy consumption to transmit a beacon message at the distance d_i.

Total Energy Consumption

Here we put together the energy analysis developed in the previous two sections. The total energy consumption is

$$\begin{aligned} E_{\mathrm{n,tot}}(n_h,\mu_c) =& T\eta_s \left[E_m\left(\frac{S_{\mathrm{tot}}}{n_h-1}\right) + E_m\left(\frac{S_{\mathrm{tot}}}{n_h-1}\right)(n_h-1)u(n_h-1) \right. \\ &\left. + n_h\left(\frac{P_r}{\mu_c} + E_{\mathrm{ca}}(\mu_c) + E_r\right)\right] + \frac{T\mu_c}{p_{\min}}\left[2E_b\left(\frac{S_{\mathrm{tot}}}{n_h-1}\right)\right. \\ &\left. + E_b\left(\frac{2S_{\mathrm{tot}}}{n_h-1}\right)(n_h-2)u(n_h-2) + n_h\left(P_wT_w + P_r(T_{ac}-T_w)\right)\right], \end{aligned} \tag{5.18}$$

where we upper bounded (5.14) and (5.17) by considering the worst distance to which data and beacon packets must be sent, which are $S_{\mathrm{tot}}/(n_h-1)$ and $2S_{\mathrm{tot}}/(n_h-1)$, and the worst reception probability $p_{\min}$.
Figure 5.6 shows the energy given by Eq. (5.18) as a function of the number of hops n_h over different wake-up rates μ_c. The total energy consumption increases with n_h given μ_c because increasing n_h implies higher wake-up rates per node for a given number of total nodes present in the network. In other words, increasing the number of hops is energy inefficient. Observe also that a low wake-up rate does not minimize the total energy consumption, because of the longer waiting time to receive a beacon message that such a rate causes. Hence, a tradeoff exists between the energy consumption for wake-up and waiting to get a beacon message. We explore this tradeoff for optimization problem in the following section.

5.5 Optimal Protocol Parameters

In this section we give the optimal protocol parameters used by Breath. Consider the reliability and delay constraints, and the total energy consumption as investigated in Sections 5.4.1, 5.4.2, and 5.4.3. The optimization problem (5.1) becomes

$$\begin{aligned} \min_{n_h,\mu_c} \quad & E_{\mathrm{n,tot}}(n_h,\mu_c) \\ \text{s.t.} \quad & \mu_c \geq \max(f_r(n_h,R_{\min}), f_d(n_h,D_{\max},\Omega)), \\ & 2 \leq n_h \leq \min\left(n_{h,r}, n_{h,d}\right), \\ & \mu_{\min} \leq \mu_c \leq \mu_{\max}, \end{aligned} \tag{5.19}$$

where the first constraint comes from (5.5) and (5.6), and the second from (5.11) and (5.12). We assume that this problem is feasible. Infeasibility means that for any $n_h = 2,\ldots,\min\left(n_{h,r}, n_{h,d}\right)$, then $\mu_c \geq \max(f_r(n_h,R_{\min}), f_d(n_h,D_{\max},\Omega)) > \mu_{\max}$, namely it is not possible to guarantee the satisfaction of the reliability and delay constraint given the application requirements. This means that the application requirements must be relaxed, so that feasibility is ensured and the problem can

be solved. The solution of this optimization problem, n_h^* and μ_c^*, is derived in the following.

By using the numerical values given for the Tmote sensors [31] for all the constants in the optimization problem, we see that the cost function of Problem (5.19) is increasing in n_h and convex in μ_c. This allows us to derive the optimal solution in two steps: for each value of $n_h = 2, \ldots, \min(n_{h,r}, n_{h,d})$, the cost function is minimized for μ_c, achieving $\mu_c^*(n_h)$. Then, the optimal solution if found in the pair $n_h, \mu_c^*(n_h)$ that gives the minimum energy consumption. We describe this procedure next.

Let n_h be fixed. From the properties the cost function of Problem (5.19), the optimal solution $\mu_c^*(n_h)$ is attained either at the minimum of the cost function or at the boundaries of the feasibility region given by the requirements on μ_c. The minimum of the cost function can be achieved by taking its derivative with respect to μ_c. To obtain this derivative in an explicit form, we assume that CSMA/CA energy consumption can be approximated by a constant value since the numerical value is smaller than other factors. Under this assumption, the minimization by the derivative is approximated by

$$\mu_e(n_h) = (p_{\min}\,\eta_s\; P_r)^{\frac{1}{2}} \left[\frac{n_h - 2}{n_h} E_b \left(\frac{2S_{\text{tot}}}{n_h - 1} \right) u(n_h - 2) + \frac{2}{n_h} E_b \left(\frac{S_{\text{tot}}}{n_h - 1} \right) + P_w T_w + P_r (T_{ac} - T_w) \right]^{-\frac{1}{2}} . \tag{5.20}$$

In Figure 5.7 we check the validity of this approximation. The figure reports the approximated minimum of the cost function as obtained by Eq. (5.20) compared to the wake-up rate that minimizes the actual energy consumption as obtained by a numerical minimization algorithm. The approximation is tight because the error is less than 2%.

Eq. (5.20) tells us that, provided n_h, an optimal solution $\mu_c^*(n_h)$ is given by $\mu_e(n_h)$ if $\mu_e(n_h)$ is in the feasible range $\mu_{\min} \leq \mu_e(n_h) \leq \mu_{\max}$ and $\mu_e(n_h)$ satisfies the two constraints of both reliability and delay requirement $\mu_e(n_h) \geq \max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega))$. If $\mu_e(n_h)$ is not in the feasible range $(\mu_e(n_h) \geq \mu_{\max}) \cup (\mu_e(n_h) \leq \mu_{\min})$ or $\mu_e(n_h)$ does not satisfies two the constraints $\mu_e(n_h) \leq \max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega))$ but $\max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega))$ is in the feasible range $\mu_{\min} \leq \max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega)) \leq \mu_{\max}$, then an optimal solution is given by $\max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega))$. Otherwise, an optimal solution is given by $\mu_{\max}$ as the best effort mode since the network does not satisfies the constraints. Therefore, for any $n_h = 2, \ldots, \min(n_{h,r}, n_{h,d})$, we compute $\mu_c^*(n_h)$. Then, the optimal solution n_h^* and μ_c^* is given by the pair $\mu_c^*(n_h)$, n_h that minimizes the cost function. This procedure to compute the optimal solution is illustrated by Algorithm 1, which summarize one of the main contribution of this chapter.

Algorithm 1: Algorithm for the computation of the optimal solution of Problem (5.19) **.**

Input: Requirements $R_{\min}, D_{\max}, \Omega$, feasible range $\mu_{\min}, \mu_{\max}$
Output: n_h^*, μ_c^*
begin
  $n_h^* \leftarrow 2$;
  if
  $\big(\mu_e(n_h^*) \geq \max(f_r(n_h^*, R_{\min}), f_d(n_h^*, D_{\max}, \Omega))\big) \cap \big(\mu_{\min} \leq \mu_e(n_h^*) \leq \mu_{\max}\big)$
  then
    $\mu_c^* \leftarrow \mu_e(n_h^*)$;
    SAT $\leftarrow true$;
  else if $\big[(\mu_e(n_h^*) \geq \mu_{\max}) \cup (\mu_e(n_h^*) \leq \mu_{\min}) \cup (\mu_e(n_h^*) \leq \max(f_r(n_h^*, R_{\min}), f_d(n_h^*, D_{\max}, \Omega)))\big] \cap \big(\mu_{\min} \leq \max(f_r(n_h^*, R_{\min}), f_d(n_h^*, D_{\max}, \Omega)) \leq \mu_{\max}\big)$ **then**
    $\mu_c^* \leftarrow \max(f_r(n_h^*, R_{\min}), f_d(n_h^*, D_{\max}, \Omega))$;
    SAT $\leftarrow true$;
  else
    $\mu_c^* \leftarrow \mu_{\max}$;
    SAT $\leftarrow fail$;
  $E \leftarrow E_{\text{n,tot}}(n_h^*, \mu_c^*)$;
  for $n_h \leftarrow 3$ **to** $\min(n_{h,r}, n_{h,d})$ **do**
    if $(\mu_e(n_h) \geq \max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega))) \cap (\mu_{\min} \leq \mu_e(n_h) \leq \mu_{\max})$ **then**
      $\mu_{\text{tmp}} \leftarrow \mu_e(n_h)$;
      SAT $\leftarrow true$;
    else if $\big[(\mu_e(n_h) \geq \mu_{\max}) \cup (\mu_e(n_h) \leq \mu_{\min}) \cup (\mu_e(n_h) \leq \max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega)))\big] \cap \big(\mu_{\min} \leq \max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega)) \leq \mu_{\max}\big)$ **then**
      $\mu_{\text{tmp}} \leftarrow \max(f_r(n_h, R_{\min}), f_d(n_h, D_{\max}, \Omega))$;
      SAT $\leftarrow true$;
    else
      $\mu_{\text{tmp}} \leftarrow \mu_{\max}$;
      SAT $\leftarrow fail$;
    $E_{\text{tmp}} \leftarrow E_{\text{n,tot}}(n_h, \mu_{\text{tmp}})$;
    if $(E > E_{\text{tmp}}) \cap$ SAT **then**
      $n_h^* \leftarrow n_h$;
      $\mu_c^* \leftarrow \mu_{\text{tmp}}$;
      $E \leftarrow E_{\text{tmp}}$;

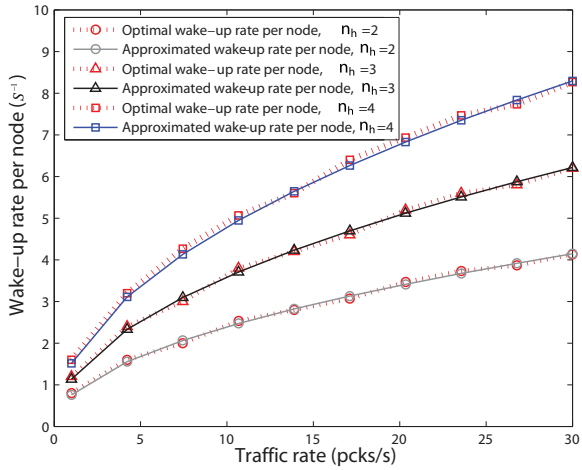


Figure 5.7: Wake-up rate that minimizes the total energy consumption and approximated wake-up rate as obtained by (5.20) for different number of hops ($n_h = 2, 3, 4$), traffic rates η_s from 1 to 30 pcks/s and $N = 15$. The y-axis was normalized by 15.

5.6 Adaptation Mechanisms

In the previous sections, we showed how to determine the optimal number of clusters and cumulative wake-up rate by solving an optimization problem. Here, we present in detail some adaptation algorithms that the sink must run to determine correctly n_h^* and μ_c^* as the traffic rate and channel conditions changes. These algorithms allow us to adapt the protocol parameters to the traffic rate and channel condition without high message overhead.

5.6.1 Traffic Rate and Channel Estimation

The sink node estimates the traffic rate η_s and the worst channel probability $p_{\min}$ of the network. To estimate the global minimum of the worst channel condition, each p_i should be estimated at a local node and sent to the sink for each link of the path $i = 1, \ldots, n_h$. This might increase considerably the packet size. To avoid this, we propose the following strategy. Consider a relay node of the i th cluster. It estimates p_i by the signal of the beacon packet. Then the nodes compares p_i with the channel condition information carried by the received data packet and selects the minimum. This minimum is then encoded in the data packet and sent with it to the next-hop node. After the sink node retrieves the channel condition of the route by receiving a data packet, it computes an average of the worst channel conditions among the last received data packets. Using this estimate, the sink solves the optimization problem running Algorithm 1. Afterwards, the return value of the algorithm, n_h^* and μ_c^*, can

be piggybacked on beacons that the sink sends toward the relays closer to the sink. Then, these protocol parameters are forwarded when the nodes wake-up and send beacons to the next cluster toward the edge nodes. During the initial state, nodes set $n_h = 2$ before receiving a beacon.

5.6.2 Wake-up Rate and Radio Power Adaptation

Once a cluster received μ_c^*, each node in the cluster must adapt its wake-up rate so that the cluster generates such a cumulative wake-up rate. We consider the natural solution of distributing μ_c^* equally between all nodes of the cluster. Let μ_k be the wake-up rate of node k, and suppose that there are l nodes in a cluster. The fair solution is $\mu_k = \mu_c^*/l$ for any node. However, a node does not know and cannot estimate efficiently the number of nodes in its cluster.
To overcome this problem, we follow the same approach proposed in [215], where an Additive Increase and Multiplicative Decrease (AIMD) algorithm leads to a fair distribution of the wake-up duties within a single cluster. Specifically, each node that is waiting to forward a data packet observes the time before the first wake-up in the forwarding region. Starting from this observation, it estimates the cumulative wake-up rate $\tilde{\mu}_c$ of the forwarding region and it compares it with the optimal value of the wake-up rate μ_c^* when a node receives a beacon. Note that the node retrieves information on n_h^*, μ_c^* and location information of the beacon node. If $\tilde{\mu}_c < \mu_c^*$ the node sends by the data packet an Additive Increase (AI) command for the wake-up rate of next-hop cluster, else it sends a Multiplicative Decrease (MD) command. Furthermore, the node updates the probability of successful transmission p_i based on the channel information using the RSSI and distance information d_k between its own location and beacon node. After the node updates the channel condition estimation, it sets the data packet transmission power to $P_t(d_k)$, and encodes the channel estimation in the packet as described in Section 5.6.1. If a data packet is received, the node retrieves information on wake-up rate update: if AI then $\mu_k = \mu_k + \theta$, else $\mu_k = \mu_k/\phi$, where θ and ϕ are control parameters. From experimental results, we obtained that $\theta = 3$ and $\phi = 1.05$ achieve good performance. The command on the wake-up rate variation is piggybacked on data packets and does not require any additional message.
However, this approach may generate a load balancing problem because of different wake-up rates among relays within a short period. Load balancing is a critical issue, since some nodes may wake-up at higher rate than desired rate of other nodes, thus wasting energy. To overcome this situation, each relay node runs a simple reset mechanism. We assign an upper and lower bound to the wake-up rate for each node. If the wake-up rate of a node is larger than the upper bound $(1+\xi)\mu_c^*(n_h^*-1)/N$ or is smaller than the lower bound $(1-\xi)\mu_c^*(n_h^*-1)/N$, then a node resets its wake-up rate to $\mu_c^*(n_h^*-1)/N$, where ξ assumes a small value and $(n_h^*-1)/N$ is an estimation of the number of nodes per cluster.

5.7 Fundamental Limits

Understanding the fundamental limits of Breath is critical for its appropriate use. This section focuses on the minimum number of relays required to support the protocol, and the minimum delay that can be set by the application.

5.7.1 Minimum Number of Nodes per Cluster

The minimum number $N_{\min}$ of nodes per cluster to support the protocol with given reliability and delay requirements is

$$N_{\min} \geq \frac{\mu_c^*}{\mu_{k,\max}} ,$$

where $\mu_{k,\max}$ is the maximum wake-up rate of node k. By considering the worst active time for the duty cycle, we have

$$\mu_{k,\max} = (T_{ac} + T_{\text{be}})^{-1} ,$$

where $T_{ac} = 30$ ms is the maximum listening time to receive a data packet and $T_{\text{be}} = 500$ ms is the maximum waiting time before receiving a beacon [31].

5.7.2 Minimum Delay

The minimum delay that the application can set is achieved by considering a very high wake-up rate per cluster. This minimizes the waiting time before receiving a beacon. Hence, by summing the delays of the CSMA/CA state and physical limits of the wireless channel, the minimum delay is

$$D_{\max} \geq n_h \, [2(m+1)(W-1)T_b + 2\,T_{\text{prop}} + t_m + t_b] ,$$

where the fist term is the maximum delay of CSMA/CA state, T_{prop} is propagation delay, and t_m and t_b are, respectively, the transmission delay of data and beacon packets. Since $T_{\text{prop}} = 0.875$ ms, $t_m = 1.5$ ms and $t_b = 0.64$ ms [31], they can be basically ignored because they are negligible with respect to other delays.

5.8 Experimental Implementation

In this section we provide an extensive set of experiments to validate the Breath protocol. The experiments enable us to assess Breath in terms of reliability and delay in the packet transmission, and energy consumption of the network both in stationary and transitionary condition. The protocol was implemented on a test bed of Tmote sensors [31], and was compared with a standard implementation of IEEE 802.15.4 [8], as we discuss next.

We consider a typical indoor environment with concrete walls. The experiments were performed in a static propagation (AWGN) and time-varying fading environment (Rayleigh), respectively:

- AWGN environment: nodes and surrounding objects were static, with minimal time-varying changes in the wireless channel. In this case, the wireless channel is well described by an additive white gaussian noise (AWGN) model.
- Rayleigh fading environment: obstacles were moved within the network, along a line of 20 m. Furthermore, a metal object was put in front of the edge node, so the edge node and the relays were not in line-of-sight. The edge node was moved on a distance of few tens of centimeters.

A node acted as edge node and generated packets periodically at different rates ($\eta_s = 5, 10$, and 15 pckt/s). 15 relays were placed to mimic the topology in Figure 5.1. The edge node was at a distance of 20 m far from the sink. The sink node collected packets and then computed the optimal solution using Algorithm 1. The delay requirement was set to $D_{\max} = 1$s and the reliability to $R_{\min} = 0.9$ and 0.95. In other words, we imposed that packet must reach destination within 1s with a probability of $R_{\min}$. These requirements were chosen as representative for control applications.
We compared Breath against an implementation of the unslotted IEEE 802.15.4 standard [8], which is similar to the randomized MAC that we use in this chapter. In such an IEEE 802.15.4 implementation, we set nodes to a fixed sleep schedule, defined by CT_{ac} were C is integer number (recall that T_{ac} is the maximum node listening time in Breath). We defined the case $\mathcal{L}$ (low sleep), were the IEEE 802.15.4 implementation is set with $C = 1$, whereas we defined the case $\mathcal{H}$ (high sleep) by setting $C = 4$. The case $\mathcal{H}$ represents a fair comparison between Breath and IEEE 802.15.4, while in the case $\mathcal{L}$ nodes are let to listen much longer time than nodes in Breath. The power level in the IEEE 802.15.4 implementation where set to -5 dBm. We set the IEEE 802.15.4 protocol parameters to default values *macMinBE* $= 3$, *aMaxBE* $= 5$, *macMaxCSMABackoffs* $= 4$. Details follows in the sequel.

5.8.1 Protocol Behavior for Stationary Requirements

In this subsection, we investigate the performance of Breath about the reliability, average delay and energy consumption that can be achieved in a stationary configuration of the requirements, i.e., during the experiment there was not change of application requirements. Data was collected out of 10 experiments, each lasting 1 hour.

Reliability

Figure 5.8 indicates that the network converges by Breath to a stable error rate lower than $1 - R_{\min}$ and hence satisfies the required reliability with traffic rate $\eta_s = 10$ pckt/s, the delay requirement $D_{\max} = 1$s, and $R_{\min} = 0.9, 0.95$. IEEE 802.15.4 $\mathcal{H}$ in AWGN channel provides the worse performance than the other protocols because of lower wake-up rate. Observe that $R_{\min} = 0.9$ in Rayleigh fading environment gives the better reliability than $R_{\min} = 0.95$ in AWGN channel due to higher

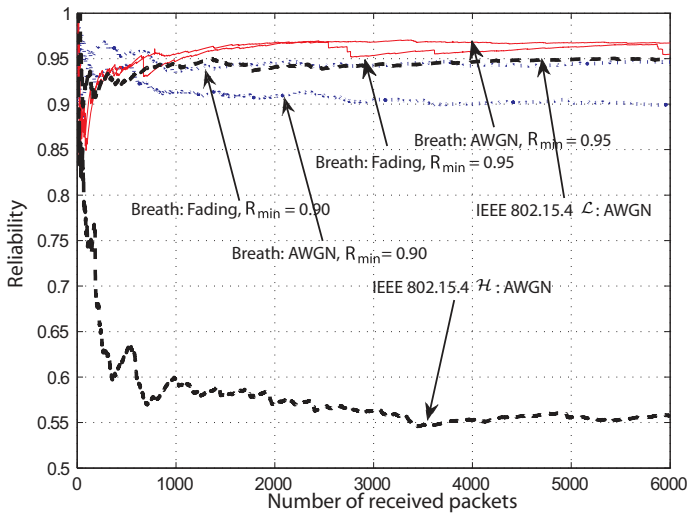


Figure 5.8: Convergence over the number of received packets of the reliability for IEEE 802.15.4 $\mathcal{L}$, $\mathcal{H}$ in AWGN, and Breath with reliability requirements $R_{\min} = 0.9, 0.95$ and traffic rates $\eta_s = 10\,\text{pckt/s}$ in AWGN and Rayleigh fading environments.

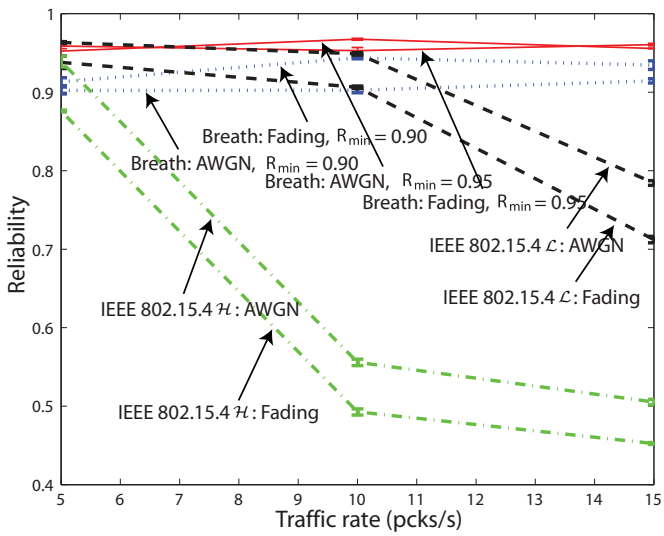


Figure 5.9: Reliability in IEEE 802.15.4 $\mathcal{L}$, $\mathcal{H}$, and Breath with requirement $R_{\min} = 0.9, 0.95$ for traffic rates $\eta_s = 5, 10, 15\,\text{pckt/s}$ in AWGN and Rayleigh fading environments. The vertical bars indicate the standard deviation as obtained out of 10 experimental runs of 1 hour each.

wake-up rate to compensate the fading channel condition. Notice that the higher fluctuation of reliability between the number of received packets 2500 and 2800 for Rayleigh fading environment with $R_{\min} = 0.95$ is due to deep attenuations in the wireless channel.
Figure 5.9 shows the reliability of Breath and IEEE 802.15.4 $\mathcal{L}$, $\mathcal{H}$ as a function of the reliability requirement $R_{\min} = 0.9, 0.95$ and traffic rate $\eta_s = 5, 10, 15\,\text{pckt/s}$ in AWGN and Rayleigh fading environments, with the vertical bars indicating the standard deviation as obtained out of 10 experimental runs of 1 hour each. Observe that the reliability is stable around the required value for Breath, and this holds for different traffic rates and environments. However, IEEE 802.15.4 $\mathcal{L}$ and $\mathcal{H}$ do not ensure the reliability satisfaction for large traffic rates. Specifically, IEEE 802.15.4 $\mathcal{H}$ shows poor reliability in any case, and performance worsen as the the environment moves from the AWGN to the Rayleigh fading. Furthermore, even though IEEE 802.15.4 $\mathcal{L}$ imposes that nodes wakes up more often, it does not guarantee a good reliability in higher traffic rates. The reason is found in the sleep schedule of the IEEE 802.15.4 case, which is independent of traffic rate and wireless channel conditions. The result is that the fixed sleep schedule is not feasible to support high traffic and time-varying wireless channels. Moreover, the fixed sleep schedule does not guarantee a uniform distribution of cumulative wake-up rate within certain time in a cluster, which means that there may be congestions. On the contrary, Breath presents an excellent behavior in any situations of traffic load and channel condition.

Delay

In Figure 5.10, the sample average of the delay for packet delivery of Breath, IEEE 802.15.4 $\mathcal{L}$ and $\mathcal{H}$ are plotted as a function of the reliability requirement $R_{\min}$ and traffic rate η_s in AWGN and Rayleigh fading environments, with the vertical bars indicating the standard deviation of the samples around the average. The sample variance of the delay exhibits similar behavior as the average. The delay meets quite well the constrains. Observe that delay decreases as the traffic rate rises. Because Breath increases linearly the wake-up rate of nodes when the traffic rate increases (see Eq. (5.5)). The delay is larger for worse reliability requirements. Note that Eq. (5.5) increases as the reliability requirement $R_{\min}$ increases. IEEE 802.15.4 $\mathcal{L}$ has lower delay than IEEE 802.15.4 $\mathcal{H}$ because nodes have higher wake-up time. Breath has an intermediate behavior with respect to IEEE 802.15.4 $\mathcal{L}$ and $\mathcal{H}$ after $\eta_s = 7$. From these experimental results, we conclude that both Breath and IEEE 802.15.4 meet the delay requirement. However, notice that the delay for IEEE 802.15.4 is related to only packet successfully received, which may be quite few.

Duty Cycle

In this section we study the energy consumption of the nodes.

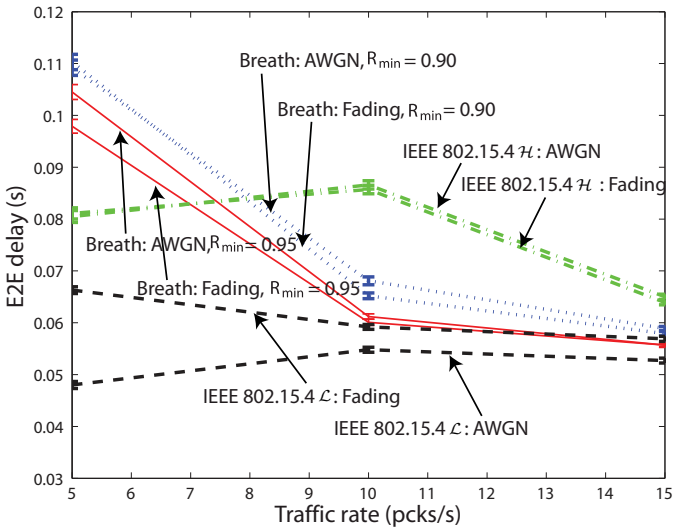


Figure 5.10: Temporal average of the delay of Breath and IEEE 802.15.4 $\mathcal{L}$, $\mathcal{H}$ with reliability requirement $R_{\min} = 0.9, 0.95$ and delay requirement $D_{\max} = 1\,\mathrm{s}$ over traffic rates $\eta_s = 5, 10, 15$ pckt/s in AWGN and Rayleigh fading environment. The vertical bars indicate the standard deviation as obtained out of 10 experimental runs of 1 hour each.

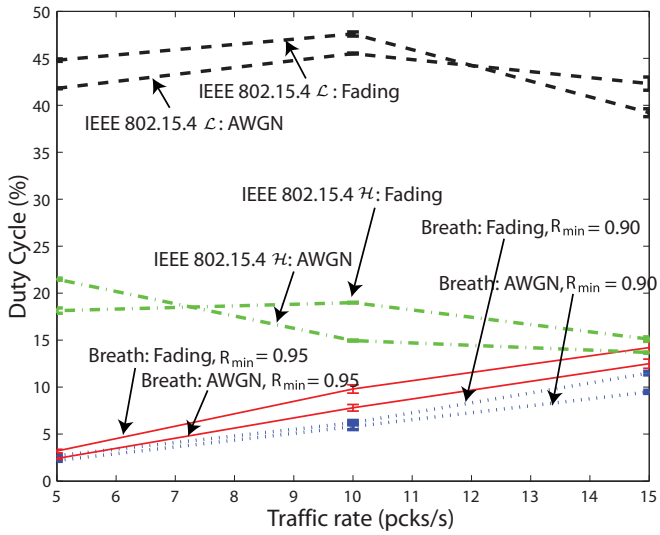


Figure 5.11: Sample average of the node's duty cycle in IEEE 802.15.4 $\mathcal{L}$, $\mathcal{H}$ and Breath with reliability requirement $R_{\min} = 0.9, 0.95$ for traffic rates $\eta_s = 5, 10, 15\,\mathrm{pckt/s}$ in AWGN and Rayleigh fading environments.

As energy performance indicator, we measured the node's duty cycle, which is the ratio of the active time of the node to the total experimental time. Obviously, the lower is the duty cycle, the better is the performance of the protocol on energy consumption.
Figure 5.11 shows the sample average of duty cycle of Breath, IEEE 802.15.4 $\mathcal{L}$ and $\mathcal{H}$ with respect to the traffic rates $\eta_s = 5$, 10, 15 pckt/s and $R_{\min} = 0.9, 0.95$, both in AWGN and Rayleigh fading environments, with the vertical bars indicating the standard deviation of the samples. Note that IEEE 802.15.4 $\mathcal{L}$ and $\mathcal{H}$ do not exhibit a clear relationship with respect to traffic rate and have almost flat duty cycle around 42% and 18%, respectively, because of fixed sleep time. Considering Breath, observe that the duty cycle increases linearly with the traffic rate for a given reliability requirement, which is explained by Eq. (5.5). Since Breath minimizes the total energy consumption on the base of a tradeoff between wake-up rate and waiting time of beacon messages (recall the analysis in Section 5.4.3), lower wake-up rates do not guarantee lower duty cycle. Observe that choosing a lower active time for the nodes of the IEEE 802.15.4 implementation would obviously obtain energy savings comparable with Breath, however, the reliability of the IEEE 802.15.4 implementation would be heavily affected (recall Figure 5.9). In other words, ensuring a duty cycle for the IEEE 802.15.4 implementation comparable with Breath would be very detrimental with respect to the reliability.
Figure 5.12 shows the experimental results for the duty cycle of each relay node for $\eta_s = 5$ pckt/s and $R_{\min} = 0.95$. A fair uniform distribution of the duty cycles among all nodes of the network is achieved. This is an important result, because the small variance of the wake-up rate among nodes signifies that duty cycle and load are uniformly distributed, with obvious advantages for the network lifetime.
Figure 5.13 reports the case of several networks, where each network corresponds to a number of relays between the edge and the sink in an AWGN environment. From the figure it is possible to evaluate how much Breath extends the network lifetime compared to IEEE 802.15.4 $\mathcal{L}$ and $\mathcal{H}$. Observe that the duty cycle is proportional to the density of nodes. Hence, the network lifetime is extended fairly by adding more nodes without creating load balancing problems.
Finally, recall that Breath uses a radio power control (Subsection 5.4.3), so that further energy savings are actually obtained with respect to the IEEE 802.15.4 implementation.

5.8.2 Protocol Behavior for Time-Varying Requirements

Performance of Breath protocol is based on the application requirements and estimation of the channel condition. In this subsection, we investigate the dynamic adaptation of Breath when the reliability $R_{\min}$ and delay $D_{\max}$ requirements change. Figure 5.14 shows the dynamic adaptability of reliability, packet delay and energy consumption when the requirements are changed for given traffic rate and number of nodes. Figures 5.14(a), 5.14(c), 5.14(e) and 5.14(b), 5.14(d), 5.14(f) present the behavior of the reliability, packet delay and average active time when the reliability

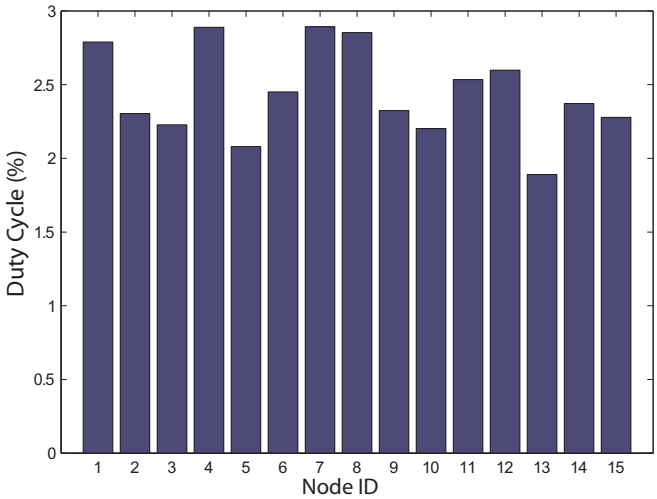


Figure 5.12: Distribution of the duty cycle in each node with $N = 15$ relays. The reliability requirement is $R_{\min} = 0.95$ and traffic rate is $\eta_s = 5\,\text{pckt/s}$.

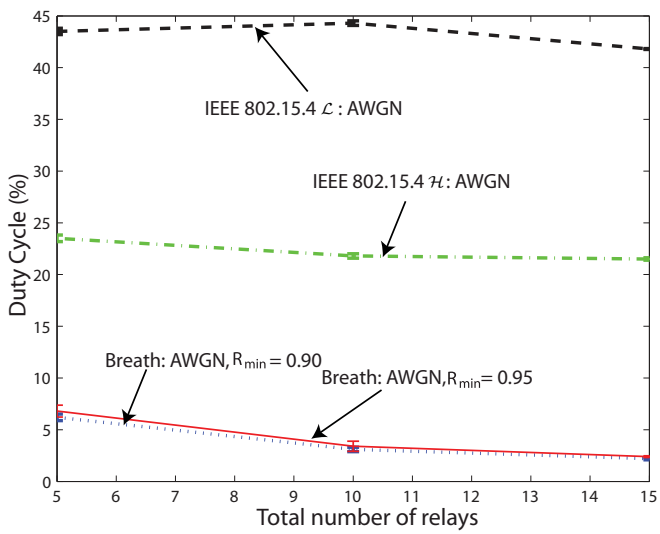
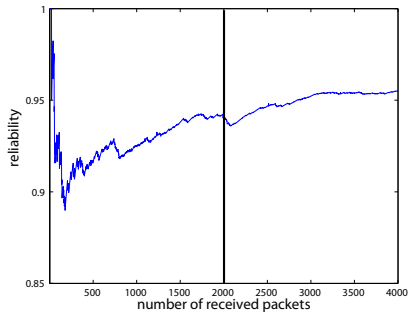
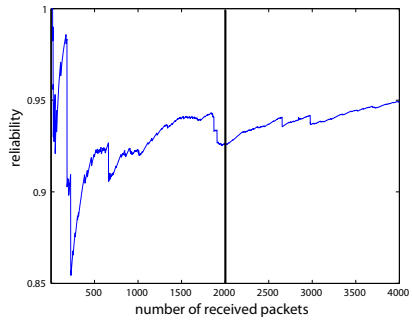


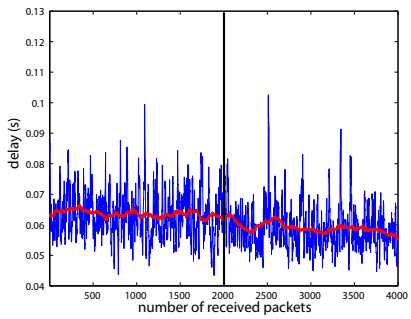
Figure 5.13: Sample average of the duty cycle in IEEE 802.15.4 $\mathcal{L}$, $\mathcal{H}$ and Breath with reliability requirement $R_{\min} = 0.9, 0.95$ and traffic rates $\eta_s = 5, 10, 15\,\text{pckt/s}$ in AWGN environment for different networks, each with a different number of relaying nodes. The vertical bars indicate the standard deviation as obtained out of 10 experimental runs of 1 hour each.



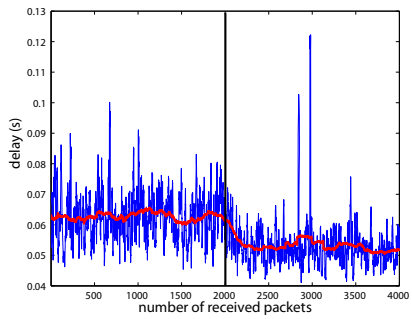
(a) Reliability behavior when $R_{\min}$ changes from 0.9 to 0.95



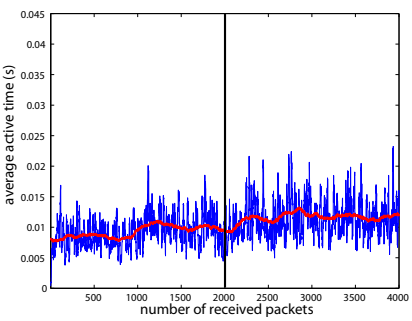
(b) Reliability behavior when $D_{\max}$ changes from 1 s to 60 ms



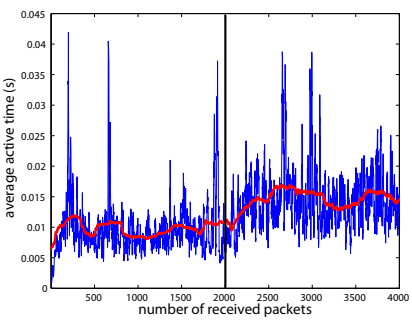
(c) Delay behavior when $R_{\min}$ changes from 0.9 to 0.95



(d) Delay behavior when $D_{\max}$ changes from 1 s to 60 ms



(e) Active time behavior when $R_{\min}$ changes from 0.9 to 0.95



(f) Active time behavior when $D_{\max}$ changes from 1 s to 60 ms

Figure 5.14: Reliability, packet delay and active time behavior for $\eta_s = 10\,\text{pckt/s}$, $N = 15$ when reliability and delay requirements vary from $R_{\min} = 0.9$ to $R_{\min} = 0.95$ and from $D_{\max} = 1\,\text{s}$ to $D_{\max} = 60\,\text{ms}$ at a time instant corresponding to the number of received packets 2000 in AWGN environment.

and delay requirements change, respectively. More specifically, the average active time is defined as the average time nodes are active. We observe performance in terms of reliability, delay, and average active time in Figures 5.14(a), 5.14(c), 5.14(e) for a reliability requirement variation. When $R_{\min}$ increases from 0.9 to 0.95 at a time corresponding to the number of received packet 2000 the reliability converges to 0.95. At the same time, packet delay decreases and average active time increase since optimal wake-up rate increases to guarantee the higher reliability requirement. Analogously, Breath adapts the network by considering the delay requirement variation in Figures 5.14(b), 5.14(d), and 5.14(f). It is clear that the average delay is under 60 ms since we consider the distribution of the delay probability. The average active time increases when the delay requirement changes due to a higher optimal wake-up rate. Hence, in the optimization problem, the delay requirement 60 ms gives a stricter constraint than reliability constraint computed at a requirement of 0.9. From this analysis, we can conclude that Breath adaptively achieves its target (i.e., minimization of power consumption) while guaranteeing the reliability and delay requirements. Furthermore, we observe clearly the tradeoff between the application requirements and energy consumption, i.e., as application requirements become strict, energy consumption increases.

5.9 Summary

We designed and implemented Breath, a protocol that is based on a system-level approach to guarantee explicitly reliability and delay requirements in wireless sensor networks for control and actuation applications. The protocol considers duty-cycle, routing, MAC, and physical layers all together to maximize the network lifetime by taking into account the tradeoff between energy consumption and application requirements for control applications.

We developed an analytical expression of the total energy consumption of the network, as well as reliability and delay for the packet delivery. These relations allowed us to pose a mixed real-integer constrained optimization problem to optimize the number of hops in the multi-hop routing, the wake-up rates of the nodes, and the transmit radio power as a function of the routing, MAC, physical layer, traffic, and hardware platform. An algorithm for the dynamic and continuous adaptation of the network operations to the traffic and channel conditions, and application requirements, was proposed.

We provided a test-bed implementation of the protocol, building a wireless sensor network with TinyOS and Tmote sensors. An experimental campaign was conducted to test the validity of Breath in an indoor environment with both AWGN and Rayleigh fading channels. Experimental results showed that the protocol achieves the reliability and delay requirements, while minimizing the energy consumption. It outperformed a standard IEEE 802.15.4 implementation in terms of both energy efficiency and reliability. In addition, Breath showed good load balancing performance, and is scalable with the number of nodes. Given its good performance, Breath is

a good candidate for many control and industrial applications, since these applications ask for both reliability and delay requirements in the packet delivery. A practical application of the protocol was illustrated in [26].

Chapter 6

Wireless Networked Control System Co-Design

In this chapter, it is shown how the proposed WSN protocols can be used in control applications. Multiple control systems are considered where the sensor measurements are transmitted to the controller over the IEEE 802.15.4 MAC protocol. The essential issues of wireless NCSs are investigated to provide an abstraction of the wireless network for a co-design approach. We first present an analytical model of the packet loss probability and delay of a IEEE 802.15.4 network. Through optimal control techniques we derive the control cost as a function of the packet loss probability and delay. Simulation results show the feasible control performance. It is shown that the optimal traffic load is similar when either the communication throughput or control cost are optimized. The co-design approach is based on a constrained optimization problem, for which the objective function is the energy consumption of the network and the constraints are the packet loss probability and delay, which are derived from the desired control cost. The co-design is illustrated through a numerical example.
The outline of the chapter is as follows. Section 6.2 defines the considered problem of control over a wireless network. In Section 6.3, we briefly remind the IEEE 802.15.4 standard and its network model. The design of the estimator and the controllers is presented in Section 6.4. In Section 6.5, we discuss the essential issues of wireless NCSs based on simulation results and propose a co-design approach. In Section 6.6, we illustrate it through numerical examples. Section 6.7 summarizes this chapter.

6.1 Motivation

Any wireless network introduces random packet losses and delays due to the harsh nature of the wireless channel, limited bandwidth, and interference generated by other wireless devices. The tradeoff between tractability and accuracy of the analytical model of a wireless network is important in order to hide the system complexity through a suitable abstraction without losing critical aspects of the network. Fur-

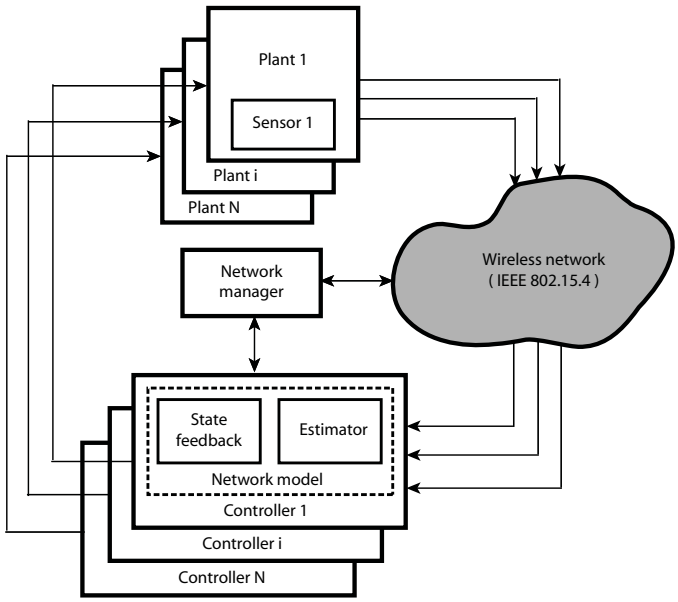


Figure 6.1: Overview of the NCS setup. N plants need to be controlled by N controllers. The wireless network closes the loop from the sensor nodes to the controllers.

thermore, WSNs require energy-efficient operation due to the limited battery power of each sensor node. Chapter 2.3 summarizes the important network quality measures and various design methods for NCSs.
In this chapter there are two original contributions:

1. We investigate the essential issues of wireless NCSs by considering the effects of wireless network on control performance.

2. We propose a co-design approach to meet the desired control cost while minimizing the energy consumption of the network.

In particular, we show the feasible control performance by considering the wireless network effects. This chapter explicitly considers both the control cost of control applications and the network performance with respect to energy consumption, which is the most important requirement of communication protocol design for WSNs. The key issue addressed here is how to derive the explicit relation between the performance of the control systems and the characteristics of a wireless network. Furthermore, the well-defined design procedure is investigated to achieve high performances in wireless NCSs.

6.2 Problem Formulation

The problem considered is depicted in Figure 6.1, where multiple plants are controlled over a WSN using the IEEE 802.15.4 protocol. N plants contend to transmit sensor measurements to the controller over a wireless network that induces packet losses and varying delays. We assume that a sensor node is attached to each plant. A contention-based IEEE 802.15.4 protocol is used to determine which sensor node accesses the wireless channel. Throughout this chapter we consider control applications where nodes asynchronously generate packets when a timer expiries. When a node sends a packet successfully or discards a packet, it stays in an idle period for $\overline{h}$ seconds without generating packets. The data packet transmission is successful if an ACK packet is received. We assume that the controller commands are always successfully received by the actuator. Many practical NCSs have several sensing channels, whereas the controllers are collocated with the actuators, as in heat, ventilation and air-conditioning control systems [143].
We consider a plant i, for $i = 1, \ldots, N$, given by a linear stochastic differential equation

$$dx(t) = Ax(t)dt + Bu(t)dt + dw(t) \tag{6.1}$$

where $x(t) \in \mathbb{R}^n$ is the plant state and $u(t) \in \mathbb{R}^m$ is the control signal. The process disturbance $w(t) \in \mathbb{R}^n$ has a mean value of zero and uncorrelated increments, with incremental covariance $R_w dt$. We neglect the plant index i to simplify notation. Let us consider the sampling of the plant with time-varying sampling period $h_k = t_{k+1} - t_k$ and delay d_k [216]. The sampling period is $h_k = \overline{h} + d_k$ where the idle period $\overline{h}$ is constant and the random delay is d_k, which is bounded $d_k \leq d_{\max}$. We assume that the random sequences $\{d_k\}$ and $\{h_k\}$ are bounded, $0 < d_k < h_k$ and $0 < h_{\min} \leq h_k \leq h_{\max}$. In addition, they are independent and have known distributions. Notice that the networked induced delay d_k is less than h_k and allows the packets to arrive at the controller in the correct order. By considering zero-order-hold, a time-varying discrete-time system is obtained

$$\begin{aligned} x_{k+1} &= \Phi_k x_k + \Gamma_0^k u_k + \Gamma_1^k u_{k-1} + w_k \\ y_k &= C x_k + v_k \end{aligned} \tag{6.2}$$

where $\Phi_k = e^{Ah_k}$, $\Gamma_0^k = \left[\int_0^{h_k - d_k} e^{As} ds\right] B$, $\Gamma_1^k = \left[\int_{h_k - d_k}^{h_k} e^{As} ds\right] B$, and v_k is a discrete-time white Gaussian noise with zero mean and variance R_v. The parameter k is the discrete time index. The initial state x_0 is white Gaussian with mean $\bar{x}_0$ and covariance P_0.
Packet loss is first modelled as a random process whose parameters are related to the behavior of the network. The measurement at the controller side is given by

$$\hat{y}_k = \begin{cases} Cx_k + v_k\,, & \gamma_k = 1\,, \\ 0\,, & \gamma_k = 0\,, \end{cases} \tag{6.3}$$

where γ_k is a Bernoulli random variable with $\Pr(\gamma_k = 1) = 1 - p$, where p is the packet loss probability which models the packet loss between the sensor and the controller.
By considering both the packet loss and delay induced by a wireless network, we introduce an augmented discrete-time state variable $z_k = \begin{pmatrix} x_k & u_{k-1} \end{pmatrix}^T$ to analyze the system. The augmented state space is

$$\begin{aligned} z_{k+1} &= \Phi_d z_k + \Gamma_d u_k + w_k \\ \hat{y}_k &= \gamma_k y_k \end{aligned} \tag{6.4}$$

where $\Phi_d = \begin{pmatrix} \Phi & \Gamma_1 \\ 0 & 0 \end{pmatrix}, \Gamma_d = \begin{pmatrix} \Gamma_0 \\ \mathbf{I} \end{pmatrix}$ and $C_d = \begin{pmatrix} C & \mathbf{0} \end{pmatrix}$.
In Figure 6.1, a network manager block is introduced to achieve an efficient control system over a wireless network. Particularly, the network manager requires an analytical model of the packet loss and delay (i.e., between the sensors and controller). Then, this model is used to design the estimator and controller that compensate for the packet loss and delay induced by the network. The network manager is based on a constrained optimization problem where the objective function, denoted by E_{tot}, is the total energy consumption of the wireless network and the constraint is the desired control cost. Hence, the constrained optimization problem of the control system is

$$\min_{h,\mathbf{V}} \quad E_{\text{tot}}(h, \mathbf{V}, \delta) \tag{6.5a}$$

$$\text{s.t.} \quad J(h, p(h, \mathbf{V}, \delta), d(h, \mathbf{V}, \delta)) \leq J_{\text{req}}\,. \tag{6.5b}$$

The decision variables are h, which is the sampling period, and $\mathbf{V}$, which are the protocol parameters of the network. δ includes the parameters of the network setup such as a network topology, length of packet, and number of nodes. $J(h, p(h, \mathbf{V}, \delta), d(h, \mathbf{V}, \delta))$ is the control cost, which is a function of the sampling period h, packet loss probability p, and delay d of the network, and J_{req} is the desired maximum control cost. We remark that the packet loss probability and delay of the network is also a function of the sampling period h, protocol parameters $\mathbf{V}$ and parameters of the network setup δ. Thus, the sampling period h affects the performance of both wireless network and control system. In (6.5b), the decision variables are feasible if they satisfy a given control cost J_{req}. Note that it is possible to pose different optimization problems under the same framework.

6.3 Wireless Medium Access Control Protocol

We consider the slotted CSMA/CA of IEEE 802.15.4 protocol [8] for a star network as described in Section 2.2. In Chapters 4 and 3, we proposed an effective analytical model of packet loss probability, packet delay, and power consumption of the

network imposed by the IEEE 802.15.4 protocol. Here, the adaptive IEEE 802.15.4 protocol of the Chapter 3 is applied to the co-design of wireless NCSs, as we show later.

6.4 Design of Estimator and Controller

In this section, we investigate how the packet loss probability and delay of the network affect the control performance. We discuss the design of an optimal feedback controller and present a control cost to analyze the NCSs described in Section 6.2. We first introduce our performance indicator as a control cost function, which is an explicit function of the sampling period h, packet loss probability p, and delay d of the network. Then, we design the estimator and controller under packet losses and delays in Section 6.4.1 and 6.4.2, respectively. This is achieved by extending the results on optimal stochastic estimation and control under packet losses in [25] with delays in [149].
Let us first define the information set under the packet loss and network induced delay as follows

$$\mathcal{I}_k = \{\mathbf{y}^k, \boldsymbol{\gamma}^k\} \tag{6.6}$$

where $\mathbf{y}^k = (y_k, y_{k-1}, \ldots, y_1)$ and $\boldsymbol{\gamma}^k = (\gamma_k, \gamma_{k-1}, \ldots, \gamma_1)$.
Consider the control cost function

$$J_M(\boldsymbol{u}^{M-1}, \bar{z}_0, P_0) = \mathbf{E}[z_M^T W_M z_M + \sum_{k=0}^{M-1} (z_k^T W_k z_k + 2 z_k^T M_k u_k + u_k^T U_k u_k)], \tag{6.7}$$

where $\bar{z}_0 = \begin{pmatrix} \bar{x}_0 & 0 \end{pmatrix}^T$, P_0 is the covariance of the initial condition, and the matrices W_k, M_k and U_k are time-invariant, symmetric and positive definite. In the following section, we introduce the estimator design.

6.4.1 Estimator Design

The estimator design is based on arguments similar to the standard Kalman filtering. Let us define the following variables

$$\begin{aligned} \hat{z}_{k|k} &= \begin{pmatrix} \mathbf{E}[x_k|\mathcal{I}_k] & u_{k-1} \end{pmatrix}^T \\ e_{k|k} &= z_k - \hat{z}_{k|k} \\ P_{k|k} &= \mathbf{E}[e_{k|k} e_{k|k}^T | \mathcal{I}_k]. \end{aligned}$$

The innovation step is given by

$$\hat{z}_{k+1|k} = \Phi_d \mathbf{E}[z_k|\mathcal{I}_k] + \Gamma_d u_k = \Phi_d \hat{z}_{k|k} + \Gamma_d u_k \tag{6.8}$$

$$e_{k+1|k} = z_{k+1} - \hat{z}_{k+1|k} = \Phi_d e_{k|k} + w_k$$

$$P_{k+1|k} = \mathbf{E}[e_{k+1|k} e_{k+1|k}^T | \mathcal{I}_k] = \Phi_d P_{k|k} \Phi_d^T + R_w \tag{6.9}$$

where w_k and $\mathcal{I}_k$ are independent and u_k is a deterministic function of $\mathcal{I}_k$. The correction step is given by

$$\begin{aligned}\hat{z}_{k+1|k+1} &= \hat{z}_{k+1|k} + \gamma_{k+1}K_{k+1}(y_{k+1} - C_d\hat{z}_{k+1|k}) \quad (6.10)\\ e_{k+1|k+1} &= z_{k+1} - \hat{z}_{k+1|k} = \Phi_d e_{k|k} + w_k\\ K_{k+1} &= P_{k+1|k}C_d^T(C_dP_{k+1|k}C_d^T + R_v)^{-1}\\ P_{k+1|k+1} &= P_{k+1|k} - \gamma_{k+1}K_{k+1}C_dP_{k+1|k} \quad (6.11)\end{aligned}$$

where we apply the standard derivation for the Kalman filter.

6.4.2 Controller Design

We introduce the feedback control law and present the finite and infinite horizon control cost functions. The cost function given by Eq. (6.7) can be expressed as

$$\begin{aligned}J_M^* = V_0(x_0) =& \bar{z}_0^T S_0 \bar{z}_0 + \mathrm{Tr}(S_0P_0) + \sum_{k=0}^{M-1}(\mathrm{Tr}((\Phi_d^T S_{k+1}\Phi_d\\ &+ W_k - S_k)\mathbf{E}_\gamma[P_{k|k}]) + \mathrm{Tr}(S_{k+1}R_w)) \quad (6.12)\end{aligned}$$

where S_k is the solution of the Riccati equation as defined in [25] and Tr denotes the trace of a square matrix. $\mathbf{E}_\gamma[\cdot]$ is the expectation with respect to the arrival sequence $\{\gamma_k\}$. The control input that minimizes the cost function of Eq (6.7) is

$$u_k = -(\Gamma_d^T S_{k+1}\Gamma_d + U_k)^{-1}\Gamma_d^T S_{k+1}\Phi_d\hat{z}_{k|k} = -L_k\hat{z}_{k|k}. \quad (6.13)$$

The expected value $\mathbf{E}_\gamma[P_{k|k}]$ is bounded by

$$\tilde{P}_{k|k} \le \mathbf{E}_\gamma[P_{k|k}] \le \hat{P}_{k|k}, \qquad \forall k \ge 0$$

where the matrices $\tilde{P}_{k|k}$ and $\hat{P}_{k|k}$ can be found in [25]. Then, it is possible to derive the bound of control cost given in Eq. (6.12). In the next section, we use two deterministic sequences $J_M^{\min}$ and $J_M^{\max}$, which bound the expected minimum cost as follows

$$\frac{1}{M}J_M^{\min} \le \frac{1}{M}J_M^* \le \frac{1}{M}J_M^{\max}, \quad (6.14)$$

and the two sequences converge to the following values:

$$\begin{aligned}J_\infty^{\max} =& \mathrm{Tr}((\Phi_d^T S_\infty \Phi_d + W_k - S_\infty)(\overline{P}_\infty - (1-p)\overline{P}_\infty C_d^T\\ &\times (C_d\overline{P}_\infty C_d^T + R_v)^{-1}C_d\overline{P}_\infty)) + \mathrm{Tr}(S_\infty R_w) \quad (6.15)\\ J_\infty^{\min} =& p\mathrm{Tr}((\Phi_d^T S_\infty \Phi_d + W_k - S_\infty)\underline{P}_\infty) + \mathrm{Tr}(S_\infty R_w) \quad (6.16)\end{aligned}$$

where,

$$\overline{P}_\infty = \Phi_d \overline{P}_\infty \Phi_d^T + R_w - (1-p)\Phi_d \overline{P}_\infty C_d^T (C_d \overline{P}_\infty C_d^T + R_v)^{-1} C_d \overline{P}_\infty \Phi_d^T$$
$$\underline{P}_\infty = p\Phi_d \underline{P}_\infty \Phi_d^T + R_w \,.$$

We remark that Eqs. (6.15) and (6.16) are explicit functions of the sampling period h, packet loss probability p, and delay d. The finite horizon cost and the cost bounds of the infinite horizon case will be used as the performance indicators in Section 6.5.1.

6.5 Co-Design Framework

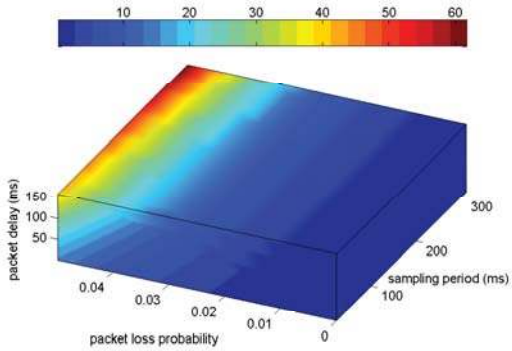
In this section, we first show the feasible control performance by taking into account realistic simulation results. Then, we study the co-design of the wireless NCS.

6.5.1 Effects of Wireless Network

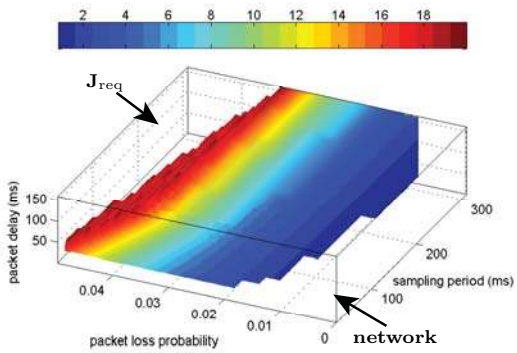
In this section, we discuss the fundamental issues of co-design of communication network and controller for wireless NCSs. The control cost (6.15) is considered as a performance indicator of the control system as described in Section 6.4. As an example we consider an unstable second-order plant in the form of (6.1) with

$$A = \begin{pmatrix} 3 & 1 \\ 0 & 1 \end{pmatrix},\ B = \begin{pmatrix} 0 \\ 1 \end{pmatrix},\ C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\ P_0 = 0.01I$$
$$W = I,\ M = 0,\ U = 0.01,\ R_w = I,\ R_v = 0.01I,$$

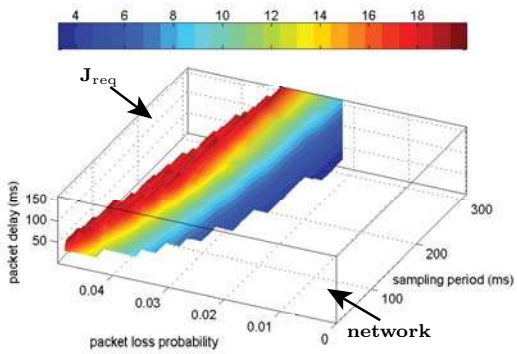
where W, M, U are assumed to be time-invariant in Eq. (6.7).
Figure 6.2 shows the feasible control cost with respect to different sampling periods, packet loss probabilities, packet delays with the simplified case and the realistic wireless networks for the different number of nodes $N = 10, 20$. Note that the simplified case does not explicitly consider the realistic network behavior i.e., independent relationship between sampling period, packet loss probability, and packet delay. In the figures, the colors show the feasible control cost. Figure 6.2(a) depicts the simplified case where longer sampling periods increase the control cost. Furthermore, we observe that packet losses at a higher sampling period are more critical than packet losses at a lower sampling period, indicating that we are sampling in a conservative way. Similarly, we derive the effects of packet delay on the control cost. Figures 6.2(b) and 6.2(c) depict the feasible region for $N = 10$ and 20 nodes, respectively. Note that we set the desired control cost $J_{\text{req}} = 20$. A point is feasible if it satisfies a given required cost, packet loss probability and delay for each sampling period. The feasible region is the set of all feasible points. In the figure, the transparent region denotes that the desired control cost is not feasible. It is natural that as the control requirement becomes strict, the infeasible region increases,



(a) Feasible control cost for the simplified case.



(b) Feasible control cost with $N = 10$.



(c) Feasible control cost with $N = 20$.

Figure 6.2: Feasible control cost over different sampling periods, packet loss probabilities, and packet delays. The colors show the control cost. Note that the scales of color bar are different in the figures.

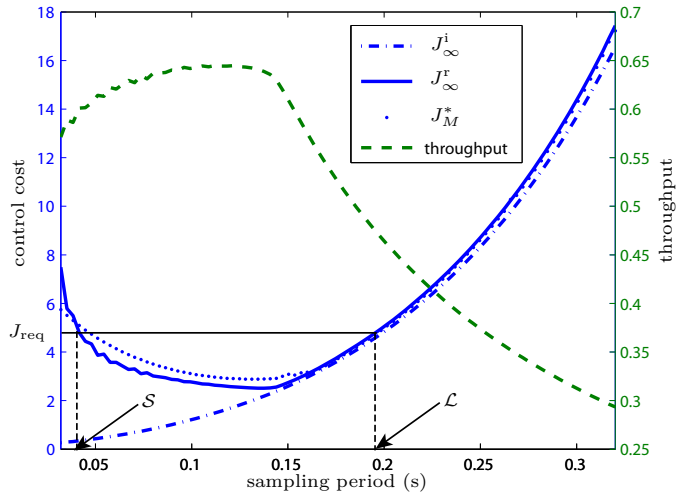


Figure 6.3: Control cost and throughput of the wireless network over different sampling periods. J^{i}_{∞} and J^{r}_{∞} refer to the cost bound $J^{\max}_{\infty}$ of the infinite horizon control cost given in (6.15) with ideal case and realistic model in [189], respectively. J^{*}_{M} denotes the finite horizon control cost given in (6.12).

since it also requires lower packet loss probability and delay of the network for lower sampling periods. Observe in Figures 6.2(b) and 6.2(c) that the packet loss probability $p \leq 0.01$ is not feasible when the sampling period is short $h \leq 0.03\,\mathrm{s}$. Since short sampling periods increase the traffic load of the network, the packet loss probability is closer to the critical packet loss probability, above which the system is unstable. Hence, it is difficult to achieve a low packet loss probability when the sampling period is short. Furthermore, by comparing Figures 6.2(b) and 6.2(c), we see that the infeasible region increases as the number of nodes increases. We remark that the infeasible region due to the wireless network starts from the origin where the sampling period $h = 0$, no packet loss $p = 0$, and no packet delay $d = 0$. No matter what communication protocol is used, the origin belongs to the infeasible region. The area and shape of the infeasible region depends on the communication protocol.

Figure 6.3 shows the control cost and communication throughput over different sampling periods. The throughput is the average rate of successful data transmission over a communication channel, which is the common objective for a communication designer. In the figure, J^{i}_{∞} and J^{r}_{∞} refers to the cost bound $J^{\max}_{\infty}$ given by Eq. (6.15) for the ideal (no packet loss and no delay) and realistic model in [189], respectively. Recall that J^{*}_{M} is the finite horizon control cost given by Eq. (6.12). The cost J^{*}_{M} follows the infinite horizon cost J^{r}_{∞} based on the realistic model. Due to the absence of packet losses and delays, the control performance when using an ideal network increases monotonically as the sampling period increases. However, when using a real network, a shorter sampling period does not minimize the control cost of the

control systems, because of the higher packet loss probability when the traffic load is high. In addition, the two curves of the cost J_∞^i and J_∞^r coincide for longer sampling periods, meaning that when the sampling period is larger, the sampling period is the dominant factor in the control cost compared to the packet loss probability and delay.

Now, let us discuss the throughput of the communication network and control cost of control systems. When we flip the throughput curve on the Y-axis, we observe a similar trend of behavior with the curve of control cost. Note that the closer the throughput is to 1, the better the utility of the wireless network. As the sampling period $h \in [0, 0.13]$ s increases, the control cost decreases and the throughput increases due to mainly high packet loss. For a longer sampling period $h > 0.15$ s, the performance of both the communication and control system degrades as the sampling period increases. The throughput decreases since the network is underutilized. We remark that the objective of both communication design and control design has a very similar trend. Hence, the optimal traffic load of the network is similar when either the communication throughput or control cost are optimized. Even though the dynamic interactions between these two objectives, throughput of the communication and control cost of control system, are critical factors for wireless NCSs, these issues are not well investigated in the previous literatures.

Let us consider a desired maximum control cost J_{req} greater than the minimum value of the control cost. Then, we have two feasible sampling periods $\mathcal{S}$ and $\mathcal{L}$ in Figure 6.3. However, the performance of the wireless network is still heavily affected by the choice of the sampling period of $\mathcal{S}$ and $\mathcal{L}$, as we discussed earlier. By choosing $\mathcal{L}$, the throughput of the network is stabilized (see details in [100]), hence, the control cost is also stabilized with respect to small perturbations of the network. Therefore, the wireless NCSs achieve good robustness for both communication and control perspective by choosing $\mathcal{L}$. Furthermore, a longer sampling period $\mathcal{L}$ leads to lower network energy consumption than the shorter sampling period $\mathcal{S}$ in [189]. Recall that the energy efficiency is one of the most critical issues for sensor nodes due to their limited battery power. This motivates our co-design approach of NCSs running over WSNs.

6.5.2 Design Procedure

We remind that the problem we consider in this chapter is how to determine the optimal sampling period h^* of control systems and the protocol parameters $\mathbf{V}^*$ of the communication protocol of an optimization problem given by Eq. (6.5). Figure 6.4 shows the proposed design flow that each control loop of the network follows. The application designer provides the parameters of network setup δ and the desired maximum control cost J_{req}. δ includes the important factors for modeling the wireless network such as a network topology, length of the packets, and the number of nodes (step 1). It is also possible that each control loop has a different desired maximum control cost J_{req}. The control designer then computes, off-line, an estimator (6.8)–(6.11) and a state feedback (6.13) according to Section 6.4 for

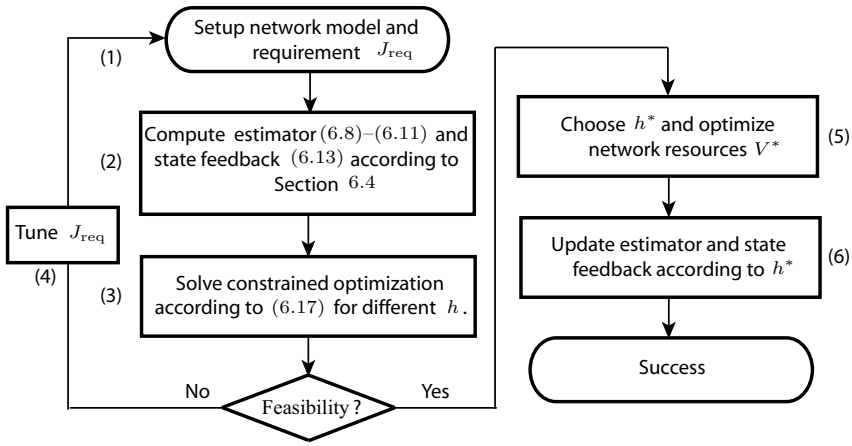


Figure 6.4: Flow diagram of co-design framework.

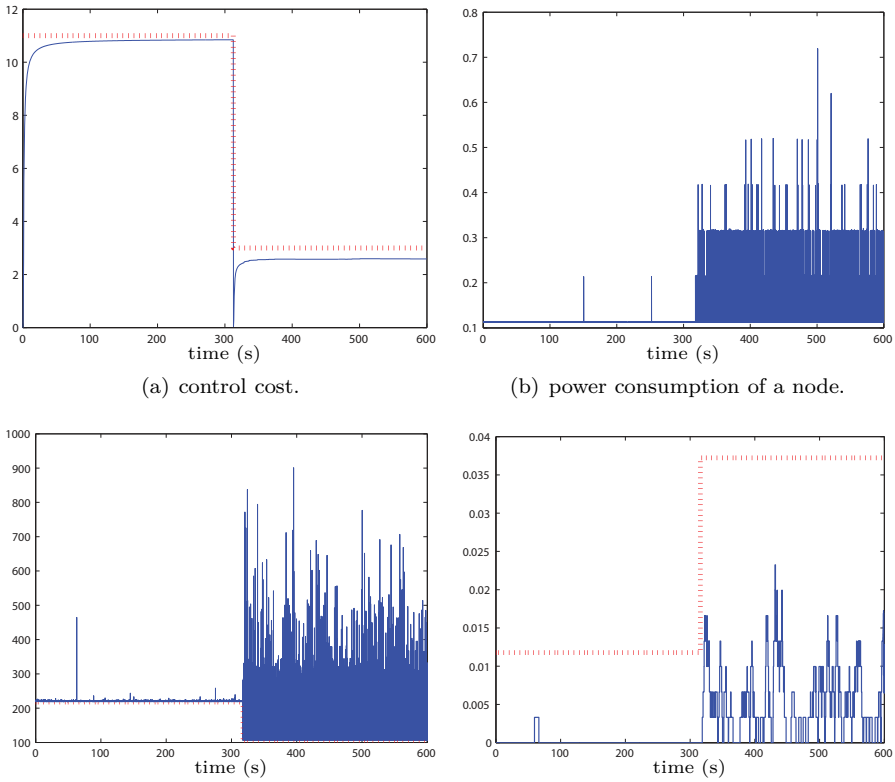
different sampling periods, packet loss probabilities, and delays (step 2). The network manager formulates and solves a constrained optimization problem, whereby the objective function is the energy consumption of the network and the constraints are the packet loss probability and delay, which are derived from J_{req} for different sampling periods (step 3). More precisely, the constrained optimization problem is formulated from (6.5) for a given sampling period h as follows

$$\min_{\mathbf{V}} \quad E_{\text{tot}}(h, \mathbf{V}, \delta) \tag{6.17a}$$

$$\text{s.t.} \quad p(h, \mathbf{V}, \delta) \leq P_{\max}\,, \tag{6.17b}$$

$$d(h, \mathbf{V}, \delta) \leq D_{\max}\,. \tag{6.17c}$$

The decision variables are the communication protocol parameters $\mathbf{V}$ depending on the network designer. The adaptive IEEE 802.15.4 protocol of Chapter 3 is applied to meet the requirements for packet loss probability and packet delay for a given sampling period. One can find a sub-optimal solution using the steps described in [217]. The network manager finds the local optimal MAC parameters $\mathbf{V}^*(h, P_{\max}, D_{\max})$ of a sub-optimization problem for a given $h, P_{\max}, D_{\max}$. Then, the optimal solution $h^*, \mathbf{V}^*$ is given by the pair h, $\mathbf{V}$ that minimizes the cost function if there are feasible solutions (step 5). Otherwise, the control designer needs to tune J_{req} since the desired control cost is not realistic (step 4). The network manager adapts the optimal sampling period h^* and the optimal protocol parameters V^* of the network (step 5). The control designer updates the estimator and the state feedback according to the optimized $h^*, p(h^*, \mathbf{V}^*, \delta), d(h^*, \mathbf{V}^*, \delta)$ (step 6).



(a) control cost. (b) power consumption of a node.

(c) interval of packet generated time. (d) packet loss probability.

Figure 6.5: Optimized control cost, power consumption of the network, interval of packet generated time, and packet loss probability of the proposed co-design approach with $N = 20$ nodes when the control requirement changes from $J_{\text{req}} = 11$ to $J_{\text{req}} = 3$ at 315 s. The particular realization is shown out of $N = 20$ nodes. The dotted line shows the requirement change of each figures.

6.6 Illustrative Example

In this section, we illustrate the proposed co-design procedure described in Section 6.5.2 through numerical examples. Figure 6.5 shows the adaptation of the requirements in terms of the sampling period, and packet loss probability of the network when the control requirement changes from $J_{\text{req}} = 11$ to $J_{\text{req}} = 3$ at 315 s. The optimal parameters $h^*, P_{\max}, D_{\max}$ are 214.4 ms, 0.012, 74.9 ms before control requirement changes, respectively. Figures 6.5(c), and 6.5(d) show that the adaptive communication protocol satisfies the requirements of h and $P_{\max}$, respectively. Note that the proposed protocol also meets the requirement of the packet delay. The high jitter of Figure 6.5(c) is mainly due to the packet loss of Figure 6.5(d). After the control requirement changes at time 315 s, the optimal parameters $h, P_{\max}, D_{\max}$

adapt to 102.4 ms, 0.037, 97.4 ms, respectively. We remark that although the requirements of packet loss probability and packet delay are less strict after the requirement changes, the sampling period decreases to meet the requirement $J_{\text{req}} = 3$. Recall that as the sampling period decreases, the packet loss probability and packet delay increase. Observe that the control cost is satisfied and the convergence of the algorithm is very fast. By comparing Figures 6.5(a) and 6.5(b), the tradeoff between the control cost and power consumption of the network is clearly observed.

6.7 Summary

The dynamic interactions between communication network and control system are critical factors to guarantee the stability of wireless NCS. In this chapter, the design framework of the WSNs is shown to be applicable to control applications. We first present how the wireless network affects the performance of NCSs by showing the feasible region of the control performance. Furthermore, the optimal traffic load of the network is similar when either the communication throughput or control cost are optimized. By considering these results, we conclude that the sampling period significantly influences not only the control performance, and throughput and energy consumption of the network, but also the robustness of the wireless NCS. A co-design between communication and control application layers is proposed for multiple control systems over the IEEE 802.15.4 wireless network. In particular, a constrained optimization problem is studied, where the objective function is the energy consumption of the network and the constraints are the packet loss probability and delay, which are derived from the desired control cost. Numerical results illustrate the efficiency of the proposed co-design approach.

Chapter 7

Conclusions and Future Work

The main contribution of this thesis is to provide a modeling, analysis, and design framework of WSN protocols for control applications. We used an analytical model-based protocol design to minimize the energy consumption of the network, while meeting the reliability and packet delay requirements of control applications. The main idea is to apply the tradeoff between the application requirements and energy consumption of the network, instead of just improving the reliability, delay or energy efficiency. In the design process, the original contribution is the derivation of analytical expressions of the energy consumption of the network, as well as reliability and delay for the packet delivery. This seems suitable for many control applications as they provide stability and performance guarantees. In particular, the contributions of this thesis are presented in four chapters. First, an adaptive IEEE 802.15.4 protocol to support energy efficient, reliable and timely communications by tuning the MAC parameters of the CSMA/CA algorithm is presented. Second, we proposed the modeling and analysis of the hybrid MAC protocol of the IEEE 802.15.4 standard based on a Markov chain model. Third, the novel protocol Breath is proposed by considering the analytical modeling of randomized routing, MAC, and duty-cycling. Eventually, the design framework of the WSNs is shown to be applicable to control applications. In this chapter, we briefly summarize each chapter of the thesis and discuss potential directions of future research.

Summary

In Chapter 3, an adaptive IEEE 802.15.4 protocol is proposed to support energy efficient, reliable and timely communications by tuning the MAC parameters. The protocol design scheme is grounded on a constrained optimization problem where the objective function is the power consumption of the network, subject to reliability and delay constraints on the packet delivery. A generalized Markov chain is proposed to model these relations by simple expressions without giving up the accuracy. The model is then used to derive an adaptive algorithm for minimizing the power consumption while guaranteeing reliability and delay constraints in the

packet transmission. The algorithm does not require any modification of the IEEE 802.15.4 standard and can be easily implemented on network nodes. An experimentally implementation and evaluation of the protocol on a test-bed with off-the-shelf wireless sensor nodes shows that the analysis is accurate, the proposed algorithm satisfies reliability and delay constraints, and the approach ensures a longer lifetime of the network under both stationary and transient network conditions.

Chapter 4 presented the modeling and analysis of the hybrid MAC of IEEE 802.15.4 combining the advantages of a CSMA/CA with contention with a TDMA without contention. Markov chains are used to model the contention access scheme and the behavior of the TDMA access scheme of the IEEE 802.15.4 protocol, validated by both theoretical analysis and Monte Carlo simulations. By using this new model, the network performance in terms of reliability, average packet delay, average queueing delay, and throughput is evaluated. It is also shown that the performance of the hybrid MAC differs significantly from what is reported previously in the literature. Furthermore, it is concluded that the tradeoff between throughput of the random access and the TDMA scheme for a fixed-length superframe is critical to maximize the throughput of the hybrid MAC.

Chapter 5 presented the novel protocol Breath based on randomized routing, MAC, and duty-cycling jointly optimized for energy efficiency. Breath is designed for WSNs where nodes must transmit information via multi-hop routing to a sink. Breath ensured a desired packet delivery and delay probability while minimizing the energy consumption of the network. The design approach relied on a constrained optimization problem, whereby the objective function is the energy consumption and the constraints are the packet reliability and delay. The challenging part is the modeling of the interactions among the layers by simple expressions of adequate accuracy, which are then used for the optimization by in-network processing. The optimal working point of the protocol is achieved by a simple algorithm, which adapts to traffic variations and channel conditions with negligible overhead. The protocol is implemented and experimentally evaluated on a test-bed with off-the-shelf wireless sensor nodes, and it has been compared with a standard IEEE 802.15.4 solution. Analytical and experimental results showed that Breath is tunable and meets reliability and delay requirements. Breath exhibited a nearly uniform distribution of the working load, thus extending network lifetime.

Chapter 6 presented how the adaptive protocol can be used in control applications. A co-design between communication and control application layers is studied by considering a constrained optimization problem presented in Chapter 3, for which the objective function is the energy consumption of the network and the constraints are the reliability and delay derived from the desired control cost. Furthermore, we investigated the essential issues of wireless networked control system by showing the feasible region of the control performance. In this way, we observed the tradeoff between communication and control performance. It is also shown that the optimal WSN traffic load is similar when either the communication throughput or control cost are optimized.

Future Work

Existing WSN research tends to focus on protocol design for specific applications and platforms, leading to a large number of protocols, mainly for monitoring applications. However, many WSNs simultaneously need to share a common communication infrastructure. Therefore, the WSN protocol needs to support not only heterogeneous platforms but also heterogeneous applications. Note that as the heterogeneity increases, developing individual protocols will become exceedingly complex and expensive. For instance, in building automation, the communication infrastructure covers all aspects of building system control including measurement and control messages of HVAC and lighting control, and alarm messages of security systems. The various applications may set different requirements on the performance of the communication infrastructure. An important question to answer is what combinations of MAC and routing protocols give good performance for applications and what parameters must be shared among different layers to improve the performance of the system. By considering the classifications in the protocol overview of Section 2.1, it is natural that the contention-based MAC protocol supports the topology-based and data-centric routing protocols and some of location-based routing protocols. However, it is not clear if and how to combine other MAC and routing protocols under other categories. One of the main reasons is that many MAC and routing protocols for WSNs cover more than the basic functionalities of its layer. Most schedule-based MAC protocols of WSNs support the packet forwarding mechanism for many-to-one communication and for even mesh communication [99]. For instance, the IEEE 802.15.4e task group suggests to use the routing protocol as the option when the packet forwarding mechanism fails. Similarly, many hierarchical-based routing protocol includes some mechanisms of MAC layer as well as routing mechanism. In particularly, the basic idea of the IEEE 802.15.4 standard [8] is similar to cluster-based protocols such as LEACH [77]. Therefore, it is not straightforward to propose good combination of MAC and routing protocols, but this problem needs to be carefully tackled in the future.

Hybrid MAC protocols are attractive since they try to take advantages of both contention-based and schedule-based communication. These protocols may support a variety of requirements for various applications. However, the critical question is how to combine the contention-based and schedule-based approach to achieve good performance. As discussed in Section 2.1.1, there are two main approaches: reservation-for-contention and partition approaches. For instance, the IEEE 802.15.4 MAC, which falls in the partition approaches, is very flexible as it allows to tune the portion of contention-based and schedule-based mechanisms depending on the application requirements. One of the interesting questions related to IEEE 802.15.4 is the feasibility and limitation of current standard for control applications. The performance of reservation-for-contention MAC protocols is still an important open problem. In addition to theoretical studies, the practical implementation of IEEE 802.15.4 standard is currently under development [218].

A generalized analytical model of the IEEE 802.15.4 standard is essential to provide

a insight understanding of the protocol. As part this work, an analytical model of single-hop network is being extended to multi-hoc network in [219].
The stability analysis of various protocols and distributed algorithm for WSNs is an interesting issue. In particular, we are working on the stability analysis of the adaptive tuning algorithms of Chapter 3.

Appendix A

Notation

A.1 Symbols

Table A.1: Main symbols used in the thesis. The italic names on the right side of the table are defined in the IEEE 802.15.4 standard [8].

Symbol	Meaning
BO	Beacon order
$D_{\max}$	Desired maximum average delay
$\mathbb{E}[D]$	Average delay for successfully received packets of CSMA/CA mechanism
$\mathbb{E}[D_q(t)]$	Average queueing delay of the GTS allocation mechanism at superframe t
E_b	Energy consumption for a beacon transmission
E_{ca}	Average energy consumption for CSMA/CA
E_m	Energy consumption for a packet transmission
E_{tot}	Total energy consumption of a node
$E_{\text{N,tot}}$	Total energy consumption of the network
E_{pck}	Average energy consumption for data packet transmission
E_r	Average energy consumption for receiving a data packet
E_{wu}	Average energy consumption for wake-up and beconing
$\mathbb{E}[\psi_s]$	Average service time for successfully received packet
$\mathbb{E}[\psi_{dc}]$	Average service time for discarded packet due to channel access failure
$\mathbb{E}[\psi_{dr}]$	Average service time for discarded packet due to retry limits
h	Length of idle-queue state without generating packets
R	Packet delivery rate, reliability
$R_{\min}$	Minimum desired probability for successful packet delivery
SO	Superframe order
S_{tot}	Distance from the edge cluster to the sink
L_c	Packet collision time with ACK mechanism
L_g	Successful packet transmission time without ACK
L_p	Total packet length including overhead and payload
L_s	Successful packet transmission time with ACK
m	Maximum number of backoffs, $macMaxCSMABackoffs$

Continued on next page

Table A.1 Continued from previous page

Symbol	Meaning
m_0	Minimum value of backoff exponent, $macMinBE$
m_b	Maximum value of backoff exponent, $macMaxBE$
n	Maximum number of retries, $macMaxFrameRetries$
n_h	Number of hops from edge cluster to sink
N	Total number of nodes of the network
N_{CAP}	Average number of successfully received non time-critical packets
N_{CFP}	Average number of successfully received GTS requests
N_{DPT}	Number of superframes in which a GTS descriptor exists, $aGTSDescPersistenceTime$
N_{GTS}	Maximum number of GTS descriptors
N_{SD}	Average number of successfully received any packets
N_{SS}	Number of slots contained in a superframe
P_c	Packet collision probability
P_{dc}	Probability that the packet is discarded due to channel access failure
P_{dr}	Probability of a packet being discarded due to retry limits
P_i	Power consumption in idle-listen state
P_r	Power consumption in receiving state
P_{sc}	Power consumption in channel sensing state
P_{sp}	Power consumption in sleep state
P_t	Power consumption in transmit state
P_w	Power consumption in wake-up state
$p_{\min}$	Minimum successful packet reception probability of physical layer for a single-hop transmission
T	Total experimental time
T_{ac}	Maximum listening time to receive a data packet
T_b	Basic time unit, $aUnitBackoffPeriod$
T_{BI}	Length of beacon interval
T_{CAP}	CAP length
T_{CFP}	CFP length
T_{SD}	Length of superframe duration
T_{SP}	Length of inactive period
T_{SS}	Length of superframe slot
T_w	Wake-up time from sleep mode
T_0	Length of superframe when SO = 0
$T_{\min}$	Minimum CAP length, $aMinCAPLength$
$T^{\star}_{\mathrm{CAP}}$	Fixed point of the CAP length
$T^{\star}_{\mathrm{CFP}}$	Fixed point of the CFP length
W	Maximum number of random backoff time
α	Busy channel probability of CCA_1
β	Busy channel probability of CCA_2
Δ_u	Maximum number of GTSs to be allocated to nodes
ϵ_i	Exponentially distributed time of intensity μ_c
ε_i	Approximation for uniformly distributed backoff time of CSMA/CA
η_d	Probability to generate a non time-critical data packet
η_p	Packet generation probability when the sampling interval is expired

Continued on next page

Table A.1 Continued from previous page

Symbol	Meaning
η_s	Total packet generation rate of periodic intervals
η_t	Packet generation probability after the node sends a packet successfully or discard a packet
Θ	Normalized throughput of the network
Ω	Required delay probability
$\underline{\theta}$	Minimum number of superframe slots for a singe GTS
λ_i	Probability of i successful requests during the CAP
μ_c	Cumulative wake-up rate of a cluster
π_k^t	Probability of state k at time t of Markov chain for the GTS allocation mechanism
ρ_t	Probability of deferred attempts due to the lack of the transmission time in a CAP
ρ_b	Probability of deferred attempts due to the lack of the backoff counter time in a CAP
τ	Channel sensing probability
φ_n	Number of time-critical data packets for each GTS request
$\psi_{sc}(n)$	Probability of successful transmission for CSMA/CA
$\psi_{sb}(\mu_c, n)$	Contention probability of n nodes for $1/\eta_s$ s

A.2 Acronyms

Table A.2: Main acronyms used in the thesis.

Acronym	Meaning
ACK	ACKnowledgement
AIMD	Additive Increase and Multiplicative Decrease
AWGN	Additive White Gaussian Noise
CAP	Contention Access Period
CCA	Clear Channel Assessment
CFP	Contention Free Period
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
FCFS	First Come First Served
GTS	Guaranteed Time Slot
LQG	Linear Quadratic Gaussian
MAC	Medium Access Control
NCS	Networked control system
PAN	Personal Area Network
PDF	Probability Density Function
PHY	PHYsical layer
RSSI	Received Signal Strength Indicator
RTS/CTS	request to send/clear to send
SINR	Signal to Interference plus Noise Ratio
TDMA	Time Division Multiple Access
WSN	Wireless Sensor Network

Appendix B

Proof of Chapter 3

B.1 Proof of Lemma 1

The proof has two steps. First, we derive the state transition probability of Markov chain. Second, the normalization condition is applied to compute the probability $b_{0,0,0}$.
The state transition probabilities associated with the Markov chain of Figure 3.1 are

$$P(i,k,j|i,k+1,j) = 1, \text{ for } k \geq 0\,, \tag{B.1}$$

$$P(i,k,j|i-1,0,j) = \frac{\alpha + (1-\alpha)\beta}{W_i}, \text{ for } i \leq m\,, \tag{B.2}$$

$$P(0,k,j|i,0,j-1) = \frac{(1-\alpha)(1-\beta)P_c}{W_0}, \text{ for } j \leq n\,, \tag{B.3}$$

$$P(Q_0|m,0,j) = (1-\eta)\,(\alpha + (1-\alpha)\beta), \text{ for } j < n\,, \tag{B.4}$$

$$P(Q_0|i,0,n) = (1-\eta)\,(1-\alpha)(1-\beta), \text{ for } i < m\,, \tag{B.5}$$

$$P(Q_0|m,0,n) = (1-\eta), \tag{B.6}$$

$$P(0,k,0|Q_0) = \frac{\eta}{W_0}, \text{ for } k \leq W_0 - 1\,. \tag{B.7}$$

Eq. (B.1) is the decrement of backoff counter, which happens with probability 1. Eq. (B.2) represents the probability of finding busy channel in CCA_1 or CCA_2 and a node selects uniformly a state in the next backoff stage. Eq. (B.3) gives the unsuccessful transmission probability after finding an idle channel in both CCA_1 and CCA_2, and a node picks uniformly a state in the next retransmission stage. Eq. (B.4) and (B.5) represent the probability of going back to the idle-queue stage due to the channel access failure and retry limits, respectively. Eq. (B.6) accounts for the traffic regime and is the probability of going back to the idle-queue stage at backoff counter m and retransmission stage n, which is given by $1-\eta$. Eq. (B.7) models the probability of going back to the first backoff stage from the idle-queue

stage. Owing to the chain regularities and Eqs. (B.1)–(B.7), we have Eqs. (3.4)–(3.6).
By the normalization condition, we know that

$$\sum_{i=0}^{m}\sum_{k=0}^{W_i-1}\sum_{j=0}^{n} b_{i,k,j} + \sum_{i=0}^{m}\sum_{j=0}^{n} b_{i,-1,j} + \sum_{j=0}^{n}\left(\sum_{k=0}^{L_s-1} b_{-1,k,j} + \sum_{k=0}^{L_c-1} b_{-2,k,j}\right) + \sum_{l=0}^{h-1} Q_l = 1\,. \tag{B.8}$$

We next derive the expressions of each term in Eq. (B.8).
From Eqs. (3.3) and (3.4), we have

$$\sum_{i=0}^{m}\sum_{k=0}^{W_i-1}\sum_{j=0}^{n} b_{i,k,j} = \sum_{i=0}^{m}\sum_{j=0}^{n} \frac{W_i+1}{2}(\alpha+(1-\alpha)\beta)^i\, b_{0,0,j} \tag{B.9}$$

$$= \begin{cases} \frac{b_{0,0,0}}{2}\left(\frac{1-(2x)^{m+1}}{1-2x}W_0 + \frac{1-x^{m+1}}{1-x}\right)\frac{1-y^{n+1}}{1-y} \\ \quad \text{if } m \le m_b - m_0 \\ \\ \frac{b_{0,0,0}}{2}\left(\frac{1-(2x)^{m_b-m_0+1}}{1-2x}W_0 + \frac{1-x^{m_b-m_0+1}}{1-x} + \right. \\ \left.(2^{m_b}+1)x^{m_b-m_0+1}\frac{1-x^{m-m_b+m_0}}{1-x}\right)\frac{1-y^{n+1}}{1-y} \\ \quad \text{otherwise,} \end{cases}$$

where $x = \alpha + (1-\alpha)\beta$ and $y = P_c(1-x^{m+1})$. Similarly,

$$\sum_{i=0}^{m}\sum_{j=0}^{n} b_{i,-1,j} = \sum_{i=0}^{m}\sum_{j=0}^{n}(1-\alpha)(\alpha+(1-\alpha)\beta)^i\, b_{0,0,j}$$

$$= (1-\alpha)\frac{1-x^{m+1}}{1-x}\,\frac{1-y^{n+1}}{1-y}\, b_{0,0,0}\,, \tag{B.10}$$

and

$$\sum_{j=0}^{n}\left(\sum_{k=0}^{L_s-1} b_{-1,k,j} + \sum_{k=0}^{L_c-1} b_{-2,k,j}\right) \tag{B.11}$$

$$= (L_s(1-P_c) + L_cP_c)(1-x^{m+1})\frac{1-y^{n+1}}{1-y}\, b_{0,0,0}\,.$$

By considering that the successful transmission and the failure events are due to the limited number of backoff stages m and the retry limit n, the idle state probability

is

$$
\begin{aligned}
Q_0 =&(1-\eta)\,Q_{h-1} + (1-\eta)\left(\sum_{j=0}^{n}(\alpha+(1-\alpha)\beta)\;b_{m,0,j} + \sum_{i=0}^{m}P_c(1-\beta)\,b_{i,-1,n}\right.\\
&\left.+\sum_{i=0}^{m}\sum_{j=0}^{n}(1-P_c)\,(1-\beta)\,b_{i,-1,j}\right)\\
=&\frac{1-\eta}{\eta}\left(\frac{x^{m+1}(1-y^{n+1})}{1-y} + P_c(1-x^{m+1})y^n\right.\\
&\left.+(1-P_c)\frac{(1-x^{m+1})(1-y^{n+1})}{1-y}\right)b_{0,0,0}\,,
\end{aligned}
\tag{B.12}
$$

where h is the idle state length without generating packets and $\sum_{l=0}^{h-1} Q_l = hQ_0$. Note that Eqs. (B.9)–(B.12) give the state values $b_{i,k,j}$ as a function of $b_{0,0,0}$. By replacing Eqs. (B.9)–(B.12) in the normalization condition given by Eq. (B.8), we obtain the expression for $b_{0,0,0}$.

Appendix C

Proofs of Chapter 4

C.1 Proof of Proposition 2

We follow two steps to compute the stationary probability. First, we derive the state transition probability of Markov chain. Second, a normalization condition is applied to compute the probability $S_{0,0,0,0}$.
First, the state transition probabilities associated to the Markov chain of Figure 4.1 are

$$P(i,j,k,0|i,j+1,k,0) = 1-\rho_b, \text{ for } k \geq 0\,, \tag{C.1}$$
$$P(i,j,k,1|i,j+1,k,1) = 1, \text{ for } k \geq 0\,, \tag{C.2}$$
$$P(i,j,k,0|i-1,0,k,0) = \frac{(1-\rho_t)(\alpha+(1-\alpha)\beta)}{W_i}, \text{ for } i \leq m\,, \tag{C.3}$$
$$P(0,j,k,0|i,0,k-1,0) = \frac{P_c(1-\rho_t)(1-\alpha)(1-\beta)}{W_0}, \text{ for } k \leq n\,, \tag{C.4}$$
$$P(0,j,k,1|i,0,k-1,1) = \frac{P_c(1-\alpha)(1-\beta)}{W_0}, \text{ for } k \leq n\,, \tag{C.5}$$
$$P(Q_0|m,0,k,0) = (1-\eta_t)(1-\rho_t)(\alpha+(1-\alpha)\beta), \text{ for } k < n\,, \tag{C.6}$$
$$P(Q_0|m,0,k,1) = (1-\eta_t)(\alpha+(1-\alpha)\beta), \text{ for } k < n\,, \tag{C.7}$$
$$P(Q_0|i,0,n,0) = (1-\eta_t)(1-\rho_t)(1-\alpha)(1-\beta), \text{ for } i < m\,, \tag{C.8}$$
$$P(Q_0|i,0,n,1) = (1-\eta_t)(1-\alpha)(1-\beta), \text{ for } i < m\,, \tag{C.9}$$
$$P(0,j,0,0|Q_0) = \frac{\eta_p}{W_0}, \text{ for } j \leq W_0 - 1\,. \tag{C.10}$$

Eq. (C.1) corresponds to the decrement of backoff counter of non-deferred transmission, which happens with probability $1-\rho_b$. Eq. (C.2) models the decrement of backoff counter after the event of deferred attempts, which happens with probability 1. Recall that we assume that the maximum number of deferred attempts of a single packet transmission is 1 in Section 4.3. Eq. (C.3) represents the probability of selecting uniformly a state in the next backoff stage of non-deferred transmis-

sion after finding busy channel in CCA_1 or CCA_2. Eqs. (C.4) and (C.5) give the probability of picking uniformly a state in the next retransmission stage after the packet collision of non-deferred and deferred transmission, respectively. Eqs. (C.6) and (C.7) represent the probability of going back to the idle-queue stage due to the channel access failure of non-deferred and deferred transmission, respectively. Eqs. (C.8) and (C.9) give the probability of going back to the idle-queue stage due to the retry limits of non-deferred and deferred transmission, respectively. Finally, Eq. (C.10) models the probability of going back to the first backoff stage from the idle-queue stage.

Next, we derive the closed form expression for the chain of Figure 4.1. Let us first consider the stationary probability of non-deferred transmission $l = 0$. Owing to the chain regularities and Eqs. (C.1) and (C.3), we have

$$S_{i,0,k,0} = y_i \, S_{i-1,0,k,0} \, ,$$

where

$$y_i = \frac{(1-\rho_t)(\alpha + (1-\alpha)\beta)}{W_i} \sum_{j=0}^{W_i-1} (1-\rho_b)^j \, , \qquad i \geq 1 \, ,$$

and $y_0 = 1$. By using the product of y_i, the stationary probability $S_{i,0,k,0}$ is rewritten as follows

$$S_{i,0,k,0} = \xi_i \, S_{0,0,k,0} \, , \tag{C.11}$$

where $\xi_i = \prod_{r=1}^{i} y_r$ and $\xi_0 = 1$. From Eqs. (C.1) and (C.3), we have

$$S_{i,j,k,0} = \frac{1}{W_i} \left(1 + \sum_{r=j+1}^{W_i-1} (1-\rho_b)^{W_i - r} \right) (1-\rho_t) x \, S_{i-1,0,k,0} \, ,$$

where $x = \alpha + (1-\alpha)\beta$. Then, we obtain

$$S_{i,j,k,0} = \varpi_{i,j} S_{i,0,k,0} \, , \tag{C.12}$$

where

$$\varpi_{i,j} = \frac{1 - (1-\rho_b)^{W_i - j}}{1 - (1-\rho_b)^{W_i}} \, .$$

From Eqs. (C.4) and (C.11), we have

$$S_{i,0,k,0} = P_c (1-\rho_t)(1-\alpha)(1-\beta) \sum_{i=0}^{m} \sum_{j=0}^{W_0-1} \frac{(1-\rho_b)^j}{W_0} S_{i,0,k-1,0} = g^k S_{0,0,0,0} \, , \tag{C.13}$$

where

$$g = P_c(1-\rho_t)(1-\alpha)(1-\beta)\sum_{i=0}^{m}\xi_i \sum_{j=0}^{W_0-1}\frac{(1-\rho_b)^j}{W_0}\,.$$

Now, we characterize the stationary probability of the deferred transmission $l = 1$. From Eqs. (C.5) and (C.11)–(C.13), we derive

$$\begin{aligned}
S_{i,0,k,1} =&(\alpha+(1-\alpha)\beta)S_{i-1,0,k,1}u(i-1) + \rho_t S_{i,0,k,0} + \rho_b \sum_{j=1}^{W_i-1} S_{i,j,k,0} \\
&+ P_c(1-\alpha)(1-\beta)\sum_{i=0}^{m} S_{i,0,k-1,1}\delta(i)u(k-1) \\
=&xS_{i-1,0,k,1}u(i-1) + \left(\rho_t + \rho_b \sum_{j=1}^{W_i-1}\varpi_{i,j}\right)\xi_i\, g^k\, S_{0,0,0,0} \\
&+ P_c(1-x)\sum_{i=0}^{m} S_{i,0,k-1,1}\delta(i)u(k-1)\,,
\end{aligned}$$

$u(i)$ is the unit step function and $\delta(i)$ is the unit delta function. For $k = 0$, we obtain

$$S_{i,0,0,1} = \zeta_i S_{0,0,0,0}\,,$$

where

$$\begin{aligned}
\zeta_i &= x^i c + x^i \sum_{r=1}^{i}\frac{z_r}{x^r}\,, \\
z_i &= \rho_t + \rho_b \sum_{j=1}^{W_i-1}\varpi_{i,j}\,, \\
c &= \zeta_0 = \xi_0 = \rho_t + \rho_b \sum_{j=1}^{W_0-1}\varpi_{0,j}\,.
\end{aligned}$$

Analogously, we derive the following recursive formula:

$$S_{i,0,k,1} = v_{i,k} S_{0,0,0,0} \tag{C.14}$$

where

$$\begin{aligned}
v_{i,k} &= \zeta_i g^k + x^i a_k\,, \\
a_k &= P_c(1-x)\sum_{i=0}^{m} v_{i,k-1}, \qquad k \geq 1\,, \\
v_{0,0} &= \rho_t + \rho_b\,\zeta_0 \sum_{j=1}^{W_0-1}\varpi_{0,j}\,,
\end{aligned}$$

and $a_0 = 0$.
By putting together Eqs. (C.5) and (C.11)–(C.13), we obtain

$$\begin{aligned} S_{i,j,k,1} =& \rho_b S_{i,j,k,0} + \frac{\rho_t}{W_i} S_{i,0,k,0} + \sum_{r=j+1}^{W_i-1} S_{i,r,k,1} + \frac{\alpha + (1-\alpha)\beta}{W_i} S_{i-1,0,k,1} u(i-1) \\ &+ \frac{P_c(1-\alpha)(1-\beta)}{W_0} \sum_{i=0}^{m} S_{i,0,k-1,1} \delta(i) u(k-1) \\ =& \left[\sum_{r=j}^{W_i-1} \rho_b \varpi_{i,r} \xi_i g^k + \rho_t \frac{W_i - j}{W_i} \xi_i g^k + x \frac{W_i - j}{W_i} v_{i-1,k} u(i-1) \right. \\ &\left. + P_c(1-x) \frac{W_0 - j}{W_0} \sum_{i=0}^{m} v_{i,k-1} \delta(i) u(k-1) \right] S_{0,0,0,0} \,. \end{aligned} \tag{C.15}$$

By considering the state transition probability of the Markov chain, we apply the normalization condition to compute the state probability $S_{0,0,0,0}$. The normalization condition is

$$\eta_d S_\sigma(\rho_{t,d}, L_{p,d}) + (1-\eta_d) S_\sigma(\rho_{t,r}, L_{p,r}) = 1 \,, \tag{C.16}$$

where

$$\begin{aligned} S_\sigma(\rho_t, L_p) =& \sum_{i=0}^{m} \sum_{j=0}^{W_i-1} \sum_{k=0}^{n} \sum_{l=0}^{1} S_{i,j,k,l} + \sum_{i=0}^{m} \sum_{k=0}^{n} \sum_{l=0}^{1} S_{i,-1,k,l} \\ &+ \sum_{k=0}^{n} \left(\sum_{j=0}^{L_s-1} \sum_{l=0}^{1} S_{+*,j,k,l} + \sum_{j=0}^{L_c-1} \sum_{l=0}^{1} S_{-*,j,k,l} \right) + \sum_{i=0}^{h-1} Q_i \,, \end{aligned}$$

$*$ is 1 for a non time-critical data packet and 2 for a GTS request of a time-critical data packet and $\rho_{t,d}$ and $\rho_{t,r}$ are the probabilities of the event of deferred transmission for a non time-critical packet with packet length $L_{p,d}$ and a GTS request for time-critical packet with request length $L_{p,r}$, respectively. In Eq. (C.16), the first and second term are related to a non time-critical data packet and a GTS request of time-critical data packet, respectively. We next derive the expressions of each term in Eq. (C.16). From Eqs. (C.11)–(C.13), we have

$$\sum_{i=0}^{m} \sum_{j=0}^{W_i-1} \sum_{k=0}^{n} S_{i,j,k,0} = \sum_{i=0}^{m} \sum_{j=0}^{W_i-1} \sum_{k=0}^{n} \varpi_{i,j} \xi_i g^k = \frac{1-g^{n+1}}{1-g} \sum_{i=0}^{m} \xi_i \sum_{j=0}^{W_i-1} \varpi_{i,j} S_{0,0,0,0} \,. \tag{C.17}$$

Similarly, from Eqs. (C.11)–(C.15), we have

$$\begin{aligned} \sum_{i=0}^{m} \sum_{j=0}^{W_i-1} \sum_{k=0}^{n} S_{i,j,k,1} =& \left[\rho_b \sum_{k=0}^{n} g^k \sum_{i=0}^{m} \xi_i \sum_{j=0}^{W_i-1} \sum_{r=j}^{W_i-1} \varpi_{i,r} + \rho_t \sum_{k=0}^{n} g^k \sum_{i=0}^{m} \xi_i \sum_{j=0}^{W_i-1} \frac{W_i - j}{W_i} \right. \\ &+ x \sum_{i=0}^{m} \sum_{j=0}^{W_i-1} \sum_{k=0}^{n} \frac{W_i - j}{W_i} v_{i-1,k} u(i-1) + P_c(1-x) \sum_{j=0}^{W_0-1} \frac{W_0 - j}{W_0} \\ &\left. \times \sum_{i=0}^{m} \sum_{k=0}^{n} v_{i,k-1} \delta(i) u(k-1) \right] S_{0,0,0,0} \,. \end{aligned} \tag{C.18}$$

From Eqs. (C.11) and (C.13), it follows

$$\begin{aligned}\sum_{i=0}^{m}\sum_{k=0}^{n} S_{i,-1,k,0} =& \sum_{i=0}^{m}\sum_{k=0}^{n}(1-\rho_t)(1-\alpha)S_{i,0,k,0} \\ =&(1-\rho_t)(1-\alpha)\frac{1-g^{n+1}}{1-g}\sum_{i=0}^{m}\xi_i S_{0,0,0,0}\,. \end{aligned} \tag{C.19}$$

Similarly, from Eq. (C.14), we obtain

$$\sum_{i=0}^{m}\sum_{k=0}^{n} S_{i,-1,k,1} = \sum_{i=0}^{m}\sum_{k=0}^{n}(1-\alpha)S_{i,0,k,1} = (1-\alpha)\sum_{i=0}^{m}\sum_{k=0}^{n} v_{i,k}S_{0,0,0,0}\,. \tag{C.20}$$

Analogously, the packet transmission state follows:

$$\begin{aligned}&\sum_{k=0}^{n}\left(\sum_{j=0}^{L_s-1}\sum_{l=0}^{1} S_{+*,j,k,l} + \sum_{j=0}^{L_c-1}\sum_{l=0}^{1} S_{-*,j,k,l}\right) \\ &= \left(L_s(1-P_c)(1-\beta) + L_cP_c(1-\beta)\right)\sum_{i=0}^{m}\sum_{k=0}^{n}\sum_{l=0}^{1} S_{i,-1,k,l}\,,\end{aligned} \tag{C.21}$$

where the last triple sum is the sum of Eqs. (C.19) and (C.20). To derive the idle-queue state probability, we first compute the state probability of the failure events due to the limited number of backoff stages m and the retry limit n and the successful transmission. From Eqs. (C.6) and (C.7), the state probability of the failure events due to the limited number of backoff stages m is

$$\begin{aligned}&\sum_{k=0}^{n}(1-\rho_t)xS_{m,0,k,0} + \sum_{k=0}^{n} xS_{m,0,k,1} \\ &= \left((1-\rho_t)x\xi_m\frac{1-g^{n+1}}{1-g} + x\sum_{k=0}^{n} v_{m,k}\right)S_{0,0,0,0}\,.\end{aligned} \tag{C.22}$$

Similarly, from Eqs. (C.8), (C.9), (C.11) and (C.13), the state probability of the failure events due to the limited number of the retry limit n is

$$\begin{aligned}&\sum_{i=0}^{m}P_c(1-\beta)S_{i,-1,n,0} + P_c(1-\beta)S_{i,-1,n,1} \\ &\quad= \sum_{i=0}^{m}P_c(1-\rho_t)(1-\alpha)(1-\beta)S_{i,0,n,0} + P_c(1-\alpha)(1-\beta)S_{i,0,n,1} \\ &\quad= \left(P_c(1-\rho_t)(1-\alpha)(1-\beta)\sum_{i=0}^{m}\xi_i g^n + P_c(1-\alpha)(1-\beta)\sum_{i=0}^{m} v_{i,n}\right)S_{0,0,0,0}\,.\end{aligned} \tag{C.23}$$

The state probability of the successful transmission follows by summing Eqs. (C.19) and (C.20):

$$\sum_{i=0}^{m}\sum_{k=0}^{W_i-1}\sum_{l=0}^{1}(1-P_c)(1-\beta)S_{i,-1,k,l}\,. \tag{C.24}$$

By putting together Eqs. (C.22), (C.23) and (C.24), the idle-queue state probability is

$$Q_0 = \frac{1-\eta_t}{\eta_p}\left((1-\rho_t)x\xi_m\frac{1-g^{n+1}}{1-g} + x\sum_{k=0}^{n} v_{m,k} + P_c(1-\rho_t)(1-\alpha)(1-\beta)\sum_{i=0}^{m}\xi_i g^n \right.$$
$$+P_c(1-\alpha)(1-\beta)\sum_{i=0}^{m} v_{i,n} + (1-P_c)(1-\beta)\left((1-\rho_t)(1-\alpha)\frac{1-g^{n+1}}{1-g}\sum_{i=0}^{m}\xi_i\right.$$
$$\left.\left.+(1-\alpha)\sum_{i=0}^{m}\sum_{k=0}^{n} v_{i,k}\right)\right) S_{0,0,0,0}\,. \tag{C.25}$$

Hence, $\sum_{i=0}^{h-1} Q_i = hQ_0$ where h is the length of idle-queue state without generating packets as described in Section 4.3. Note that Eqs. (C.17)–(C.21) and (C.25) give the state values $S_{i,j,k,l}$ as a function of $S_{0,0,0,0}$. By replacing Eqs. (C.17)–(C.21) and (C.25) in the normalization condition given by Eq. (C.16), we obtain the expression for $S_{0,0,0,0}$.

C.2 Proof of Proposition 4

To compute the average delay, we need some intermediate technical steps. In particular, we characterize (a) the expected value of the backoff delay due to busy channel and (b) the expression of the delay due to collision after two successful CCAs. We first address the average backoff delay due to busy channel in the following.

Let d_i be the random time associated to the successful CCAs of a packet at the i-th backoff stage. Denote by $\mathscr{A}_i$ the event of two successful CCAs at time $i+1$ after i-th events of unsuccessful CCAs. Let $\mathscr{A}$ be the event of successful CCAs within the total attempts m. Then, the backoff delay for two successful CCAs after the i-th unsuccessful attempt is

$$d = \sum_{i=0}^{m} \mathbb{1}_{\mathscr{A}_i|\mathscr{A}}\, d_i\,,$$

where $\mathbb{1}_{\mathscr{A}_i|\mathscr{A}}$ is 1 if $\mathscr{A}_i|\mathscr{A}$ holds, and 0 otherwise.

By considering the deferred attempt, we divide the events of two successful CCAs at time $i+1$, given i previous unsuccessful CCAs, as follows: (a) the event of successful CCAs without any deferred attempts during $i+1$ attempts (b) the event of successful CCAs at the deferred attempt (c) the event of successful CCAs after the deferred attempts during i previous unsuccessful attempts before time $i+1$. We assume that the maximum number of deferred attempts for packet transmission is one, i.e., every node needs to transmit a packet within $2T_{\text{BI}}$. In the following, we derive the average backoff delay of these three events.

First, we determine the event of successful CCAs without any deferred attempts during $i+1$ attempts. There are two events which depend on the previous deferred attempt before the current $i+1$ attempts as follows: the event of successful CCAs

without any deferred attempts and with a deferred attempt before current $i+1$ attempts.
Let $\mathscr{B}_{0,i}$ be the event of successful CCAs without any deferred attempts before and during $i+1$ attempts with $i \in \{0,\dots,m\}$. The probability of such an event is

$$\Pr[\mathscr{B}_{0,i}] = \varepsilon_i \sum_{k=0}^{i} (1-\rho_t) \sum_{j=0}^{W_i-1} \frac{(1-\rho_b)^j}{W_i} (1-\alpha)(1-\beta)\,, \tag{C.26}$$

where

$$\varepsilon_i = \begin{cases} \prod_{r=0}^{i-1} \epsilon_r & \text{if } i \geq 1\,, \\ 1 & \text{otherwise}\,, \end{cases}$$

$$\epsilon_r = (1-\rho_t) \sum_{j=0}^{W_r-1} \frac{(1-\rho_b)^j}{W_r} \max\left(\alpha\,, (1-\alpha)\beta\right)\,. \tag{C.27}$$

The term ϵ_r gives an approximation for the probability (C.26) for analytical tractability. For an accurate model, see our previous work [189] in the manuscript. In Eq. (C.26), the first term ε_i models i previous unsuccessful attempts without any deferred attempts. Note that the terms $1-\rho_t$ and $1-\rho_b$ give the probability of non-deferred attempts due to the lack of the remaining slot times for packet transmission and of the remaining slot times during backoff time, respectively. $\mathscr{B}_0$ is the event of successful CCAs without any deferred attempts before and during the maximum $m+1$ times:

$$\Pr[\mathscr{B}_0] = \sum_{i=0}^{m} \Pr[\mathscr{B}_{0,i}]\,. \tag{C.28}$$

Consider the attempt of CCA of the i-th channel sensing. Then, the random CCA delay $d_{\mathscr{B}_0}$ within $m+1$ attempts can be described as

$$d_{\mathscr{B}_0} = \begin{cases} \Phi_0 + 2L_{\text{sc}}, & \text{if } \mathscr{B}_{0,0}|\mathscr{B}_0; \\ \Phi_0 + 2L_{\text{sc}} + \Phi_1 + 2L_{\text{sc}}, & \text{if } \mathscr{B}_{0,1}|\mathscr{B}_0; \\ \quad\vdots & \\ \sum_{k=0}^{m} \Phi_k + (m+1)2L_{\text{sc}}, & \text{if } \mathscr{B}_{0,m}|\mathscr{B}_0; \end{cases}$$

where Φ_k is the random backoff time at $k+1$-th attempts. $2L_{\text{sc}}$ is the successful sensing time and $i2L_{\text{sc}}$ is the unsuccessful sensing time due to busy channel during CCAs. Note that we consider the worst case, i.e., a failure of the second sensing (CCA_2), which implies that $L_{\text{sc}} = T_b$ and that each sensing failure takes $2L_{\text{sc}}$ in Eq. (C.26). Recall that a node transmits the packet when the backoff counter is 0 and two successful CCAs are detected, see [8] in the manuscript. We can rewrite $d_{\mathscr{B}_0}$ as

$$d_{\mathscr{B}_0} = \sum_{i=0}^{m} \left[\sum_{k=0}^{i} \Phi_k + (i+1)2L_{\text{sc}} \right] \mathbb{1}_{\mathscr{B}_{0,i}|\mathscr{B}_0}\,. \tag{C.29}$$

The expectation of Φ_k can be computed by recalling the uniform distribution of backoff time:

$$\mu_{\Phi_k} = \sum_{l=0}^{W_k-1} \frac{l}{W_k} T_b \,. \tag{C.30}$$

Now, it is possible to compute the average value of $d_{\mathscr{B}_0}$ as

$$\mu_{\mathscr{B}_0} = \sum_{i=0}^{m} \frac{\rho_{\mathscr{B}_{0,i}}}{\Pr[\mathscr{B}_0]} \,, \tag{C.31}$$

where

$$\rho_{\mathscr{B}_{0,i}} = \varepsilon_i \sum_{k=0}^{i} (1-\rho_t) \sum_{j=0}^{W_i-1} \frac{(1-\rho_b)^j}{W_i} (1-\alpha)(1-\beta) \left(\mu_{\Phi_k} + (i+1)2L_{\text{sc}}\right) \,,$$

and recall that the first term is given in Eq. (C.26). Note that the normalization comes by considering all possible events of successful attempts.
Let $\mathscr{B}_{1,i}$ be the event of successful CCAs without any event of deferred attempt during $i+1$ attempts, but with an event of deferred attempt before $i+1$ attempts. The probability of such an event is

$$\Pr[\mathscr{B}_{1,i}] = \sum_{k=0}^{i} \alpha^{i-k} ((1-\alpha)\beta)^k (1-\alpha)(1-\beta) \,. \tag{C.32}$$

With a similar way of Eq. (C.29), we can define $d_{\mathscr{B}_1}$ as

$$d_{\mathscr{B}_1} = \sum_{i=0}^{m} \left[\sum_{k=0}^{i} \Phi_k + (i+1)2L_{\text{sc}} \right] \mathbb{1}_{\mathscr{B}_{1,i}|\mathscr{B}_1} \,. \tag{C.33}$$

By using the expectation of Φ_k given in Eq. (C.30), the average value of $d_{\mathscr{B}_1}$ with the maximum number of times $m+1$ is

$$\mu_{\mathscr{B}_1} = \sum_{i=0}^{m} \frac{\rho_{\mathscr{B}_{1,i}}}{\Pr[\mathscr{B}_1]} \,, \tag{C.34}$$

where

$$\rho_{\mathscr{B}_{1,i}} = \sum_{k=0}^{i} \alpha^{i-k} ((1-\alpha)\beta)^k (1-\alpha)(1-\beta) \left(\mu_{\Phi_k} + (i+1)2L_{\text{sc}}\right) \,,$$
$$\Pr[\mathscr{B}_1] = \sum_{i=0}^{m} \Pr[\mathscr{B}_{1,i}] \,.$$

Here, we have considered the worst case of sensing time given by $2L_{\text{sc}}$.
Second, we derive the average backoff delay of the event of successful CCAs at the $i+1$-th attempt with an event of deferred attempt. We remind that an event of deferred attempt is due to two reasons: (i) lack of the remaining time slots for

packet transmission, which happens with the probability ρ_t, and (ii) lack of the remaining time slots during the backoff time, which happens with probability ρ_b. Let $\mathscr{C}_{0,i}$ be the event of successful CCAs after the event of deferred attempt due to the lack of the remaining slot times for packet transmission at the $i+1$-th attempt. $\mathscr{C}_0$ is the successful event of $\mathscr{C}_{0,i}$ during the maximum $m+1$ times. The probability of such an event $\mathscr{C}_{0,i}$ is

$$\Pr[\mathscr{C}_{0,i}] = \varepsilon_i \sum_{k=0}^{i} \rho_t \sum_{j=0}^{W_i-1} \frac{(1-\rho_b)^j}{W_i} (1-\alpha)(1-\beta)\,. \tag{C.35}$$

The random CCA delay within $m+1$ attempts of the event $\mathscr{C}_0$ is

$$d_{\mathscr{C}_0} = \sum_{i=0}^{m} \left[\sum_{k=0}^{i} \Phi_k + T_{\mathrm{SP}} + L_{\mathrm{tx}} + \Phi_i + (i+1)2L_{\mathrm{sc}} \right] \mathbb{1}_{\mathscr{C}_{0,i}|\mathscr{C}_0}\,. \tag{C.36}$$

The average value of $d_{\mathscr{C}_0}$ is

$$\mu_{\mathscr{C}_0} = \sum_{i=0}^{m} \frac{\rho_{\mathscr{C}_{0,i}}}{\Pr[\mathscr{C}_0]}\,, \tag{C.37}$$

where

$$\begin{aligned} \rho_{\mathscr{C}_{0,i}} =& \varepsilon_i \sum_{k=0}^{i} \rho_t \sum_{j=0}^{W_i-1} \frac{(1-\rho_b)^j}{W_i} (1-\alpha)(1-\beta) \\ & \times \left(\mu_{\Phi_k} + T_{\mathrm{SP}} + L_{\mathrm{tx}} + \mu_{\Phi_i} + (i+1)2L_{\mathrm{sc}}\right)\,, \\ \Pr[\mathscr{C}_0] =& \sum_{i=0}^{m} \Pr[\mathscr{C}_{0,i}]\,, \end{aligned}$$

and the expectation of Φ_k in Eq. (C.30), the total number of time slots that are needed for a single transmission L_{tx} in Eq. (4.1) and ϵ_r in Eq. (C.27).
Similarly, we denote by $\mathscr{C}_{1,i}$, the event of successful CCAs at the event of deferred attempt due to the lack of the remaining slot times for backoff time counter at the $i+1$-th attempt. $\mathscr{C}_1$ is the successful event of $\mathscr{C}_{1,i}$ during the maximum $m+1$ times. The probability of such an event $\mathscr{C}_{1,i}$ is

$$\Pr[\mathscr{C}_{1,i}] = \varepsilon_i \sum_{k=0}^{i} \left(1 - \sum_{j=0}^{W_i-1} \frac{(1-\rho_b)^j}{W_i}\right) (1-\alpha)(1-\beta)\,. \tag{C.38}$$

The random CCA delay within $m+1$ attempts of the event $\mathscr{C}_1$ is

$$d_{\mathscr{C}_1} = \sum_{i=0}^{m} \left[\sum_{k=0}^{i} \Phi_k + T_{\mathrm{SP}} + \Phi_i + (i+1)2L_{\mathrm{sc}} \right] \mathbb{1}_{\mathscr{C}_{1,i}|\mathscr{C}_1}\,. \tag{C.39}$$

The average value of $d_{\mathscr{C}_1}$ is

$$\mu_{C_1} = \sum_{i=0}^{m} \frac{\rho_{\mathscr{C}_{1,i}}}{\Pr[\mathscr{C}_1]}\,, \tag{C.40}$$

where

$$\rho_{\mathscr{C}_{1,i}} = \varepsilon_i \sum_{k=0}^{i} \left(1 - \sum_{j=0}^{W_i-1} \frac{(1-\rho_b)^j}{W_i}\right)(1-\alpha)(1-\beta) \times (\mu_{\Phi_k} + T_{\mathrm{SP}} + \mu_{\Phi_i} + (i+1)2L_{\mathrm{sc}}) ,$$

$$\Pr[\mathscr{C}_1] = \sum_{i=0}^{m} \Pr[\mathscr{C}_{1,i}] .$$

Third, we derive the average backoff delay of the event of successful CCAs at the time $i+1$ given i previous unsuccessful CCAs but one of i unsuccessful attempts with an event of deferred attempt. With a similar way of previous events $\mathscr{C}_0$ and $\mathscr{C}_1$, we consider two different reasons of events of deferred attempt.
Let $\mathscr{F}_{0,i}$ be the event of successful CCAs at the time $i+1$ after an event of deferred attempt during i previous unsuccessful attempts due to the lack of the remaining slot times for packet transmission. $\mathscr{F}_0$ is the successful event of $\mathscr{F}_{0,i}$ during the maximum $m+1$ times. The probability of such an event $\mathscr{F}_{0,i}$ is

$$\Pr[\mathscr{F}_{0,i}] = \sum_{f=0}^{i-2} \sum_{h=0}^{i-2-(f-1)} \varepsilon_f \sum_{k=0}^{f} (\widetilde{\alpha}_f + \widetilde{\beta}_f)\alpha^{i-2-(f-1)-h} \times ((1-\alpha)\beta)^h (1-\alpha)(1-\beta)u(i-2) , \tag{C.41}$$

where

$$\widetilde{\alpha}_r = \rho_t \sum_{j=0}^{W_r-1} \frac{(1-\rho_b)^j}{W_r}\alpha ,$$

$$\widetilde{\beta}_r = \rho_t \sum_{j=0}^{W_r-1} \frac{(1-\rho_b)^j}{W_r}(1-\alpha)\beta .$$

The random CCA delay within $m+1$ attempts of the event $\mathscr{F}_0$ is

$$d_{\mathscr{F}_0} = \sum_{i=0}^{m} \left[\sum_{f=0}^{i-2} \sum_{k=0}^{f} \Phi_k + T_{\mathrm{SP}} + L_{\mathrm{tx}} + \Phi_f + (i+1)2L_{\mathrm{sc}}\right] \times \mathbb{1}_{\mathscr{F}_{0,i}|\mathscr{F}_0} . \tag{C.42}$$

The average value of $d_{\mathscr{F}_0}$ is

$$\mu_{\mathscr{F}_0} = \sum_{i=0}^{m} \frac{\rho_{\mathscr{F}_{0,i}}}{\Pr[\mathscr{F}_0]} , \tag{C.43}$$

where

$$\rho_{\mathscr{F}_{0,i}} = \sum_{f=0}^{i-2} \sum_{h=0}^{i-2-(f-1)} \varepsilon_f \sum_{k=0}^{f} (\widetilde{\alpha}_f + \widetilde{\beta}_f)\alpha^{i-2-(f-1)-h}((1-\alpha)\beta)^h(1-\alpha)(1-\beta) \times \left(\mu_{\Phi_k} + T_{\mathrm{SP}} + L_{\mathrm{tx}} + \mu_{\Phi_f} + (i+1)2L_{\mathrm{sc}}\right) u(i-2) ,$$

$$\Pr[\mathscr{F}_0] = \sum_{i=0}^{m} \Pr[\mathscr{F}_{0,i}] .$$

Similarly, we denote by $\mathscr{F}_{1,i}$, the event of successful CCAs at the time $i+1$ after a deferred attempt during i previous unsuccessful attempts due to the lack of the remaining slot times for backoff time counter. $\mathscr{F}_1$ is the successful event of $\mathscr{F}_{1,i}$ during the maximum $m+1$ times. The probability of such an event $\mathscr{F}_{1,i}$ is

$$\begin{aligned}\Pr[\mathscr{F}_{1,i}] =& \sum_{f=0}^{i-2}\sum_{h=0}^{i-2-(f-1)} \varepsilon_f \sum_{k=0}^{f} (\widehat{\alpha}_f + \widehat{\beta}_f)\alpha^{i-2-(f-1)-h} \\ & \times ((1-\alpha)\beta)^h (1-\alpha)(1-\beta) u(i-2)\,, \end{aligned} \tag{C.44}$$

where

$$\begin{aligned}\widehat{\alpha}_r &= \left(1 - \sum_{j=0}^{W_r-1} \frac{(1-\rho_b)^j}{W_r}\right)\alpha\,, \\ \widehat{\beta}_r &= \left(1 - \sum_{j=0}^{W_r-1} \frac{(1-\rho_b)^j}{W_r}\right)(1-\alpha)\beta\,. \end{aligned}$$

The random CCA delay within $m+1$ attempts of the event $\mathscr{F}_1$ is

$$d_{\mathscr{F}_1} = \sum_{i=0}^{m}\left[\sum_{f=0}^{i-2}\sum_{k=0}^{f} \Phi_k + T_{\mathrm{SP}} + \Phi_f + (i+1)2L_{\mathrm{sc}}\right] \mathbb{1}_{\mathscr{F}_{1,i}|\mathscr{F}_1}\,. \tag{C.45}$$

The average value of $d_{\mathscr{F}_1}$ is

$$\mu_{\mathscr{F}_1} = \sum_{i=0}^{m} \frac{\rho_{\mathscr{F}_{1,i}}}{\Pr[\mathscr{F}_1]}\,, \tag{C.46}$$

where

$$\begin{aligned}\rho_{\mathscr{F}_{1,i}} =& \sum_{f=0}^{i-2}\sum_{h=0}^{i-2-(f-1)} \varepsilon_f \sum_{k=0}^{f} (\widehat{\alpha}_f + \widehat{\beta}_f)\alpha^{i-2-(f-1)-h}((1-\alpha)\beta)^h(1-\alpha)(1-\beta) \\ & \times \left(\mu_{\Phi_k} + T_{\mathrm{SP}} + \mu_{\Phi_f} + (i+1)2L_{\mathrm{sc}}\right) u(i-2)\,, \\ \Pr[\mathscr{F}_1] =& \sum_{i=0}^{m} \Pr[\mathscr{F}_{1,i}]\,. \end{aligned}$$

In the following, we consider the packet loss due to collision. We remind that each node transmits a packet when the channel sensing is successful within the maximum number of m backoff stages. Hence, we derive the average packet delay based on the average backoff delay for the event of successful CCAs. By considering the event of deferred attempt, we categorize the events of successful transmission at the time $k+1$ given k previous packet collisions as follows: (a) the event of successful packet transmission without any deferred attempts (b) the event of successful packet transmission at the event of deferred attempt (c) the event of successful packet transmission after an event of deferred attempt during k previous unsuccessful attempts.

First, the average delay of the successful packet transmission at the time $k+1$ after k events of packet collisions without any events of deferred attempts is

$$\Gamma_k = (1-P_c)P_c^k \Pr[\mathscr{B}_0]^{k+1} \left((k+1)\mu_{\mathscr{B}_0} + L_s + kL_c\right) , \tag{C.47}$$

$$P_k = (1-P_c)P_c^k \Pr[\mathscr{B}_0]^{k+1} , \tag{C.48}$$

where $\Pr[\mathscr{B}_0]$ in Eq. (C.28), $\mu_{\mathscr{B}_0}$ in Eq. (C.31), and L_s and L_c are the packet successful transmission time and the packet collision time given in Eqs. (2.3) and (2.4), respectively. A packet transmission is successful with probability $1-P_c$, or collide with probability P_c given by Eq. (C.48).
Second, the average delay of the successful transmission at the deferred attempt of the time $k+1$ after k packet collisions is

$$\hat{\Gamma}_{*,k} = (1-P_c)P_c^k \Pr[\mathscr{B}_0]^k \Pr[*] \left(k\mu_{\mathscr{B}_0} + \mu_* + L_s + kL_c\right) , \tag{C.49}$$

$$\hat{P}_{*,k} = (1-P_c)P_c^k \Pr[\mathscr{B}_0]^k \Pr[*] , \tag{C.50}$$

where $*$ is one of the events $\{\mathscr{C}_0, \mathscr{C}_1, \mathscr{F}_0, \mathscr{F}_1\}$ given in Eqs. (C.34), (C.40), (C.43), and (C.46), respectively.
Third, the average delay of the successful packet transmission at the time $k+1$ given k previous events of packet collisions but one of k events of packet collisions with an event of deferred attempts is

$$\begin{aligned}\check{\Gamma}_{*,k} =& (1-P_c)P_c^k \sum_{r=0}^{k-1} \Pr[\mathscr{B}_0]^{k-1-r} \Pr[*] \Pr[\mathscr{B}_1]^{r+1} \left((k-1\right. \\ & \left. -r)\mu_{\mathscr{B}_0} + \mu_* + (r+1)\mu_{\mathscr{B}_1} + L_s + kL_c\right) u(k-1) , \end{aligned} \tag{C.51}$$

$$\check{P}_{*,k} = (1-P_c)P_c^k \sum_{r=0}^{k-1} \Pr[\mathscr{B}_0]^{k-1-r} \Pr[*] \Pr[\mathscr{B}_1]^{r+1} u(k-1) , \tag{C.52}$$

where $*$ is one of the events $\{\mathscr{C}_0, \mathscr{C}_1, \mathscr{F}_0, \mathscr{F}_1\}$ given in Eq. (C.34), (C.40), (C.43), and (C.46), respectively.
Now, we are in the position to derive the average delay for successfully received packets. By normalizing Eqs. (C.47), (C.50), and (C.52), the expected value of delay is

$$\begin{aligned}\mathbb{E}[D] =& \sum_{k=0}^{n} \Big(\Gamma_k + \hat{\Gamma}_{\mathscr{C}_0,k} + \hat{\Gamma}_{\mathscr{C}_1,k} + \hat{\Gamma}_{\mathscr{F}_0,k} + \hat{\Gamma}_{\mathscr{F}_1,k} \\ & + \left(\check{\Gamma}_{\mathscr{C}_0,k} + \check{\Gamma}_{\mathscr{C}_1,k} + \check{\Gamma}_{\mathscr{F}_0,k} + \check{\Gamma}_{\mathscr{F}_1,k} \right) u(k-1) \Big) P_{\text{tot}}^{-1} , \end{aligned} \tag{C.53}$$

where

$$\begin{aligned}P_{\text{tot}} =& \sum_{k=0}^{n} P_k + \hat{P}_{\mathscr{C}_0,k} + \hat{P}_{\mathscr{C}_1,k} + \hat{P}_{\mathscr{F}_0,k} + \hat{P}_{\mathscr{F}_1,k} \\ & + \left(\check{P}_{\mathscr{C}_0,k} + \check{P}_{\mathscr{C}_1,k} + \check{P}_{\mathscr{F}_0,k} + \check{P}_{\mathscr{F}_1,k} \right) u(k-1) . \end{aligned}$$

C.3 Proof of Proposition 5

We follow two steps to compute the average queueing delay of GTS allocation mechanism. We first derive the expected delay $D_{i,j,t}$ of j *new requests* that observe a queue size of i waiting requests at superframe t. Then, the expected delay experienced by j *new requests* arriving at superframe t is computed by considering $D_{i,j,t}$. Now, we compute the expected delay $D_{i,j,t}$. If the *new requests* are $j < \omega_j$ (i.e., $\upsilon_j = 0$), then

$$D_{i,j,t} = \frac{T^t_{\mathrm{CAP}}}{2} + \sum_{k=t}^{t+\upsilon_i} T^{k,k+1}_{\mathrm{BI}} + \underline{\theta} T^{t+\upsilon_i+1}_{\mathrm{SS}} \left(\omega_i + \frac{j+1}{2} \right), \tag{C.54}$$

where υ_i, υ_j is the quotient and ω_i, ω_j is the remainder of *old* and *new requests*, respectively. The average arrival delay is half of T_{CAP}. The offset $\sum_{k=t}^{t+\upsilon_i} T^{k,k+1}_{\mathrm{BI}}$ takes into account the delay of i *old requests*. In addition, the average service delay is half of the service time of $j+1$ *new requests* with a single GTS duration $\underline{\theta} T^{t+\upsilon_i+1}_{\mathrm{SS}}$ for each request. Note that ω_i remainders are considered as an offset before the GTS allocation of j *new requests*.
If j *new requests* are more than ω_j (i.e., $\upsilon_j > 0$), then

$$\begin{aligned}
&D_{i,j,t} \\
&= \frac{\Delta_u - \omega_i}{j} \left[T^t_{\mathrm{CAP}} \left(1 - \frac{\Delta_u - \omega_i}{2j} \right) + \sum_{k=t}^{t+\upsilon_i} T^{k,k+1}_{\mathrm{BI}} + \underline{\theta} T^{t+\upsilon_i+1}_{\mathrm{SS}} \left(\omega_i + \frac{\Delta_u - \omega_i + 1}{2} \right) \right] \\
&\quad + \frac{\Delta_u}{j} \sum_{l=t+\upsilon_i+1}^{t+\upsilon_i+\upsilon_j-1} \left[T^t_{\mathrm{CAP}} \left(1 - \frac{\Delta_u - \omega_i}{j} - \frac{(l-(t+\upsilon_i+1))\Delta_u}{j} - \frac{\Delta_u}{2j} \right) + \sum_{k=t}^{t+\upsilon_i} T^{k,k+1}_{\mathrm{BI}} \right. \\
&\quad \left. + \sum_{k=t+\upsilon_i+1}^{l} T^{k,k+1}_{\mathrm{BI}} + \frac{\underline{\theta} T^{l+1}_{\mathrm{SS}} (\Delta_u + 1)}{2} \right] u(\upsilon_j - 2) \\
&\quad + \frac{\omega_j}{j} \left[T^t_{\mathrm{CAP}} \left(1 - \frac{\Delta_u - \omega_i}{j} - \frac{(\upsilon_j - 1)\Delta_u}{j} - \frac{\omega_j}{2j} \right) + \sum_{k=t}^{t+\upsilon_i+\upsilon_j} T^{k,k+1}_{\mathrm{BI}} \right. \\
&\quad \left. + \frac{\underline{\theta} T^{t+\upsilon_i+\upsilon_j+1}_{\mathrm{SS}} (\omega_j + 1)}{2} \right].
\end{aligned} \tag{C.55}$$

It is possible to categorize the delay into three groups (first, middle, last group). The different groups of delay are also combined with the arrival, offset, and service delay components. The delay of the first group is the delay of first allocated GTSs $\Delta_u - \omega_i$ out of j *new requests* at superframe $t + \upsilon_i + 1$, see Figure 4.3. The average arrival delay of the first group considers the ratio $1 - (\Delta_u - \omega_i)(2j)^{-1}$ since the arrival time of j *new requests* are uniformly distributed in the CAP. The offset $\sum_{k=t}^{t+\upsilon_i} T^{k,k+1}_{\mathrm{BI}}$ takes into account the delay of i *old requests*. With a similar approach

of $j < \omega_j$, the average of the service delay considers ω_i remainders and $\Delta_u - \omega_i$ *new requests*.
The delay of the middle group is the delay between the first and last allocated GTSs $(\upsilon_j - 1)\Delta_u$ out of j *new requests* from the superframe $t + \upsilon_i + 2$ to $t + \upsilon_i + \upsilon_j$, see Figure 4.3. Hence, the number of allocated GTSs is the maximum number of GTSs Δ_u out of j *new requests*. The average arrival delay of the middle group considers the term $1 - (\Delta_u - \omega_i)j^{-1} - (l - (t + \upsilon_i + 1))\Delta_u j^{-1} - \Delta_u(2j)^{-1}$ since other requests of the number $\Delta_u - \omega_i$ of the first group and the number $(l - (t + \upsilon_i + 1))\Delta_u$ of the middle group obtain GTSs before Δ_u *new requests* out of j. Note that the number of allocated GTS is dependent on the time progress l in a superframe unit. With a similar approach used in the case $j < \omega_j$, we consider Δ_u *new requests* to derive the aveage service delay.
The delay of the last group is the delay of last allocated GTSs ω_j out of j *new requests* at superframe $t + \upsilon_i + \upsilon_j + 1$. The average arrival delay of the last group considers the term $1 - (\Delta_u - \omega_i)j^{-1} - ((\upsilon_j - 1)\Delta_u)j^{-1} - \omega_j(2j)^{-1}$ since other requests of the number $\Delta_u - \omega_i$ of the first group and the number $(\upsilon_j - 1)\Delta_u$ of the middle group obtain GTSs before ω_j remainders out of j *new requests*. By a similar approach to the case $j < \omega_j$, we consider ω_j remainders in order to compute the average service delay.
By considering the Eqs. (C.54) and (C.55), the expected delay experienced by j *new requests* arriving at the superframe t is

$$\begin{aligned}\mathbb{E}[D_q(t)] = &\sum_{i=0}^{\Delta_u - 1}\sum_{j=1}^{\bar{\lambda}} \pi_i^t \frac{\lambda_j}{\sum_{k=1}^{\bar{\lambda}} \lambda_k} \left(D_{i,j,t}\, u(q-j) + D_{i,q,t}\, u(j-(q+1))\right) \\ &+ \sum_{i=\Delta_u}^{q}\sum_{j=1}^{\bar{\lambda}} \pi_i^t \frac{\lambda_j}{\sum_{k=1}^{\bar{\lambda}} \lambda_k} \left(D_{i,j,t}\, u(q-i+\Delta_u-j)\right. \\ &\left. + D_{i,q-i+\Delta_u,t}\, u(j-(q-i+\Delta_u+1))\right) \\ &+ \pi_{\bar{q}}^t \sum_{j=1}^{\bar{\lambda}} \frac{\lambda_j}{\sum_{k=1}^{\bar{\lambda}} \lambda_k} \left(D_{q,j,t}\, u(\Delta_u - j) + D_{q,\Delta_u,t}\, u(j-(\Delta_u+1))\right), \end{aligned} \tag{C.56}$$

where $\bar{\lambda}$ is the maximum number of requests. The first term of Eq. (C.56) gives the average delay of *new requests* when the *old requests* $i \leq \Delta_u - 1$. By considering the Markov chain, if there are more than q *new requests*, then the extra requests out of q arrivals will be dropped. The unit function takes into account the queue size q. The second term considers the average delay when the *old requests* $\Delta_u \leq i \leq q$. Note that the Markov chain represents the feasible number of *new requests* without the dropped requests. By a similar approach to first term, it is possible to consider the maximum number of *new requests*, $q - i + \Delta_u$, without dropped requests when i *old requests* wait in the queue. The third term computes the average delay at the

dropped state $\bar{q}$. At the dropped state $\bar{q}$, Δ_u *new requests* are considered since the queue size of *old requests* is q.

C.4 Proof of Lemma 4

Let us denote $T^{\star}_{\mathrm{CAP}}$ and $T^{\star}_{\mathrm{CFP}}$ the fixed point of (4.22). We derive the stochastic mean convergence condition of these iterations. The geometric series is used to compute the condition of convergence and divergence of the iterations of the CAP and CFP lengthes. The iteration of Eq. (4.22) is rewritten as follows

$$T^{t+1}_{\mathrm{CAP}} = T_{\mathrm{SD}} - \frac{(1-\eta_d)N_{\mathrm{SD}}T_{\mathrm{SS}}}{T_{\mathrm{SD}}}T^{t}_{\mathrm{CAP}}\,.$$

Note that the convergence of an infinite series is not changed by insertion or removal of finite number of terms. We remark that the iteration follows an oscillatory behavior since the slope of iteration is negative. Then, the CAP length is rewritten as the following sum of geometric series

$$T^{\infty}_{\mathrm{CAP}} = T_{\mathrm{SD}} \sum_{k=0}^{\infty} \left(-\frac{(1-\eta_d)N_{\mathrm{SD}}T_{\mathrm{SS}}}{T_{\mathrm{SD}}} \right)^{k}\,.$$

This geometric series converges to

$$T^{\star}_{\mathrm{CAP}} = \frac{T^2_{\mathrm{SD}}}{T_{\mathrm{SD}} + (1-\eta_d)N_{\mathrm{SD}}T_{\mathrm{SS}}}\,, \tag{C.57}$$

for $(1-\eta_d)N_{\mathrm{SD}}T_{\mathrm{SS}} < T_{\mathrm{SD}}$ and the CAP length is saturated to

$$T^{\star}_{\mathrm{CAP}} = T_{\mathrm{SD}} - \Delta_u T_{\mathrm{SS}}\,, \tag{C.58}$$

for $(1-\eta_d)N_{\mathrm{SD}}(T_{\mathrm{SD}} - \Delta_u T_{\mathrm{SS}})/T_{\mathrm{SD}} \geq \Delta_u$. If the number of requests is greater than Δ_u, then the CAP length is $T_{\mathrm{SD}} - \Delta_u T_{\mathrm{SS}}$. By considering the convergence condition (C.57) and the saturated condition (C.58), the stability of the CAP and CFP in a superframe are as follows

$$\begin{array}{ll} \text{convergence} & \text{if } N_{\mathrm{SD}} < \frac{N_{\mathrm{SS}}}{1-\eta_d}\,, \\ \text{saturation} & \text{if } N_{\mathrm{SD}} \geq \frac{N_{\mathrm{SS}}\Delta_u}{(1-\eta_d)(N_{\mathrm{SS}}-\Delta_u)}\,, \\ \text{oscillation} & \text{if } \frac{N_{\mathrm{SS}}}{1-\eta_d} \leq N_{\mathrm{SD}} < \frac{N_{\mathrm{SS}}\Delta_u}{(1-\eta_d)(N_{\mathrm{SS}}-\Delta_u)}\,. \end{array}$$

Bibliography

[1] A. Willig, K. Matheus, and A. Wolisz. Wireless technology in industrial networks. *Proceedings of the IEEE*, 93(6):1130–1151, 2005.

[2] A. Wheeler. Commercial applications of wireless sensor networks using zigbee. *IEEE Communications Magazine*, 45(4):70 –77, 2007.

[3] *Emerson Process Management*. `http://www.emersonprocess.com`.

[4] On World. *Smart Energy Homes: A Market Dynamics Report*, 2010. `http://onworld.com/smartenergyhomes/`.

[5] ZigBee Alliance. *ZigBee Specification*, 2005. `http://www.caba.org/standard/zigbee.html`.

[6] On World. *ZigBee Crosses the Chasm: A Market Dynamics Report on IEEE 802.15.4 and ZigBee*, 2010. `http://onworld.com/zigbee/`.

[7] ABI research. *Wireless Sensor Networking Markets*, 2010. `http://www.abiresearch.com/research/1003936`.

[8] IEEE. *IEEE 802.15.4 standard: Wireless Medium Access Control and Physical Layer Specifications for Low-Rate Wireless Personal Area Networks*, 2006. `http://www.ieee802.org/15/pub/TG4.html`.

[9] Gartner. *Gartner's Hype Cycle Special Report*, 2010. `http://www.gartner.com/technology/research/hype-cycles/index.jsp`.

[10] E. Witrant, P. Di Marco, P. Park, and C. Briat. Limitations and performances of robust control over WSN: UFAD control in intelligent buildings. *IMA Journal of Mathematical Control and Information*, 2010.

[11] KTH. *Projects in Networked Control Systems*, 2010. `http://www2.ee.kth.se/web_page/netcon`.

[12] O. P. Vidal. Smart office: Wireless sensor network for energy monitoring and user profiling. Master's thesis, KTH, 2010.

[13] European Environment Agency. *Electricity Consumption*, 2008. `www.eea.europa.eu/data-and-maps/indicators/en18-electricity-consumption-1`.

[14] European Environment Agency. *Use of freshwater resources*, 2009. http://www.eea.europa.eu/data-and-maps/indicators/use-of-freshwater-resources.

[15] California Energy Commission. *2008 Building Energy Efficiency Standards for California*, 2008. http://www.energy.ca.gov/title24/2008standards/index.html.

[16] S. M. Amin and B. F. Wollenberg. Toward a smart grid. *IEEE Power and Energy Magazine*, 3(5):34–38, 2005.

[17] Stockholm City Development Administration. *Stockholm Royal Seaport*. http://www.stockholmroyalseaport.com/.

[18] Quanser. *Inverted Pendulum*. http://www.quanser.com/english/downloads/products/Linear/IP02_PIS_031108.pdf.

[19] Quanser. *Coupled Water Tanks*. http://www.quanser.com/english/downloads/products/Specialty/CoupledTanks_PIS_031708.pdf.

[20] *ISA SP-100*. http://www.isa.org/.

[21] A. Willig. Recent and emerging topics in wireless industrial communication. *IEEE Transactions on Industrial Informatics*, 4(2):102–124, 2008.

[22] J. R. Moyne and D. M. Tilbury. The emergence of industrial control networks for manufacturing control, diagnostics, and safety data. *Proceedings of the IEEE*, 95(1):29–47, 2007.

[23] João P. Hespanha, Payam Naghshtabrizi, and Yonggang Xu. A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162, 2007.

[24] W. Zhang, M. S. Branicky, and S. M. Phillips. Stability of networked control systems. *IEEE Control Systems Magazine*, 21(1):84–99, 2001.

[25] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poola, and S. Sastry. Foundations of control and estimation over lossy networks. *Proceedings of the IEEE*, 95(1):163–187, 2007.

[26] E. Witrant, P. Park, M. Johansson, C. Fischione, and K. H. Johansson. Predictive control over wireless multi-hop networks. In *IEEE MSC*, 2007.

[27] A. Speranzon, C. Fischione, and K. H. Johansson. Distributed and collaborative estimation over wireless sensor networks. In *IEEE CDC*, 2006.

[28] A. Speranzon, C. Fischione, K. H. Johansson, and A. Sangiovanni-Vincentelli. A distributed minimum variance estimator for sensor networks. *IEEE Journal on Selected Areas in Communications, Special Issue on Control and Communication*, 95(4):609–621, 2008.

[29] E. Anderson M. Buettner, G. Yee and R. Han. X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks. In *ACM SenSys*, 2006.

[30] G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh. Fidelity and yield in a volcano monitoring sensor network. In *USENIX OSDI*, 2006.

[31] Moteiv, San Francisco, CA. *Tmote Sky Data Sheet*, 2006. `http://www.moteiv.com/products/docs/tmote-sky-datasheet.pdf`.

[32] Crossbow, Milpitas, CA. *MICA motes*. `http://www.xbow.com`.

[33] P. Park, P. Di Marco, C. Fischione, and K. H. Johansson. Adaptive IEEE 802.15.4 protocol for reliable and timely communication. *IEEE/ACM Transactions on Networking*, 2010. Submitted.

[34] C. Intanagonwiwat and R. Govindan and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *ACM MobiCom*, 2000.

[35] P. Buonadonna, D. Gay, J. Hellerstein, W. Hong, and S. Madden. TASK: Sensor network in a box. Technical report, Intel Research Lab Report, 2007.

[36] R. Steigman, and J. Endresen. Introduction to WISA and WPS, WISA-wireless interface for sensors and actuators and WPS-wireless proximity switches. *White paper*, 2004. `http://www.eit.uni-kl.de/litz/WISA.pdf`.

[37] I. Demirkol, C. Ersoy, and F. Alagoz. MAC protocols for wireless sensor networks: a survey. *IEEE Communications Magazine*, 44(4):115 – 121, 2006.

[38] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung. MAC essentials for wireless sensor networks. *IEEE Communications Surveys and Tutorials*, 12(2):222–248, 2010.

[39] K. Langendoen and A. Meier. Analyzing MAC protocols for low data-rate applications. *ACM Transactions on Sensor Networks*, 7(2):1–40, 2010.

[40] J. N. Al-Karaki and A. E. Kamal. Routing techniques in wireless sensor networks: a survey. *IEEE Transactions on Wireless Communications*, 11(6):6–28, 2004.

[41] K. Akkaya and M. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325 – 349, 2005.

[42] H. Luo, Y. Liu, and S. K. Das. Routing correlated data in wireless sensor networks: A survey. *IEEE Network*, 21(6):40 –47, 2007.

[43] K. Pavai, A. Sivagami, and D. Sridharan. Study of routing protocols in wireless sensor networks. In *ACT*, 2009.

[44] T. P. Lambrou and C. G. Panayiotou. A survey on routing techniques supporting mobility in sensor networks. In *IEEE MSN*, 2009.

[45] P. Levis, A. Tavakoli, and S. Dawson-Haggerty. Overview of existing routing protocols for low power and lossy networks. Technical report, IETF, 2009.

[46] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Communication*, 7(5):16–27, 2000.

[47] J. Rabaey, M. Ammer, J. J. da Silva, D. Patel, and S. Roundy. PicoRadio supports ad hoc ultra-low power wireless networking. *IEEE Computer*, 33(7):42–48, 2000.

[48] G. Pei and C. Chien. Low power TDMA in large wireless sensor networks,. In *IEEE MILCOM*, 2001.

[49] K. Arisha, M. Youssef, and M. Younis. Energy-aware TDMA-based MAC for sensor networks. In *IMPACCT*, 2002.

[50] V. Rajendran, K. Obraczka, and J. Garcia-Luna-Aceves. Energy-efficient, collision-free medium access control for wireless sensor networks. In *ACM SenSys*, 2003.

[51] L. van Hoesel and P. Havinga. A lightweight medium access protocol (LMAC) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches. In *INSS*, 2004.

[52] S. Coleri-Ergen and P. Varaiya. Pedamacs: Power efficient and delay aware medium access protocol for sensor networks. *IEEE Transactions on Mobile Computing*, 5(7):920–930, 2006.

[53] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic, and T. F. Abdelzaher. MMSN: Multi-frequency media access control for wireless sensor networks. In *IEEE INFOCOM*, 2006.

[54] N. Burri, P. von Rickenbach, and R. Wattenhofer. Dozer: ultra-low power data gathering in sensor networks. In *ACM/IEEE IPSN*, 2007.

[55] H. K. Le, D. Henriksson, and T. Abdelzaher. A practical multi-channel media access control protocol for wireless sensor networks. In *ACM/IEEE IPSN*, 2008.

[56] K. Pister and L. Doherty. TSMP: Time synchronized mesh protocol. In *DSN*, 2008.

[57] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *ACM MobiCom*, 2001.

[58] J. L. Hill and D. Culler. Mica: a wireless platform for deeply embedded networks. *IEEE Micro*, 22(6):12–24, 2002.

[59] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava. Optimizing sensor networks in the energy-latency-density design space. *IEEE Transactions on Mobile Computing*, 1(1):70–80, 2002.

[60] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *ACM SenSys*, 2004.

[61] E. Y. Lin, J. Rabaey, and A. Wolisz. Power-efficient rendezvous schemes for dense wireless sensor networks. In *IEEE ICC*, 2004.

[62] A. El-Hoiydi and J. D. Decotignie. WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks. In *IEEE ISCC*, 2004.

[63] R. Musaloiu-E., C. M. Liang, and A. Terzis. Koala: Ultra-low power data retrieval in wireless sensor networks. In *ACM/IEEE IPSN*, 2008.

[64] W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12(3):493–506, 2004.

[65] T. Van Dam and K. Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. In *ACM Sensys*, 2003.

[66] P. Lin, C. Qiao, and X. Wang. Medium access control with a dynamic duty cycle for sensor networks. In *IEEE WCNC*, 2004.

[67] G. Lu, B. Krishnamachari, and C. S. Raghavendra. An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks. In *IEEE IPDPS*, 2004.

[68] G. S. Ahn, E. Miluzzo, A. T. Campbell, S. G. Hong, and F. Cuomo. Funneling-MAC: A localized, sink-oriented MAC for boosting fidelity in sensor networks. In *ACM SenSys*, 2006.

[69] W. Ye, F. Silva, and J. Heidemann. Ultra-low duty cycle MAC with scheduled channel polling. In *ACM SenSys*, 2006.

[70] G. Halkes and K. Langendoen. Crankshaft: An energy-efficient MAC-protocol for dense wireless sensor networks. In *IEEE EWSN*, 2007.

[71] I. Rhee, A. Warrier, M. Aia, J. Min, and M. L. Sichitiu. Z-MAC: A hybrid MAC for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 16(3):511 –524, 2008.

[72] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. Collection tree protocol. In *ACM SenSys*, 2009.

[73] S. Moeller, A. Sridharan, B. Krishnamachari, and O. Gnawali. Routing without routes: the backpressure collection protocol. In *ACM/IEEE IPSN*, 2010.

[74] W. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. In *ACM/IEEE MobiCom*, 1999.

[75] R. C. Shah and J. Rabaey. Energy aware routing for low energy ad hoc sensor networks. In *IEEE WCNC*, 2002.

[76] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva. Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking*, 11(1):2 – 16, 2003.

[77] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *HICSS*, 2000.

[78] A. Manjeshwar and D. P. Agarwal. TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In *IEEE IPDPS*, 2001.

[79] S. Lindsey and C. Raghavendra. PEGASIS: Power-efficient gathering in sensor information systems. In *IEEE Aerospace Conference*, 2002.

[80] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang. A two-tier data dissemination model for large-scale wireless sensor networks. In *ACM MobiCom*, 2002.

[81] O. Younis and S. Fahmy. HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 3(4):366 – 379, 2004.

[82] V. Rodoplu and T. H. Ming. Minimum energy mobile wireless networks. *IEEE Journal of Selected Areas in Communications*, 17(8):1333–1344, 1999.

[83] Y. Xu, J. Heidemann, and D. Estrin. Geography-informed energy conservation for ad-hoc routing. In *ACM MobiCom*, 2001.

[84] Y. Yu, D. Estrin, and R. Govindan. Geographical and energy-aware routing: A recursive data dissemination protocol for wireless sensor networks. Technical report, UCLA Comp. Sci. Dept., 2001.

[85] M. Zorzi and R. R. Rao. Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance. *IEEE Transactions on Mobile Computing*, 2(4):349–365, 2003.

[86] E. Felemban, C. G. Lee, and E. Eylem. MMSPEED: Multipath multi-speed protocol for QoS guarantee of reliability and timeliness in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 5(6):738–754, 2006.

[87] A. Awad, R. German, and F. Dressler. P2P-based routing and data management using the virtual cord protocol (VCP). In *ACM MobiHoc*, 2008.

[88] P. Park, C. Fischione, A. Bonivento, K. H. Johansson, and A. Sangiovanni-Vincentelli. Breath: an adaptive protocol for industrial control applicationa using wireless sensor networks. *IEEE Transactions on Mobile Computing*, 2011. To appear.

[89] V. Rajendran, J. Garcia-Luna-Aceves, and K. Obraczka. Energy-efficient, application-aware medium access for sensor networks. In *IEEE MASS*, 2005.

[90] A. Barroso, U. Roedig, and C. Sreenan. uMAC: an energy-efficient medium access control for wireless sensor networks. In *IEEE EWSN*, 2005.

[91] A. El-Hoiydi. Spatial TDMA and CSMA with preamble sampling for low power ad hoc wireless sensor networks. In *IEEE ISCC*, 2002.

[92] M. Miller and N. Vaidya. A MAC protocol to reduce sensor network energy consumption using a wakeup radio. *IEEE Transactions on Mobile Computing*, 4(3):228–242, 2005.

[93] H. Wang, X. Zhang, F. Nait-Abdesselam, and A. Khokhar. DPS-MAC: An asynchronous MAC protocol for wireless sensor networks. In *HIPC*, 2007.

[94] X. Shi and G. Stromberg. SyncWUF: An ultra low-power MAC protocol for wireless sensor networks. *IEEE Transactions on Mobile Computing*, 6(1):115–125, 2007.

[95] Y. Sun, O. Gurewitz, and D. Johnson. RI-MAC: A receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks. In *ACM SenSys*, 2008.

[96] P. Dutta, S. Dawson-Haggerty, Y. Chen, C. M. Liang, and A. Terzis. Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless. In *ACM SenSys*, 2010.

[97] J. Son, J. Pak, and K. Han. A MAC protocol using separate wakeup slots for sensor network. In *ICCSA*, 2006.

[98] L. Bernardo, R. Oliveira, M. Pereira, M. Macedo, and P. Pinto. A wireless sensor MAC protocol for bursty data traffic. In *IEEE PIMRC*, 2007.

[99] IEEE. *IEEE 802.15 task group 4e: Wireless Medium Access Control and Physical Layer Specifications for Low-Rate Wireless Personal Area Networks*, 2010. `http://www.ieee802.org/15/pub/TG4e.html`.

[100] R. Rom and M. Sidi. *Multiple access protocols: performance and analysis*. Springer-Verlag, 1990.

[101] L. Bao and J. J. Garcia-Luna-Aceves. A new approach to channel access scheduling for ad hoc networks. In *ACM MobiCom*, 2001.

[102] O. D. Incel, S. Dulman, and P. Jansen. Multi-channel support for dense wireless sensor networking. In *EuroSSC*, 2006.

[103] HART Communication Foundation. *Wirelesshart data sheet*, 2007. `http://www.hartcomm2.org/hartprotocol/wirelesshart/wirelesshartmain.html`.

[104] I. Chlamtac and S. Kutten. Tree-based broadcasting in multihop radio networks. *IEEE Transactions on Computers*, 36(10):1209 – 1223, 1987.

[105] M. Ali, T. Suleman, and Z. Uzmi. MMAC: A mobility-adaptive, collision-free MAC protocol for wireless sensor networks. In *IEEE IPCCC*, 2005.

[106] W. Lee, A. Datta, and R. Cardell-Oliver. FlexiMAC: A flexible TDMA-based MAC protocol for fault-tolerant and energy-efficient wireless sensor networks. In *IEEE ICON*, 2006.

[107] R. Jurdak, P. Baldi, and C. Lopes. Energy-aware adaptive low power listening for sensor networks. In *INSS*, 2005.

[108] L. Kleinrock and Y. Yemini. An optimal adaptive scheme for multiple access broadcast communication. In *IEEE ICC*, 1978.

[109] R. Binder. A dynamic packet switching system for satellite broadcast channels. In *IEEE ICC*, 1975.

[110] W. Crowther, R. Rettberg, D. Walden, S. Ornstein, and F. Heart. A system for broadcast communication: Reservation-aloha. In *IEEE HICSS*, 1973.

[111] L. G. Roberts. Dynamic allocation of satellite capacity through packet reservation. In *AFIPS*, 1973.

[112] F. A. Tobagi and L. Kleinrock. Packet switching in radio channels: Part III - polling and (dynamic) split-channel reservation multiple-access. *IEEE Transactions on Communications*, 24(8):832–845, 1976.

[113] S. Tasaka and Y. Ishibashi. A reservation protocol for satellite packet communication - a performance analysis and stability considerations. *IEEE Transactions on Communications*, 32(8):920– 927, 1984.

[114] D. Tsai and J.F. Chang. Performance study of an adaptive reservation multiple access technique for data transmissions. *IEEE Transactions on Communications*, 34(7):725–727, 1986.

[115] M. Fine and F. A. Tobagi. Demand assignment multiple access schemes in broadcast bus local area networks. *IEEE Transactions on Computers*, 33(12):1130–1159, 1984.

[116] C. W. Pyo and H. Harada. Throughput analysis and improvement of hybrid multiple access in IEEE 802.15.3c mm-wave WPAN. *IEEE Journal on Selected Areas in Communications*, 27(8):1414 – 1424, 2009.

[117] W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12(3):493–506, 2004.

[118] H. Pham and S. Jha. An adaptive mobility-aware MAC protocol for sensor networks (MS-MAC). In *IEEE MASS*, 2004.

[119] P. Havinga and G. Smit. E2MAC: an energy efficient MAC protocol for multimedia traffic. Technical report, University of Twente, 1998.

[120] I. Rhee, A. Warrier, J. Min, and L. Xu. DRAND: Distributed randomized TDMA scheduling for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 8(10):1384–1396, 2009.

[121] J. Moy. Open shortest path first protocol. Technical report, IETF, 1998.

[122] T. Clausen and P. Jacquet. Optimized link state routing protocol (OLSR). Technical report, IETF, 2003.

[123] D. B. Johnson, D. A. Maltz, and J. Broch. DSR: the dynamic source routing protocol for multihop wireless ad hoc networks. *Ad hoc networking*, pages 139–172, 2001.

[124] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc on-demand distance vector (AODV) routing. Technical report, IETF, 2003.

[125] A. Manjeshwar and D. P. Agarwal. Apteen: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In *IEEE IPDPS*, 2002.

[126] *IPv6 over Low power WPAN (6lowpan).* `http://www.ietf.org/dyn/wg/charter/6lowpan-charter.html`.

[127] Internet Engineering Task Force (IETF). *Routing Over Low power and Lossy networks.* `http://www.ietf.org/dyn/wg/charter/roll-charter.html`.

[128] IEEE. *IEEE 802.11 standard: Wireless LAN Medium Access Control and Physical Layer Specifications*, 1999. `http://www.ieee802.org/11`.

[129] OnWorld. *WSN for Smart Industries: A Market Dynamics Report*, 2007. `http://www.onworld.com/smartindustries/index.html`.

[130] Frost & Sullivan. *Wireless Devices in the Factory Automation - An Overview of Adoption Trends*, 2008. http://www.frost.com/prod/servlet/frost-home.

[131] Bluetooth SIG. *Bluetooth Specification*. http://www.bluetooth.com.

[132] S. Lee, K. C. Lee, M. H. Lee, and F. Harashima. Integration of mobile vehicles for automated material handling using profibus and IEEE 802.11 networks. *IEEE Transactions on Industrial Electronics*, 49(3):693 –701, 2002.

[133] R. Moraes, F. Vasques, P. Portugal, and J. A. Fonseca. VTP-CSMA: A virtual token passing approach for real-time communication in IEEE 802.11 wireless networks. *IEEE Transactions on Industrial Informatics*, 3(3):215 –224, 2007.

[134] *International Standard ISO/IEC 8802-2:1998: Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 2: Logical Link Control*. http://grouper.ieee.org/groups/802/2/.

[135] V. Srivastava and M. Motani. Cross-layer design: A survey and the road ahead. *IEEE Communications Magazine*, 43(12):112–119, 2005.

[136] Texas Instruments. *CC2420 datasheet*. http://focus.ti.com/lit/ds/symlink/cc2420.pdf.

[137] Crossbow. *TelosB Device*. http://www.xbow.com/.

[138] Texas Instruments. *MSP430 datasheet*. http://focus.ti.com/docs/prod/folders/print/msp430f123.html.

[139] R. Cristescu, B. Beferull-Lozano, M. Vetterli, and R. Wattenhofer. Network correlated data gathering with explicit communication: NP-completeness and algorithms. *IEEE/ACM Transactions on Networking*, 14(1):41–54, 2006.

[140] P. Ogren, E. Fiorelli, and N. E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, 2004.

[141] C. Meng, T. Wang, W. Chou, S. Luan, Y. Zhang, and Z. Tian. Remote surgery case: robot-assisted teleneurosurgery. In *IEEE ICRA*, 2004.

[142] J. Garcia, F. R. Palomo, A. Luque, C. Aracil, J. M. Quero, D. Carrion, F. Gamiz, P. Revilla, J. Perez-Tinao, M. Moreno, P. Robles, and L. G. Franquelo. Reconfigurable distributed network control system for industrial plant automation. *IEEE Transactions on Industrial Electronics*, 51(6):1168–1180, 2004.

[143] T. Arampatzis, J. Lygeros, and S. Manesis. A survey of applications of wireless sensors and wireless sensor networks. In *IEEE MCCA*, 2005.

[144] L. A. Montestruque and P. J. Antsaklis. On the model-based control of networked systems. *Automatica*, 39(10):1837–1843, 2003.

[145] P. Seiler and R. Sengupta. Analysis of communication losses in vehicle control problems. In *IEEE ACC*, 2001.

[146] G. Walsh, H. Ye, and L. Bushnell. Stability analysis of networked control systems. *IEEE Transactions on Control Systems Technology*, pages 2876–2880, 1999.

[147] Rogelio Luck and Asok Ray. An observer-based compensator for distributed delays. *Automatica*, 26(5):903–908, 1990.

[148] M. Yu, L. Wang, G. Xie, and T. Chu. Stabilization of networked control systems with data packet droupout via switched system approach. In *IEEE CACSD*, 2004.

[149] J. Nilsson. *Real-Time control systems with delays*. PhD thesis, Lund Institute of Technology, 1998.

[150] G. N. Nair, F. Fagnani, S. Zampieri, and R. J. Evans. Feedback control under data rate constraints: An overview. *Proceedings of the IEEE*, 95(1):108–137, 2007.

[151] D. Henriksson and A. Cervin. Optimal on-line sampling period assignment for real-time control tasks based on plant state information. In *IEEE CDC*, 2005.

[152] X. Liu and A. J. Goldsmith. Wireless network design for distributed control. In *IEEE CDC*, 2004.

[153] C. Ramesh, H. Sandberg, and K. H. Johansson. Multiple access with attention-based tournaments for monitoring over wireless networks. In *ECC*, 2009.

[154] N. Elia and S. K. Mitter. Stabilization of linear systems with limited information. *IEEE Transactions on Automatic Control*, 46(9):1384–1400, 2001.

[155] W. S. Wong and R. W. Brockett. Systems with finite communication bandwidth constraints-II: Stabilization with limited information feedback. *IEEE Transactions on Automatic Control*, 44(5):1049 –1053, 1999.

[156] A. Khaladj, M. Rahgozar, and N. Yazdani. Reducing the variations in delay in IEEE 802.11 DCF. In *IEEE APCC*, 2005.

[157] F. Stann and J. Heidemann. RMST: Reliable Data Transport in Sensor Networks. In *IEEE SNPA*, 2003.

[158] S. Pollin, M. Ergen, S. C. Ergen, B. Bougard, F. Catthoor, A. Bahai, and P. Varaiya. Performance analysis of slotted carrier sense IEEE 802.15.4 acknowledged uplink transmissions. In *IEEE WCNC*, 2008.

[159] G. Bianchi. Performance analysis of the IEEE 802.11 distributed cordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, 2000.

[160] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas. IEEE 802.11 packet delay – a finite retry limit analysis. In *IEEE GLOBECOM*, 2003.

[161] Z. Hadzi-Velkov and B. Spasenovski. Saturation throughput-delay analysis of IEEE 802.11 in fading channel. In *IEEE ICC*, 2003.

[162] O. Tickioo and B. Sikdar. Queueing analysis and delay mitigation in IEEE 802.11 random access MAC based wireless networks. In *IEEE INFOCOM*, 2004.

[163] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma. Performance of reliable transport protocol over IEEE 802.11 wireless LAN: Analysis and enhancement. In *IEEE INFOCOM*, 2002.

[164] J. Zheng and M. L. 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*, 42(6):140–146, 2004.

[165] A. Koubaa, M. Alves, and E. Tovar. A comprehensive simulation study of slotted csma/ca for IEEE 802.15.4 wireless sensor networks. In *IEEE WFCS*, 2006.

[166] S. Pollin, M. Ergen, S. C. Ergen, B. Bougard, L. V. D. Perre, F. Catthoor, I. Moerman, A. Bahai, and P. Varaiya. Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer. In *IEEE GLOBECOM*, 2006.

[167] P. K. Sahoo and J. P. Sheu. Modeling IEEE 802.15.4 based wireless sensor network with packet retry limits. In *IEEE PE-WASUN*, 2008.

[168] Z. Tao, S. Panwar, G. Daqing, and J. Zhang. Performance analysis and a proposed improvement for the IEEE 802.15.4 contention access period. In *IEEE WCNC*, 2006.

[169] J. Misic and V. B. Misic. Access delay for nodes with finite buffers in IEEE 802.15.4 beacon enabled PAN with uplink transmissions. *Computer Communications*, 28(10):1152–1166, 2005.

[170] J. Misic, S. Shaf, and V. B. Misic. Performance of a beacon enabled IEEE 802.15.4 cluster with downlink and uplink traffic. *IEEE Transactions on Parallel and Distributed Systems*, 17(4):361–376, 2006.

[171] C. Y. Jung, H. Y. Hwang, D. K. Sung, and G. U. Hwang. Enhanced Markov chain model and throughput analysis of the slotted CSMA/CA for IEEE 802.15.4 under unsaturated traffic conditions. *IEEE Transactions on Vehicular Technology*, 58(1):473–478, 2009.

[172] I. Ramachandran, A. K. Das, and S. Roy. Analysis of the contention access period of IEEE 802.15.4 MAC. *ACM Transactions on Sensor Networks*, 3(1):641–651, 2007.

[173] F. Cali, M. Conti, and E. Gregori. IEEE 802.11 protocol: design and performance evaluation of an adaptive backoff mechanism. *IEEE Journal on Selected Areas in Communications*, 18(9):1774 –1786, 2000.

[174] P. K. Sahoo, J. P. Sheu, and Y. C. Chang. Performance evaluation of wireless sensor network with hybrid channel access mechanism. *Journal of Network and Computer Applications*, 32(4):878–888, 2009.

[175] M. Martalo, S. Busanelli, and G. Ferrari. Markov chain-based performance analysis of multihop IEEE 802.15.4 wireless networks. *Performance Evaluation*, 66(12):722–741, 2009.

[176] J. Misic, J. Fung, and V. B. Misic. Interconnecting 802.15.4 clusters in master-slave mode: queueing theoretic analysis. In *IEEE ISPAN*, 2005.

[177] J. Misic and R. Udayshankar. Slave-slave bridging in 802.15.4 beacon enabled networks. In *IEEE WCNC*, 2007.

[178] K. Yedavalli and B. Krishnamachari. Enhancement of the IEEE 802.15.4 MAC protocol for scalable data collection in dense sensor networks. In *ICST WiOPT*, 2008.

[179] R. Bruno, M. Conti, and E. Gregori. Optimization of efficiency and energy consumption in p-persistent CSMA-based wireless LANs. *IEEE Transactions on Mobile Computing*, 1(1):10 – 31, 2002.

[180] Q. Pang, S. C. Liew, J. Y. B. Lee, and V. C. M. Leung. Performance evaluation of an adaptive backoff scheme for WLAN: Research articles. *Wirel. Commun. Mob. Comput.*, 4(8):867–879, 2004.

[181] V. Bharghavan, A. J. Demers, S. Shenker, and L. Zhang. MACAW: A media access protocol for wireless LAN's. *ACM SIGCOMM*, 1994.

[182] B. Bensaou, Y. Wang, and C. C. Ko. Fair medium access in 802.11 based wireless ad-hoc networks. In *ACM MobiHoc*, 2000.

[183] J. G. Ko, Y. H. Cho, and H. Kim. Performance evaluation of IEEE 802.15.4 MAC with different backoff ranges in wireless sensor networks. In *IEEE ICCS*, 2006.

[184] A. C. Pang and H. W. Tseng. Dynamic backoff for wireless personal networks. In *IEEE GLOBECOM*, 2004.

[185] A. Giridhar and P. R. Kumar. Toward a theory of in-network computation in wireless sensor networks. *IEEE Communication Magazine*, pages 97–107, 2006.

[186] IEEE. *IEEE 802.15 task group 4b: Wireless Medium Access Control and Physical Layer Specifications for Low-Rate Wireless Personal Area Networks*, 2010. `http://www.ieee802.org/15/pub/TG4b.html`.

[187] Enocean Alliance. *Enocean Products*. `http://www.enocean.com/`.

[188] Riga Development. *WiSuite — The Freedom and Simplicity of Wireless Control*. `http://www.wisuite.com/`.

[189] P. Park, P. Di Marco, P. Soldati, C. Fischione, and K. H. Johansson. A generalized Markov chain model for effective analysis of slotted IEEE 802.15.4. In *IEEE MASS*, 2009.

[190] SICS. *Contiki Operating System*. `http://www.sics.se/contiki/about-contiki.html`.

[191] KTH. *Implementation of adaptive IEEE 802.15.4 protocol*. [Online] Available: `http://www.ee.kth.se/~pgpark/code/adap-wpan.zip`.

[192] P. Park, P. Di Marco, C. Fischione, and K. H. Johansson. Adaptive IEEE 802.15.4 protocol for energy efficient, reliable and timely communications. Technical report, KTH, 2009. `http://www.ee.kth.se/~pgpark/papers/adaptive_wpan_opt.pdf`.

[193] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[194] J. A. Storer. Special issue on lossless data compression. *Proceedings of the IEEE*, 88(11):1685–1688, 2000.

[195] F. Cali, M. Conti, and E. Gregori. Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit. *IEEE/ACM Transactions on Networking*, 8(6):785 – 799, 2006.

[196] R. Jain, D. Chiu, and W. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. Technical report, Digital Equipment Corporation, 1984.

[197] G. Lu, B. Krishnamachari, and C. S. Raghavendra. Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks. In *IEEE IPCCC*, 2004.

[198] N. F. Timmons and W. G. Scanlon. Analysis of the performance of IEEE 802.15.4 for medical sensor body area networking. In *IEEE SECON*, 2004.

[199] L. Changle, L. Huan-Bang, and R. Kohno. Performance evaluation of IEEE 802.15.4 for wireless body area network WBAN. In *IEEE ICC Workshops*, 2009.

[200] A. Koubaa, M. Alves, and E. Tovar. Energy and delay trade-off of the GTS allocation mechanism in IEEE 802.15.4 for wireless sensor networks. *International Journal of Communication Systems*, 20(7):791 – 808, 2006.

[201] A. Koubaa, M. Alves, and E. Tovar. i-GAME: An implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks. In *ECRTS*, 2006.

[202] L. Cheng, X. Zhang, and A. G. Bourgeois. GTS allocation scheme revisited. *Electronics Letters*, 43(18):1005–1006, 2007.

[203] S. E. Yoo, D. Y. Kim, M. L. Pham, Y. M. Doh, E. C. Choi, and J. D. Huh. Scheduling support for guaranteed time services in IEEE 802.15.4 low rate WPAN. In *IEEE RTCSA*, 2005.

[204] C. Na, Y. Yang, and A. Mishra. An optimal GTS scheduling algorithm for time-sensitive transactions in IEEE 802.15.4 networks. *Computer Networks*, 52(13):2543–2557, 2008.

[205] Y. K. Huang, A. C. Pang, and H. N. Hung. An adaptive GTS allocation scheme for IEEE 802.15.4. *IEEE Transactions on Parallel Distribbuted Systems*, 19(5):641–651, 2008.

[206] D. Gross and C. M. Harris. *Fundamentals of Queueing theory*. Wesley interscience, 1998.

[207] Y. Ma and J. H. Aylor. System lifetime optimization for heterogeneous sensor networks with a hub-spoke topology. *IEEE Transactions on Mobile Computing*, 3(3):286–294, 2004.

[208] Texas Instruments. *CC2431 Data Sheet*. `http://focus.tij.co.jp/jp/lit/ds/symlink/cc2431.pdf`.

[209] S. Pollin, M. Ergen, S. C. Ergen, B. Bougard, L.V. Perre, I. Moerman, A. Bahai, P. Varaiya, and F. Catthoor. Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer. *IEEE Transactions on Wireless Communication*, 7(9):3359–3371, 2008.

[210] Y. Xiao and J. Rosdahl. Throughput and delay limits of IEEE 802.11. *IEEE Communication Letters*, 6(8):355–357, 2002.

[211] H. Zhai, Y. Kwon, and Y. Fang. Performance analysis of IEEE 802.11 MAC protocols in wireless LANs: Research articles. *Wireless Communications and Mobile Computing*, 4(8):917–931, 2004.

[212] P. Park, P. Di Marco, C. Fischione, and K. H. Johansson. Accurate delay analysis of slotted IEEE 802.15.4 for control applications. Technical report, KTH, 2010.

[213] D. Lymberopoulos and A. Savvides. XYZ: a motion-enabled, power aware sensor node platform for distributed sensor network applications. In *ACM/IEEE IPSN*, 2005.

[214] P. Park, C. Fischione, A. Bonivento, K. H. Johansson, and A. Sangiovanni-Vincentelli. Breath: a self-adapting protocol for wireless sensor networks in control and automation. In *IEEE SECON*, 2008.

[215] J. Van Greuen, D. Petrovic, A. Bonivento, J. Rabaey, K. Ramchandran, and A. Sangiovanni-Vincentelli. Adaptive sleep discipline for energy conservation and robustness in dense sensor networks. In *IEEE ICC*, 2004.

[216] K. J. Aström and B. Wittenmark. *Computer-Controlled Systems*. Prentice Hall, 1997.

[217] P. Park. Protocol design for control applications using wireless sensor networks. Technical report, KTH, 2009. Licentiate thesis.

[218] KTH. *IEEE 802.15.4 Implementation based on TKN15.4 using TinyOS*. `http://tinyos.cvs.sourceforge.net/viewvc/tinyos/tinyos-2.x-contrib/kth/tkn154-gts/`.

[219] P. Di Marco, P. Park, C. Fischione, and K. H. Johansson. Analytical modelling of IEEE 802.15.4 for multi-hop networks with heterogeneous traffic and hidden terminals. In *IEEE GLOBECOM*, 2010.