

Copyright © 2004, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

**A UNIFIED DATA-LINK ENERGY MODEL
FOR WIRELESS SENSOR NETWORKS**

by

Lizhi Charlie Zhong

Memorandum No. UCB/ERL M04/12

21 May 2004

**A UNIFIED DATA-LINK ENERGY MODEL
FOR WIRELESS SENSOR NETWORKS**

by

Lizhi Charlie Zhong

Memorandum No. UCB/ERL M04/12

21 May 2004

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

**A UNIFIED DATA-LINK ENERGY MODEL
FOR WIRELESS SENSOR NETWORKS**

by

Lizhi Charlie Zhong

B.S. (Tsinghua University) 1993
M.S. (New Jersey Institute of Technology) 1995

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy
in

Engineering-Electrical Engineering
and Computer Sciences

in the

GRADUATE DIVISION

of the

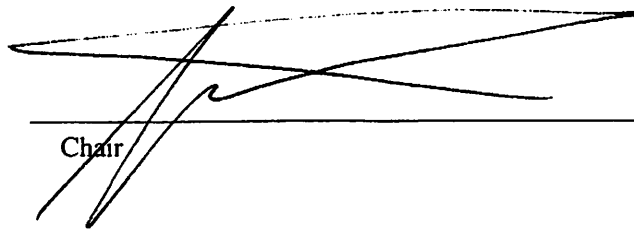
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Jan M. Rabaey, Chair
Professor Pravin Varaiya
Professor Dorit Hockbaum

Spring 2004

The dissertation of Lizhi Charlie Zhong is approved:


Chair _____ Date 5/18/04

Kevin Baranig
_____ Date 5/19/04

Don't Hu
_____ Date 5/20/04

University of California, Berkeley

Spring 2004

**A UNIFIED DATA-LINK ENERGY MODEL
FOR WIRELESS SENSOR NETWORKS**

Copyright © 2004

by

Lizhi Charlie Zhong

ABSTRACT

A UNIFIED DATA-LINK ENERGY MODEL FOR WIRELESS SENSOR NETWORKS

by

Lizhi Charlie Zhong

Doctor of Philosophy in Engineering-Electrical Engineering and Computer Sciences

University of California, Berkeley

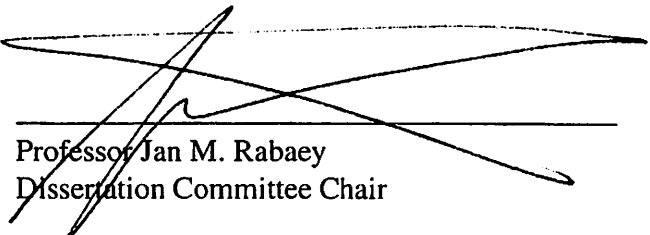
Professor Jan M. Rabaey, Chair

Low power consumption is the most important requirement in wireless sensor networks. Many questions arise in the design of such low-power wireless networks. How many channels are needed? Is it better to use random access or dedicated channel assignment? What is the proper data rate for the radio? When is power control effective and when should power management be used instead? How low can the power consumption be for given Quality of Service (QoS) requirements?

Analytical models provide deep insight into these key issues. For this reason, a framework of analytical models has been established in this dissertation for wireless sensor networks. Special attention is given to the energy models for the data-link layer, because its energy consumption dominates the overall energy consumption. The data-link layer models and the abstracted models of the interacting layers are integrated in this framework to expose the dependence of the overall power consumption on the design parameters of wire-

less sensor networks. Being integrated is what makes this framework unique. In fact, this integrated framework is very useful to the designers of wireless sensor networks: it provides venues and guidelines for energy reduction and design improvements. Furthermore, it can be a place to raise and address fundamental questions. However, the existence of the closed feedback loops makes the integration of the models very hard. Methodology of solving this closed-loop problem is given in this dissertation. The conditions imposed by the convergence on individual parts are also specified.

The integrated modeling framework presented in this dissertation uses an analytical approach, which provides designers better insight into the low-power design of the data-link layer than simulations or experiments. It is superior to other analytical approaches on that it considers the interactions between the components, enabling designers to reduce the overall power consumption even further. The validity of the models in this framework has been verified using OMNET++ network simulations. In addition, the guidelines resulting from this framework have already had a real impact on the design of an actual sensor network.



Professor Jan M. Rabaey
Dissertation Committee Chair

I want to dedicate this thesis to my dear wife, Han Pan and my lovely children, Patrick and Isabel.

Table of Contents

List of Figures	v
List of Tables	vi
1 INTRODUCTION	1
1.1 Emerging Wireless Networks	1
1.1.1 High-speed wireless networks	1
1.1.2 Low-power wireless networks	3
1.2 Wireless Sensor Networks Applications	3
1.2.1 Impact to the society	4
1.2.2 Smart home	6
1.2.3 Scientific research	8
1.2.4 Medical applications	9
1.2.5 Military applications	9
1.3 Sensor Networks Characteristics	12
1.4 Projects on Sensor Networks	12
1.5 Motivation	14
1.6 Existing Work	18
1.6.1 Low-power data-link designs	18
1.6.2 Modeling approaches	20
2 FUNCTIONAL DECOMPOSITION	23
2.1 The Unified Modeling Language	23
2.2 Functional Descriptions	24
2.3 Abstracted Models	25
2.3.1 Network traffic model	25
2.3.2 Physical layer model	33
2.3.3 Channel models	38
3 ANALYTICAL MODELS FOR THE DATA-LINK LAYER	43
3.1 Media Access Control Models	43
3.1.1 ALOHA	44
3.1.2 Carrier sense multiple access	44
3.1.3 Multi-channel media access control	46
3.2 Error Control Model	47
3.3 Link Layer Models	49
3.3.1 Link layer 1: positive acknowledgment	49
3.3.2 Link layer 2: extended negative acknowledgment	51
3.4 Power Management Model	52

3.4.1	The full duty cycle receiver	53
3.4.2	The cycled receiver	54
3.4.3	The preamble sampling	57
3.4.4	The wakeup radio	59
3.4.5	Multi-channel power management	60
3.5	Power Control Model	60
4	INTEGRATION OF MODELS	63
4.1	Model Interfaces	63
4.1.1	Media access control model interface	64
4.1.2	Error control model interface	64
4.1.3	Link layer model interface	65
4.1.4	Power management model interface	65
4.1.5	Power control model interface	67
4.2	Interactions between the Models	67
4.3	Closed Loop Problem	72
4.3.1	Fixed point theorems	72
4.3.2	Optimization after integration	75
5	VERIFICATION OF THE MODELS	78
5.1	Design Metrics	78
5.2	Verification of the Models	80
6	ANALYSIS AND DESIGN	83
6.1	Impact to the Physical Layer Designs	84
6.1.1	Impact of the radio data rate	84
6.1.2	Channel hop scheme and its analysis	86
6.2	Impact to the Data-link Layer Designs	90
6.2.1	Comparison of the wakeup radio to the cycled receiver	90
6.2.2	The joint design of the power control and the power management	91
6.3	Impact to the Network Layer Design	97
6.3.1	Introduction and motivation	97
6.3.2	Description of the generic sensor node radio architecture	99
6.3.3	Selected coding approaches	102
6.3.4	Definition of the single hop asymmetric structure	106
6.3.5	Comparison to the multi-hop structures	109
6.3.6	Summary	111
6.4	Future Research	112
6.4.1	Power-efficient designs	113
6.4.2	Fundamental bounds on power consumption	113
6.4.3	The network lifetime	113
7	CONCLUSION	116

References

List of Figures

1.1	The overall system	17
2.1	Network used to collect network traces	27
2.2	Network traffic at the center node	28
2.3	Network traffic at the center node (zoom in)	29
2.4	Network traffic at the edge node	30
2.5	Network trace from the test bed	30
2.6	The transmit and receive chain	34
2.7	The operation modes	36
2.8	Packet success rate as function of time	39
2.9	Gilbert-Elliott model	40
2.10	Impact of burst length on average packet error rate	42
3.1	β as function of bit error rate	48
3.2	UML sequence diagram of a four-way handshake	50
3.3	The cycled receiver	54
3.4	The preamble sampling	58
4.1	Media access control model	64
4.2	Error control model	65
4.3	The link layer model	66
4.4	Power management model	66
4.5	Power control model	67
4.6	Interactions between the data-link layer components	68
5.1	Network used to verify the models	81
5.2	Comparison of the numerical result to the simulation result	81
5.3	Average delay	82
6.1	Impact of the radio data rate	85
6.2	Impact of the network density on the outage probability	89
6.3	Comparison between the wakeup radio and the cycled receiver	91
6.4	Optimal hop distance as a function of the network density	94
6.5	Total transmit power as function of range	103
6.6	Breakdown by the components	104
6.7	Coding power as function of the distance to the Shannon bound	105
6.8	Power consumption in the BASE	107
6.9	Comparison to the multi-hop approach	111

List of Tables

1.1	Comparison between several wireless networks	13
1.2	Comparison of different modeling techniques	21
3.1	Design examples modeled	43
6.1	Three design zones based on network density	97

Acknowledgments

First I want to thank Professor Jan M. Rabaey, my thesis advisor, for years of support, encouragement and valuable advices. He is a true leader. As his student, I also learn to be one. In fact, working in his group, I have had more opportunities than ever to speak in public and to interact people from different fields. I would also like to thank Professor Kanan Ramchandran, Professor Dorit Hochbaum, Professor Jean Walrand and Professor Pravin Varaiya for providing important feedback on my research.

I spent the first couple of years of my graduate school on taking courses. I thank every professor in these classes for what he has taught me. In particular, I would like to thank Professor Michael Honig, a visiting professor from Northwestern University, for teaching the best communications course I have ever had and for being my reference in my recent faculty applications. I also want to thank Professor Edward A. Lee for teaching me the fixed point theorems, which have become the founding stone in my research. In these classes, I became friends with many people too, who have made my school life ever more wonderful. My thanks go to all of them: Lixia Zhou, Roy You, Rahul C. Shah, Chunlong Guo, Wei Mao, Carol Yeo, Gia, Jason, Christine, Sharon, Joyce, Karen, Cindy, Sun, Nancy, Orchuma Sakdamnuson, Amit Mahajan, Xiaoyu and Li Weng. In addition, I want to thank my SmartRoamer team members: Yuhong Xiong, Yashesh Shroff and Conie Song, for giving me a chance to prove myself in this high-tech venture.

The Berkeley Wireless Research Center (BWRC) is where I spent most of my time doing research. The people in the center are family to me. I would like to give my thanks

to everyone in the Pico Radio group for collaborating with me on this exciting project. I would also like to thank Professor Adam Wolisz, a visiting professor from TU Berlin, for offering me many constructive suggestions. Martin Kubisch and Andreas Willig, also from TU Berlin, have given me the impression that every student from TU Berlin is smart and hard-working. I have learned a great deal from them during the long discussions I had with them. I had a great summer in 2001 too, working with Xu Mei, Gerald Wang, Johnathan Reason and Ruth Wang to build the current test bed. The staff in Cory and BWRC are just fabulous. I can not thank Ruth Gjerde enough for all she has done for me and I want to acknowledge Tom Boot for the excellent service he provided.

My colleagues in STMicroelectronic Inc. are great too. I highly appreciate the help and support from Bhusan Gupta, Thomas Descamps, Engling Yeo, Varghese George, Srikanth Murror, Peimin Chi, Fred Raynal, Fred Revenu, Jim Xie, Benjamin Coates and Lana Scoville.

Lastly, but certainly not the least, my family has played a very important role in my graduate student life. I would love to thank my dear wife, Han Pan, for making me the happiest man in the world. My parents have given me both emotional and financial support throughout my life. They even volunteered to taken care of my children recently, so that I could focus on my dissertation. Patrick and Isabel, my son and daughter, are so adorable that I want to thank them for the many happy moments they have brought to my life.

CHAPTER 1

INTRODUCTION

1.1 Emerging Wireless Networks

Over the past few years, many wireless networks have emerged. Some of them are pushing for very high data rates, while the others focus more on ultra low power consumption.

1.1.1 High-speed wireless networks

As wireless internet access becomes increasingly popular, users are demanding ever higher and higher data rates. To support high data rates, either more bandwidth must be allocated, or advanced signal processing has to be used to improve the spectral efficiency.

Ultra wide band (UWB) for example is allocated a huge trunk of spectrum. The IEEE 802.15.3a uses the UWB as its physical layer (PHY) to provide data rates up to 480 Mbps (at 5 m) in a personal area network (PAN). The application there is to implement the very popular IEEE 1394 interface over a wireless link.

Advanced coding has been used to get closer to the famous Shannon bound. Other ways to improve the spectral efficiency include the use of multiple-input multiple output (MIMO) systems and orthogonal frequency division multiplexing (OFDM).

The IEEE 802.15.3, also known as WiMedia, uses Trellis Coded Modulation (TCM) to provide data rates up to 55 Mbps in the 2.4 GHz band at ranges up to 50 m, enabling high

rate multimedia transfer over a PAN.

The IEEE 802.11a/g uses both OFDM and convolutional codes to provide data rates up to 54 Mbps over a local area network (LAN). However, the IEEE 802.11n is the state of art in the 802.11 family. It provides at least 100 Mbps effective throughput measured at the media access control (MAC) interface. The throughput, different from the radio data rate, is what an end user really cares about. As a matter of fact, preliminary simulations indicate that to obtain a throughput of 100 Mbps, the radio data rate must be at least 300 to 400 Mbps. There are currently two candidates for the PHY of the IEEE 802.11n. The first one employs a MIMO system, which uses M antennas at the transmitter and M antennas at the receiver and provides M times the peak throughput of a single-input single output (SISO) system without increasing the frequency bandwidth. The other candidate, wide band adaptive OFDM, increases the bandwidth from 20 MHz (as in 802.11a) to multiples of 20 MHz up to 80 MHz. To achieve high data rates, both candidates use Low Density Parity Check (LDPC) codes, which can come close to the Shannon bound within 0.1 dB. An IEEE 802.11n network will replace the 100-BaseT network, widely used in enterprises. It will also be used in Hotspots. But the coolest thing about 802.11n is that it eliminates the cable between a portable Liquid Crystal Display (LCD) TV and cable outlet. As a result, one can take his portable LCD TV to his backyard or kitchen even though the cable outlet is in his living room. In fact, even a High Definition Television (HDTV) signal can be carried in a 802.11n network.

Providing high data rates over a even longer distance (e.g. wide area network (WAN))

is much harder to do but makes it possible for users to access the Internet from almost anywhere. Third generation (3G) cellular wireless system can deliver data rates up to 384 kbps by using Turbo codes. The IEEE 802.20, on the other hand, is able to provide data rates up to 3 Mbps over a comparable distance through the use of LDPC codes and OFDM.

1.1.2 Low-power wireless networks

Power consumption has always been a concern for wireless networks. In certain applications, this concern surpasses the needs for high data rates. In fact, some emerging wireless networks are targeted at low data rate applications, where they can use simple radios and aggressive power management to achieve ultra low power consumption.

The Zigbee/IEEE 802.15.4 for example is targeted at simple, low cost, low power, low data rate (2 kbps to 200 kbps) applications like home control, building automation and industrial automation.

The wireless sensor network is yet another type of emerging wireless network where low power consumption is more important than high data rates. It will be discussed in great details in the rest of this chapter.

1.2 Wireless Sensor Networks Applications

The idea of a sensor network is simple enough. Hundreds of sensors are spread out. Each of them will measure the temperature, light intensity, humidity, noise level and airflow speed etc. in its locality. But the beauty of the sensor network is that all these nodes can be

self-organized into a network, over which they can do cooperative processing to accomplish tasks that they can not do individually.

The idea of sensor network applies directly to the office building environment, where sensors can work together to create a micro-climate based on personal settings. Or they can be used to minimize the energy consumption of a building without sacrificing the comfort of its users. Similar things can be done inside a car, airplane, bus or train, ... any public transportation or public place one can name.

But the applications for sensor networks go far beyond these. It will be shown in the following paragraphs that they have a big impact on our society, change the way we live, or they can help with scientific research. In addition, they are used in medical and military applications. There is no doubt that wireless sensor network is becoming a very important component of wireless data networks.

1.2.1 Impact to the society

Due to the energy crisis, people have become more concerned about their utility bills than a fine tuning of their personal climate. As a result, the projects on building environment have migrated from optimizing users' comfort to minimizing their spendings on utilities while still maintaining a desired level of comfort. Study shows that an energy distribution infrastructure is severely impacted by the peak usage of utilities. If the capacity of a utility company is below this peak, ridiculously higher prices need to be paid to buy energy from a neighboring state. This cost eventually passes on to the end users. A strat-

egy called “demand response” tries to address this problem from both the utility company and end user side. A utility company will dynamically adjust its utility prices based on the instantaneous demand for its energy. A meter is installed in every user’ home, which has access to the real-time utility prices. Also needed in a user’s home is a sensor network with controllers, sensors and actuators. The controller will send commands to the actuators based on the readings from the sensors, the energy price at the time and the comfort level a user sets. For example, if the average temperature is above what a user feels comfortable with, and the price for electricity is not very high, the controller will turn on the air conditioning. For errands that can tolerate a high latency, the controller can schedule them to be done at the time when the energy price is low. For instance, the controller will start the laundry machine in the evening. Thus the “demand response” operation, made possible by the use of sensor networks, results in a better balanced energy distribution and consumption infrastructure, saving money for both utility companies and end users.

Traffic jam is a big headache to the commuters, but sensor networks can be used to alleviate this problem. Sensors can be installed along a freeway to measure the speed of the vehicle passing by. This information is sent via the sensor network to a central controller. The information obtained after processing the raw data can help drivers to see the traffic ahead of him, who can take an alternative route to avoid the traffic jam.

Firebug, as another example, is a Global Position Satellite (GPS) enabled wireless thermal sensor. Firebugs can be simply dropped to the ground and self-organized into a wireless network. The data they collect, when coupled with real-time burn model and helicopter-

based real-time topographic and ground cover mapping, can return visualization and adaptive modeling of observed phenomena to the in situ firefighters, helping them put off fires quickly.

Sensors can be used in the interactive museums as well to track which exhibit attracts more people. This information will be used to guide the placement of exhibits. What is more, when users experiment with a exhibit, the data collected by the sensors can be sent via the network to the gateway to the Internet, enabling those data be available on-line for later viewing or reporting.

Lastly, sensor networks are used in asset management to track the location of a particular item, especially where the distance is so long that radio frequency identification (RFID) can not be used. For instance, a car dealer may use a sensor network to locate his cars.

1.2.2 Smart home

Not only do wireless sensor networks change our society, but they also change the way we live. They can be used in our homes, with sensors embedded in appliances, hidden in walls. This ambient intelligence leads to a “smart” home. One can see from the following examples that sensor networks in our homes make us more comfortable and more secure. They also bring a lot of convenience to our lives.

Networked refrigerators are already on the market. A LCD display can be found in such a refrigerator. A user can use it to either watch his favorite TV show or surf the web to search for real-time grocery prices, health-nutrition tips or cooking information. With

a sensor network, such a refrigerator may become even smarter. A person can remotely find out what is in his refrigerator. What is more, the refrigerator may even suggest recipes based on the food inside. Once the user has chosen a recipe, the refrigerator can also help him locate the ingredients that he already has at the different places of his home and notify him of the additional ingredients to buy on his way home. One more thing such a smart refrigerator can do is to alert a person when the expiration date of a food package is close.

Sensors can make other appliances smarter too. A smart microwave can read the chips embedded in a food packaging to figure out by itself the weight and type of food to defrost. A smart dining-ware can tell its user if he has had enough nutrition from a meal, or if he has already had too much fat.

A wireless sensor network can also create a micro-climate in a home just as in an office environment. Together with video cameras and an alarm system, it can detect an intruder and notify the police of the incident. It is also able to send a warning to a person either when his children are doing something dangerous, or when there is a fire in a room. For people who really love music and would like to have music following them when they walk from one room to another, sensors can detect their motion and turn on the speakers in the second room automatically.

Sensor networks can also be used to create a net surfing experience impossible before. Local sensor networks can recreate the environment a web page intends to bring one to, be it smells, strong wind or cold weather. Similarly, a game designer can put in all the details in a computer game, allowing a sensor network to recreate the scene in a player's

local environment. In fact, the S controller for Microsoft Xbox (a game console) has a vibration feature: during an action game, a player will feel the vibration if he gets hit by his opponent. To many gamers, this is a really neat feature. Sensor networks will certainly go one step further and make gaming a more rich experience. In the future, not only will a gamer get high-definition three-dimensional (3D) graphics and Dolby Digital sound, but he will also get a “U feel” gaming experience.

1.2.3 Scientific research

Sensors can be used in scientific research to take measurements where it is difficult or dangerous, if not impossible, for human beings to go. A wireless sensor network makes it possible to have many of these sensors around and makes it easier for a scientist to get data from the sensors. Consequently, a scientist can study the behavior of his object in a more detailed manner.

Sensors have been installed on redwood trees to measure the temperature, humidity, as well as the velocity of the water flow within the redwood trees. These data enable the scientists to study the Maritime Fog’s impact on the water relations of a redwood forest. However, currently the sensors are all “wired” and therefore can only sample a very small fraction of what need to be sampled. Thus a wireless sensor network is in demand here.

The structural health monitoring of the Golden Gate Bridge is another application to which a wireless sensor network is essential. Sensors can monitor vibration of bridge and detect unusual behavior by wind, earthquake or local damage. A wireless network of these

sensors makes high fidelity sampling a reality, where high accuracy, high frequency with low jitter and large amount of data are required.

Other places where wireless sensor networks are used to do scientific research include the outer space and the desert. They are used underwater too.

1.2.4 Medical applications

In one medical application, tiny CMOS image sensors are sent into the body of a patient, each of which takes a picture of a part of an infected organ. Through a wireless sensor network, these local views are sent to a central controller to produce a global view of the organ, which makes it possible for doctors to inspect the infected organ without any surgery. In some cases, the controller can even use this information to control a robot arm in a robot-aided surgery.

Wireless sensor networks are now used in more and more medical applications. A doctor can remotely monitor the health of his patients thanks to the networks of medical sensors in their homes. As a result, home health care becomes a feasible option for more patients. Many patients like home health care because it eliminates the huge cost associated with staying in a hospital and gives patients the comfort of their homes.

1.2.5 Military applications

Wireless sensor networks in the battlefield extend the vision and hearing of a soldier: information retrieved by either the sensors carried by himself and other soldiers or satellites and airplanes is relayed to him, giving him a better knowledge of the environment he is and

the enemy he is against. Soldiers can communicate with his neighbors silently and organize themselves into a wireless network based on military operations. Rich information can be exchanged over such a network, leading to a much better collaboration between the soldiers.

Military applications have their own requirements for wireless sensor networks. The range of each transmission must be short to minimize the possibility of being eavesdropped. The power consumption shall be low to ensure the radio every soldier carries is small, lightweight and can be operated for a very long time. Global synchronization should not be a requirement for proper network operation, so a network can be setup on the fly anywhere in the battlefield and the operation of the network is robust even when soldiers come in and out of this network. A central control point is not desired, since the network would have a single point of failure. In general, military applications would like to see a very robust network. As a matter of fact, the inherent redundancy in many wireless sensor networks further increases the reliability of the communications and makes wireless sensor network a good candidate for many military applications.

It is not the intent of this thesis to enumerate every sensor network application. The key message from the previous paragraphs is that wireless sensor networks are very useful and important, which provides the motivation for the research documented in this dissertation. A final comment on the applications is that, a "sensor" in a sensor network does not have to be limited to a mechanic device that measures certain parameters. It can be anything that has certain knowledge of its local environment. Just like the traditional sensors, these

"sensors" can be self-organized into a network. According to Metcalfe's Law, the value of a network is proportional to the square of the number of "sensors" in it. There is no doubt that these "sensors" can accomplish a lot more by forming a sensor network. Design used in a traditional sensor network can be ported easily to these more generalized sensor networks. For example, a complicated decoder consumes a lot of power and cuts down the battery life. An idea is to replace a complicated decoder with many simpler decoders. Certain block codes can be decoded in parallel. Hence each "sensor" will decode one parallel path and together they can accomplish a much more complicated decoding task. This decoding strategy is called network decoding. With this new definition of "sensor", almost everything can be thought of as a form of sensor network. One can imagine how diverse the sensor network applications can be.

Despite the diversity of the applications in sensor networks, the following requirements are true for all of them: low power consumption, low cost, wireless and self-configuring. Among these requirements, the low power consumption requirement is the most important. It would be a maintenance nightmare having to replace the batteries of hundreds of sensors every day or even every month. In some applications, the sensors are buried inside a wall so it is impossible to take them out to replace their batteries. In yet some other applications, the sensors are used just once and live only as long as their batteries can support. So it is very important to keep the power consumption low. If the power consumption can be kept sufficiently low ($< 100\mu W$), power obtained through energy scavenging techniques will be enough to keep the sensors self-powered. The outcome is very exciting: not only is

the maintenance effort greatly reduced, but the network operation is also more robust, free from the impact of dead nodes. In addition, nodes are lighter and smaller.

1.3 Sensor Networks Characteristics

A wireless sensor network (WSN) is very different from some other popular wireless networks: cellular network, wireless local area network (WLAN) and Bluetooth. Compared to these other wireless networks, a wireless sensor network has much more nodes in a network and the distance between the neighboring nodes is much shorter. On the other hand, the application data rate in a wireless sensor network is much lower too. Due to these characteristics, the power consumption of a sensor must be and can be made much lower. To keep the cost of the entire sensor network down, the cost of each sensor needs to be very low, in the order of cents. Size is important too. A smaller size makes it easier for a sensor to be embedded in the environment it is in. Lastly, in many sensor networks, the data collected is highly correlated in time and/or space. This inherent redundancy can be either reduced or exploited to improve the energy efficiency. It can also provide additional reliability. Table 1.1 compares the characteristics of these networks.

1.4 Projects on Sensor Networks

Wireless Sensor network has been and will still be an active area of research. Many universities and research labs have initiated projects on wireless sensor networks. Center for Information Technology Research in the Interest of Society (CITRIS) for example

involves many universities in the research of sensor networks. The University of California, Los Angeles (UCLA) is one of the first universities that started research on sensor networks. Pottie' group, Kaiser's group and Rockwell Science Center initiated the research on wireless integrated network sensors (WINS) in 1993. Srivastava's and Estrin's groups also turned to sensor networks several years later and have developed algorithms for media access control(MAC) and class based routing.

At the University of California, Berkeley (UCB), more than ten faculty members are involved with different aspects of sensor networks. To name a few, Culler's group and Intel Labs have built the TinyOS, a lightweight operating system that is very well-suited for sensor networks and have made efforts on the design of the MAC and the network layer as well as the development of locationing algorithms. Pister's group launched the Smart Dust project in 1999. The project deals with the application of tiny sensors dropping out of an airplane, taking measurements on their way to the ground. These sensors are made small and robust. Optical communication is used between them. A radio frequency (RF) mote

Table 1.1: Comparison between several wireless networks

Types	Number of nodes	Range	Data rate	Mobility
Cellular	Large	Long	Medium	High
WLAN	Small	Medium	High	Medium
Bluetooth	Small	Short	Medium	Low
WSN	Large	Very short	Low	Low
Types	Power	Cost	Size	Redundancy
Cellular	High	High	large	low
WLAN	Medium	Medium	Medium	low
Bluetooth	Low	Low	Small	low
WSN	Very low	Very low	Very small	high

has also been built using the Chipcon cc1000 radio and other off-the-shelf components. Its size is very small and its average power consumption is reasonably low. In fact, it has been used in the test beds of many projects, including those of UCLA and Intel Labs. Rabaey's group in the Berkeley Wireless Research Center (BWRC) started the Pico Radio project in 1999 too and has done extensive research on every aspect of sensor network, from protocol design to radio design and energy scavenging techniques.

Chandrakasan's group in MIT is very active in this area as well. The group has focused on applications with clusters of sensors. Centralized control is assumed and leader selection is used to rotate the leader position. Global synchronization is used in these applications.

Besides these major projects, many individuals have contributed a rich amount of information to all aspects of sensor networks.

1.5 Motivation

It has been shown that wireless sensor networks can be used in many applications and the research on them is very active. What problem in wireless sensor networks does this Ph.D. research try to address? What is the motivation behind? As mentioned earlier, very low power consumption is crucial to wireless sensor networks. However, it is no easy task to bring the overall power consumption of a sensor to a very low level. Designers of wireless sensor networks have attempted to optimize the power consumption of a component of the overall system in an isolated fashion, only to find that when the power consumption of one component is pushing down, that of another component goes up. It has become evident

to the designers that how the overall power consumption is related to the design parameters of a wireless sensor network is the most important problem in the design of these networks. This research will attempt to solve this very problem.

A closer look at this problem indicates that a sensor node system can be compared to a box with hundreds of knobs. Its overall power consumption is the output of this box and its design parameters are the different knobs. It is not obvious how to tune these many knobs to achieve low power consumption. Examples of these design parameters are the transmit power level and the duty cycle. It is well known that an optimal power level exists which minimizes the total transmit energy consumption. However, if a network is dense enough, one can reduce the overall energy consumption by using a higher transmit power level. The reason is as follows: as the transmit power level increases, more candidates will be available for data forwarding. Now one can lower the duty cycle of these candidates such that statistically a node still has roughly the same number of active candidates as before. Consequently, the energy consumption in channel monitoring reduces and the total energy consumption too if the network is sufficiently dense. Similarly, interdependencies exist between the overall power consumption and other design parameters. The complexity of these interactions is at such a magnitude that one cannot comprehend them without proper modeling.

Models are simplifications of complex systems and can reveal relevant information about them. Thus models will be essential to sensor networks as SPICE models have been to integrated circuits.

How can models be used to reveal the dependency of the overall power consumption on the design parameters of a wireless sensor network?

A two-step approach to the above problem will be taken. In the first step, functional decomposition is performed:

- A complex system like a sensor node, first needs to be divided into smaller but simpler subsystems based on functionality. An iterative process using the unified modeling language (UML) helps one to arrive at the decomposition shown in Fig. 1.1.
- An analytical model is then developed for each component. The energy consumption of the data-link layer (DLL) dominates the total energy consumption, thus this research will focus on the data-link layer and use higher abstraction on the other layers.

The second step is the integration of the models of these components. This integration (often ignored by other designers) is essential for one to link the total power consumption to the design parameters.

- First the interface of every model is defined. These interfaces are not design-dependent even though the models are.
- Next the interactions between the components are modeled.
- Lastly, the closed feedback loops (in Fig.1.1 there exists a closed loop between the media access control (MAC) and the link layer, shown with a thick line) are solved using the fixed-point theorems and an “optimization after integration” technique.

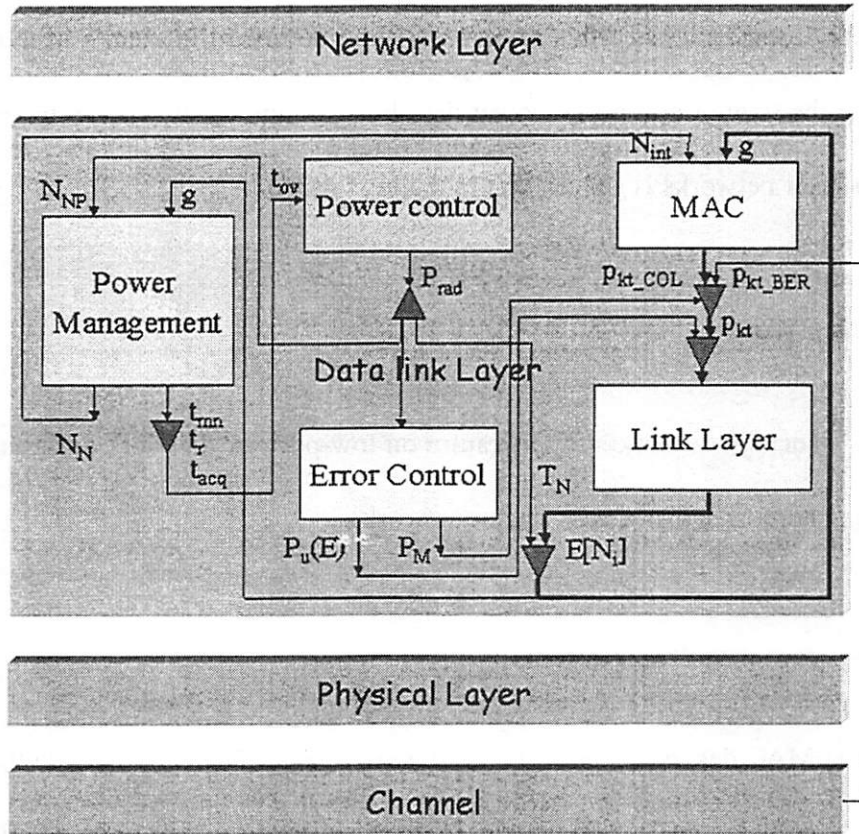


Figure 1.1: The overall system

This approach results in an integrated data-link energy model for wireless sensor networks, which exposes the dependence of the overall energy consumption on the design parameters of a wireless sensor network, especially those of the data-link layer. Based on the insights gained from this framework, guidelines for energy deductions and design improvements have been derived. In fact, these guidelines have already had a real impact on the design of an actual sensor network. To illustrate this point, several emerging designs developed at the BWRC will be described at the end of this dissertation.

The main contributions of this research will be described in details in the following chapters. But the state of art on low-power data-link designs and the modeling techniques of wireless sensor networks is given first in the next section.

1.6 Existing Work

Below is a survey of the existing literature on low-power data-link designs as well as the modeling approaches in wireless sensor networks.

1.6.1 Low-power data-link designs

Low power MAC design is one of the most active areas. The main focus in this area is to reduce the collision rate, so the energy in retransmissions can be reduced. One way to reduce collisions is to coordinate the use of the channel through scheduling. Centralized scheduling was used in [2] while the distributed versions can be found in [3] and [4]. Another way to reduce collisions is to use multiple channels. [5] uses random access in each channel, while [6]-[9] use dedicated channel assignment. For example, a node's locally unique identification (ID) [1] is used as the channel for transmission or reception in [9].

Power control (PC) is starting to receive more and more attention in sensor networks. The issue is how to achieve the connectivity at the minimum energy cost. [10] presented a distributed power control algorithm that guarantees the connectivity while ensuring the path from each node to a designated master has the minimum power consumption. [11]

starts from a connected network and reduces the transmit power level as much as possible while maintaining the connectivity. [12] provided a distributed algorithm, using which not only is a network connected, but a large percentage of nodes in the network are also bi-connected. [10] assumes accurate location information while [11] and [12] requires directional information. Two simple power control algorithms that do not require location or directional information were presented in [13]. They do not guarantee the connectivity, instead they adjust the power level on per node basis to make sure the number of neighbors of every node is within a given range.

The biggest power saving in sensor networks actually comes from the power management (PM). Since a node in a sensor network is inactive most of the time, the energy consumption when it is inactive dominates the total energy dissipation. For this reason, many have looked into how to turn off a node when it is not active. Scheduling ([2]-[4] and [14]) helps a node know when to sleep and when to wake up. But delay is a big problem in these schemes besides the overhead in handshakes. [5] proposed the idea of a wakeup radio, which can monitor the channel for the main radio but at a much lower energy cost. As a result, the main radio can be turned off until the wakeup radio detects the activity in the channel and wakes it up. Compared with the scheduling algorithms, this approach has very small delay and overhead. [15] used a dedicated control channel as a wakeup radio to the data channel. The power management was also done on the control channel using an approach similar to the cycled receiver discussed in Chapter 3. [16] turns off a node not only when it is inactive, but also when there are neighbors who can do the job for it.

There has also been quite a bit of research on the joint design of several DLL components. [17] pointed out that when the transmit power level is increased, the link quality improves. As a result, the required number of retransmissions goes down. So there is a tradeoff in setting the power level. [2]-[4] used scheduling in both the MAC and the PM. [9] uses the local ID as the channel, allowing the local address and the MAC to share the same overhead. [18] on the other hand assigns a power level and a channel to every link. This link-based power control algorithm brings down the average power consumption. It also works together with the MAC to reduce the interferences. In addition, the PC and the MAC can share the overhead of assigning the power level and channel. In Chapter 6, a joint design of the power control and the power management is proposed. It follows the idea in [16] and does combined tuning of the transmit power level and the duty cycle of the cycled receiver: the transmit power level is increased on purpose to get more candidates so that the duty cycle of each candidate can be reduced.

1.6.2 Modeling approaches

The modeling approaches used in the above papers and other papers in the literature to evaluate the performance of a design can be categorized into three classes: simulations, experiments and analytical approaches. Their pros and cons are summarized in Table 1.2. Analytical approaches provide designers much more insight to the design of the DLL within much less time than simulative or experimental ones. They also enable designers to do design optimization. They are the preferred approaches if available.

Table 1.2: Comparison of different modeling techniques

Modeling techniques	Accuracy	Insight	Time to develop	Cost	Controllability
Close form	Low	High	Days	Low	High
Numerical	Medium	Medium	Days	Low	High
Simulation	High	Low	Months	Low	Medium
Test bed	Highest	Low	1+ years	High	Low

The simulative approaches are used in many of the above papers [2][3][5]-[7][9]-[11][13][14][16]-[18]. In these papers, although the algorithms proposed were implemented in either a discrete-event simulator or MATLAB, the other interacting components were not simulated. [11] as an exception, simulated the MAC, the network and application layers besides the power control algorithm it proposed. But the effect of these other components or layers on the PC was not shown in the simulation results provided.

[4] and [16] collected data from the test beds using the Rene motes and the TinyOS developed at the UCB. [19] did performance analysis using the Pico Radio test bed. Even though the test beds have the entire protocol stack and a realistic channel, the interactions between the components were not well studied in any of the above papers.

As for the analytical models, [20][21] gave a good survey of many access protocols. But most of the papers in the survey focused on only the analysis of the throughput and average delay. [22] provided energy analysis for the simplest access protocols (ALOHA). [15] did energy analysis for the two-channel PM mentioned above. [23] provided analytical bounds on the average number of transmissions for a MAC that uses the receiver to synchronize all the potential transmitters. [24] obtained upper bounds on the lifetime of sensor networks,

but the work was mainly done for the power control component only.

The above survey indicates that many designers have simply designed a component and simulated it. Very often they did not model the other components or layers that have impact on the component they designed. Some designers have a channel model or network model in their simulations, but the interactions between those and the component they design are never studied. There are some papers on the joint design of several components/layers. These approaches are quite ad-hoc. In other words, these works lack a systematic approach. The analytical models seen so far are made for a stand-alone component only. It has not yet been seen an integrated framework that can allow designers to bring all the DLL components together with the network layer model, the PHY model and the channel model.

The rest of this dissertation is organized as follows. In Chapter 2, 3 and 4, the functional decomposition and integration procedures of the unified modeling approach will be described. The analytical models built will be verified in Chapter 5, after which the analysis done using this framework is presented and several designs that benefit from this analysis will be introduced.

CHAPTER 2

FUNCTIONAL DECOMPOSITION

2.1 The Unified Modeling Language

The UML is an Industry-standard language that helps users to specify, visualize, and document models of software systems, including their structure and design, in a way that meets the design requirements. In this research, the design requirements were first specified in the UML, followed by the system level use cases and an initial decomposition (represented with a class diagram in the UML)[27]. After the subordinate use cases were developed for these subsystems, if functional dependence between the subsystems were detected, the original decomposition was modified. For example, the power management was later realized to be an important function of the data-link layer. As a result, a PM component was added to the original class diagram. Through this iterative process, a decomposition (Fig. 1.1) was obtained which made it easier for us to develop the models and study the interactions between them.

As shown in Fig.1.1, a system is first divided into several subsystems: the application layer, network layer, data-link layer, physical layer and channel. The breakdown is similar to the Open Systems Interconnection (OSI) reference model. The data-link layer is further divided into several components, such as the media access control, error control, link layer, power management and power control.

The goal of this process is to convert a complex system into simpler systems that are

easier to deal with. Below the functions of these layers and components are briefly described.

2.2 Functional Descriptions

The application layer implements different sensor network applications. It configures the functionality of every node in the network. The network layer does routing and aggregation.

The data-link layer handles everything within one hop. The media access control (MAC) is responsible for managing the usage of channels such that collisions can be controlled. The error control (EC) component detects and /or corrects errors in packets. The link layer specifies the handshakes and handles retransmissions. The function of the power management (PM) component is to turn off a node as much as possible. It also takes care of the synchronization between a sender and its receivers. The power control (PC) component sets the radiated power level. The power control is usually done for one or more of the following reasons: to maintain the connectivity, to reduce the power consumption or to reduce the interference.

The physical layer (PHY) is responsible for sending and receiving a bit stream over a channel. It does acquisition and demodulation/modulation. Although the channel is not part of a design, it is part of the overall system. It attenuates the strength of a signal and adds interferences and noise to the signal.

Once a complex system is divided into simpler subsystems, analytical models are built

for these subsystems. Naturally these models will be tied to the particular designs. For illustration purposes, some commonly used designs (in sensor networks) are presented here for each of these subsystems. In every example, the design is briefly described and only the results of the corresponding model is presented. It needs to be emphasized here that the other designs can also be modeled in a similar way.

2.3 Abstracted Models

The overall system is shown in Fig. 1.1. In this section, the abstracted models for the layers outside the DLL are presented. The analytical models of the DLL will be covered in the next chapter.

2.3.1 Network traffic model

The network traffic is defined as the packets sent from the network layer to the DLL. A network traffic model not only specifies the traffic rate, but also the traffic pattern.

The network traffic model is very important for the computation of the collision rate (in the MAC) as well as the average percentage of time a node spends in the monitor mode (in the power management component). Unfortunately the network traffic depends on the application scenarios, the network design as well as the MAC design. It is unlikely to have a single network traffic model even for sensor networks. So different network traffic models may be obtained for different protocol stacks. Below a model is presented for the network traffic generated by the indoor environment control application, a network layer design

using probabilistic routing [1] and a MAC using carrier sense multiple access (CSMA).

In this application, there are controllers and sensors in the network. The controllers initiate queries and the sensors receiving them periodically report data to the corresponding controllers. The network layer uses a multi-hop routing scheme based on probabilistic forwarding. The probability to use a particular neighbor to forward a packet is based on both the energy needed to go through it to the final destination and its remaining battery life. For instance, if the energy needed to go through a neighbor to the destination is lower, the packet is more likely to be forwarded to this neighbor. On the other hand, if the battery life of this neighbor is low, it makes more sense to forward the packet to other neighbors instead. The MAC uses CSMA, which is described in details in Chapter 3. Basically a random delay is introduced by this MAC to let different nodes to use the channel at different times.

2.3.1.1 Network traces

Network traces from both OMNET++ network simulations and the test bed were collected. Both have the applications and network designs mentioned above. The network simulations were done on a network with 100 nodes (Fig. 2.1). The nodes are randomly placed. Among them there is one controller (node 0, at the right edge of the network) that initiates queries to all other nodes every two minutes. All the other nodes are sensors. Most of them report temperature every five minutes. A small number of them (randomly placed) report light intensity data every thirty seconds.

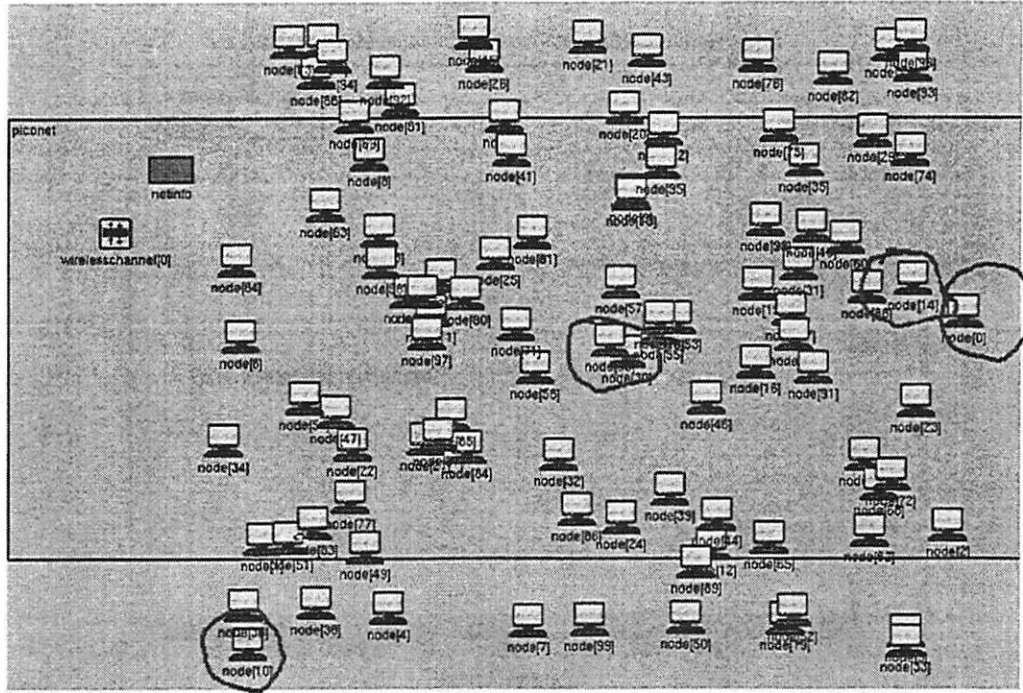


Figure 2.1: Network used to collect network traces

Packet arrival times were recorded at a node in the center of the network (node 90) and a node at the edge of the network (node 10). Fig. 2.2 shows the distribution of the inter-arrival time between the network packets. The data was collected at node 90. It can be seen from Fig. 2.2 that the probability density function (PDF) of the packet inter-arrival time is similar to an exponential PDF. But a spike can also be noticed at 30 seconds, which is the period of light intensity data. This indicates the network traffic has a periodic component in it. Other than this spike, most of the packet inter-arrival times are within one second. When zooming in this zone (Fig. 2.3), one can see the PDF is very close to an exponential PDF of rate 0.05 packets per second.

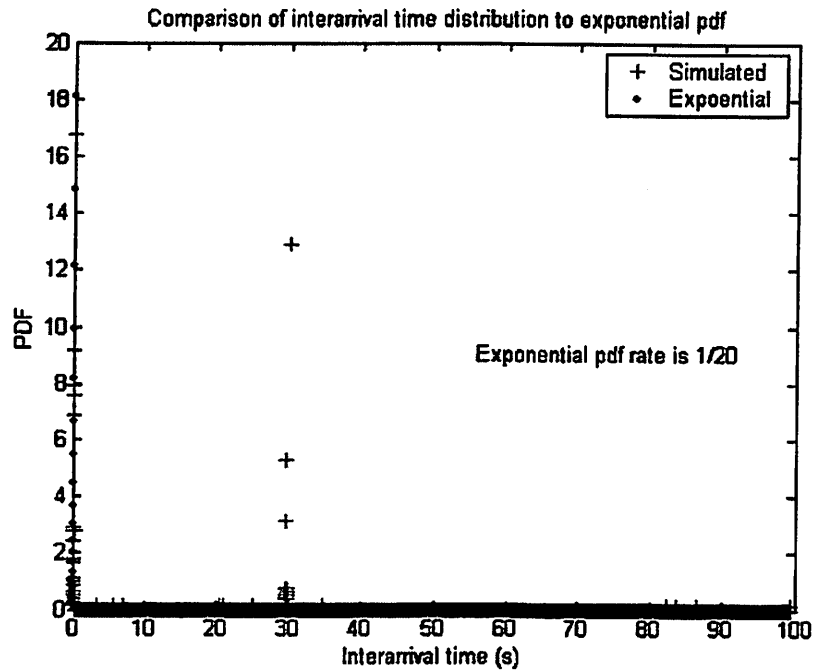


Figure 2.2: Network traffic at the center node

Fig. 2.4 shows the PDF of the packet inter-arrival time at node 10. Not only does one see spikes at 30 seconds, but also at the multiples of 30 seconds. The envelope of the spikes goes down exponentially. This is because the packet from a light sensor may not go through node 10 every time due to the probabilistic routing used. In fact, the further away the sensor is from node 10, the larger the inter-arrival time of its packets at node 10. On the other hand, the number of times that its packets go through node 10 is smaller as well. As a result, the inter-arrival time of the packets from different light sensors at node 10 is at the multiples of 30s. In addition, the larger the inter-arrival time, the less likely it is. If the spikes were removed, the PDF would be very close to an exponential PDF, but at a slower

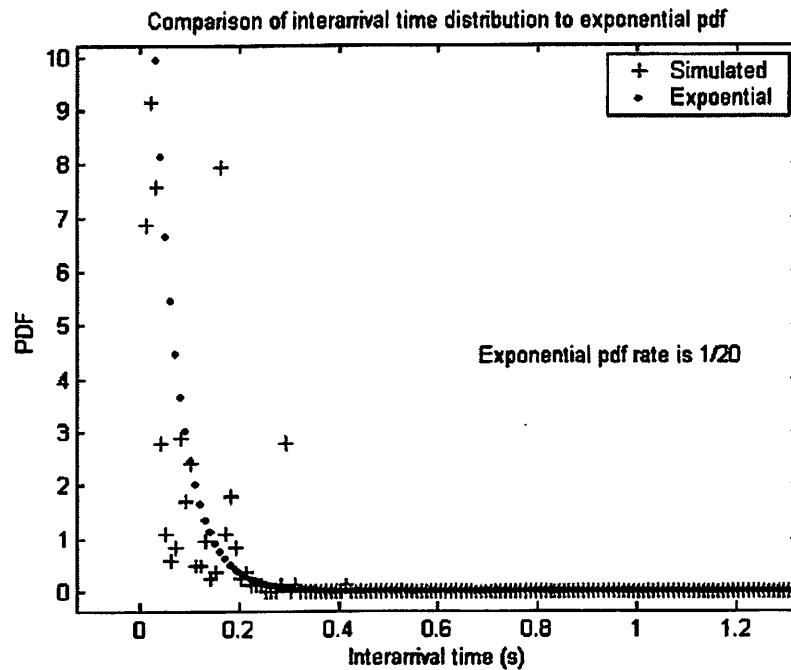


Figure 2.3: Network traffic at the center node (zoom in)

rate, 0.014 packets per second (can be seen after zooming in the one-second zone). This is because node 10 is at the edge of the network where less traffic goes through.

The Pico Radio test bed was also used to record the network traces [19]. The network used has one controller and ten sensors. Fig. 2.5 (taken from [19]) shows the distribution of the time between activities. Here an activity is to send or to receive a packet. One can see from Fig. 2.5 that as the time between activities increases, its probability goes down exponentially.

In summary, the network traces collected from both network simulations and the Pico Radio test bed indicate that the PDF of the inter-arrival time of the network packets is very

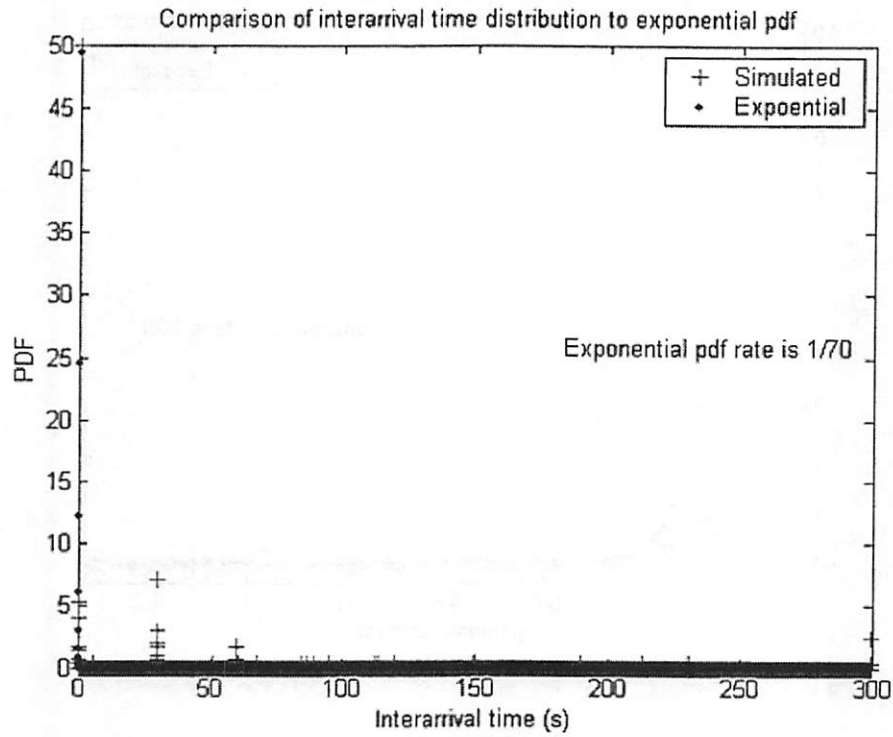


Figure 2.4: Network traffic at the edge node

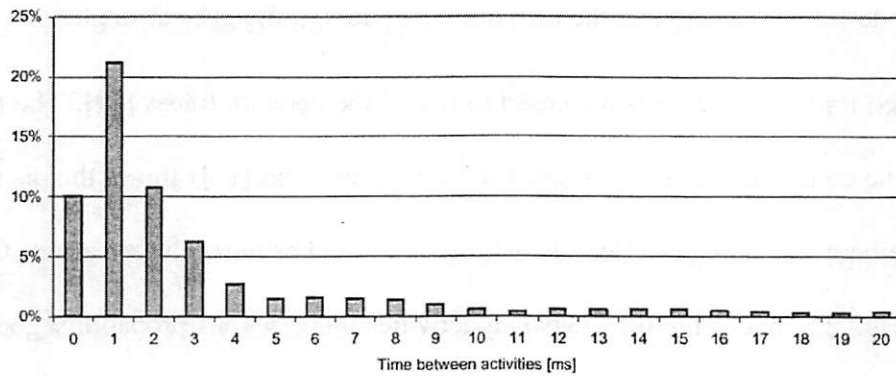


Figure 2.5: Network trace from the test bed

similar to an exponential PDF, but it also has some spikes at the multiples of the periods of the application data.

2.3.1.2 Network traffic model

The network traffic model is built on the network traces presented in the previous section, which indicate that the network traffic is very similar to a Poisson process, but it also has a periodic component.

This phenomenon is explained as follows: in general, the network traffic is the combination of two components: the application traffic generated by the node itself and the forwarding traffic coming from its neighbors. For the indoor environment control application, the application traffic is periodic. This explains why there are spikes at the periods of the application data. As the result of this periodic application traffic, the forwarding traffic has a periodic component too. Due to the probabilistic routing used, the inter-arrival time may also be the multiples of the periods. In fact, if every node was synchronized, the inter-arrival time of the network packets would only be the multiples of the periods. However, the MAC introduces a random delay during backoff. In addition, the clocks of the different nodes are not synchronized. Consequently, the inter-arrival time can have any value. The number of packets that a node needs to forward within a time interval is the count of the network flows going through it in this interval. Here a network flow is defined as the route between a source and its final destination. Assuming the probability for a network flow to go through a particular node is small but equal, the count of the network flows going

through a node is binomially distributed and approaches Poisson when the number of the network flows is very large. Since there are typically many nodes in a sensor network, there are a large number of network flows in it. This explains why the forwarding traffic is very similar to a Poisson process in sensor networks.

Based on these observations, the network traffic is modeled to consist of a deterministic and a random component.

For the indoor environment control application, the deterministic component is the traffic with inter-arrival time at the multiples of the periods. The random component is assumed to be a Poisson process.

The deterministic component causes two neighboring nodes either always use a channel at the same time, or never. As a result, they always collide or never. But the good thing about the deterministic component is that one can predict the arrival time of the packets so he can alleviate the effect of this component on collisions by the use of local scheduling or by adding a long enough random delay. The number of collisions as a result of using a real network traffic are compared with that using a pure Poisson process. Even though backoff and clock drift add some random delay, that delay is not long long enough, so the number of collisions when real network traffic is used is still significantly higher than that of a Poisson process. But if the response time of every sensor is separated by 5 ms, the number of collisions when a real network traffic is used comes very close to that of a Poisson process. Therefore, if the effect of the deterministic component can be eliminated as above, the network traffic can be assumed to be a Poisson process in the collision rate

calculation.

In short, an example on how to model the network traffic is given above. Network traffic as a result of different applications or network and MAC designs can also be modeled in a similar way.

2.3.2 Physical layer model

An abstracted PHY model gives the power profile, defined as the power consumption in each operation mode of a PHY.

There are three key things in the modeling of a PHY: first the major components that consume most of the power must be identified. Secondly it needs to be found out how many operation modes a PHY has. Lastly the impact of the data rate on the power consumptions of the components of a PHY shall be understood.

As an example, a special PHY used in sensor networks is described. As shown in Fig. 2.6, its transmit chain includes a power amplifier (PA), an oscillator [31] and a digital base band (DBB) (the encoder is a simple cyclic redundancy check encoder). Its receive chain includes a RF frontend (including an envelope detector) and a DBB. On-off Keying (OOK) modulation is used inside the DBB.

This PHY has five operation modes, which are shown in Fig. 2.7 (taken from [30]). The transmit (TX) mode and the power-off (IL) mode are among the operation modes. The other three operation modes are related to the DBB in the receive chain. When a receiving node monitors the channel to look for incoming packets or a sending node does carrier

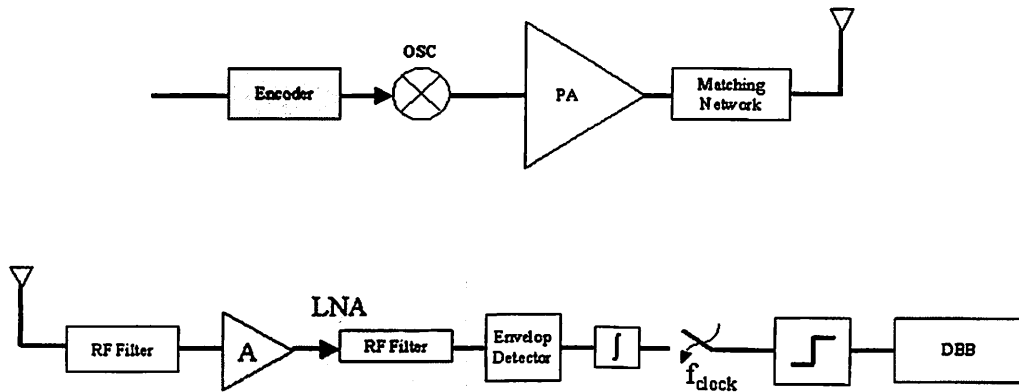


Figure 2.6: The transmit and receive chain

sensing, the DBBs in their receive chains do the same thing: measure the received signal strength and compare it to a threshold. This mode is called the monitor (MN) mode. When a packet is received, the DBB first needs to process the PHY header in the packet to do synchronization and acquisition. This mode is called the acquisition (AQ) mode. After the header is processed, the DBB demodulates the rest of the packet. This mode is called the receive (RX) mode.

2.3.2.1 The power profile

The power profile is obtained from the power consumptions of the components in each operation mode. Since no advanced codes are used, the power consumption of the DBB in the transmit chain is negligible. Because OOK is used, the PA needs to be on only half the time. Hence the average power consumption of the PA is:

$$P_{T,PA} = \frac{1 + \chi}{2} \cdot \frac{P_{rad}}{\eta} \quad (2.1)$$

where P_{rad} is the radiated power level, η is the power efficiency of the PA and χ is the ratio of the power consumption of the PA between the off and on state. Besides the oscillator, there are other components (e.g. buffer) in the transmit chain whose power consumption is independent of the radiated power level. For the sake of simplicity, $P_{T,osc}$ is used to denote the sum of the power consumption of the oscillator and the power consumption of all those components.

Thus the power profile of this PHY is obtained as:

$$P_T = \frac{1 + \chi}{2} \cdot \frac{P_{rad}}{\eta} + P_{T,osc} \quad (2.2)$$

$$P_{R,m} = P_{R,RF} + P_{R,BB,m} \quad (2.3)$$

$$P_{R,acq} = P_{R,RF} + P_{R,BB,acq} \quad (2.4)$$

$$P_{R,r} = P_{R,RF} + P_{R,BB,r} \quad (2.5)$$

$$P_{off} = 0 \quad (2.6)$$

where P_T , $P_{R,m}$, $P_{R,acq}$, $P_{R,r}$ and P_{off} are the power consumption of this PHY in the transmit, monitor, acquisition, receive and power-off mode, $P_{R,RF}$ is the power consumption of the RF frontend in the receive chain and $P_{R,BB,m}$, $P_{R,BB,acq}$, $P_{R,BB,r}$ are the power consumption of the receive chain DBB in the monitor, acquisition and receive mode.

2.3.2.2 Impact of the radio data rate

The radio data rate has different impact on the power consumption of different components. The power consumption of the RF frontend in the receive chain is relatively insensitive to the data rate, since a high-speed A/D is not used in this PHY. The same is

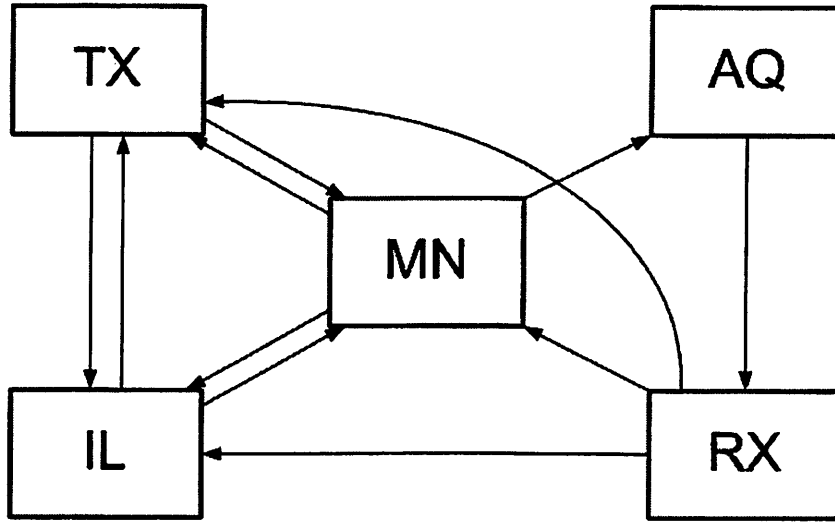


Figure 2.7: The operation modes

true for the power consumption of the oscillator. The power consumption of the DBB in the receive chain increases linearly with the data rate since its clock frequency is proportional to the data rate. Lastly the power consumption of the PA increases only sub-linearly with the data rate. This is explained as follows.

The radiated power level required to reach a receiver at distance d is:

$$P_{rad} = \frac{(4\pi f_c)^2 \cdot L' \cdot S}{G_t \cdot G_r \cdot c^2 \cdot d_0^{n-2}} \cdot d^n \quad (2.7)$$

where n is the path loss exponent, S is the receiver sensitivity, d_0 is the close-in reference distance, G_t is the transmitter antenna gain, G_r is the receiver antenna gain, L' is the system loss factor, c is the speed of the light, and f_c is the carrier frequency in Hertz.

The receiver sensitivity is the minimum received power needed at the input of the envelope detector for a target bit error rate (BER). The input Signal to Noise Ratio (SNR) is

therefore:

$$SNR_{in} = \frac{S}{k \cdot T \cdot BW \cdot NF} \quad (2.8)$$

where k is Boltzmann's constant, T is the absolute temperature in kelvin, BW is the bandwidth of the noise, NF is the system noise figure. The envelope detector is a nonlinear detector, so it changes the SNR of the signal that goes into it [32]. The relationship between its output SNR, SNR_{out} , and SNR_{in} can be approximated by the square law [29]:

$$SNR_{out} = SNR_{in}^2 \quad (2.9)$$

The output of this detector is the input to the demodulator in the DBB. Modulation scheme and error control codes determine the $\frac{E_b}{N_0}$ required for a target BER. The SNR_{out} is related to this $\frac{E_b}{N_0}$ as:

$$SNR_{out} = \frac{E_b}{N_0} \cdot \frac{R}{BBW} \quad (2.10)$$

where BBW is the data filter bandwidth.

If the radio data rate is increased by k times, from (2.7)-(2.10), P_{rad} only has to be increased by \sqrt{k} times to maintain the same target BER. Therefore P_{rad} only increases sub-linearly with the data rate.

But it should be noted that the radio data rate can not be increased infinitely. The envelope detector used in this PHY suffers from the inter symbol interference at higher data rates. The highest data rate the detector can be used at is about 160 kbps. On the other hand, if a more complex modulation scheme is used to increase the data rate beyond 160 kbps, P_{rad} may increase linearly with the data rate. In addition, going to a more complex modulation raises the noise floor.

In general, other PHYs may have different components or operation modes. The impact of the radio data rate on them may be different too. However, one can always model them by specifying the components, the operation modes and the impact of the data rate.

2.3.3 Channel models

A channel model will provide the average packet error rate from the channel impairment p_{kt_BER} .

Two different channel models are used as examples here. One is the independent channel model, where the channel is independent between bits. The other is the Gilbert-Elliott model, which models the fading channel.

2.3.3.1 Independent channel model

For a packet of length L , the probability that the packet fails is the probability that at least one bit in the packet is wrong. Assuming the channel is independent from bit to bit, one has:

$$p_{kt_BER} = 1 - (1 - p_b)^L \quad (2.11)$$

where p_b is the average BER of the channel.

2.3.3.2 Gilbert-Elliott channel model

Fig. 2.8 (taken from [19]) shows how the channel quality varies with time in an indoor environment. The measurements were taken inside the BWRC using the Pico Radio test bed. When there were many people near the radios, the link quality deteriorated due to the

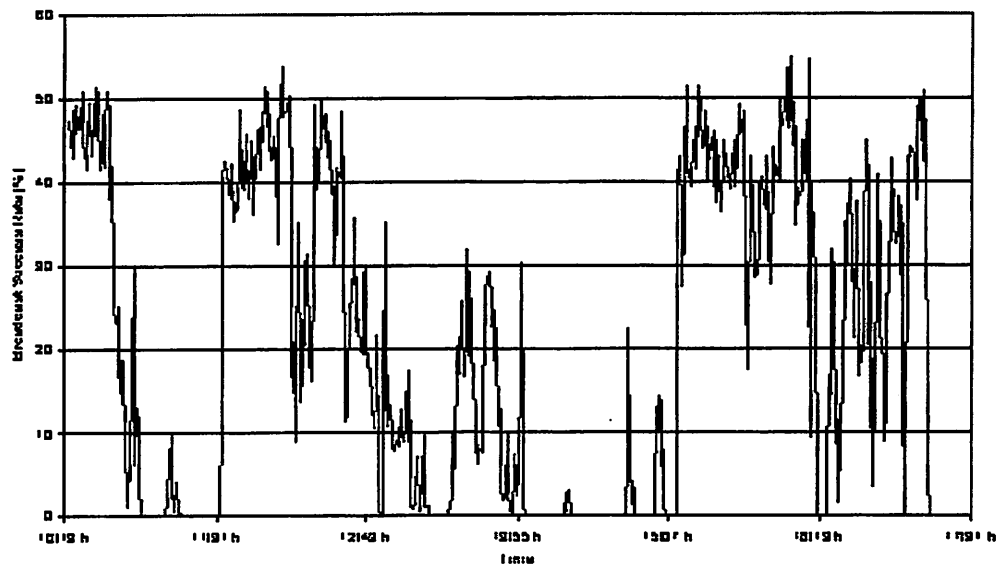


Figure 2.8: Packet success rate as function of time

loss of the line of sight. But after those people left, the link quality increased, corresponding to a much higher packet success rate.

The simple radios used in sensor networks (to reduce cost and power) often result in a channel that is either very good or very bad.

Therefore, a two-state discrete-time Markov chain called Gilbert-Elliott model [26] is used to model this channel. The model, shown in Fig. 2.9, has two states: the good and bad state. p is the transition probability from the good state to the bad state and q is the transition probability from the bad state to the good state.

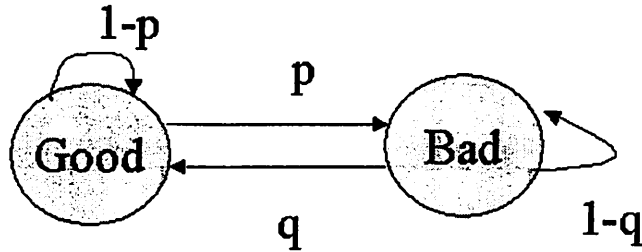


Figure 2.9: Gilbert-Elliott model

The stationary probability in each state can be obtained as:

$$\pi(bad) = \frac{p}{p+q} \quad (2.12)$$

$$\pi(good) = \frac{q}{p+q} \quad (2.13)$$

The channel in each state is an independent channel. If the BER in the good state is denoted as p_{bG} and that in the bad state as p_{bB} , the average BER is:

$$p_b = \pi(good) \cdot p_{bG} + \pi(bad) \cdot p_{bB} \quad (2.14)$$

Usually $p_{bG} \ll p_{bB}$ and the bad state corresponds to a deep fade. So the Gilbert-Elliott model captures the bursty behavior of a fading channel.

The average burst length, defined as the average time the channel stays in the bad state, can be computed to be

$$E[bl] = \frac{1}{q} \quad (2.15)$$

The simplified Gilbert-Elliott model further assumes that errors only occur in the bad state. As a result, a packet is good only if it starts in the good state and stays in the good

state, which happens with a probability of $\pi(\text{good}) \cdot (1 - p)^{L-1}$. Consequently, the average packet error rate is:

$$\begin{aligned} p_{kt.BER} &= 1 - \pi(\text{good}) \cdot (1 - p)^{L-1} \\ &= 1 - (1 - p_b) \cdot \left(1 - \frac{p_b}{1 - p_b} \cdot \frac{1}{E[bl]}\right)^{L-1} \end{aligned} \quad (2.16)$$

In general, the average packet error rate is:

$$p_{kt.BER} = 1 - \sum_{L_g=0}^L (1 - p_{bG})^{L_g} \cdot (1 - p_{bB})^{L-L_g} \quad (2.17)$$

where L_g is the number of bits of a packet in the good state. It is not easy to find the probability mass function of L_g .

By making $p_{bG} = 0$ and $p_{bB} = 1$, the simplified Gilbert-Elliott model encourages the use of the channel in the good state. So it favors designs that are opportunistic: use the channel only when it is in the good state. In fact, it makes sense to do so. Even though $p_{bB} < 1$ in the real life, attempting to use the channel in the bad state for transmissions or retransmissions wastes a lot of energy and increases the offered load of the channel. Thus the simplified Gilbert-Elliott model is a reasonable channel model to use.

One can see the impact of the average burst length on the average packet error rate in Fig. 2.10. The independent channel model corresponds to the case when the burst length is 1. It can be seen that the average packet error rate drops exponentially as the average burst length increases. So for the same average BER, the independent channel model is more conservative than the Gilbert-Elliott model.

In summary, two channels are modeled in this section. Other channel models exist, for

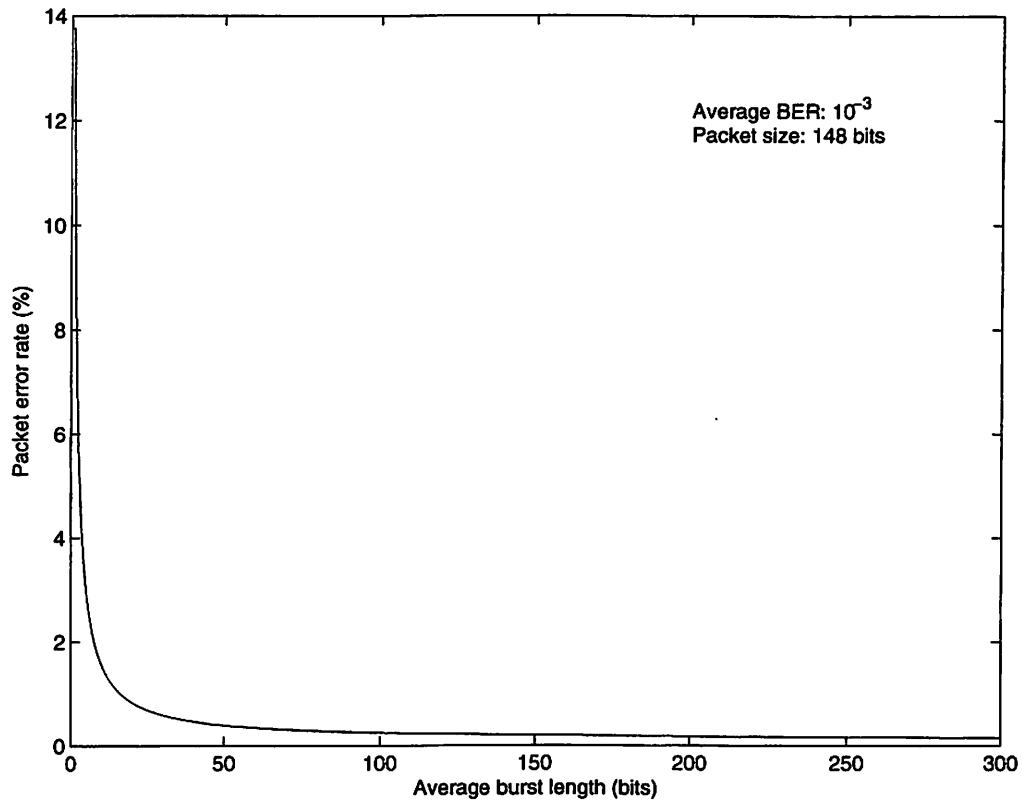


Figure 2.10: Impact of burst length on average packet error rate

instance the Rayleigh fading channel model. In any case, the goal of a channel model is to determine the average BER. This concludes the presentation of the abstracted models. In the next chapter, models will be built for the components inside the DLL.

CHAPTER 3

ANALYTICAL MODELS FOR THE DATA-LINK LAYER

This chapter continues the discussion on functional decomposition but focuses on the models for the DLL components. For each DLL component, one or more designs have been developed and modeled. Table 3.1 summarizes these design options. The description of these designs and their corresponding models will be presented in this chapter. These models, once built, will be integrated together with the abstracted models discussed in Chapter 2, providing important information on the relationship between the total power consumption and the design parameters.

3.1 Media Access Control Models

MAC controls the collisions, so every MAC model computes the collision rate. As examples, models of several commonly used MAC designs are given here. The first one

Table 3.1: Design examples modeled

DLL	Design Examples
MAC	ALOHA, CSMA with hidden terminals, Multi-channel MAC
EC	CRC
Link	Positive Acknowledgment, Extended NACK
PM	The full duty cycle receiver, the cycled receiver, the preamble sampling, the wakeup radio, Multi-channel PM
PC	The optimal power level

is ALOHA. The second one is carrier sense multiple access (CSMA) and the last one is a multi-channel MAC.

3.1.1 ALOHA

In ALOHA, basically no access control is performed. A node will access the channel whenever it gets a packet from the network layer.

Assuming both the network traffic and the retransmission traffic are Poisson processes, the collision rate ($p_{kt.COL}$) for a packet of size L is

$$p_{kt.COL} = 1 - \exp \left\{ - \left[\sum_{i=1}^{N_{msg}} (g_i \cdot L_i) + \sum_{i=1}^{N_{msg}} g_i \cdot L \right] \cdot \frac{N_{int}}{R} \right\} \quad (3.1)$$

Here N_{msg} is the number of message or packet types, g_i is the offered load of the packets of type i , N_{int} is the number of interferers and R is the radio data rate.

3.1.2 Carrier sense multiple access

In CSMA, a node monitors the channel before each transmission. If the channel is idle (i.e. no carrier is sensed), the node sends its packet. Otherwise it waits until the channel becomes available, after which it will perform a random back off.

There are two sources of collisions in CSMA. First, collisions are caused by the hidden terminals. When a node is sending a packet to another node, nodes outside the coverage area of the sender, but inside the coverage area of the receiver, can not sense the ongoing transmission and will cause collisions at the receiver if they transmit at this time. These nodes are called the hidden terminals. For them, CSMA is the same as ALOHA. Thus one

can get the collision rate for the hidden terminals from (3.1) by replacing N_{int} with N_h , the number of hidden terminals.

Secondly, collisions may also happen due to the propagation delay τ . There are two such cases. The first case is: a node has a packet to send within τ seconds of the start of the ongoing transmission. Due to the propagation delay, the node is not able to sense the current transmission. Consequently, it will go ahead to send its packet, causing collisions. The collision rate in this case is given by [25]. Let T_d be the time that a packet of size L uses the channel, when $\frac{\tau}{T_d}$ approaches 0, the collision rate in [25] goes to 0. In sensor networks, the range of a radio is typically short, hence $\tau \ll T_d$. For example, if the range is 10 m, τ is only 33 ns. On the other hand, given a 100 kbps radio and a packet of 200 bits, T_d is 2 ms. Therefore the collision rate is negligible in this case.

The second case is: a node finishes its back off before the other nodes, so it gets the channel. But another node finishes its back off within τ seconds of the first one. The collision rate due to back off is $1 - (1 - \frac{\tau}{W})^{N_b}$. W is the back off window size and N_b is the number of nodes performing back off. The above collision rate can be approximated by $\frac{\tau}{W}N_b$, when τ is small compared with W . In sensor networks. W is usually in the order of ms, so this collision rate is very small as well.

One can conclude from the above discussions that in sensor networks, the dominant source of collisions in CSMA is the hidden terminals. Therefore, he can approximate the collision rate in CSMA by the collision rate for hidden terminals.

$$p_{kt.COL} = 1 - \exp \left\{ - \left[\sum_{i=1}^{N_{msg}} (g_i \cdot L_i) + \sum_{i=1}^{N_{msg}} g_i \cdot L \right] \cdot \frac{N_h}{R} \right\} \quad (3.2)$$

Since N_h is typically less than a half of N_{int} , the collision rate of CSMA is generally lower than that of ALOHA.

3.1.3 Multi-channel media access control

A multi-channel MAC uses more than one channel to do the access control. In each channel, a traditional single-channel MAC (e.g. CSMA) is used. In general, if there are N_{ch} number of channels, one way is to assign N_c of them to the control messages and the rest of them to the data packets. Another way is to mix the control messages and data packets in every channel. It is not difficult to compute the collision rate in either case. In the following, CSMA is assumed to be used in each channel.

If the control messages are separated with the data packets, the collision rate of a control packet can be computed using (3.2) with the offered load being the offered load of all the control messages divided by N_c . One can compute the collision rate of a data packet in a similar way.

If the control messages are mixed with the data packets in every channel, one just need to divide the offered load by N_{ch} in (3.2) to get the collision rate. As one can see, the availability of additional channels reduces the traffic in each channel, resulting in a lower collision rate.

Three MAC designs widely used in wireless sensor networks are modeled in this section. From these three examples, one can see a MAC model is to compute the collision rate based on a set of parameters. Some of these parameters are specified by the other layers

(e.g. radio data rate), while others are determined by the other DLL components (e.g. the offered load). The message here is that no matter what a MAC design is, a model can be built for it, specifying the relationship between the collision rate and other parameters.

3.2 Error Control Model

As described in Chapter 2, the EC component is to detect and/or correct errors in a packet. An example of error detection codes is given here, while forward error correction (FEC) codes will be discussed in Chapter 6.

One of the widely used EC designs in wireless networks is the cyclic redundancy check (CRC) code. The CRC encoder and decoder can be easily implemented using shift registers, so the energy overhead of its circuitries is minimum. The CRC is very effective at detecting errors, especially the burst errors. Below the error detection probability of the CRC is derived.

For CRC- k , all error bursts of length no larger than k can be detected. However, error bursts of length $k + 1$ may not be detected. The probability of not being detected is $\frac{1}{2^{k-1}}$. Error bursts of even larger length have actually smaller probability of not being detected, which is given by $\frac{1}{2^k}$. It should be noted here that the detectable bursts of length larger than k include the end-round bursts of length no larger than k . The average probability of an undetected error is computed as:

$$P_u(E) = \frac{1}{2^k} \cdot \beta \quad (3.3)$$

where β is related to the channel and the packet size. It can be evaluated as follows for a

channel that is independent from bit to bit:

$$\beta = \frac{1 - (1 - p_b)^L - L \cdot p_b \cdot (1 - p_b)^{L-1} \cdot (1 - 2 \cdot p_b)}{1 - (1 - p_b)^{L+k}} \quad (3.4)$$

Here L is the packet size and p_b is the average BER. It is easy to show that if $p_b \leq \frac{1}{2}$, $\beta \leq 1$, so the average probability of an undetected error is upper bounded by $\frac{1}{2^k}$. Normally p_b in the wireless channel is in the range of 10^{-4} to 10^{-3} . In this range, the upper bound is quite loose. This can be seen in Fig. 3.1. Fig. 3.1 also shows that β is not very sensitive to k . For the simplified Gilbert-Elliott channel model, β can also be obtained using the similar approach.

As seen from this example, the outputs of a EC model include the average probability of

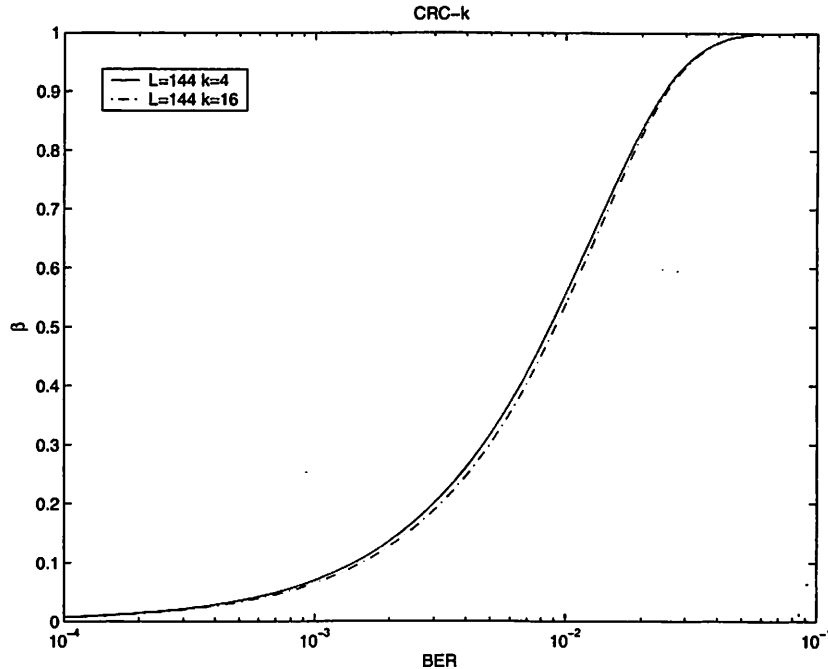


Figure 3.1: β as function of bit error rate

an undetected error. The size of a packet is obviously modified through coding. In general, the power consumption of the encoder and decoder are the outputs of a EC model as well, which can not be ignored if FEC is used. The probability of a codeword error is another output of the EC model if FEC is used.

3.3 Link Layer Models

There are many ways to design a link layer. For illustration purposes, two examples are provided in this section.

3.3.1 Link layer 1: positive acknowledgment

A four-way handshake is used in this link layer, as shown in Fig. 3.2. Before sending the actual data, a node first sends a setup request to its receiver. If the receiver is ready, it will respond with the ready message. Normal data transfer and acknowledgment (ACK) then follows. Assuming the receiver is always ready, the ready message is a positive acknowledgment for the setup message, as ACK is a positive acknowledgment for the data packet.

The number of transmissions for the setup, ready, data and ACK packet are denoted as N_{setup} , N_{ready} , N and N_{ACK} correspondingly, while p_{pkt_setup} , p_{pkt_ready} , p_{pkt_d} and p_{pkt_a} denote the corresponding probabilities for the setup, ready, data and ACK packet to fail and to be detected by the EC component. If the pair of the setup packet and the ready packet is defined as a setup session and the pair of the data packet and the ACK packet as

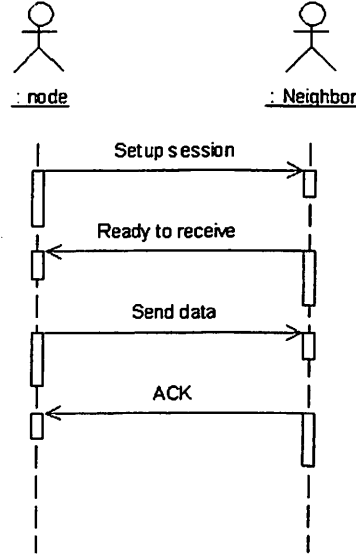


Figure 3.2: UML sequence diagram of a four-way handshake

a data session, $p_{kt.ss}$ and $p_{kt.s}$ are probabilities for a setup session and data session to fail.

Assuming the channel is independent between the packets, and a receiver is always ready, the average number of transmissions for each type of message can be obtained as below:

$$E[N_{setup}] = \frac{1 - p_{kt.ss}^M}{1 - p_{kt.ss}} \quad (3.5)$$

$$E[N_{ready}] = E[N_{setup}] \cdot (1 - p_{kt.setup}) \quad (3.6)$$

$$E[N] = \frac{1 - p_{kt.s}^M}{1 - p_{kt.s}} \cdot (1 - p_{kt.ss}^M) \quad (3.7)$$

$$E[N_{ACK}] = E[N] \cdot (1 - p_{kt.d}) \quad (3.8)$$

$$p_{kt.ss} = 1 - (1 - p_{kt.setup}) \cdot (1 - p_{kt.ready}) \quad (3.9)$$

$$p_{kt.s} = 1 - (1 - p_{kt.d}) \cdot (1 - p_{kt.a}) \quad (3.10)$$

M is the maximum number of transmissions of the setup packet here, which is the same as that of the data packet, but they can be made different.

3.3.2 Link layer 2: extended negative acknowledgment

Improvements have been made to the link layer design in the previous subsection to make it more robust. First, even though positive acknowledgment is still used for the setup packets, negative acknowledgment (NACK) is used for the data packets instead. The reason is as follows. It has not been considered so far that a receiver may not be ready. The receiver may be in a state expecting only messages from a third node. Or it may not be able to get a channel to respond. In both cases, even if there are no collisions, the packet transmitted still fails. With a multi-channel MAC, there is an additional possibility that a receiver may be on a different channel. When ACK is used, a sender pushes data to its receiver. If the receiver is not ready for one of the above reasons, the sender wastes energy in transmissions and retransmissions. Furthermore, these actions increase the offered load of the channel, resulting in higher collision rate. But with NACK, the receiver pulls data from the sender instead, so it is always ready when the sender responds to it with data.

The second improvement made is to send a group of ready or NACK packets instead of only one ready packet or ACK as in the first link layer design. This is done to make the reverse link more robust, so unnecessary retransmissions due to the failure of the ready or NACK packets can be reduced.

The model for this link layer is also to find out the average numbers of transmissions

for all packets, which are summarized below:

$$E[N_{setup}] = \frac{1 - p_{kt.ss}^M}{1 - p_{kt.ss}} \quad (3.11)$$

$$E[N_{ready}] = E[N_{setup}] \cdot (1 - p_{kt.setup}) \cdot \frac{1 - p_{kt.ready}^{M_r}}{1 - p_{kt.ready}} \quad (3.12)$$

$$E[N] = \frac{1 - p_{kt.s}^M}{1 - p_{kt.s}} \cdot (1 - p_{kt.ss}^M) \quad (3.13)$$

$$E[N_{ACK}] = (E[N] - 1 + p_{kt.ss}^M) \cdot p_{kt.d} \cdot \frac{1 - p_{kt.NACK}^{M_n}}{1 - p_{kt.NACK}} \quad (3.14)$$

$$p_{kt.ss} = 1 - (1 - p_{kt.setup}) \cdot (1 - p_{kt.ready}^{M_r}) \quad (3.15)$$

$$p_{kt.s} = p_{kt.d} \cdot (1 - p_{kt.NACK}^{M_n}) \quad (3.16)$$

where N_{NACK} is the number of transmissions of NACK, $p_{kt.NACK}$ is the probability for a NACK to fail and to be detected by the EC component, M_r is the maximum number of ready packets sent per session and M_n is that of NACK packets.

One can learn from the above two examples that the average number of transmissions of all packets can be determined from the probabilities of detected failure of the packets. Different link layer designs only make the relationship between the inputs and the outputs different.

3.4 Power Management Model

In the following, several PM designs used in sensor networks are introduced, together with their models. Again there can be many other ways to design a PM, but the point here is:

Whatever a PM design is, a model can be built, which will be integrated with the models of the other components or layers, shedding light on the relationship between the total power consumption and the design parameters, including those of the PM component.

3.4.1 The full duty cycle receiver

The most popular PM design is not to do power management at all. In reference to the duty cycle receivers discussed later on, this scheme is called the full duty cycle receiver (FDCR), because a receiver is always on in this scheme. In fact, the receiver is on even if there is no event in a network, consuming a lot of power in the monitor mode. However, in this scheme, the synchronization between a sender and its receiver is easy since the receiver is always on. As a result, the delay due to the synchronization process is small. When the network traffic rate is high, the overhead of synchronization does not pay off so the FDCR is indeed the best. Its results are summarized below:

$$t_t = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} (g_i \cdot L_i) \quad (3.17)$$

$$t_r = N_{NP} \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} [g_i \cdot (L_i - L_{OHP})] \quad (3.18)$$

$$t_{acq} = N_{NP} \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} g_i \cdot L_{OHP} \quad (3.19)$$

$$t_{mn} = 1 - (t_t + t_r + t_{acq})$$

Here t_t , t_r , t_{acq} and t_{mn} are the corresponding average percentage of time a node spends in the transmit, receive, acquisition and monitor mode, L_{OHP} is the size of the PHY overhead and N_{NP} is the number of physical neighbors.

One can see from the above results when the network traffic rate is low, t_{mn} will be the highest. At the same time, the radio range is short in sensor networks. Consequently, the power consumption in the monitor mode dominates the total power consumption.

3.4.2 The cycled receiver

The cycled receiver is one of those PM designs seeking to reduce the power consumption in the monitor mode [28]. Fig. 3.3 shows how it works. A receiver is only on for a short amount of time T_{on} in every cycle T . So when there is no event in a network, the receiver can sleep most of the time. But here arises a problem, for when a sender has a packet for this receiver, the receiver may not be on at the time. So before each data transmission, a sender needs to send a series of beacons to its receiver until the receiver wakes up and responds. This is how a sender and its receiver get synchronized, after which the normal data transfer procedure follows.

The cycled receiver reduces the power consumption of a receiver, at the cost of a sender. The penalty a sender has to pay is the power consumption needed to send the additional

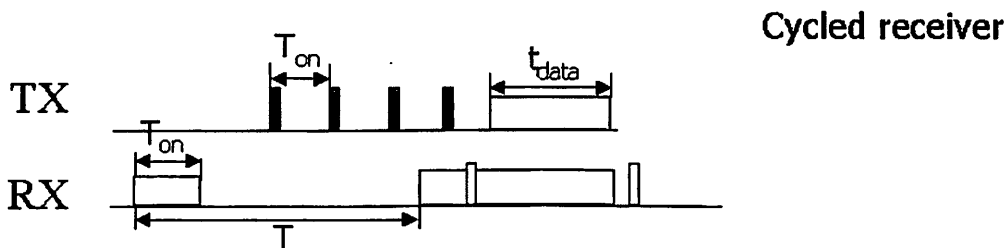


Figure 3.3: The cycled receiver

beacons and to monitor the channel until its receiver wakes up. But the sender pays this penalty only when it has a packet to send. In sensor networks, the network traffic rate is low, so the cycled receiver has lower power consumption than the FDCR.

The original scheme shown in Fig. 3.3 has been improved to increase the number of candidates who can forward the received packet. Instead of sending the packet to a particular receiver, a sender can send its packet to the first available receiver in a target direction $(-\theta_{max}, \theta_{max})$. Not only does this help to reduce the synchronization time, but it also increases the reliability. The model for this improved scheme is described next.

N_N , the number of neighbors active during an access window starting from the time a sender has a packet to send, is computed as:

$$N_N = \frac{T_{on} + t_{access}}{T} \cdot N_{NP} \quad (3.20)$$

where t_{access} is the access window width. Assuming the neighbors are uniformly distributed in the coverage area and all active neighbors in the target direction are candidates, the number of candidates, $N_{candidates}$, is:

$$N_{candidates} = \frac{N_N}{\pi} \cdot \theta_{max} \quad (3.21)$$

Assuming the network traffic is a Poisson process, the number of beacons a sender needs to transmit until the candidate i wakes up (N_i), is a random variable uniformly distributed between 1 and $\frac{T}{T_{on}}$ given that the sender has at least one packet from the network layer.

Hence

$$E[\min(N_i)] \approx \frac{1}{N_{candidates} + 1} \cdot \frac{T}{T_{on}}, \quad (3.22)$$

when $\frac{T}{T_{on}} \gg 1$. It follows that the mean time that a sender needs to wait until the first candidate wakes up (t_w) is:

$$E[t_w] = \frac{T}{N_{candidates} + 1}. \quad (3.23)$$

Based on these results, the average percentage of time a node spends in each operation mode is derived as:

$$t_t = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} (g_i \cdot L_i) + g_{setup} \cdot \frac{T}{2 \cdot \left(\frac{N_N}{\pi} \cdot \theta_{max} + 1\right)} \quad (3.24)$$

$$t_r = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} [g_i \cdot (L_i - L_{OHP})] + N_{NP} \cdot \frac{T_{on}}{T} \cdot \frac{L_{setup}}{R} \cdot g_{setup} \quad (3.25)$$

$$t_{acq} = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} g_i \cdot L_{OHP} \quad (3.26)$$

$$t_{mn} = \frac{T_{on}}{T} + g_{setup} \cdot \left(\frac{T_{on}}{2} + \frac{T}{2 \cdot \left(\frac{N_N}{\pi} \cdot \theta_{max} + 1\right)} \right) \quad (3.27)$$

Here g_{setup} is the offered load of the setup packets and θ_{max} is the maximum angle of the target direction.

One can see from the above equations that when T increases, the time in the transmit mode and part of the time in the monitor mode go up since the sender has to send more beacons and spend more time to monitor the channel. On the other hand, its receiver can sleep more, reducing its part of monitoring time. What is more, there will be fewer nodes awake in the neighborhood to overhear the beacons (the beacons are broadcasted). So an optimal T exists and it can be found by using an “optimization after integration” technique described in Chapter 4.

One can also see from the equations that an optimal T_{on} exists which is very close to 0.

But T_{on} in the cycled receiver can not be made arbitrarily small. It is lower bounded by the time needed to send and receive a beacon.

3.4.3 The preamble sampling

The preamble sampling goes a step further in the tradeoff between the energy spent in a sender and its receiver, allowing one to reduce T_{on} by nearly 30 times. Fig. 3.4 shows how it works. A sender precedes every data packet with a preamble (or busy tone). A receiver wakes up periodically to sample a preamble. If the receiver detects a preamble, it will stay on to receive the forthcoming data packet. Since the sampling of a preamble is the same as carrier sense, T_{onp} can be made as small as one-bit duration.

The results for the preamble sampling are given below:

$$t_t = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} (g_i \cdot L_i) + g_{data} \cdot T_p \quad (3.28)$$

$$t_r = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} [g_i \cdot (L_i - L_{OHP})] + (N_N - 1) \cdot g_{data} \cdot \frac{L_{data} - L_{OHP}}{R} \quad (3.29)$$

$$t_{acq} = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} g_i \cdot L_{OHP} + (N_N - 1) \cdot g_{data} \cdot \frac{L_{OHP}}{R} \quad (3.30)$$

$$t_{mn} = \frac{T_{onp}}{T_p} + N_N \cdot g_{data} \cdot \frac{T_p}{2} \quad (3.31)$$

The preamble sampling is better than the cycled receiver when the network traffic rate is low, because its T_{onp} can be made much smaller than the T_{on} used in the cycled receiver. It is better for broadcasting traffic too. As long as the length of the preamble is no less than the sampling period, every neighbor can detect the ongoing transmission. But for the cycled receiver, there is no guarantee that every neighbor will get the beacon unless the

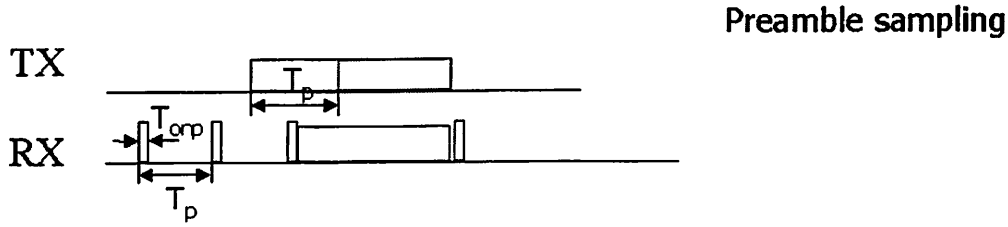


Figure 3.4: The preamble sampling

sender sends a busy tone for a duration of T .

The drawback of the preamble sampling is that the problem of overhearing (every neighbor is waken up even though the data packet is meant for only one of them) or false alarm is a lot worse. This becomes a problem when a network is very dense. The overhead of the preamble sampling on the sender side is fixed, no matter when the receiver wakes up. On the other hand, the overhead of the cycled receiver (the improved version) on the sender side is closely related to the time between the instant when the sender has a packet to send and the instant when the first receiver wakes up. This time is a random number with mean of $\frac{T}{2}$ if there is only one candidate. But if there are more candidates, as shown in (3.23), the mean goes down with the number of candidates, $N_{candidates}$, as $\frac{T}{N_{candidates}+1}$. So the cycled receiver can benefit from the increase of the number of candidates while the preamble sampling can not. Therefore, the cycled receiver outperforms the preamble sampling when a network is very dense.

From the above discussion, one can also know the N_N in the preamble sampling is:

$$N_N = N_{NP} \quad (3.32)$$

Similarly an optimal T_p exists and can be calculated easily in close-form as:

$$T_{p-opt} = \sqrt{\frac{P_{R.m} \cdot T_{onp}}{(P_T + N_N \cdot \frac{P_{R.m}}{2}) \cdot g_{data}}} \quad (3.33)$$

3.4.4 The wakeup radio

In the wakeup radio approach, an extra radio is used to constantly monitor the channel and wake up the main radio when a packet is coming [5]. This extra radio, called the wakeup radio, can be designed to consume much less power than the main radio, since it does not need to have all the functions of a regular radio. So even though it is constantly on, the pay off is justified. Just like the FDCR, the receiver is always available to receive the packets, so the delay is much smaller than that of the duty cycle receivers. But some kind of synchronization is still expected between a sender and the wakeup radio at its receiver, adding energy and delay.

The results for the wakeup radio scheme are listed below:

$$t_t = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} (g_i \cdot L_i)$$

$$t_r = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} [g_i \cdot (L_i - L_{OHP})] \quad (3.34)$$

$$t_{acq} = \frac{1}{R} \cdot \sum_{i=1}^{N_{msg}} g_i \cdot L_{OHP} \quad (3.35)$$

t_{mn} of the main radio will be zero. But the power consumption of the wakeup radio, P_{wakeup} , will be added to the total average power consumption instead.

So far it has been assumed that only the intended receiver will be waken up. If a designer does not put the destination information into the wakeup signal to simplify the

design of the wakeup radio, P_{wakeup} will reduce at the cost of more false alarms. The results in this case are the same as those of the FDCR with the exception of t_{mn} .

3.4.5 Multi-channel power management

If there are more than one channel available, the handshakes in the link layer can be used to further reduce the overhearing (false alarm): the control messages use a dedicated channel and a receiver turns on its data channels only if a setup session (in the control channel) is successful. As a result, overhearing only happens in the control channel. The results presented earlier can be modified slightly to only consider overhearing of the control messages.

Even though many commonly used PM designs in wireless sensor networks are described in this section, it is not the intent here to enumerate every PM design for WSNs. In fact, there may be other PM designs. The goal of this section is to show an analytical model can be built for a PM design, which is the first step of a two-step modeling approach to the most important problem in WSNs. It will be shown in Chapter 4 that all these different PM models have in fact one universal interface.

3.5 Power Control Model

The PC component, as said in Chapter 2, sets the radiated power level. A PC model is therefore to establish the relationship between this power level and other parameters.

Here a particular power control design is used as an example. This design is based on

the algorithm proposed in [11]. Instead of attacking directly the problem of obtaining connectivity, which is a much harder problem to solve, [11] assumes a network is connected and reduces the radiated power level as much as possible while still maintaining the connectivity. The problem with this scheme is that when a network is dense, the radiated power level is reduced so much that the number of hops needed to go from a source to its destination will be so big that the total energy from the source to the destination is actually higher. But the algorithm in [11] can be improved by setting a minimum radiated power level. This threshold is chosen to be the optimal radiated power level, defined as the radiated power level associated with the minimum total power consumption.

How does one model such a PC design? First it needs to be pointed out that the optimal power level is fixed and same for all nodes. Furthermore, when a network is dense enough, most of the nodes in the network will be operating near the optimal power level. This has been verified by network simulations with a network density of average 0.25 nodes per square meter and with nodes randomly placed, a typical scenario in sensor networks. For the sake of simplicity, every node is assumed to use the same fixed power level in the PC model.

This chapter concludes the description of the functional decomposition step of the two-step modeling approach. One can not help noticing that after the decomposition, it has become a much more manageable task to develop the analytical models for the resulting components and layers. Indeed, the examples in this chapter and Chapter 2 demonstrate this point. However, one can not stop here: These models must be integrated for one to see

how the overall power consumption is related to the design parameters. This will be done in the next chapter.

CHAPTER 4

INTEGRATION OF MODELS

This chapter is the most important chapter in this dissertation. It includes the main contributions of this research.

After developing models for each component, it is crucial to integrate them together to link the total power consumption to the design parameters of these components.

It turns out that this integration procedure can be made relatively independent of the actual designs used for the components. As in the integration of software components, the interface of every model needs to be decided first in this integration process. As a matter of fact, an interface can be easily obtained by looking at the inputs and outputs of an analytical model and finding the union of inputs/outputs for various designs. Once the interfaces are determined, the interactions between the models can be studied. The total power consumption can be evaluated from the design parameters after solving the closed loops.

4.1 Model Interfaces

To integrate the models, their interfaces have to be defined first. The interface of a model specifies its inputs and outputs. An interface should not be tied to only particular designs, instead it should be true for all designs. One can verify the examples in the previous chapters do comply with the interfaces specified here.

4.1.1 Media access control model interface

The block diagram for the MAC is shown in Fig. 4.1. N_{int} is the number of interferers, g is the offered load, R is the radio data rate, N_{ch} is the number of channels provided by the PHY and L are the sizes of the packets from the network layer and the components inside the DLL, for example the link layer and PC. The output of the MAC is the collision rate.

4.1.2 Error control model interface

Fig. 4.2 is its block diagram. P_{RCV} is the received signal power level, p_b is the channel BER before coding and L_{info} is the size of a packet excluding its PHY overhead and the redundant bits added by the EC. The outputs of the EC include the power consumption of the DBB (of the PHY) in the receive modes $P_{R,BB}$, the probability for an undetected error

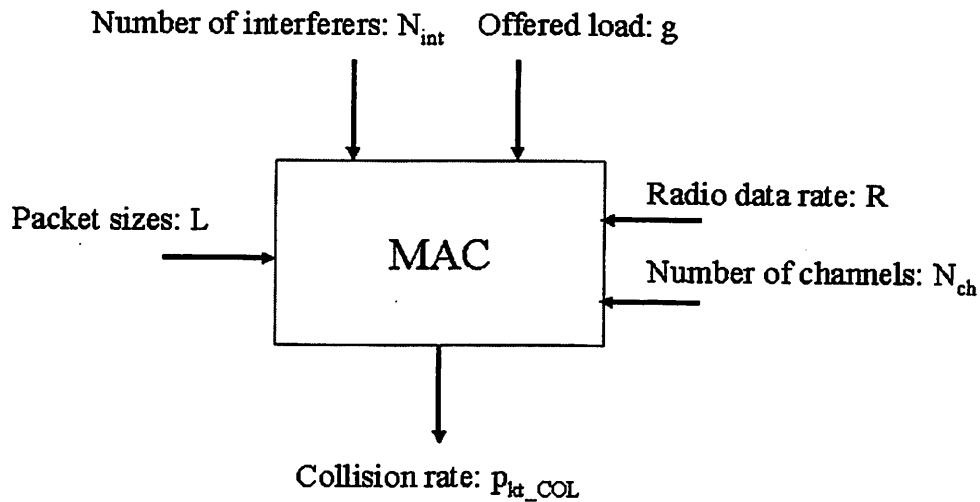


Figure 4.1: Media access control model

$P_u(E)$, the probability of a codeword error P_M and the total size of a packet L .

4.1.3 Link layer model interface

Fig. 4.3 is the black box view of any link layer. Its input is the probability of the detected failure for a packet and its output is the average number of transmissions of that packet.

4.1.4 Power management model interface

The component view of the PM is shown in Fig. 4.4. N_{NP} is the number of physical neighbors, g is the offered load, t_{access} is the access window width, R is the radio data rate, t_{cs} is the time needed to sense a carrier, P_{wakeup} is the power consumption of a wakeup radio, L_{OHP} is the size of the PHY header and L are the packet sizes. The PM has two outputs: the number of active neighbors N_N and the average percentage of time a node

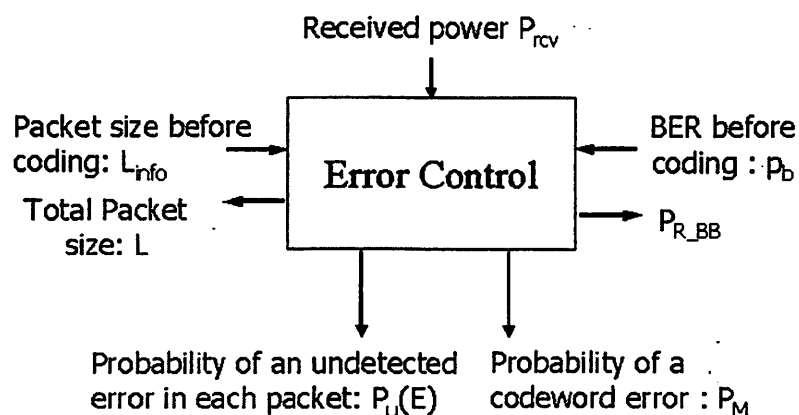
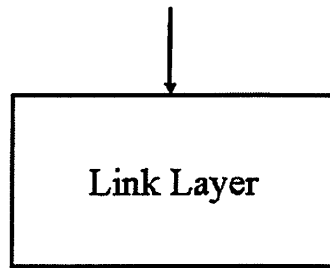


Figure 4.2: Error control model

Probability of the detected failure for each packet



Average number of transmissions

Figure 4.3: The link layer model

spends in each operation mode.

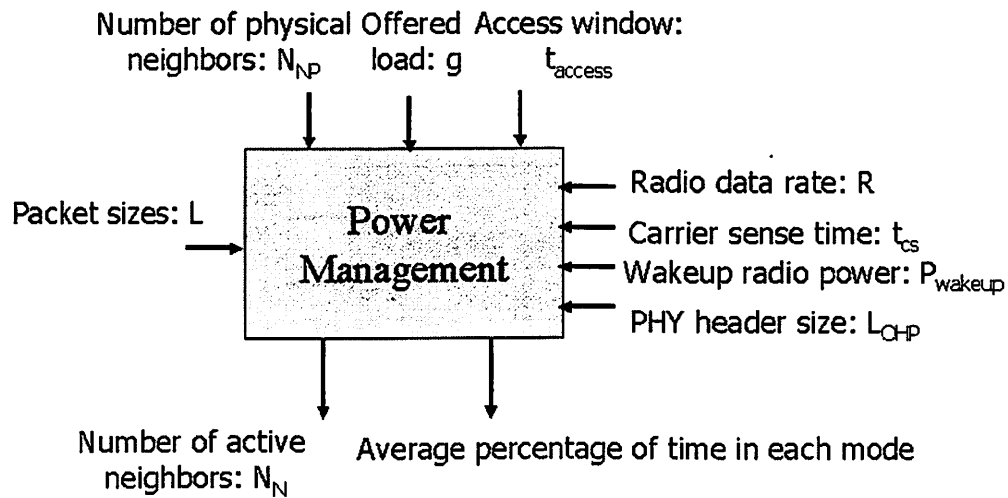


Figure 4.4: Power management model

4.1.5 Power control model interface

Its block diagram is shown in Fig. 4.5. D is the network density, G is related to the antenna gain, power efficiency of the power amplifier and the receiver sensitivity, n is the path loss exponent, P_R is the power consumption in receive modes, t_{rx} and t_{ov} are the time a node spends in receiving or overhearing. The PC outputs the radiated power level.

4.2 Interactions between the Models

The interactions between the DLL models are summarized in Fig. 4.6. An arrow from model A to B suggests that B depends on A.

In the following paragraphs, the interactions between each model and the other models are discussed in details. The MAC model is inspected first. The MAC's output and that of the channel (and also that of the EC if FEC is used) jointly determine the average packet

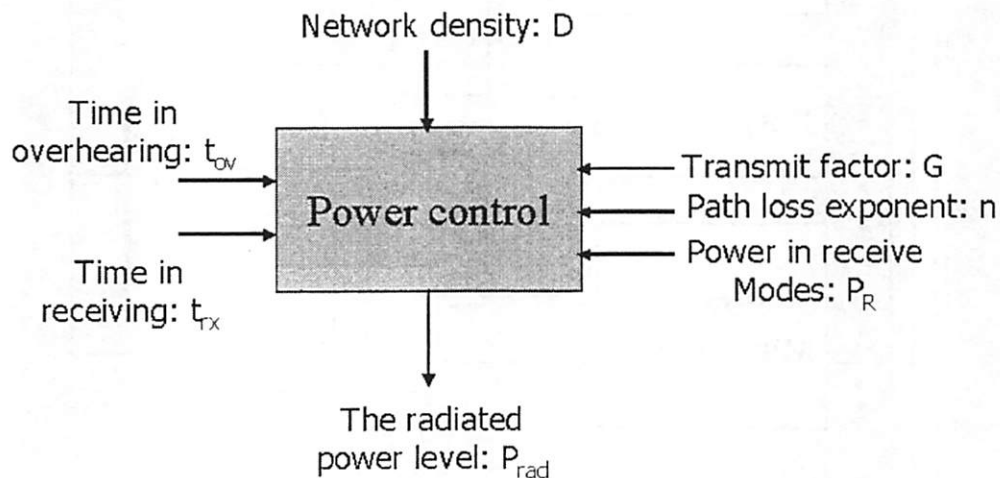


Figure 4.5: Power control model

error rate.

Assuming collisions happen independent of the channel errors, one has:

$$p'_{kt} = p_{kt_BER} + p_{kt_COL} - p_{kt_BER} \cdot p_{kt_COL} \quad (4.1)$$

Among the packets corrupted, a fraction of them will be undetected, hence the probability of detected failure for a packet is:

$$p_{kt} = p'_{kt} \cdot (1 - P_u(E)) \quad (4.2)$$

which is the input to the link layer model (Fig. 1.1). Therefore the link layer model depends on the MAC model.

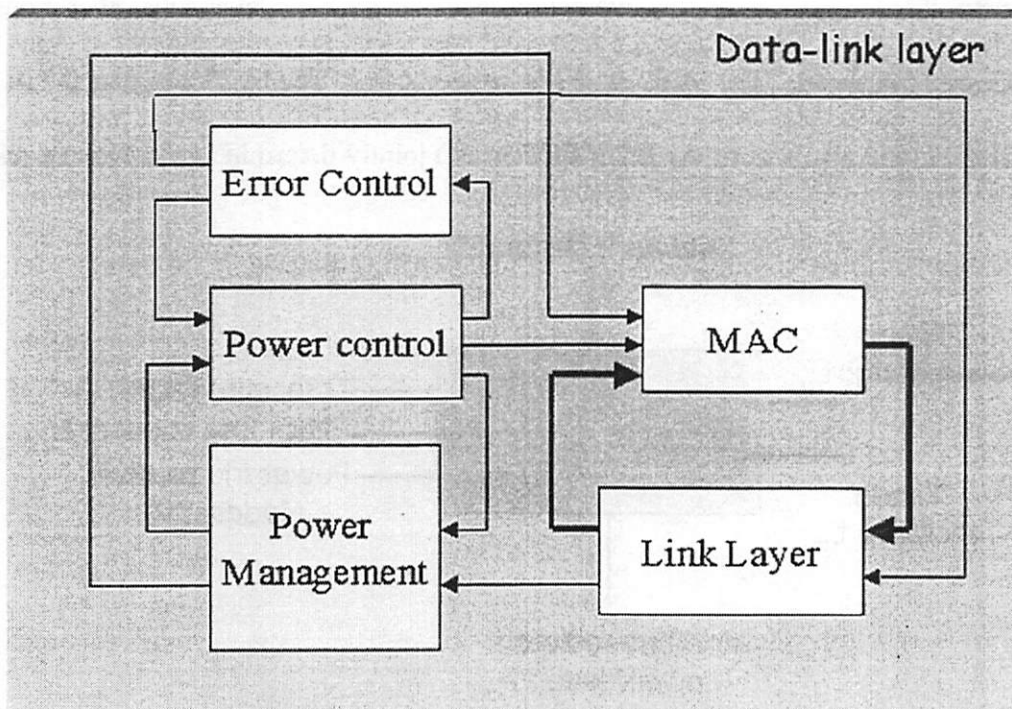


Figure 4.6: Interactions between the data-link layer components

As for the link layer model, its output, the average number of transmissions of a packet, influences the offered load for that packet:

$$g_i = T_N \cdot E[N_i] \quad (4.3)$$

where T_N is the average number of network events per second per node or the network traffic density. T_N can be determined from the output of the PC model, which will be discussed shortly.

At this point, it needs to be pointed out that the offered load is the input to both the MAC model and the PM model (Fig. 1.1). This implies that both models depend on the link layer model and the PC model. As a result, there is a closed loop between the MAC model and the link layer model (Fig. 4.6).

The PC model outputs the radiated power level, which in turn determines the hop distance:

$$d_H = \sqrt{\frac{P_{rad} \cdot G_t \cdot G_r \cdot c^2 \cdot d_0^n}{(4\pi f_c)^2 \cdot L' \cdot S}} \quad (4.4)$$

where n is the path loss exponent, d_0 is the close-in reference distance, G_t is the transmitter antenna gain, G_r is the receiver antenna gain, L' is the system loss factor, c is the speed of the light, S is the receiver sensitivity and f_c is the carrier frequency in Hertz. The hop distance further specifies the following two parameters: the number of the physical neighbors N_{NP} and the traffic density. N_{NP} given by:

$$N_{NP} = \pi \cdot D \cdot d_H^2 \quad (4.5)$$

is the input to the PM model. On the other hand, the traffic density, which both the MAC

model and the PM model depend on, can be computed from the hop distance as:

$$T_N = E[APDR] \cdot \frac{\bar{d}}{d_H} \quad (4.6)$$

where APDR is the average application data rate, or the average number of application events per second per node, and \bar{d} is the mean distance between a source and a destination.

Furthermore, one can calculate the received signal power level at distance r , the input to the EC model, as:

$$P_{rcv} = \frac{P_{rad} \cdot G_t \cdot G_r \cdot c^2 \cdot d_0^{n-2}}{(4\pi f_c)^2 \cdot L \cdot r^n} \quad (4.7)$$

It follows that the MAC, PM and EC model depend on the PC model. One of the outputs of PM model is the average percentage of time a node spends in each mode, from which one can compute the time the node spends in overhearing and receiving, the two inputs to the PC model. However it must be noted here that only for certain PC designs, the output of the PC model, the radiated power level, depends on these two inputs. So only for those designs, there is a closed loop between the PM model and the PC model. An example of such PC designs is: the radiated power level is always set as the locally optimal power level for given t_{rx} and t_{ov} . As good as it seems, this design is not the best design for the PC, which will be discussed in more details in subsection 4.3.2.

Another output of the PM model is the number of active neighbors. As with the PC model, this output depends on the inputs: N_{NP} and g only for some PM designs. One such design is to always use the locally optimal duty cycle for given N_{NP} and g . Again it will be shown in subsection 4.3.2 that this constraint actually drives the design away from the real optimum.

Lastly, the EC model has several outputs: the probability of a codeword error P_M influences the average packet error rate while the probability for an undetected error $P_u(E)$ impacts the probability of detected failure for a packet. Hence the link layer model depends on the EC model. The EC model also influences the power consumption of the DBB (of the PHY) in the receive modes $P_{R,BB}$, which in turn determines the power consumption in these receive modes, the input to the PC model. Therefore there will be a closed loop between the PC model and EC model if a PC design makes the radiated power level depend on the power consumption in the receive modes.

By now one has seen that the interactions between the components are so complex that the availability of proper models is really appreciated. In particular, there is at least one closed feedback loop involved: the loop of the MAC model and the link layer model. For certain PC designs and PM designs, there are also a closed loop between the PC and the PM models, a closed loop between the PC model and the EC model and a bigger closed loop containing all of the above three loops (Fig. 4.6). In the next subsection, the fixed point theorems are used to address the closed loop problem. In subsection 4.3.2, a technique called "optimization after integration" is introduced to simplify this problem: it will be shown that many closed loops can be opened.

4.3 Closed Loop Problem

4.3.1 Fixed point theorems

The fixed point theorems are ideal for the closed loop problem mentioned earlier. In the following it will be shown how they can be used to solve the closed loops.

Definition 1: (partial order) A relation " \leq " is a partial order on a set S if it has:

1. Reflexivity: $a \leq a \forall a \in S$.
2. Antisymmetry: $a \leq b$ and $b \leq a \implies a = b$.
3. Transitivity: $a \leq b$ and $b \leq c \implies a \leq c$.

A partially ordered set is also called a poset. A set with a bottom is a poset.

Definition 2: (chain) A chain is a fully ordered subset of a partially ordered set.

Definition 3: (CPO) A set is a complete partially ordered (CPO) set if every chain Y in the set has a lowest upper bound, denoted by $\bigvee(Y)$, and $\bigvee(Y)$ is inside the set.

Definition 4: A function f is monotonic if $f(x_1) \leq f(x_2)$ for any $x_1 \leq x_2$.

Definition 5: A function f is F-continuous if for every chain Y in a CPO, $\bigvee(f(Y)) = f(\bigvee(Y))$.

Theorem 1 *The set which the packet error rate vector belongs to is a CPO.*

Proof: Since the packet error rate is a probability, the vector of the packet error rates of all packet types belongs to the set $[0, 1]^{N_{msg}}$, where N_{msg} is the number of packet types. This set is real-valued, closed, and partially ordered with $(0, 0, \dots, 0)$ (denoted as $\mathbf{0}$) as the

bottom and $(1, 1, \dots, 1)$ as the top. From Definition 2, every chain \vec{X}_n in $[0, 1]^{N_{msg}}$ is a non-decreasing sequence. A non-decreasing sequence bounded by $(1, 1, \dots, 1)$ has a limit, which is also the lowest upper bound of the sequence. In addition, $[0, 1]^{N_{msg}}$ is closed, so the limit is in the set. In conclusion, every chain in $[0, 1]^{N_{msg}}$ has a lowest upper bound and the lowest upper bound is inside the set, so $[0, 1]^{N_{msg}}$ is a CPO. ■

Theorem 2 *A function f is also F-continuous if it is both monotonic and continuous.*

Proof: Given a chain \vec{x}_n in $[0, 1]^{N_{msg}}$, $f(\vec{x}_n)$ is also a non-decreasing sequence in $[0, 1]^{N_{msg}}$ since f is monotonic. From the proof of Theorem 1, a non-decreasing sequence in $[0, 1]^{N_{msg}}$ has a limit and the limit is its upper bound. Therefore,

$$\lim_{n \rightarrow \infty} f(\vec{x}_n) = \bigvee (f(\vec{x}_n))$$

and

$$\lim_{n \rightarrow \infty} \vec{x}_n = \bigvee (\vec{x}_n).$$

f being a continuous function implies

$$\lim_{n \rightarrow \infty} f(\vec{x}_n) = f(\lim_{n \rightarrow \infty} \vec{x}_n).$$

It follows that

$$\bigvee (f(\vec{x}_n)) = \lim_{n \rightarrow \infty} f(\vec{x}_n) = f(\lim_{n \rightarrow \infty} \vec{x}_n) = f(\bigvee (\vec{x}_n)).$$

By Definition 5, f is a F-continuous function. ■

Theorem 3 (Brouwer's theorem) *Every continuous mapping f of a closed n -ball to itself has a fixed point.*

Theorem 4 (*Banach's fixed point theorem*) *If X is a CPO with bottom \perp , and $f : X \rightarrow X$ is F -continuous, then f has a least fixed point x and one can find x constructively by obtaining the lowest upper bound of the following chain: $\{\perp, f(\perp), f(f(\perp)), \dots\}$.*

Next, these fixed point theorems are used to solve the closed loop problem, starting with the case there is only one closed loop, which is the loop containing the MAC and the link layer model. In this case, f is continuous. By Brouwer's theorem, at least one fixed point exists. In addition, f is monotonic in most cases (except when p_{kt_NACK} and/or p_{kt_setup} are close to 1). The intuition behind this conclusion is as follows: when the packet error rates for all packets \vec{X} increase, generally the numbers of transmissions for all packets go up. As a result, the channel gets more crowded and the packet error rates for all packets $f(\vec{X})$ will be higher. If f is both monotonic and continuous, by Banach's fixed point theorem, one can always find the least fixed point of the packet error rate vector by starting at $\mathbf{0}$ and keeping applying function f to it. This can be easily done using MATLAB. In summary, to solve the closed loop problem, it is only required that the MAC function and the link layer function are monotonic and continuous. Exactly what designs are used for them does not matter.

When there are more than one closed loops, one can still use the same approach to find the fixed point.

The constraint on each individual component is that the function that maps the inputs to the outputs needs to be both monotonic and continuous.

In another word, the least fixed point can be found if the designs for all DLL compo-

nents satisfy this constraint. Since for all designs, the corresponding functions are continuous, the real constraint is that these functions need to be monotonic. The MAC designs described in Chapter 2 satisfy this constraint; the link layer designs satisfy the constraint if p_{kt_NACK} and/or p_{kt_setup} are not close to 1. A subset of PM designs also has monotonic functions. For instance, if a PM design uses a locally optimal duty cycle based on given N_{NP} and g , when the input g increases, the duty cycle increases, resulting in a higher number of active neighbors (output of the PM model). However, the PC model is not always monotonic. If a PC design is used which locally optimizes the radiated power level based on the time a node spends in overhearing t_{ov} etc., when t_{ov} increases, the locally determined optimal power level will decrease instead to reduce the number of nodes overhearing.

There are many closed loops in Fig. 4.6, can some of them be opened? The answer is yes if one can choose designs that open the loops from inside the blocks. The next question is if these designs consume more power than those who do not open the loops. The answer is no if an “optimization after integration” technique is used. With this technique, all the closed loops in Fig. 4.6 can be opened except the one shown with a thick line. In addition, the resulting total power consumption is the lowest one can get.

4.3.2 Optimization after integration

As mentioned earlier in this thesis, many designers tend to design a component in an isolated fashion. The optimizations they did were therefore at the local level, resulting in a constraint between the inputs and outputs of the component. If the optimizations were

not done at the local level, many of the closed loops could be opened. The fixed point theorems can be used to deal with the remaining closed loops and integrate the models to compute the overall power consumption, after which the overall power consumption can be optimized by choosing the design parameters accordingly. In contrast to a local optimum, a global optimum is obtained here. The above technique, called “optimization after integration”, takes advantage of the **integrated** nature of this modeling framework, and attempts to remove the constraints in the models as much as possible before integration: it will remove every constraint unless the constraint is the result of the laws of physics or the resulting design (after removing the constraint) is impractical.

Here an example is used to show how this is done. In this example, the MAC design is CSMA, the link layer uses extended NACK, and the PM is the cycled receiver. All these designs are described in Chapter 3. The optimization goal used is the average power consumption per node, which is defined to be the total power consumption of all the nodes in the network divided by the number of nodes. Given the power profile from the PHY and the average percentage of time a node spends in each mode from the PM, the average power consumption per node can be easily computed as:

$$E[P] = P_T \cdot t_t + P_{R,r} \cdot t_r + P_{R,acq} \cdot t_{acq} + P_{R,m} \cdot t_{mn} \quad (4.8)$$

One first sets a radiated power level, from which N_{NP} and T_N can be determined using (4.4)-(4.6). Next he picks a T for the cycled receiver and computes N_N using (3.20). For given T_N and N_N , the iterative method in the previous subsection is used to find the least fixed point for (3.2) and (3.11)-(3.16). Once the fixed point is found, the average number

of transmissions for every type of packet is known, so the offered load for a packet can be obtained using (4.3). Now (3.24)-(3.27) can be used to compute the average percentage of time a node stays in each mode, after which the average power consumption can be computed from (4.8). Following the steps outlined above, one can calculate the average power consumption for every pair of (P_{rad}, T) . Consequently, he can numerically determine the optimal radiated power level and duty cycle that result in the lowest average power consumption.

In the above example, only one closed loop needs to be solved. Why does this technique lead to the lowest overall power consumption? In constrained optimization, the fewer constraints an optimization problem has, the more optimal its solution will be.

Since the “optimization after integration” technique removes the constraints as much as possible, the resulting overall power consumption can only be lower.
--

CHAPTER 5

VERIFICATION OF THE MODELS

In this chapter, the models are first used to obtain several important design metrics. Afterward, the models are verified by comparing their results on these design metrics with those from the network simulations.

5.1 Design Metrics

Many design metrics of interest can be obtained after the integration of the models. For example, the **average power consumption** can be obtained from (4.8). Quality of Service (QoS) metrics, including the average packet loss rate and average delay, are important in sensor networks too. One must reduce power consumption without sacrificing the QoS. Consequently, it is important to evaluate the average packet loss rate and delay together with the average power consumption. The modeling framework introduced in the preceding chapters can provide QoS metrics as well. Several simple examples are given below to make this point.

First, the **average packet loss rate** for a link layer using extended NACK (described in Chapter 3) is evaluated. If the probability that a data packet fails is denoted as $p_{kt.d}'$, one can get:

$$p_{kt.d} = p_{kt.d}' \cdot (1 - P_u(E)).$$

The packet loss rate for the above link layer is obtained as:

$$p_{kt_loss} = p_{kt_ss}^M + (1 - p_{kt_ss}^M) \cdot [p_{kt_s}^M + (1 - p_{kt_s}^M) \cdot (p_{kt_d}' \cdot P_u(E) + p_{kt_d} \cdot p_{kt_NACK}^{M_n})] \quad (5.1)$$

The energy per useful bit is a metric that reflects the overall effect of the energy consumption and packet loss rate. It is defined as the ratio between the average energy spent in a cycle and the number of information bits successfully sent in the cycle:

$$E_{bu} = \frac{E[P]}{T_N \cdot L_N \cdot (1 - p_{kt_loss})} \quad (5.2)$$

where L_N is the network packet size.

As another example, the average delay t_D associated with a design using the CSMA and the link layers in Chapter 3 can also be computed. If t_{D_setup} and t_{D_data} denote the delay in the setup phase and data phase correspondingly, t_D is computed as:

$$t_D = t_{D_setup} + t_{D_data}.$$

t_{D_setup} and t_{D_data} can be evaluated as:

$$t_{D_setup} = E[N_{setup}] \cdot \left(\frac{L_{setup} + L_{ready}}{R} + \frac{W}{2} \cdot M_{bk} \cdot P_{setup_busy} \right) + (E[N_{setup}] - 1) \cdot \left(T_s + \frac{T_{sr}}{2} \right) \quad (5.3)$$

$$t_{D_data} = E[N] \cdot \frac{L_{data}}{R} + (E[N] - 1) \cdot \left(T_d + \frac{W}{2} \cdot M_{bk} \cdot P_{data_busy} \right) \quad (5.4)$$

where W is the back off window size, M_{bk} is the maximum number of back offs, P_{setup_busy} (or P_{data_busy}) is the probability that the channel is busy when a setup (or data) packet is sent. When a setup packet needs to be retransmitted, the sender waits T_s time and additional random time up to T_{sr} . Similarly, T_d is the time between data packet retransmissions.

For other designs, the QoS metrics can also be derived accordingly.

5.2 Verification of the Models

In the analytical models, a number of assumptions have been made, some of which might be simplifications. Examples of such are:

- Retransmissions are Poisson distributed
- Channel is independent between packets
- Collisions are independent from channel errors
- Receiver is always ready

The OMNET++ network simulation framework was used to verify the models. In the network simulations, none of the above assumptions are made. The network traffic model is still assumed to be a Poisson process and the channel model is the simplified Gilbert-Elliott model, same as those used in the analytical models. For both approaches, the same PHY and the same DLL are used. The DLL uses a two-channel CSMA as the MAC, extended NACK as the link layer, the CRC as the EC, the wakeup radio as the PM and it uses the optimal power level in the PC. A network of twenty four nodes is simulated for an hour. Fig. 5.1 shows the topology of the network in OMNET++. The two icons with a bar are the two channels used in the MAC.

Fig. 5.2 shows the energy per useful bit obtained from both approaches. It can be seen that even though these two represent different modeling approaches, their results match

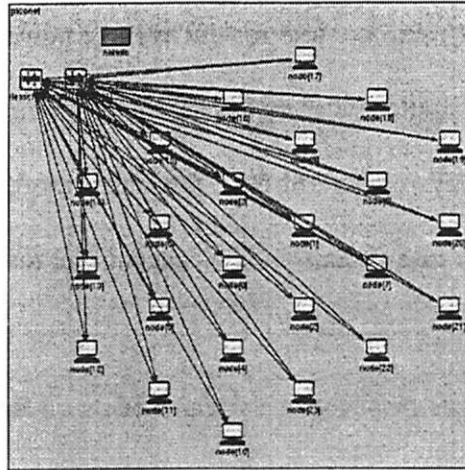


Figure 5.1: Network used to verify the models

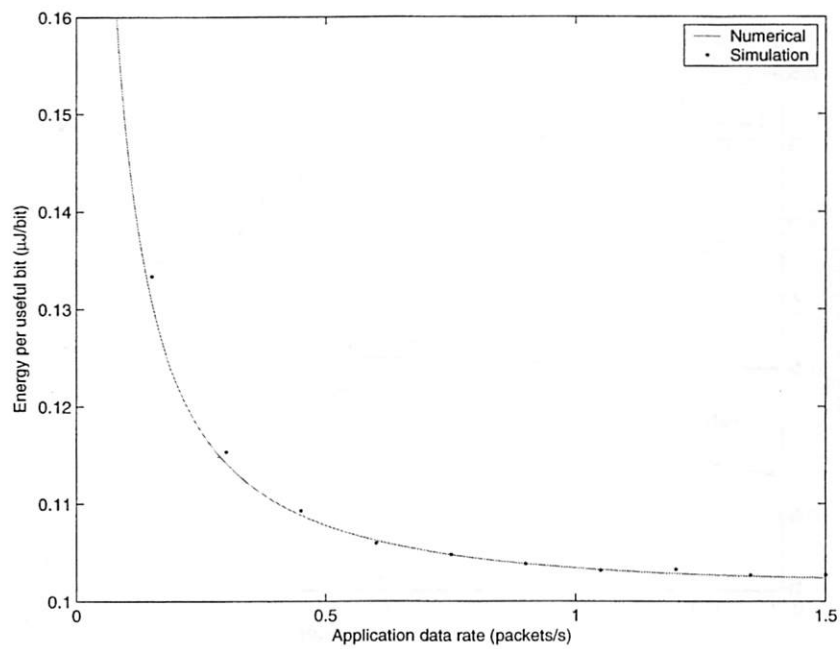


Figure 5.2: Comparison of the numerical result to the simulation result

each other quite well. The same are true for the average power consumption and average packet loss rate used to compute this energy per useful bit metric.

Lastly, the average delay is obtained from these two modeling approaches (shown in Fig. 5.3). Here the traffic rate (x axis) is T_N . Again, the results of the two approaches match each other quite well.

In conclusion, the results from the analytical models on four different design metrics all come very close to those from the network simulations. As a result, one can have confidence on these models. In the next chapter, analysis will be done using these models and the guidelines obtained from the analysis will be shown to have a big impact on the design of a wireless sensor network.

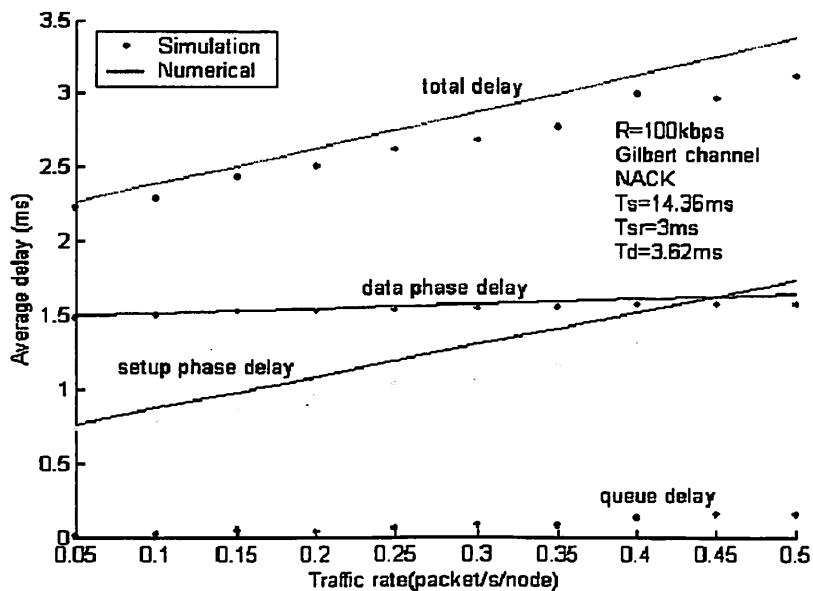


Figure 5.3: Average delay

CHAPTER 6

ANALYSIS AND DESIGN

The advantage of the established framework is that a rich amount of information can be obtained. The information obtainable includes but is not limited to:

- The impact of any design parameter on the energy or QoS metrics
- Comparison between two designs
- Leads to more power-efficient designs

There is no doubt that this framework gives designers deep insight into the design of wireless sensor networks.

But the next question is what a designer can do with it. Can he use the rich amount of information provided to make a big impact to the design of wireless sensor networks?

The answer is yes. This framework helps designers on the the selection of design options and the choice of design parameters. It also enables them to use the information from existing designs to invent new ones and verify the performance of those new designs. Furthermore, how to manipulate the information is up to the designers, who can be very creative. The outcome of using this framework is therefore far beyond what have been mentioned so far. In the following, several examples are given to demonstrate the impact of this framework on the design of the different parts of a wireless sensor network.

6.1 Impact to the Physical Layer Designs

6.1.1 Impact of the radio data rate

Before this unified framework is established, designers were confronted with many questions that they do not have enough information to answer. For example, one question is what data rate should the radio uses. On one side, the radio becomes more complicated to support a higher data rate; on the other side, a node can reduce its duty cycle with a higher data rate. A designer can hardly make a decision at this level. This framework however can aid him in reaching a decision for any given radio architecture, as illustrated in the following example.

The choice of the radio data rate has impact on many parts of a system, including the PHY, the PM and the MAC. The integrated modeling framework developed can be used to demonstrate the overall effect of the radio data rate on the average power consumption, given obviously the modeling constraints. The results are shown in Fig. 6.1.

Chapter 2 has discussed the impact of the radio data rate on the PHY used in this analysis. The results are repeated here: the power consumption of the RF frontend in the receive chain and that of the oscillator [31] are relatively insensitive to the data rate. The power consumption of the digital base band (DBB) in the receive chain increases linearly with the data rate. Lastly the power consumption of the power amplifier (PA) increases only sub-linearly with the data rate.

The effect of the data rate on the PM is as follows: when the radio data rate is increased, a node will spend less time in transmitting and receiving. When the proper power

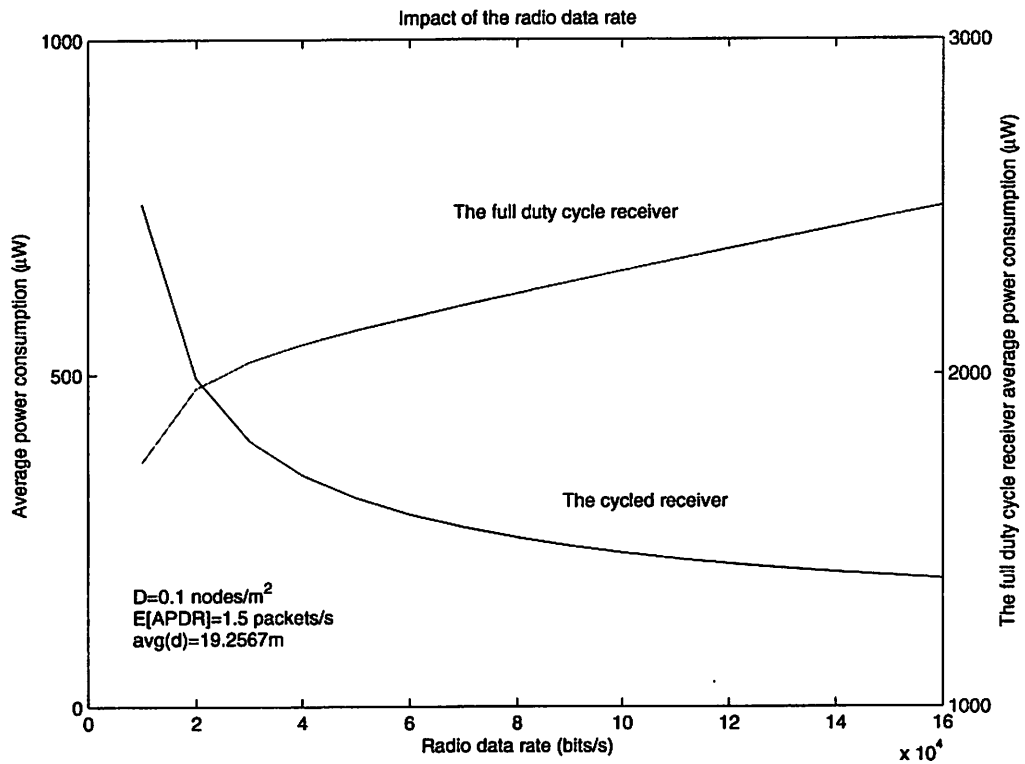


Figure 6.1: Impact of the radio data rate

management is used, the time in the monitor mode will be reduced too.

The data rate influences the MAC too: the higher the radio data rate, the less time will a packet occupy the channel, resulting in less collisions.

So when the data rate is increased, on one hand the power in each mode increases; on the other hand, the time that a node stays on will be reduced and less energy will spend on retransmissions. One cannot see the overall effect without using an integrated modeling framework.

Fig. 6.1 shows that for the cycled receiver the overall effect is that the average power

consumption goes down when the data rate increases. But the opposite is true for the full duty cycle receiver (FDCR). This is because the time a nodes spends in the monitor mode dominates in the FDCR and does not decrease with the data rate. Consequently, the average power consumption goes up instead. One may conclude from Fig. 6.1 that with proper power management, using the highest radio rate possible is always a good thing. However, as pointed out in Chapter 2, the radio data rate can not be increased infinitely. The envelope detector used in this PHY suffers from the inter symbol interference at higher data rates. The highest data rate the detector can be used at is about 160 kbps. On the other hand, if a more complex modulation scheme is used to increase the data rate beyond 160 kbps, P_{rad} may increase linearly with the data rate. In addition, going to a more complex modulation raises the noise floor.

The above example uses the developed framework to study the overall effect of the data rate. In general, this unified framework enables a system architect to understand the impact of any key design parameter, who can then go one step further to make the right design tradeoff.

6.1.2 Channel hop scheme and its analysis

The following design makes the tradeoff between the application layer and radio design, to which this integrated modeling framework is essential.

The use of a simple LC oscillator enables the full integration of DBB and RF frontend on a single chip, the result of which is a very low cost and low power radio. Keeping the

total cost of a sensor network the same, more sensors can be implemented to either have higher fidelity sampling, or perform finer control. However, this simple LC oscillator has such a large frequency variation that a receiver's carrier frequency may be very different from that of a sender, making the communications between them difficult. Therefore the end-to-end packet loss rate has become a big concern.

Earlier results from the modeling effort of a cycled receiver suggest that using the first available candidate provides a lower packet loss rate than using a fixed candidate. There a virtual channel between a sender and its receiver is defined. When the receiver is powered off, the virtual channel is considered bad. In general, when the virtual channel between a sender and a receiver is bad, it makes sense for the sender to use other candidates who have a good virtual channel. Going back to the problem of using a simple LC oscillator, here a virtual channel is regarded bad if the carrier frequency of the sender differs from that of a receiver by an amount higher than the signal bandwidth.

Therefore the concept of a virtual channel can be generalized as follows: the virtual channel between two nodes is considered bad if any of the following is true:

- the received packet is corrupted due to fading
- collisions happen
- the receiver is powered off
- the receiver's carrier frequency differs from that of the sender by an amount higher than the signal bandwidth

Similar to the case of a cycled receiver, a sender should send its packet to the first candidate whose virtual channel is not bad. The details of this design are as follows: a sender listens for the beacons from its neighbors. When a beacon is received, it assumes the channel between the neighbor and itself is good. Among these neighbors, it will send the packet to the one who is in the direction of the final destination and whose beacon is received first. So only one neighbor will forward the packet. It will follow the same procedure above to send the packet to the next node, until the final destination is reached.

Obviously the more candidates are available, the lower the packet loss rate. Hence in the application layer, the design parameter, network density, has a direct impact on the probability of outage, defined as the end-to-end packet loss rate. Fig. 6.2 shows the probability of outage as a function of the network density. The values of the parameters used are: every sensor will generate data at an average application data rate of 1.5 packets per second. The average distance between a sender and its final destination is 19 meters. Furthermore, the maximum angle of the target direction is 60 degrees. The ratio between the maximum frequency variation of the LC oscillator and the signal bandwidth is 100. Lastly, the hop distance is set to 10 meters.

It can be seen from Fig. 6.2 that the probability of outage goes down from more than 90% to well below 10% when the network density increases by 10 times from one node per square meter. From Fig. 6.2 one can also identify the minimum network density required for a given outage probability. For example, to maintain a outage probability of 10%, a density of at least six nodes per square meter is needed. With lower duty cycle, this thresh-

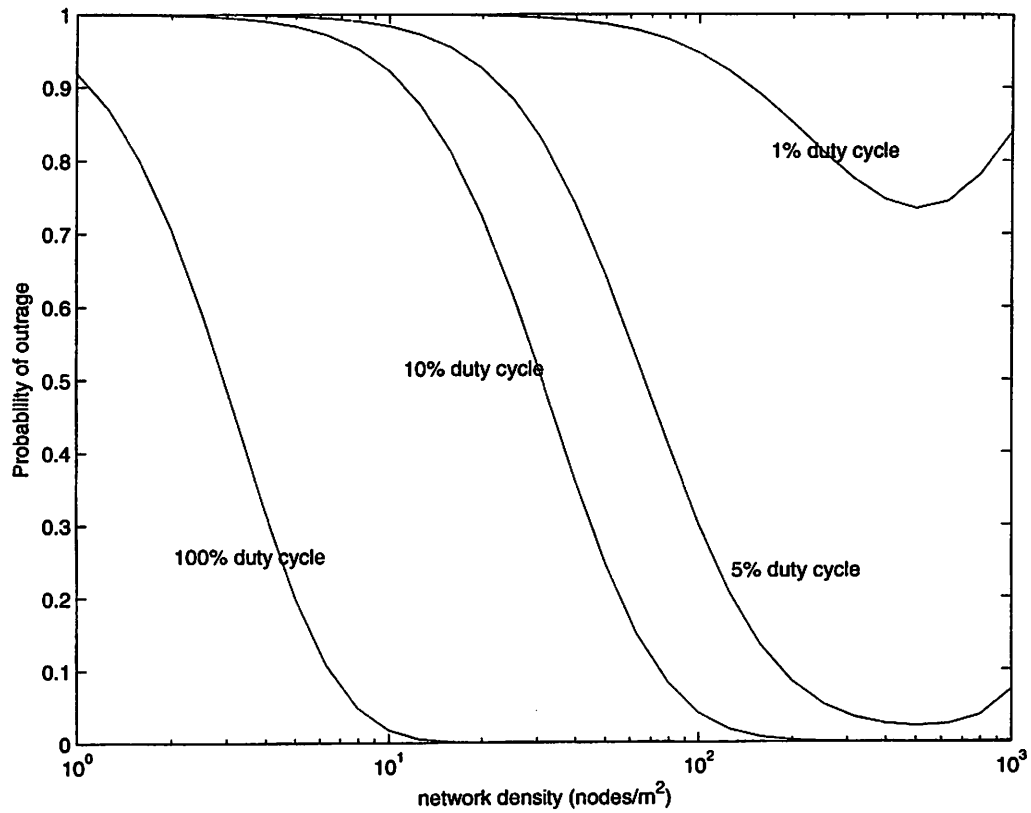


Figure 6.2: Impact of the network density on the outage probability

old increases because the effective density reduces. Consequently, a tradeoff can be made between the power consumption (duty cycle), cost (density) and QoS (outage probability) based on application requirements. The framework developed can help designers making this tradeoff.

6.2 Impact to the Data-link Layer Designs

6.2.1 Comparison of the wakeup radio to the cycled receiver

When a designer comes up with a new design, he always needs to compare it to the best of the existing designs. More than often, one design is better than the other only under certain conditions and worse otherwise. This modeling framework allows a designer to evaluate his design of a component in the context of other components. Additionally it can show under what conditions his design is better than an existing one. Here the comparison of two PM designs is used as an example.

Fig. 6.3 shows under what conditions the cycled receiver is better than the wakeup radio. For example, when the network density is $1 \text{ node}/m^2$ and the average application data rate is 1.5 packets/s, the cycled receiver is better than the wakeup radio if the extra radio used to do wakeup consumes more than $80\mu W$. In other words, the Y axis of Fig. 6.3 is this threshold. It changes with the average application data rate and the network density. When the application data rate is lower, it is easier for the cycled receiver to beat the wakeup radio. The cycled receiver can also take advantage of the increased network density. If a network is more dense, for the same radiated power level, there are more candidates available, enabling the cycled receiver to reduce the duty cycle of every candidate. As a result, the power consumption of the cycled receiver is lower if a network is more dense.

One can see from the above example that if the extra radio used to do wakeup is to be designed, this framework can point out the threshold below which the power consumption of the extra radio has to be for the wakeup radio scheme to be better than the cycled receiver

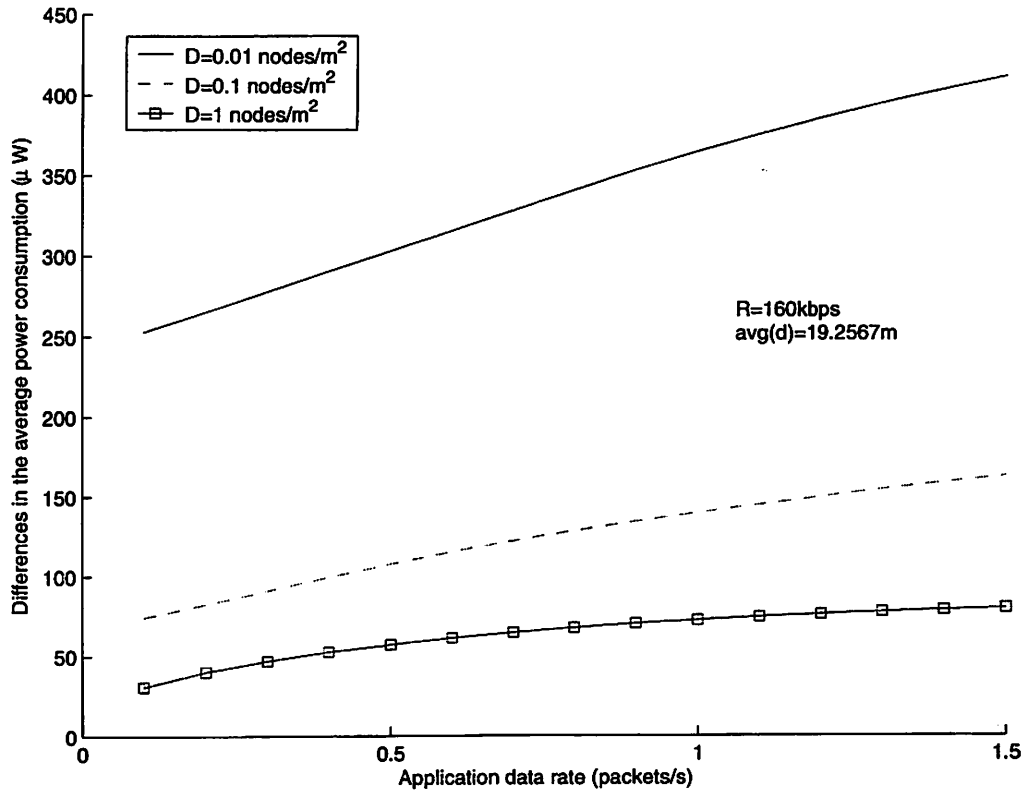


Figure 6.3: Comparison between the wakeup radio and the cycled receiver

scheme. Generally speaking, this modeling framework is able to help designers specify the design requirements for any particular scheme.

6.2.2 The joint design of the power control and the power management

This example demonstrates that the framework is able to help designers to invent new algorithms.

6.2.2.1 The PC-dominated interval and PM-dominated interval

The modeling framework is used to provide designers with information on how to design power control and power management in different network density regions. Using the cycled receiver as the PM design and assuming N_N , the number of active neighbors, remains constant, one can rewrite (4.8) in terms of d_H and T as:

$$E[P] = \kappa \cdot d_H^{n-1} + \varsigma \cdot d_H + \frac{\rho}{d_H} + \frac{\alpha}{T} \quad (6.1)$$

κ is called the transmit coefficient and is related to the antenna gain, power amplifier efficiency and receiver sensitivity. ς , named as the overhearing factor, is related to the network density and the power consumption of the PHY in the receive (RX) mode $P_{R,r}$. ρ , the energy floor, is related to the power consumption of the oscillator and that of the PHY in all three receive (RX, MN and AQ) modes. In addition, all of them are related to the application data rate, the average end-to-end distance between a source and its destination (\bar{d}) and the radio data rate R . What one can see from (6.1) is that as d_H increases, the power consumption in transmitting and overhearing increases while the power consumption in receiving reduces because the number of hops has become less. α is related to only $P_{R,mn}$ and T_{on} . It is independent of both d_H and T , while κ , ς and ρ are independent of d_H but depend on T . For the sake of illustration, they are assumed to be independent of T for the time being.

The sum of the first three terms in (6.1) relates to d_H only, so it is called **the PC-controlled power consumption**. Similarly the last term in (6.1) is called **the PM-controlled power consumption**. When a network is infinitely dense, the last term disappears because

T is very large. The optimal hop distance d_{Hopt} is defined as the hop distance that results in the lowest overall power consumption while ensuring a connected network. The d_{Hopt} when a network is infinitely dense is denoted as $d_{Hopt.inf}$, which corresponds to the lowest PC-controlled power consumption.

For any network, d_{Hopt} and $d_{Hopt.inf}$ divide d_H into three intervals.

- If d_H is below $d_{Hopt.inf}$, as it increases, T can be made larger without sacrificing the number of active neighbors, hence the PM-controlled power consumption will go down. The PC-controlled power consumption decrease as well since d_H is getting closer to $d_{Hopt.inf}$.
- If d_H is in the interval between $d_{Hopt.inf}$ and d_{Hopt} , as it moves away from $d_{Hopt.inf}$ and gets closer to d_{Hopt} , the PC-controlled power consumption increases, but the amount of this increase should be less than the corresponding reduction in the PM-controlled power consumption so that the sum of the two still reduces. This interval is hence called **the PM-dominated interval**.
- Lastly if d_H is greater than d_{Hopt} , the increase in the PC-controlled power consumption is higher than the decrease in the PM-controlled one, as d_H becomes larger. This interval is therefore called **the PC-dominated interval**.

In summary, d_{Hopt} in comparison to $d_{Hopt.inf}$ tells a person information about where power control and power management dominates. To gain insight to the design of the two for different network densities, this modeling framework is used to produce Fig. 6.4, which

highlights the comparison of d_{Hopt} to $d_{Hopt.inf}$ for different network densities. In Fig. 6.4, the area above the d_{Hopt} v.s. density curve is the PC-dominated interval. The area between the curve and the dashed line is the PM-dominated interval. It should be noted that in this numerical result κ , ζ and ρ are not assumed to be independent of T , nor is N_N assumed to be a constant. The cycled receiver uses the optimal duty cycle associated with each d_H .

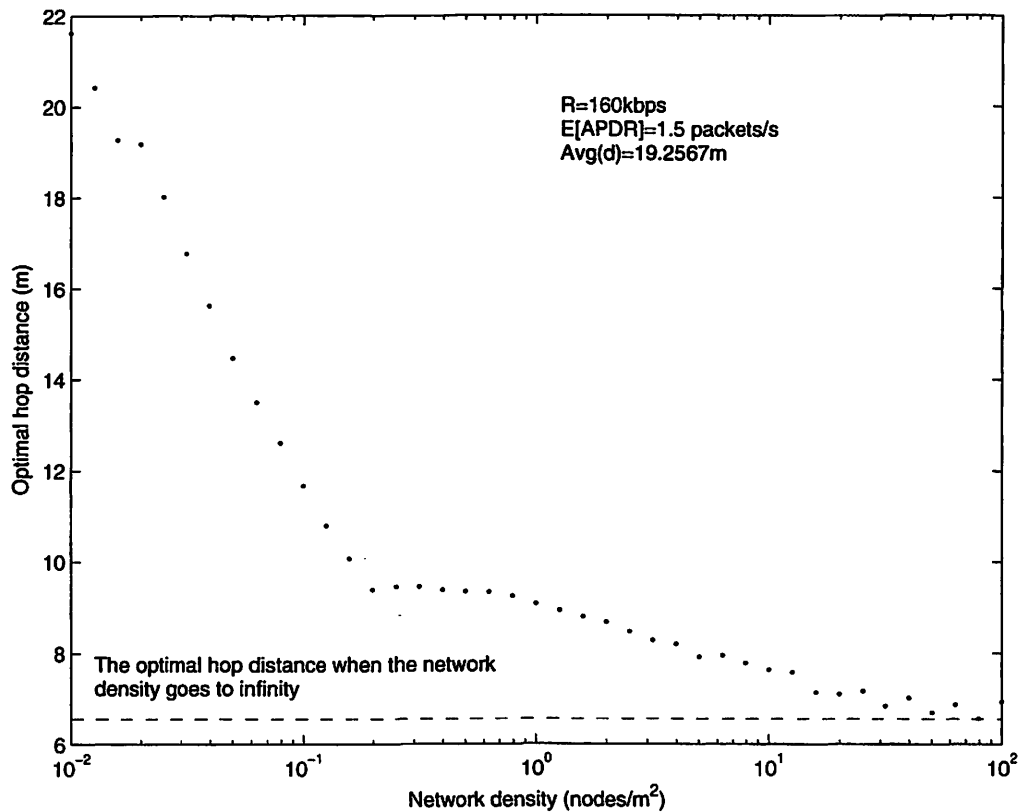


Figure 6.4: Optimal hop distance as a function of the network density

6.2.2.2 The three design zones

Fig. 6.4 shows the influence of the network density on d_{Hopt} . One can see from Fig. 6.4 that there exist three areas: high slope linear, small slope and almost flat.

A network is sparse in the first area, where connectivity requirement dominates. In this area, the hop distance that results in the lowest overall power consumption, computed using the optimization after integration technique in Chapter 4, is smaller than d_{min} , the minimum hop distance needed for connectivity. Hence $d_{Hopt} = d_{min}$. This explains why d_{Hopt} changes linearly with the network density in this area. For any hop distance d_H ensuring network connectivity: $d_H \geq d_{min}$ (i.e. it is in the PC-dominated interval), so one always see "the power control dominates" in this case. Furthermore, d_{min} is relatively high in sparse networks and not known in advance. Consequently, the design in this area should be as follows: power control is essential and it must be very accurate. PM plays a very small role here since the PC-controlled power consumption dominates.

The curve stabilizes in the third area where a network is very dense. This will be explained in the next paragraph. For now it needs to be pointed out that the PM-dominated interval is very narrow in this case. However a node can still operate with a d_H in this interval since the network density is so high that the node needs not worry about the connectivity. In fact, it is not a good idea to use a d_H much higher than $d_{Hopt.inf}$, since the d_H will be in the PC-dominated interval where the return from increasing d_H is not justified. Based on these observations, one should design the PC in this area such that it sets a fixed power level for every node. This power level corresponds to $d_{Hopt.inf}$. However, the duty

cycle of the cycled receiver must be accurately tuned to control the number of active neighbors. The duty cycle will be very low since a node has many neighbors and can afford to allow each of them to sleep longer.

The most interesting networks are the ones with intermediate densities. Fig. 6.4 shows d_{Hopt} for these networks are higher than $d_{Hopt.inf}$ and the amount of increase reduces as a network becomes more and more dense. This is because when d_H increases, more candidates will be available in the neighborhood, enabling one to reduce the duty cycle of each candidate further. As a result, the PM-controlled power consumption in (6.1) will be reduced, contributing to the reduction of the overall power consumption. But as density increases, the PM-controlled power consumption for each candidate is already very low at $d_{Hopt.inf}$, leading to smaller headroom for one to increase d_H while still reducing the overall power consumption. In this area, both the PM-dominated interval and the PC-dominated interval exist. One has seen so far that in sparse or very dense networks, the PC sets d_H based on either connectivity or $d_{Hopt.inf}$. In both cases, The PC sets d_H independent of the PM. The PM simply works with a neighborhood already determined by d_H and adjusts the duty cycle to meet the minimum N_N requirement. However in the area of intermediate density, the collaboration of the PC and the PM is necessary to optimize the overall power consumption: the PC increases d_H on purpose for the PM to leverage on the increased number of candidates. This framework is very well suited to addressing the interactions between the two: the optimization after integration technique has been very useful in determining both the right power level and duty cycle to use in this area.

Table 6.1: Three design zones based on network density

Zones	Characteristics	Design
Sparse	$d_{Hopt} = d_{min}$	Accurate power control
Intermediate	Between the two	The power level needs to be set together with the duty cycle
Very dense	d_{Hopt} is within 10% of $d_{Hopt.inf}$	The power level is set in disregard to the duty cycle. PM is used to reduce N_N afterward

In conclusion, this modeling framework suggests a design must be different for different network density. Table 6.1 summaries the designs for these three zones. One can see from Fig. 6.4 that at the application data rate of 1.5 packets per second, the intermediate zone includes networks whose densities are between $0.1 \text{ nodes}/m^2$ and $10 \text{ nodes}/m^2$. Most of sensor networks fall into this zone where the co-design of the power control and power management can achieve lower energy consumption.

6.3 Impact to the Network Layer Design

In the following example, the established framework is used to make the tradeoff between the EC, PC, MAC components and the network layer.

6.3.1 Introduction and motivation

The vast majority of the reported sensor network research is devoted to rather complex multi-hop structures. On the other hand multiple real applications (e.g. WiFi) have a rather simple, frequently single hop topology. In addition it is very easy to enumerate multiple advantages from using single hop structures:

- Lower power consumption as will be shown in this section
- Simpler protocol stack: no network layer and no need for complex routing
- Easier global synchronization
- Less packet loss due to no collisions (since time division multiple access (TDMA) can be used) and fewer hops
- Lower latency from having only one hop and no backoff
- Better support to a centralized application (e.g. the indoor environment control application)
- Possibility of using many centralized algorithms (e.g. the controller-aided locationing)

On the other hand the common wisdom in wireless sensor networks says: energy economy is best achieved by the usage of low power radios with very short range (up to a few meters), thus coverage of distances exceeding this very limited range calls for multi-hop configurations as the only option!

In this thesis it is claimed that under realistic assumptions as to architecture of the modern sensor networks radios and their computation facilities, the usage of highly efficient coding (close to the Shannon limit) is feasible and can allow for significant extension of the range of efficient communication. As a result a new approach to design WSNs is

proposed: single hop asymmetric structure (SHAS) and the borders of their applicability will be explored.

The rest of this section is organized as follows. First the generic sensor node radio architecture is described, the major blocks given and their energy consumption discussed. Next selected coding approaches will be discussed, especially the asymmetric codes having low encoding complexity and high decoding complexity. After this, the SHAS is presented and compared to the traditional approaches where no forward error control (FEC) is used and multi-hop structures must be adopted.

6.3.2 Description of the generic sensor node radio architecture

Fig. 2.6 shows the components consuming most of the power in the transmit chain: the power amplifier (PA), oscillator and encoder. A PA amplifies the input signal to the desired radiated power level. Its power consumption is therefore closely related to the radiated power level. The efficiency of a PA is not a constant. In fact, it reduces significantly as the radiated power level gets lower. Typically, the rated efficiency of a PA is its efficiency at the peak power level. But the rated efficiency of a PA with a lower peak power level is only slightly lower than that of another PA with a higher peak power level, assuming each PA is optimized for its own peak power level. For the sake of simplicity, the rated PA efficiency is assumed to be the same for different peak power levels.

To compute the power consumption of the PA from the $\frac{E_b}{N_0}$ required for a target BER, a transmitter model, a channel model and a receiver model are needed. For instance, (2.1) is

the model for the particular transmitter described in Chapter 2 and (2.7) is the large-scale path loss channel model. (2.8)-(2.10) are the models for the receiver in Chapter 2. They are used to compute the receiver sensitivity needed for a given $\frac{E_b}{N_0}$:

$$S = \sqrt{\frac{E_b}{N_0} \cdot \frac{R}{BBW}} \cdot k \cdot T \cdot BW \cdot NF \quad (6.2)$$

where R is the information data rate. So (2.1), (2.7) and (6.2) together can compute the PA power consumption from $\frac{E_b}{N_0}$.

An oscillator generates the transmitted carrier frequency as well as the local oscillator signal. Its power consumption is independent of the distance or the coding used.

An encoder encodes information so as to deliver it reliably while using a lower radiated power level. The complexity of an encoder, represented by the gate count of its circuitry, determines its power consumption. For the same clock frequency and supply voltage, the higher the gate count of an encoder, the more power will it consume. The power consumption of an encoder has two components: dynamic power and leakage power. The dynamic power is consumed when a gate switches, charging a previously discharged load. It can be roughly estimated by $C \cdot V^2 \cdot f_{clk}$, where C is the capacitance of the load, whose value is proportional to the gate count, V is the supply voltage and f_{clk} is the clock frequency, which can be made equal to or even smaller than the incoming data rate R . More accurate estimate can be obtained through Register Transfer Language (RTL) simulation if a circuit is available. Among many things, the probability of actual gate switching is more accurately assessed in a RTL simulation than in a hand calculation.

The leakage power is consumed even when a gate does not switch. This is because there

is always a leakage current flowing through the gate. At low data rates, the leakage power consumption dominates the total power consumption of an encoder. Therefore it is crucial to keep the leakage power consumption in check. The use of low-leakage (high V_T) device can help reduce the leakage power consumption. Another way to reduce the leakage power consumption is to turn off the circuit completely between incoming data bits and only store some important state information. The way to estimate the leakage power consumption is to go to the technology file of the fabrication process used (e.g. 0.13 μm) to find the leakage power consumption for a gate, from which and the gate count the total leakage power consumption can be obtained. Once again, RTL simulation can provide a much more accurate estimate of the leakage power consumption than this hand calculation.

Fig. 6.5 shows how the total transmit power consumption changes with the distance between a sender and its receiver. Here a reconfigurable convolutional encoder (made by STMicroelectronics Inc.) with constraint length $K = 7$ and rate $1/2$ is used. Soft-decision Viterbi decoding is used at the receiver side. The encoder has 456 gates. RTL simulation indicates that the dynamic power consumption of this encoder is only $2\mu\text{W}$ at the clock frequency of 100kHz , while its leakage power consumption is $67\mu\text{W}$. Thus the power consumption of this encoder is $69\mu\text{W}$. A non-reconfigurable version of this encoder can be implemented using only 100 gates, bringing down the power consumption to only $15\mu\text{W}$. What is more, leakage control techniques mentioned above can be used to further reduce the leakage power consumption. The RF frontend described in Chapter 2 is used to produce this figure and other figures in this section. The noise bandwidth is 300 kHz, the data filter

bandwidth is 150 kHz and the system noise figure is 20 dB. These parameters correspond to a receiver sensitivity of -93 dBm at 100 kbps when no FEC is used. The receiver sensitivity increases when FEC is used, since the required $\frac{E_b}{N_0}$ for a target BER reduces. For example, the receiver sensitivity becomes -98.5 dBm at 100 kbps for the convolutional code just mentioned. Fig. 6.6 shows the breakdown of power consumption among the PA, oscillator and encoder. The oscillator [31] itself consumes 300 μW , while the other components (e.g. buffer) in the transmit chain consume a total of 200 μW . For the sake of simplicity, P_{T_osc} is used denote the sum (i.e. 500 μW).

6.3.3 Selected coding approaches

The information theory specifies the fundamental limit on what coding can do: to achieve reliable communication, with as small an error probability as desired, the minimum $\frac{E_b}{N_0}$ required is $\ln 2$. This is known as the Shannon bound, which is the benchmark for accessing code performance. Advanced coding nowadays can get very close to this bound. For example, Turbo codes can get within 1dB of the Shannon bound, while Low-Density Parity-Check (LDPC) codes can get within 0.1dB of the bound.

Even though the decoders for these advanced codes are normally very complicated, their encoders are much simpler. Generally speaking, an asymmetric code is a code whose encoder is much simpler than the corresponding decoder. Many codes are asymmetric codes. Convolutional codes for instance are asymmetric codes. The same are true for Turbo codes and LDPC codes. For example, the gate count of a LDPC decoder is typically

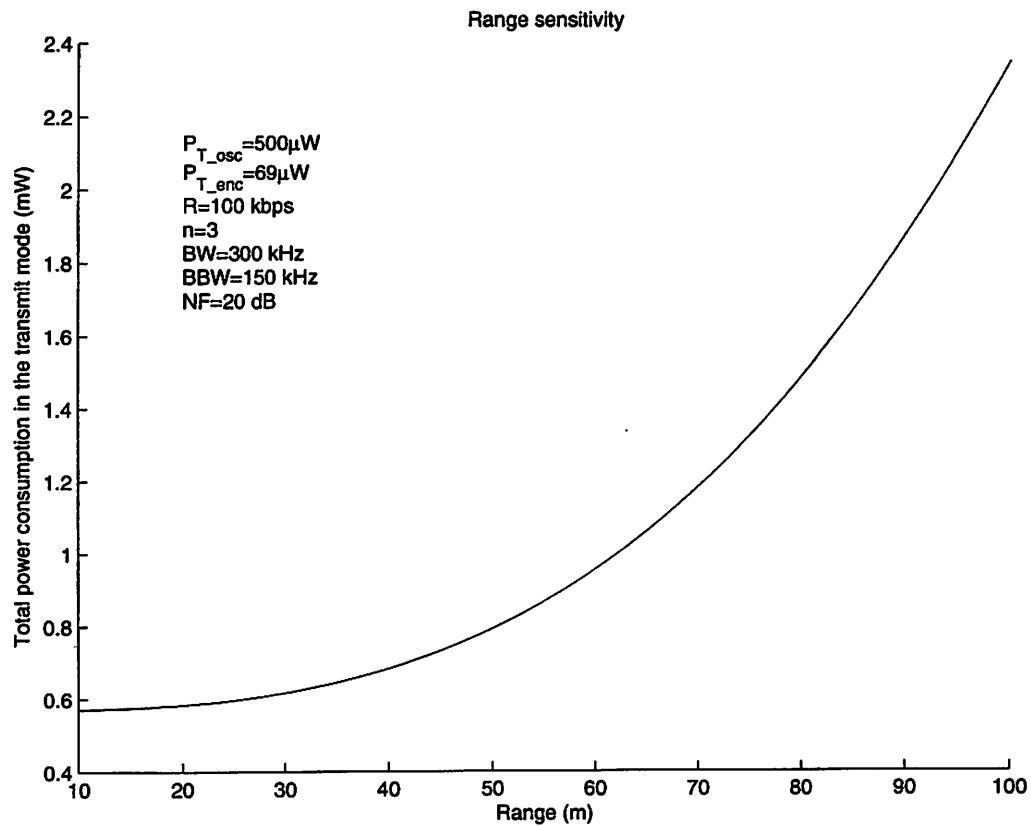


Figure 6.5: Total transmit power as function of range

five times of that of a corresponding encoder.

If only the power consumption in the transmit chain is concerned, the choice of coding depends on the tradeoff between the power consumption of the PA and encoder. As a code gets closer to the Shannon bound, lower radiated power level is needed to achieve the same BER at the receiver. Therefore the power consumption of the PA can be reduced (assuming of course a different PA is used, which is optimized for the new peak power level). On the other hand, the corresponding encoder becomes more complicated, requiring a higher

number of gates. As a result, the power consumption of the encoder goes up. Hence, an optimal point exists between the Shannon bound and where no FEC is used at all.

While it is easy to compute the power consumption of a PA using (2.1), (2.7) and (6.2), it is quite difficult to obtain the number for an encoder since very little data has been published on the gate counts of various encoders. Fig. 6.7 shows how the power consumption of a PA changes as a function of the distance to the Shannon bound. Here the radiated power level is set to assure a BER of 10^{-3} . When the distance to the Shannon bound be-

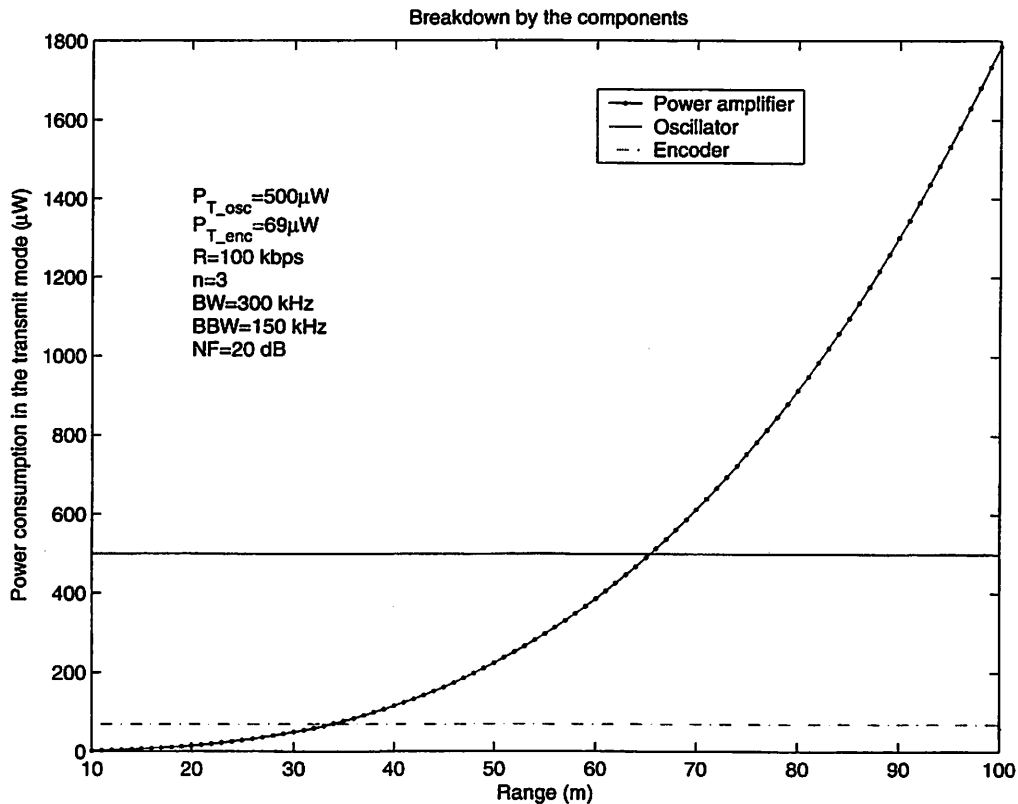


Figure 6.6: Breakdown by the components

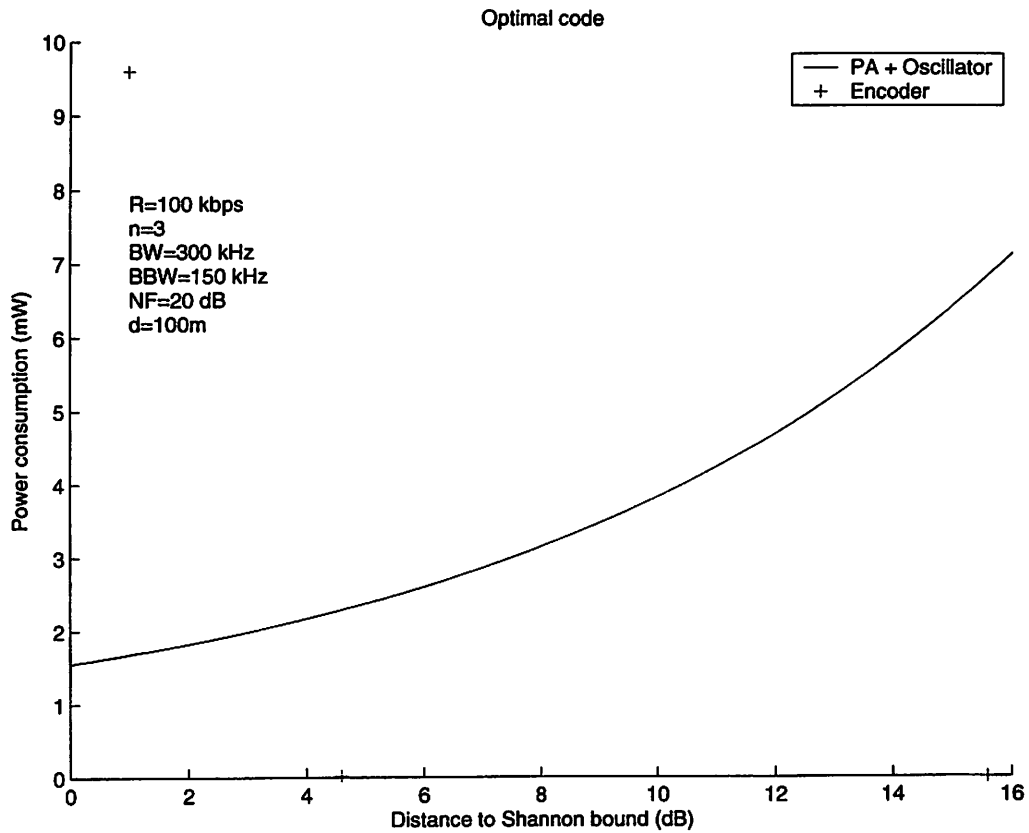


Figure 6.7: Coding power as function of the distance to the Shannon bound

comes larger, higher radiated power level is required to maintain this BER. Consequently, the power consumption of the PA increases as the distance to the Shannon bound increases. Also shown in Fig. 6.7 are the power consumptions of three encoders whose data are available. These encoders include a LDPC encoder by Flarion Technologies, the convolutional encoder mentioned earlier and a cyclic redundancy check (CRC) encoder. The LDPC encoder has 128 parallel paths and requires 64k gates. It can be seen from Fig. 6.7 that the convolutional code outperforms the uncoded case. Hence the optimal point must be within

4.6 dB of the Shannon bound. But exactly where the optimal point lies strongly depends on the implementation. As new architectures emerge, enabling the use of fewer gates to implement the same encoder, and as leakage reduction techniques are used, the optimal point will be very likely to move toward the Shannon bound.

6.3.4 Definition of the single hop asymmetric structure

A SHAS consists of a BASE and a set of sensors/actuators called hereafter satellites. The BASE has a good power supply and is thus NOT energy limited. The satellites are - as usually assumed within sensor networks - strongly energy limited. It is assumed that satellites communicate in one hop with the BASE. Below the uplink and the downlink (as seen from the BASE) are discussed separately.

In the downlink, no FEC is used and only CRC is used to trigger retransmissions. This is done on purpose, so that only a simple decoder is needed in a satellite. To cover a long distance, the BASE must use a high radiated power level. The power consumption of the BASE in the transmit mode is plotted in Fig. 6.8 as function of the distance it covers. The BASE can use an even higher power level, such that the receiver in a satellite is almost passive (low receiver sensitivity), greatly reducing the power consumption of a satellite in the receive mode.

In the uplink, advanced coding is used to extend the range of a satellite so that it can reach the BASE with a single hop (at a reasonable energy cost). The code used has to be asymmetric, that is the power consumption of the encoder must be much lower than

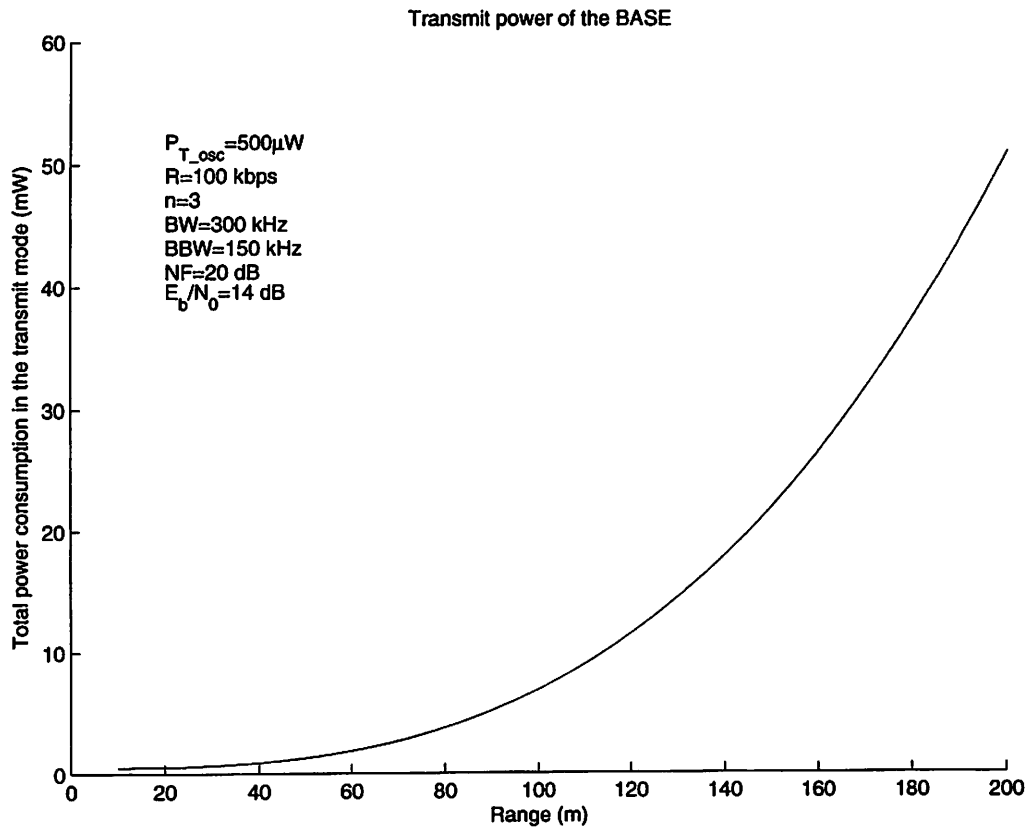


Figure 6.8: Power consumption in the BASE

that of the decoder. The optimal code (if found) discussed in the previous subsection is suggested to be used here. At the same time, the receiver sensitivity of the BASE may also be increased in order to extend the range of a satellite. One way to increase the receiver sensitivity is to increase the RF amplifier gain. Although the power consumption of the RF amplifier (in the receive chain) is higher, since the BASE has a good power supply, this is not an issue. Another way to increase the receiver sensitivity is to reduce the thermal noise by lowering the temperature in the BASE. Since the thermal noise is proportional to the

temperature, the use of cooling techniques (already used in many personal computers) in the BASE is justified.

With SHAS, the BASE takes the penalty of the higher energy cost, enabling the satellites to run at very low power levels. The complicated decoder required to receive the uplink signal resides in the BASE, so its power consumption is not a concern. Similarly, the BASE can afford to transmit at a high power level since it has a unlimited power supply. In addition, one can take this game further by adjusting the receiver sensitivity of both sides accordingly.

The SHAS also opens the door for MAC designs which are not feasible in multi-hop structures. It can use either TDMA or polling: the BASE polls particular satellites for data. Both designs eliminate the collisions as well as interferences within the network.

The satellites only need to use moderate radiated power levels thanks to the advanced coding they use. Since an asymmetric code is used, the power consumption of the encoder is much lower than that of a corresponding decoder. Therefore the power consumption of a satellite can still be reasonably low. With TDMA, each satellite will wait for its scheduled slots to transmit its data to the BASE and can sleep at other times. The use of TDMA also reduces the number of retransmissions needed by the satellites (due to no collisions). In addition, functional equivalent zones can be defined such that only one satellite in a zone needs to transmit and all others can sleep to reserve energy. Therefore, even if the transmit energy cost of a satellite was high due to the long distance between the BASE and itself, it did not need to transmit very often - enabling it to have a very long battery life.

6.3.5 Comparison to the multi-hop structures

If the distance between the BASE and a satellite is long, the transmit power needed can still be many times higher than the case when the distance is shorter (e.g. one hop of the multi-hop case). However, in a multi-hop approach, not only does a packet need to be forwarded multiple times, but more power is also consumed on receiving the packet at every hop. For instance, if there are N hops between the BASE and a satellite, N transmissions and $N - 1$ receptions (excluding that of the BASE since it has a unlimited power supply) will be needed. In reality, a different MAC needs to be used for the multi-hop case since global synchronization is not feasible. For example, carrier sense multiple access (CSMA) can be used. If so, there will be collisions. As a result, more energy will be spent on retransmissions in the multi-hop case. In addition, in a multi-hop structure, there will be the overhead of RTS/CTS (to deal with the hidden terminals) or the handshakes in the power management component to synchronize a sender and its receiver. Furthermore, the time a node spends in the monitor mode in the multi-hop case is going to be much longer than that of the single hop case since a node in the multi-hop case needs to be on even when no packet is sent. Lastly, for the same end-to-end packet loss rate, the required packet loss rate per link in the single hop case can be many times higher than that in the multi-hop case. Consequently, the SHAS can either have fewer retransmissions or transmit at a lower radiated power level, both resulting in energy savings.

On the other hand, multi-hop structures have their own advantages. First, spatial diversity can be exploited. When the channel is bad between the BASE and a satellite (uplink),

the satellite can send its data to another satellite first, the channel to whom is good at the time. Especially in the case of slow fading, time diversity provided by a FEC code is not enough. Secondly, aggregation can be done in multi-hop structures to reduce the total traffic.

In the following, the SHAS is compared to the multi-hop case where no FEC is used. For the sake of simplicity, the required BER is assumed to be the same in both cases. Additionally, it is assumed that there are no collisions and only data packets are sent. It is also assumed no time is spent in overhearing or monitoring in the multi-hop case. It should be noted here that in reality the power consumption of the multi-hop case is higher than what is computed here. The receiver sensitivity of the BASE is assumed to be the same as that of the satellites in the SHAS. So the power consumption of the SHAS can be made even lower if the BASE has higher receiver sensitivity. A more comprehensive analysis using the established modeling framework will be done in the future to incorporate all these factors.

A convolutional code (soft decision, $K=7$, $1/2$ rate) is used in this numerical result. It can be seen from Fig. 6.9 that the SHAS outperforms the multi-hop case at distances up to 175 meters. For many applications, this range is good enough. The reason that the SHAS has lower total power consumption at short ranges is that the use of advance coding has reduced the power consumption's sensitivity to the range (but still exponential). Another reason is the overhead of the multi-hop case (i.e. $N - 1$ receptions) is not small compared to the transmit power needed for short ranges. But as the range goes beyond 175 meters, the exponential growth of power consumption in the SHAS surpasses the linear growth in

the multi-hop case. Hence, the SHAS should not be used if the distance is too long. As noise level increases, this threshold will reduce. On the other hand, the use of a better code makes this threshold higher.

6.3.6 Summary

Results from the modeling framework indicate that when a better code is used in the EC component, the required radiated power level for a given BER (at a receiver) reduces.

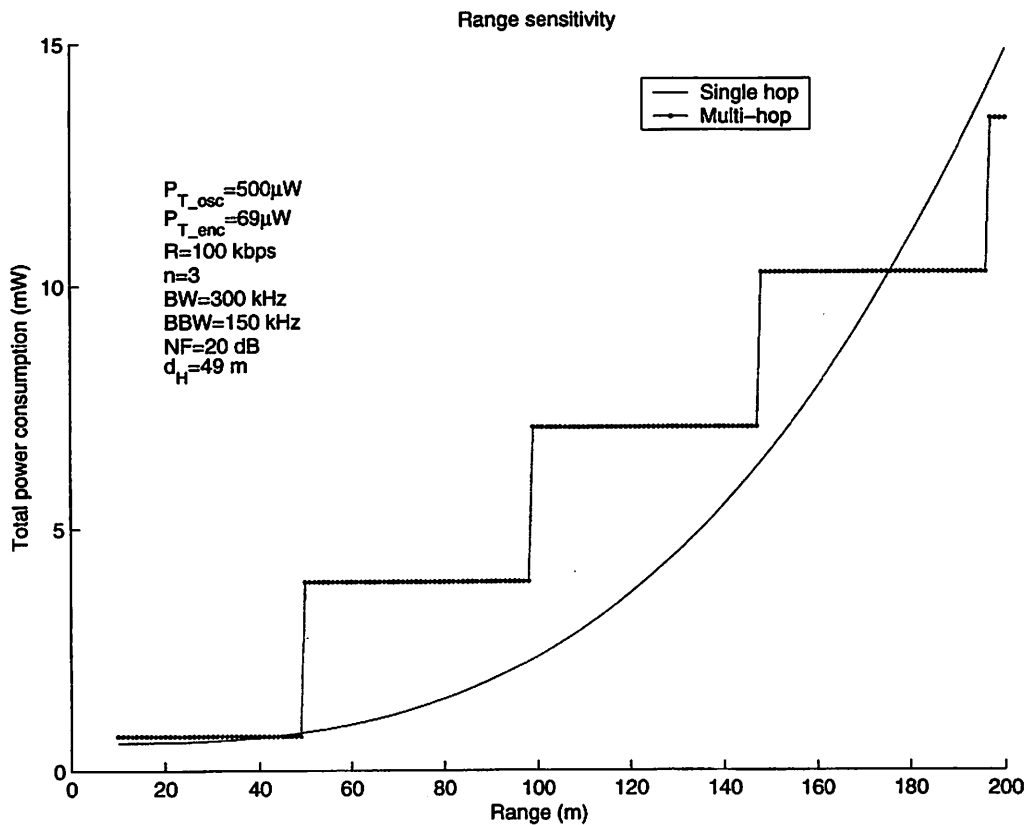


Figure 6.9: Comparison to the multi-hop approach

The implication is a satellite can extend its range so a packet of its can reach the BASE in fewer hops, which results in a lower packet loss rate, a lower delay and reduced power consumption on forwarding and receiving. On the other hand, the power consumption of the corresponding encoder increases. When a satellite can reach the BASE with a single hop, a network layer becomes unnecessary and a simpler protocol stack can be used. Based on these results, a single hop asymmetric approach has been proposed for wireless sensor networks. Its power consumption compares favorably to the traditional approach using multi-hop structures. As technology progresses, power consumption of an encoder can be made very low, even though its code performance is very close to the Shannon limit. The noise in the RF frontend can also be reduced. When these come true, the additional benefits of a single hop structure will make it the preferred choice for many wireless sensor networks.

Several examples have been given to demonstrate the impact of the established framework on the design of the PHY, DLL as well as the network layer. The key message from these examples is that this integrated modeling framework can help produce many new design ideas and make a big impact to the design of wireless sensor networks.

6.4 Future Research

Before concluding this thesis, a few things will be said on the future perspectives of this research.

6.4.1 Power-efficient designs

A integrated modeling framework has been established in this dissertation. It gives designers a lot of insight into the design of WSN. Why not take advantage of this insight? It has been shown in the previous section a number of new designs have been generated from this research. Looking forward, one can expect that a lot more power-efficient designs will be produced from this framework.

6.4.2 Fundamental bounds on power consumption

A fundamental question in WSN is that for a given QoS, how low the power consumption can be made. Is there a fundamental bound, just as the Shannon bound for error control codes?

Can this bound be achieved by using a code near the Shannon limit? How much of the existing information theory results can be used to obtain this bound?

This dissertation has built a strong mathematic foundation on top of which fundamental problems like the one above can be raised and answered. The analytical modeling framework established here has paved the way for others to pursue this fundamental problem in the years to come.

6.4.3 The network lifetime

For a wireless network, the battery life is obviously very important. However it is not the ultimate measure of network performance. A node in the network can be out of service

because its battery is used up, but the network operation may still go on. This has to do with the notion of functionally equivalent zones mentioned in several places of this thesis. From the application point of view, some nodes may be equivalent in terms of functionality. Consequently, even if one node dies, other functional equivalent nodes can still replace it. So what matters to a wireless network is the network lifetime, not the battery life of a particular node.

On the other front, the availability of energy scavenging techniques enables a node to regain energy from its local environment. As a result, a node will never run out of energy. Does it mean the network lifetime is infinite? The answer is no. Even though a node can always get energy from its environment, the rate at which the energy is generated is finite. As a result, if a node consumes its power faster than the rate at which its energy can be regenerated, it has to be out of service temporarily, which may impact the network operation.

In general, one can not talk about the network lifetime without specifying the corresponding QoS. For the same network, a user can choose to tolerate a higher latency or packet loss rate in exchange for longer network life.

The network lifetime is defined as the time a network can operate above a specified level of QoS.

For example, a WSN can last for five years if the required packet loss rate is below 1%, but it can last fifteen years if the required packet loss rate becomes 10%.

The network lifetime depends on the application data rate, initial energy supply, the

rate at which the energy can be regenerated. It also depends on the network density, QoS requirement, and area of function equivalence. In addition, it depends on the energy consumption of every node.

A high level analytical model may be built, using which the network lifetime can be uniquely determined from the above parameters. But for the time being, the network lifetime can only be measured in network simulations or experiments. It is not clear for the moment what is the close form relationship between the network lifetime and these parameters. Finding this relationship is a very exciting research topic for the future.

CHAPTER 7

CONCLUSION

Wireless sensor network has been and will still be an active area of research. However, power consumption is crucial to such a type of network. An integrated modeling framework is presented in this dissertation, revealing the overall power consumption's dependency on the design parameters of the wireless sensor networks, especially those of the data-link layer. Using the insight obtained, designers have developed several very power-efficient designs. In the future, more such designs can be produced. The framework can also become a place to raise and address fundamental questions. Lastly, efforts need to be made to optimize the network lifetime of a wireless sensor network.

References

- [1] L. C. Zhong, J. M. Rabaey, C. Guo and R. Shah, "Data link layer design for wireless sensor networks," *Proc. MILCOM 2001*, vol.1, p.352-6, October 2001.
- [2] K. M. Sivalingam, M. B. Srivastava, P. Agrawal and J. Chen, "Low power access protocols based on scheduling for wireless and mobile ATM networks," *1997 IEEE 6th International Conference on Universal Person Communications Record*, San Diego, CA, USA, 12-16 Oct. 1997.
- [3] K. Sohrabi and G. Pottie, "Performance of a novel self-organization protocol for wireless ad-hoc sensor networks", *50th IEEE Vehicle Technology Conference*, The Netherlands, September 1999.
- [4] W. Ye, J. Heidemann and D. Estrin, "An energy efficient MAC protocol for wireless sensor networks", *IEEE Infocom 02*, June 2002.
- [5] L. C. Zhong, R. Shah, C. Guo and J. M. Rabaey, "An ultra-low power and distributed access protocol for broadband wireless sensor networks", *Networld+Interop: IEEE broadband wireless summit*, Las Vegas, May 2001.
- [6] S. Lee and D. Cho, "Distributed reservation CDMA for wireless LAN", *IEEE GLOBECOM '95*, Singapore, 13-17 Nov. 1995.
- [7] D. Lopez-Rodriguez and R. Perez-Jimenez, "Distributed method for channel assign-

- ment in CDMA based ad hoc wireless local area networks”, *1999 IEEE MTT-S International Topical Symposium on Technologies for Wireless Applications*, Vancouver, BC, Canada, 21-24 Feb. 1999.
- [8] M. Joa-Ng and I. Lu, “Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network”, *IEEE INFOCOMM 1999*, New York, NY, USA, 21-25 March 1999.
- [9] C. Guo, L.C. Zhong and J.M. Rabaey, “Low power distributed MAC for ad hoc sensor radio networks”, *IEEE Globecom 01*, November 2001.
- [10] V. Rodoplu and T. Meng, “Minimum energy mobile wireless networks”, *IEEE J. Selected Areas in Communications*, vol.17,no.8, pp.1333-1344, August 1999.
- [11] R. Wattenhofer, L. Li, P. Bahl and Y.Wang, “Distributed topology control for power efficient operation in multihop wireless ad hoc networks”, *Proc. INFOCOM 2001*, April 22-26 2001.
- [12] S. Borbash and E. Jennings, “Distributed topology control algorithm for multihop wireless networks”, *Proc. IJCNN 02*, vol.1, pp. 355-360, May 2002.
- [13] M. Kubisch, H.Karl, A.Wolisz, L.C. Zhong and J.M.Rabaey, “Distributed algorithms for transmission power control in wireless sensor networks”, *WCNC 03*, March 2003.
- [14] G. Pei and C. Chien, “Low power TDMA in large wireless sensor network”, *Proc. MILCOM 2001*, vol.1, pp.347-51, October 2001.

-
- [15] C. Schurgers, V. Tsiatis, S. Ganeriwal and M. Srivastava, "Optimizing sensor networks in the energy-latency-density design space", *IEEE Trans. Mob. Comp.*, vol.1, pp.70-80, January-March 2002.
- [16] Y.Xu, S.Bien, Y.Mori, J.Heidemann and D.Estrin, "Topology control protocols to conserve energy in wireless ad hoc networks" *submitted to IEEE Trans. Mob. Comp.*, January 2003.
- [17] J. P. Ebert and A.Wolisz, "Combined tuning of RF power and medium access control for WLANs", *Proc. IEEE Intl. Workshop on Mobile Multimedia Communications*, pp. 74-82, San Diego, November 1999.
- [18] C. Guo, *Ph.D. Thesis*, UC Berkeley, 2004.
- [19] R. Stutz, "Performance analysis and optimization of 2.4GHz multi-hop, wireless self-configurable network", *M.S. Report*, UC Berkeley, 2003.
- [20] R. Rom and M. Sidi, *Multiple Access Protocols-Performance and Analysis*, Springer-Verlag, 1990.
- [21] N. Abramson, *Multiple Access Communications*, IEEE Press, New York.
- [22] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks", *IEEE ICC 2002 Computer Society Workshop on VLSI*, New York, NY, USA, 28 April -2 May 2002.

-
- [23] S. Khanna, S. Sarkar and I. Shin, "An energy measurement based collision resolution protocol", *International Teletraffic Congress (ITC18)*, Berlin, Germany, September 2003.
- [24] M. Bhardwaj and A.P. Chandrakasan, "Bounding the life time of sensor networks via optimal role assignments", *Proc. INFOCOM 02*, pp. 1587-1596, New York, June 2002.
- [25] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: part I-carrier sense multiple-access modes and their throughput-delay characteristics", *IEEE Trans. Comm.*, vol. 23, pp. 1400-1416, December 1975.
- [26] E. N. Gilbert, "Capacity of a burst-noise channel", *Bell Systems Tech. Journal*, vol. 39, pp.1253-1266, September 1960.
- [27] L. C. Zhong, "The pico radio data-link layer architecture", *Technical Document*, [http://www.bwrc.berkeley.edu/czhong/documents/data link layer architecture v2.pdf](http://www.bwrc.berkeley.edu/czhong/documents/data%20link%20layer%20architecture%20v2.pdf), June 2001.
- [28] L. C. Zhong, J. M. Rabaey and A. Wolisz, "An integrated data-link energy model for wireless sensor networks", *ICC 2004*, Paris, France, June 20-24, 2004.
- [29] U. G. Schuster, "Simulation and analysis of a low-power radio link for sensor node networks", *M.S. Report*, UC Berkeley, 2002.

-
- [30] E. Lin, A. Wolisz and J. M. Rabaey, "Power-efficient rendezvous schemes for dense wireless sensor networks", *ICC 2004*, Paris, France, June 20-24, 2004.
- [31] B. P. Otis and J. M. Rabaey, "A 300 μ W 1.9GHz CMOS Oscillator utilizing micro-machined resonators", *Proceedings of the European Solid-state Circuits Conference*, Florence, Italy, Sept. 24-26, 2002.
- [32] MAXIM, "Calculating the sensitivity of an ASK receiver", *Application Notes APP2815*, http://www.maxim-ic.com/appnotes.cfm/appnote_number/2815, Nov. 05, 2003.