Protocol Design and Performance Evaluation of Wake-up Radio enabled IoT Networks

Debasish Ghose

Protocol Design and Performance Evaluation of Wake-up Radio enabled IoT Networks

Doctoral Dissertation for the Degree *Philosophiae Doctor (PhD)* at the Faculty of Engineering and Science, Specialization in Information and Communication Technology

University of Agder
Faculty of Engineering and Science
2019

Doctoral Dissertations at the University of Agder 225
ISBN: 978-82-7117-924-3
ISSN: 1504-9272

Printed by 07 Media
Oslo

Dedicated to my parents

Dilip Kumar Ghose and Tapati Ghose

Abstract

As the Internet of things (IoT) is becoming a reality in almost every corner of our society, a tremendous amount of research and development efforts are being made from both academia and industry. In many IoT applications, energy-efficiency of IoT devices/sensor nodes is one of the most important performance metrics. Early research efforts have focused on reducing the energy consumption of sensor nodes through duty-cycling (DC) based medium access control (MAC) mechanisms. However, DC MAC protocols suffer from overhearing and idle listening during the active state of a cycle as well as long delay caused by the inactive state of the cycle. Recently, a paradigm shift from DC MAC to wake-up radio (WuR) has been envisaged. In a WuR enabled node, a wake-up receiver (WuRx) is attached to the main radio (MR) and these two components together form an IoT device. While the MR is only responsible for data communication and it sleeps most of the time, WuRxs are always on listening to the channel continuously. It is demonstrated that the power consumption of a WuRx is at the magnitude of 1000 times lower than that of the MR. Despite this overwhelming advantage, WuR enabled IoT networks suffer from wake-up call (WuC) collision, overhearing, and longer latency for WuC decoding. In this dissertation, we focus on data communication protocols to overcome the limitations of such networks. The objectives of this doctoral dissertation is twofold, i.e., MAC protocol design and performance evaluation via analytical models and simulations.

To reduce WuC collision, four MAC protocols have been proposed for synchronous and asynchronous WuR enabled IoT networks respectively. These protocols employ various techniques, such as clear channel assessment and backoff before WuC transmission for collision avoidance. Furthermore, to diminish overhearing that occurs when a WuR node listens to WuCs which is not intended to it, we propose an energy-efficient WuC address decoding scheme. Moreover, an efficient data transmission scheme which eliminates the necessity of a separate data transmission phase is proposed.

In a multi-hop WuR enabled IoT network, we propose a lightweight relay selection scheme which does not need a route establishment phase and keeps load

balancing among relay nodes.

The performance of the proposed schemes and protocols is evaluated via Markov chain based analysis and discrete-event simulations. As a whole, the protocol design and performance evaluation work presented in this thesis advances the state-of-the-art techniques towards the development and deployment of WuR enabled IoT networks.

Preface

This dissertation is a result of the research work carried out at the Department of Information and Communication Technology (ICT), University of Agder (UiA), Grimstad, Norway, from May 2015 to December 2018. During my Ph.D. study, my main supervisor has been Professor Frank Y. Li, University of Agder, and my co-supervisor has been Professor Vicent Pla, Universitat Politècnica de València (UPV), Spain. From November 2017 to December 2017, I visited UPV as a visiting researcher, hosted by Professor Vicent Pla. From January 2018 to June 2018, I was a visiting research scholar at Rice University, USA. My host professor at Rice University was Professor Edward Knightly.

My academic visit to UPV was sponsored by the University of Agder and the visit to Rice University was jointly financed by the University of Agder and the NorTex project through the International Partnerships for Excellent Education and Research (INTPART) program funded by the Research Council of Norway and the Norwegian Center for International Cooperation in Education.

Production note: LaTeX has been adopted as the tool for writing this dissertation, as well as the papers produced during my Ph.D. study. The mathematical calculations and simulation results are obtained by using MATLAB.

Acknowledgments

First of all, I would like to express my intense gratitude to my supervisor, Professor Frank Y. Li. Without his assistance and guidance, this endeavor would not have been possible. His encouragement, enthusiasm, and vision always motivated me to do fruitful research. I am deeply indebted to him for his meticulous reviews of my manuscripts. His comments and feedback always helped to ensure the manuscripts with high quality. During this Ph.D. journey, he has been not only an advisor but also a guardian. His contribution in my personal and professional development is enormous. During the ups and downs of the journey, he is the only person who always came with a solution and showed the next logical step.

I am grateful to my co-supervisor Professor Vicent Pla for his support and guidance in my research. Many thanks to him for hosting my research visit at UPV. I am amazed by his hospitality during my research stay in Valencia. His insightful comments and feedback always helped to improve the quality of the manuscripts. A special thanks to Professor Edward Knightly for providing me an opportunity to visit Rice University which is a prestigious university with top research quality. It was a lifetime experience which changed my vision towards applied research.

I would like to extend my thanks to my co-authors, Dr. Lakshmikanth Guntupalli, Dr. Luis Tello-Oquendo, Min Zhang, and Anders Frøytlog for their cooperation and contributions. I wish to thank the Ph.D. coordinators at the Faculty of Engineering and Science, UiA, Emma Elisabeth Horneman, Tonje Sti, and Kristine Evensen Reinfjord for their administrative support. I also would like to thank all the professors at the Department of ICT who always encouraged and motivated me.

I am also grateful to my teammates Dr. Indika Balapuwaduge, Thilina Nuwan Weerasinghe, and Associate Professor Lei Jiao for their feedback, help, and encouragements through these years. I am thankful to my office-mates Saeed Rahimi Gorji and Harsha Sandaruwan Gardiyawasam Pussewalage for their cooperation. I really enjoyed the technical as well as non-technical discussions that we had in the office. Such discussions energized me especially when I felt tired. Many thanks to Associate Professor Souman Rudra, Dr. Manika Rudra, Dr. Vimala Nunavath, and Ridwan Bin Zafar for their support throughout the Ph.D. journey.

I am deeply indebted to my parents for being so far away from them. Taking this opportunity, I express my eternal gratitude to them for their blessings, love, understanding, and support. Last but not least, I am grateful to my wife, Smriti Kana Roy, who always supported, motivated, and encouraged me in this endeavor.

Many thanks to all of you for encouragement!

Debasish Ghose
December 2018
Grimstad, Norway

List of Publications

The author of this dissertation is the first author and the principal contributor of all the included papers listed below. Papers A-D in the first set of the following list are selected to represent the main research achievements and are reproduced as Part II of this dissertation. The other papers which are listed in the second set are complementary to the main focus.

Papers Included in the Dissertation

Paper A D. Ghose and F. Y. Li, "Enabling Backoff for SCM Wake-up Radio: Protocol and Modeling," *IEEE Communications Letters*, vol. 21, no. 5, pp. 1031-1034, May 2017.

Paper B D. Ghose, F. Y. Li, and V. Pla, "MAC Protocols for Wake-up Radio: Principles, Modeling and Performance Analysis," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 5, pp. 2294-2306, May 2018.

Paper C D. Ghose, A. Frøytlog, and F. Y. Li, "Enabling Early Sleeping and Early Data Transmission in Wake-up Radio-enabled IoT," *Computer Networks*, vol. 153, pp. 132-144, Apr. 2019.

Paper D D. Ghose, L. Tello-Oquendo, F. Y. Li, and V. Pla, "Lightweight Relay Selection in Multi-hop Wake-up Radio Enabled IoT Networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Dec. 2018.

Other Publications Not Included in the Dissertation

Paper 5 D. Ghose and F. Y. Li, "Enabling Retransmissions for Achieving Reliable Multicast Communications in WSNs," in *Proc. IEEE Vehicular Technology Conference (VTC)-Spring*, May 2016.

Paper 6 D. Ghose and F. Y. Li, "Priority-oriented Multicast Transmission Schemes for Heterogeneous Traffic in WSNs," in *Proc. IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep. 2016.

Paper 7 M. Zhang, D. Ghose, and F. Y. Li, "Does Wake-Up Radio Always Consume Lower Energy Than Duty-Cycled Protocols?" in *Proc. IEEE Vehicular Technology Conference (VTC)-Fall*, Sep. 2017.

Paper 8 D. Ghose, A. Frøytlog, and F. Y. Li, "Reducing Overhearing Energy in Wake-up Radio-Enabled WPANs: Scheme and Performance," in *Proc. IEEE International Conference on Communications (ICC)*, May 2018.

Paper 9 M. Zhang, D. Ghose, and F. Y. Li, "Collision Avoidance in Wake-Up Radio Enabled WSNs: Protocol and Performance Evaluation," in *Proc. IEEE International Conference on Communications (ICC)*, May 2018.

Paper 10 L. Guntupalli, D. Ghose, F. Y. Li, and M. Gidlund, "Energy Efficient Consecutive Packet Transmissions in Receiver-Initiated Wake-up Radio Enabled WSNs," *IEEE Sensors Journal*, vol. 18, no. 11, pp. 4733-4745, Jun. 2018.

Contents

List of Figures

List of Tables

List of Abbreviations

3GPP	Third Generation Partnership Project
5G	Fifth Generation Mobile and Wireless Communications
ACK	Acknowledgment
ADP-WuR	Adaptive Wake-up Radio
AODV	Ad Hoc On-demand Distance Vector
APP	Application
ATS	About to Send
BER	Bit Error Rate
BLE	Bluetooth Low Energy
BO	Backoff
BoWuR	BO-enabled WuR
BPSK	Binary Phase Shift Keying
BS	Base Station
CCA	Clear Channel Assessment
CMOS	Complementary Metal Oxide Semiconductor
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear to Send
CW	Contention Window
DC	Duty Cycling
DCF	Distributed Coordination Function
DIFS	DCF Interframe Space
DSL	Digital Subscriber Line
DTMC	Discrete Time Markov Chain
DW-MAC	Demand Wake-up Medium Access Control
EDT	Early Data Transmission
eMBB	Enhanced Mobile Broadband
eNB	Evolved Node B
ES	Early Sleeping

FAD	Full Address Decoding
FSK	Frequency Shift Keying
gNB	5G Node B
ICT	Information and Communications Technology
IoT	Internet of Things
IP	Internet Protocol
ISM	Industrial, Scientific, and Medical band
LR	Lightweight Relay
LTE	Long Term Evolution
MAC	Medium Access Control
MCU	Micro-controller Unit
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
mMTC	Massive Machine Type Communication
MTC	Machine Type Communication
MR	Main Radio
NAV	Network Allocation Vector
NB	Node B
NR	New Radio
OOK	On-Off Keying
OPWUM	Opportunistic Wake-up Medium Access Control
OSI	Open Systems Interconnection
PCB	Printed Circuit Board
PDR	Packet Delivery Ratio
PER	Packet Error Rate
PRACH	Physical Random Access Channel
PTW	Pipelined Tone Wake-up
PW-MAC	Predictive Wake-up Medium Access Control
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RAT	Radio Access Technology
RBCR	Relay Selection Based Cooperative Routing
RF	Radio Frequency
RI	Receiver Initiated
RI-MAC	Receiver Initiated Medium Access Control
RRC	Radio Resource Control
RTM	Radio Triggered Sensor Medium Access Control
RTS	Request to Send

SCH	Scheduling Frame
SCM	Sub-carrier Modulation
SIFS	Short Interframe Space
S-MAC	Sensor Medium Access Control
SNR	Signal-to-Noise Ratio
STEM	Sparse Topology and Energy Management
TDMA	Time Division Multiple Access
TI	Transmitter Initiated
TICER	Transmitter Initiated Cycled Receiver
UART	Universal Asynchronous Receiver-Transmitter
URLLC	Ultra-Reliable Low Latency Communication
Wi-Fi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WRTA	WuRx-Optimised Routing and Topology Optimization Approach
WSN	Wireless Sensor Network
WuC	Wake-up Call
WuR	Wake-up Radio
WuRx	Wake-up Receiver
WuTx	Wake-up Transmitter
X-MAC	Short Preamble Medium Access Control
XOR	Exclusive Or

PART I

Chapter 1

Introduction

In this chapter, we first give a brief introduction to the Internet of things (IoT) and massive IoT. Then the importance of energy-efficient communications in IoT is highlighted. The motivation of this Ph.D. dissertation is discussed along with the overview of the research questions. Furthermore, the goals and approaches are explained and how the dissertation is structured is also outlined.

1.1 IoT and Massive IoT

The IoT refers to a network infrastructure which links physical objects, i.e., 'things' to the global Internet through the exploitation of data capture and communication capabilities [3]. IoT devices usually have identities, physical attributes, virtual perceptions, and data communication capabilities. As an emerging technology, IoT is playing a pivotal role towards the digital transformations of many societal sectors such as transportation, healthcare, industry, and agriculture.

More recently, the research and development focus on IoT is moving towards the direction of massive IoT, a new terminology referring to the 10's of billions of devices, objects, and machines that require ubiquitous connectivity even in the most remote locations, like sensors buried deep underground [4]. According to an Ericsson report, around 29 billion connected devices are forecast by 2022, of which around 18 billion will be related to IoT [5]. When connecting potentially 1+ million IoT devices per km^2, cellular networks, particularly the 5^{th} generation (5G) mobile and wireless communications, appear as the most popular solution. One important reason for this popularity is the ubiquitous coverage that 5G brings.

To provide ubiquitous connectivity, IoT devices can be connected to the Internet via either a cellular network connection, i.e., direct third generation partnership project (3GPP) connection [6] or an indirect non-3GPP connection [7]. A high-

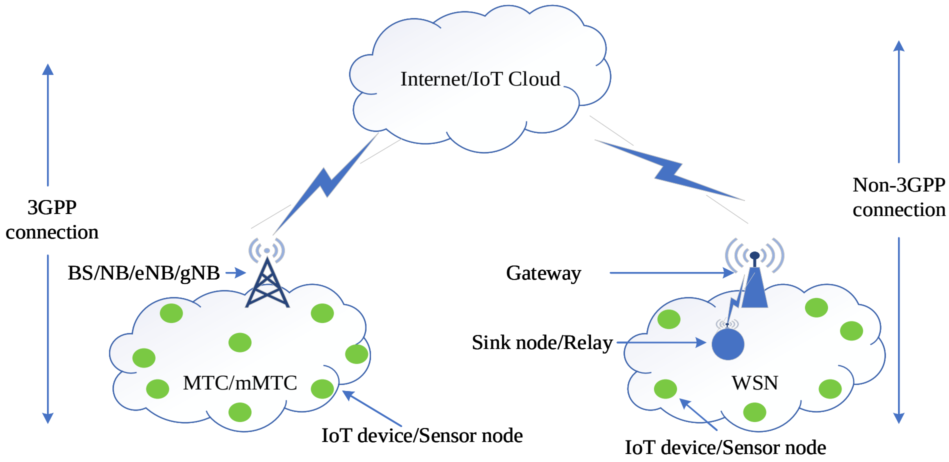


Figure 1.1: An overview of IoT networks with two types of connections.

level overview of the IoT architecture is shown in Fig. 1.1.

1.1.1 3GPP Connections

3GPP connections provide cellular network based radio access technologies (RATs) in which IoT devices can connect to the global Internet via a base station (BS)/ Node B (NB), Evolved Node B (eNB)/ 5G new radio (NR) base station (gNB). Examples of 3GPP connections are 2G, 3G, 4G, and 5G cellular networks. The ubiquitous coverage of 5G new radio will unleash the possibilities of many future IoT applications based on machine type communication (MTC) and massive MTC (mMTC). Indeed, mMTC is one of the three major use cases envisaged by 5G [8].

1.1.2 Non-3GPP Connections

Non-3GPP connections include wireless fidelity (Wi-Fi), digital subscriber line (DSL), worldwide interoperability for microwave access (WiMAX) which are standardized by non-3GPP communities like IEEE. For instance, IEEE 802.15.4e is specially designed to connect IoT devices with the Internet through gateways. To extend the capability of IEEE 802.11, IEEE 802.11ah (also known as Wi-Fi Halow) is standardized to connect IoT devices to the Internet over long distances. It operates in the 900 MHz spectrum to broaden the network size and coverage to much larger than today's Wi-Fi networks.

1.1.3 Key Enabling Technologies of IoT

MTC/mMTC and wireless sensor networks (WSNs) are two key enabling technologies for IoT thanks to the rapid development of sensor technology and its great potential in various application scenarios. In a WSN, multiple IoT devices [1] collaborate with each other to facilitate data exchange through wireless communications.

1.2 Energy-Efficiency in IoT Networks/WSNs

Long lifetime of sensor devices/WSNs is one of the essential requirements in many IoT applications. For instance, it is expected that the node lifetime should be 10+ years for IoT devices [9] [6]. Thus it is imperative to develop techniques that can improve the energy-efficiency of WSNs to fulfill the requirement of IoT services.

Typically, a WSN consists of a single or multiple clusters. A cluster consists of numerous child nodes and a sink. The primary role of child nodes is to collect information about surrounding environment (e.g., temperature, humidity, light intensity, and vibration) and send the gathered data to the sink. Such data transmissions can be performed in a time-driven or event-triggered manner. The sink acts as a data aggregator and sends the collected data to the gateway. All child nodes are low-cost, low-power devices equipped with limited sensing, communication, and processing capabilities. These resource-constrained sensor devices are typically powered by batteries.

For achieving energy-efficiency in WSNs, medium access control (MAC) mechanisms play an essential role since power utilization activities are mainly coordinated by MAC protocols [10]. From the last decade, it has become a norm to adopt duty cycling (DC) MAC to reduce energy wastage or improve energy-efficiency of WSNs by letting sensor nodes sleep in a periodic or an aperiodic manner. More specifically, a duty cycle consists of an active and a sleep period. Nodes do not perform any communication during the sleep period. Many DC MAC protocols have been proposed in the literature to improve energy-efficiency in WSNs. An overview of DC MAC is available in [11].

DC MAC protocols are mainly categorized as synchronous and asynchronous MAC, based on the sleep schedule of the sensor nodes. In synchronous DC MAC, all nodes in the network follow the same sleep schedule, wake up at the same time, and data communication occurs when both sending and receiving nodes are active. On the other hand, nodes in asynchronous DC MAC do not need to maintain

[1] Throughout the dissertation WSN/WSN node and IoT/IoT device are interchangeably used.

synchronization, i.e., nodes in the network do not follow the same sleep schedule. However, the intended receiver has to be in the active state when data transmission happens.

Regardless of the type of operation mode, i.e., synchronous or asynchronous, DC MAC shows significant performance enhancement in terms of energy-efficiency. However, such improvements are achieved at a cost of longer latency and lower channel utilization. In addition, overhearing and idle listening are two major problems of DC MAC protocols. *Idle listening occurs when a node listens to the channel but there is no ongoing communication and overhearing happens when a node overhears the packets of the other nodes which are not intended to it*. Overhearing and idle listening occur during the active period of a DC MAC protocol when nodes wait to exchange control and/or data messages.

1.3 From DC MAC to Wake-up Radio

As mentioned above, idle listening and overhearing are two significant contributors to the total energy consumption of DC MAC operated nodes. To diminish overhearing and idle listening, a paradigm shift from DC MACs to wake-up radio (WuR) has been envisaged recently [12]. In a WuR, an ultra-low power, secondary radio called wake-up receiver (WuRx) is attached to the main radio (MR) to monitor channel status actively, i.e., to detect incoming wake-up calls (WuCs). Whereas the MR of a WuR enabled node sleeps most of the time and wakes up when it is needed to transmit or receive a WuC or/and the data frames. In this way, idle listening of MRs is diminished by enabling a WuR. It is worth mentioning that the energy consumption of WuRx is at the μW or nW level whereas the energy consumption of MR is at the mW level [12].

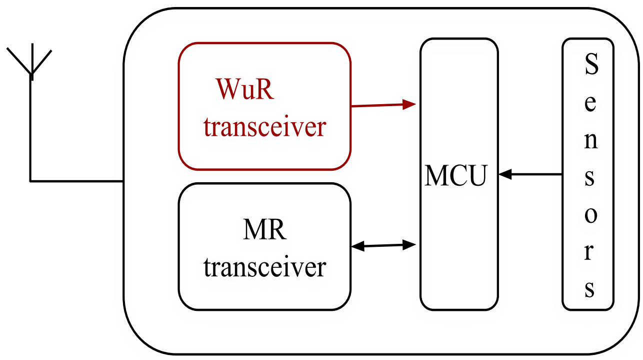


Figure 1.2: The composition of a WuR enabled sensor node/IoT device.

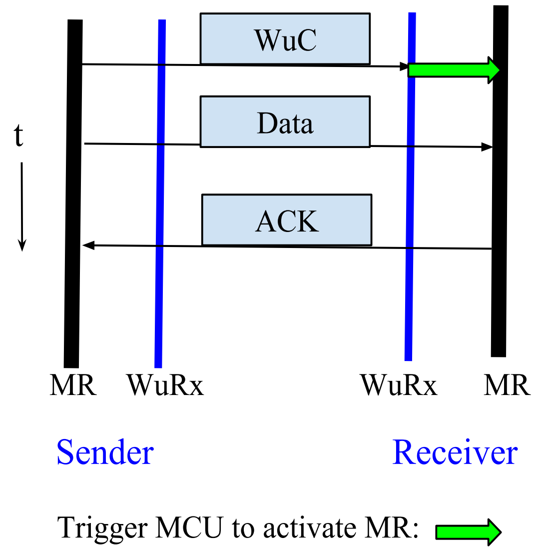


Figure 1.3: The operating principle of WuR.

A WuRx can decode and validate an address embedded in a WuC message and wake up the targeted node. Such a self-address decoding capability of WuR mitigates the overhearing problem. In addition, since the WuRx is always active and switches on its MR merely when necessary, it can operate purely in an on-demand manner. The time needed to activate an MR is at the ms level or less and this feature solves another problem, i.e., long latency which is caused by DC MAC.

The node architecture and working principle of WuR are shown in Figures 1.2 and 1.3 respectively. Based on the WuR design principle, a WuC is transmitted from the MR/wake-up transmitter (WuTx) of the sending node. A WuC frame contains a preamble and the address of the intended node. Upon receiving a WuC, the WuRx demodulates, decodes, and validates the address. If the address of the received WuC matches with its own, it then interrupts the micro-controller unit (MCU) of the node to switch on the MR from the sleep mode to perform data communication. Based on the MAC protocol embraced for data communication, the MR may follow a backoff (BO) procedure or directly send data without doing BO. A data communication finishes with a positive acknowledgment from the receiver. After that, the sender and receiver go to the sleep mode but keep their WuRxs always active to listening the channel.

Considering that WuCs are transmitted through wireless medium, the unintended nodes also receive the same WuCs. Thus, the unintended neighboring nodes will also decode and validate the address of a WuC. When a node identifies that the WuC is not intended to it, it does not generate any trigger to its MCU and remains active with its WuRx listening to the channel. Depending on the WuR design principle, WuCs and data transmission can be sent through different channels or over a single channel. When both data and WuC are sent via a single channel, different modulation schemes are used to distinguish them. WuCs are generally modulated with the simplest form of amplitude shift keying, i.e., on-off keying (OOK). The main reason of adopting OOK for WuC transmission is that it allows a sender to remain silent while transmitting ‘0’s, thus conserving energy. Moreover, it requires low implementation cost, i.e., low-processing power. Depending on the design of a main radio, different modulation schemes can be adopted. For example, binary phase shift keying (BPSK) or quadrature phase shift keying (QPSK) modulation is used to modulate data and ACK packets in IEEE 802.15.4. When a data or ACK frame is transmitted, WuRxs are not able to decode them. The WuRxs treat the data or ACK packet as unintended signal and stay silent during the data communication cycle.

WuR is suitable for both synchronous and asynchronous networks. A synchronous WuR network refers to a network where state transition occurs based on the mini-slots/symbol durations and the occurrence of each state transition (e.g., from the sleep state to active state) is aligned to the mini-slot boundaries. However, not all operations of WuR occur by mini-slots in an asynchronous network. For example, a WuR can wake up (sleep to active state transition) at any instant of time whenever it generates a packet. In other words, a state transition in an asynchronous WuR network does not always occur at the mini-slot boundary.

1.3.1 Categories of WuR Operations

The operation of WuR-enabled IoT networks can be divided into two categories, i.e., transmitter-initiated (TI) and receiver-initiated (RI). In an RI-WuR, a receiver initiates communication and sends a periodic or aperiodic WuC to gather information from WuR nodes. RI-WuR is suitable for mission-driven *data collection* scenarios. In a TI-WuR, a transmitter sends WuC to a receiver node, i.e., the sender initiates communication by transmitting a WuC to wake up the intended receiver. TI-WuR is beneficial for event-triggered *data reporting* scenarios. The research focus of this dissertation is on TI based WuR enabled networks.

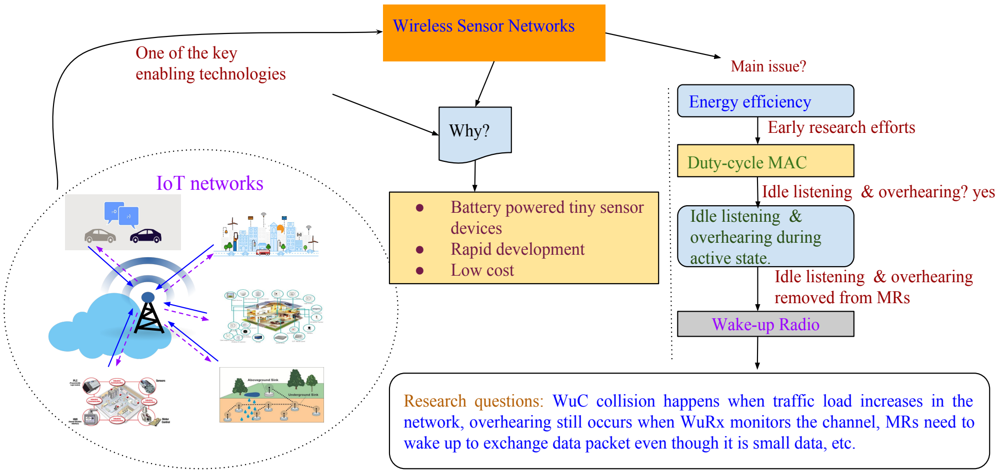


Figure 1.4: IoT and WuR: From topic to research questions.

1.4 Motivation and Research Questions

MAC layer protocols specify how sensor nodes access the radio medium. Therefore, a prime task for MAC protocol design is how to utilize the shared channel in a fair and efficient manner while considering improving energy-efficiency and reducing latency.

Various WuR design and MAC protocols have been proposed to improve the lifetime of WuRs. However, very little work has been done to ensure reliable WuC transmission when traffic load is higher, especially in TI-WuR networks. A bottleneck of most of the existing TI-WuR MAC protocols is collisions among concurrent WuC transmissions.

Although overhearing of MRs is completely eliminated, overhearing still occurs in the WuRx mode when the unintended WuRxs decode and validate the address of a WuC. No existing work focused on reducing overhearing of WuRx and evaluated the performance of WuR-enabled IoT networks. Thus it has become imperative to design new protocols to improve the performance of WuR networks.

Furthermore, energy-efficient relay selection is essential when a WuR node sends collected information to the sink over multiple-hops. Existing relay selection schemes (e.g., [13]) do not meet the low computational capability of WuRxs even though it is one of the primary requirements for WuR protocol design.

A logical flow of the research motivation and challenges is illustrated Fig. 1.4. Motivated by the observation that WuR techniques present a convincing direction for facilitating energy-efficient IoT applications and that more research efforts are needed within this topic, this Ph.D. dissertation takes an effort to address a few research challenges and advance the state-of-the-art techniques in WuR networks.

More precisely, this Ph.D. thesis addresses the following research questions.

- Question 1: In a synchronous TI-WuR network, reliable WuC transmission is expected to be maintained while embracing ultra-low power consumption. In this case, how to design a TI-WuR MAC protocol that reduces WuC collision probability under heavy traffic load conditions?
- Question 2: In an asynchronous network, traffic load varies from time to time and TI-WuR MAC protocols should support such traffic variation. Then, how to develop TI-WuR MAC protocols which support traffic variation while maintaining low latency and higher reliability?
- Question 3: Overhearing is still not eliminated from WuRxs. Moreover, WuC transmission prior to data transmission increases latency. How to further reduce overhearing energy consumption and delay when the to-be-transmitted data is very small?
- Question 4: In a multi-hop WuR network, route establishment between sender and receiver before actual data transmission consumes a significant amount of energy. How to develop a scheme which reduces energy consumptions of data communication for a WuR enabled multi-hop network?

1.5 Research Objectives

The main objective of this dissertation is to design protocols which improve the performance of WuR-enabled IoT networks. Although the performance can be improved through various techniques performed across all seven layers of the open systems interconnection (OSI) model, the scope of this Ph.D. dissertation is mainly lean to the MAC layer. More specifically, the main tasks of this dissertation are to propose protocols and schemes for TI-WuR networks by considering the research challenges mentioned above and develop analytical and simulation framework to evaluate the performance. More specifically,

- Aim 1: To develop a TI-MAC protocol for synchronous WuR networks that ensures reliable transmissions under heavy traffic load conditions.
- Aim 2: To design TI-MAC protocols for asynchronous WuR networks which consider random packet arrivals and support traffic variations.
- Aim 3: To develop efficient transmission schemes which further reduce overhearing and latency for small data transmission in WuR networks.

- Aim 4: To develop an energy-efficient relay selection scheme which decreases latency and energy consumption in multi-hop WuR networks.
- Aim 5: To develop analytical frameworks to evaluate the performance of the developed protocols and schemes. The accuracy of the developed analytical models should also be validated.

1.6 Research Contributions

In a nutshell, the contributions of this thesis work are twofold, i.e., protocol design and performance evaluation via analytical frameworks and simulations. Briefly, the major contributions are outlined as follows.

- Protocol design: One TI-MAC protocol for synchronous WuR networks; three TI-MAC protocols for asynchronous WuR networks; one lightweight relay selection scheme for multi-hop WuR networks; and two data transmission mechanisms to enable early sleeping and early data transmission are proposed.
- Modeling and validation: Four analytical frameworks to model the proposed schemes through discrete time Markov chain (DTMC) modeling and queuing theory are developed. The accuracy of the analytical frameworks is validated through custom made discrete-event simulations in MATLAB.

In addition, the early sleeping and early data transmission schemes are validated via a small-scale testbed that is implemented at the University of Agder (UiA).

The results of this study have been published or under review in three journals and four conferences. Among them, four papers which are published or under review at IEEE Communication Letters, IEEE Transactions on Industrial Informatics, Computer Networks (under review), and IEEE Global Communications Conference (GLOBECOM'18) are included in Part B of the thesis.

Table 1.1: An Overview of the Thesis Organization.

Part I					Part II			
Question	Ch. 2	Ch. 3	Ch. 4	Ch. 5	Paper A	Paper B	Paper C	Paper D
Q1	√				√			
Q2		√				√		
Q3			√				√	
Q4				√				√

† The check mark (√) specifies the appearance of particular question and where the solution has been presented.

1.7 Thesis Outline

The dissertation is organized into two parts. Part I contains an overview of the work carried out throughout this Ph.D. study and Part II includes a collection of four published or submitted papers, which are mentioned in the list of publications. In addition to the introduction chapter presented above, the following chapters are included.

- Chapter II presents several MAC mechanisms for synchronous networks and introduces one novel scheme which is originally proposed in Paper A.
- Chapter III discusses various MAC schemes for asynchronous networks and presents three MAC protocols which can handle variable traffic load in asynchronous WuR networks.
- Chapter IV highlights the overhearing and longer latency problems of WuR networks and introduces two data transmission schemes known as early sleeping (ES) and early data transmission (EDT).
- Chapter V summarizes the different categories of relay selection schemes for multi-hop WuR networks and presents a lightweight relay selection scheme.
- Chapter VI concludes the dissertation and highlights a few potential future research directions relevant to the topic addressed in this dissertation.

Chapter 2

A WuR MAC Protocol for Synchronous Networks

In this chapter, we first discuss the operating principle of several MAC protocols for synchronous networks and then propose a backoff mechanism with a fixed contention window (CW) size prior to WuC transmissions for the purpose of WuC collision avoidance. Such a mechanism improves network throughput, especially under saturated traffic condition. Lastly, we provide a brief introduction of the developed analytical framework.

2.1 A Few Popular Synchronous MAC Protocols

In a synchronous network, nodes maintain synchronization by adopting an identical clock with a fixed phase alteration. Examples of such networks include Wi-Fi and synchronous DC WSNs. In this section, a few MAC protocols for synchronous networks are summarized.

2.1.1 Contention-based MAC Protocols

In a Wi-Fi network, carrier sense multiple access with collision avoidance (CSMA/CA) is adopted where a node performs clear channel assessment (CCA) to check the channel status whenever it has a packet to transmit. If the channel is idle for a distributed coordination function (DCF) interframe space (DIFS) time duration, the station starts a BO procedure. When multiple nodes find the channel being idle, these nodes start BO at the same instant. In this way, the network maintains synchronization in the DCF mode. If *only one node* selects the smallest BO slot, it wins the channel access competition and transmits its data packet successfully if the channel is error-free. A data transmission cycle ends when an ACK from the receiver is received. If multiple nodes select the same lowest BO slot, then a collision occurs in the network. For the collided packet, no ACK is generated. After the ACK timeout period, the involved nodes follow the same procedure for BO, however, based on a larger CW size according to the exponential BO procedure.

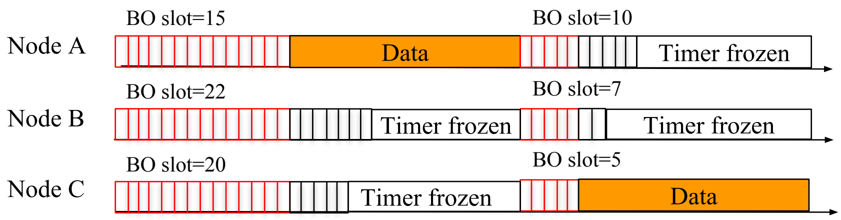


Figure 2.1: An overview of the CSMA/CA BO procedure.

The operation of CSMA/CA is shown in Fig. 2.1 where nodes A, B, and C compete for channel access. At the first round of channel access competition, node A wins the competition and finishes its data transmission. The remaining nodes freeze their BO and start counting again when the channel becomes idle. Next, node C wins and transmits its data. The same procedure continues until all nodes finish their data transmissions. Such a scheme improves channel utilization by reducing collisions among simultaneous transmissions.

2.1.2 Synchronous DC-MAC protocols

Sensor-MAC (S-MAC) [14] is one of the popular synchronous protocols that improves energy-efficiency of WSNs by reducing idle listening. To eliminate idle listening, it allows a node to follow a *periodic active and sleep schedule*. S-MAC uses in-band control messages to reduce contention latency. The underlying working principle of S-MAC is shown in Fig. 2.2.

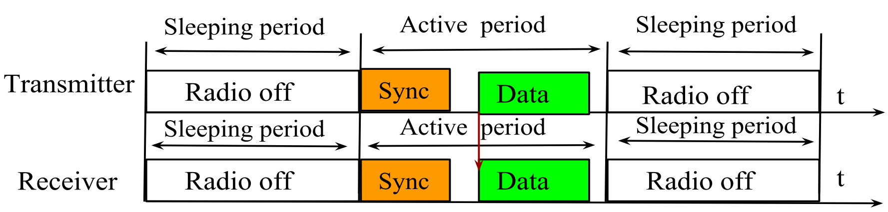


Figure 2.2: The working principle of S-MAC.

All nodes in S-MAC keep their transmitter switched off during the sleep period and activate at the same time instant to check whether any neighboring node intends to perform communication or not. If any node has data to transmit, it first follows a contention procedure, i.e., it selects a random slot from the CW and broadcasts a request to send (RTS) message as soon as its backoff timer expires. The intended node replies with clear to send (CTS) message. Other nodes synchronize their sleep schedule by decoding the value specified in the network allocation vector (NAV) of the RTS or CTS packet. When multiple nodes contend for sending RTS, the node which selects the lowest backoff wins the competition and transmits its RTS message. If no node has data to exchange in the active period, one of the neighboring nodes distributes a sleep schedule to synchronize all the nodes about the sleep schedule. In this way, S-MAC reduces collision and overhearing.

DW-MAC [15] is another representative synchronous MAC protocol. It reduces latency by adopting a novel schedule algorithm which allows a neighboring node to wake up and perform data communication during the sleep period [15]. Such an on-demand wake-up and data transmission scheme improves channel utilization and reduces latency.

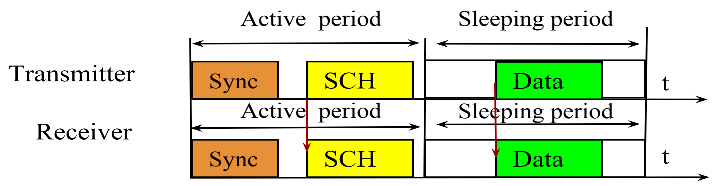


Figure 2.3: The working principle of DW-MAC.

In DW-MAC, each cycle is divided into three parts, i.e., a sync, data, and sleep period. The clock synchronization of all the neighboring nodes is done mainly in the sync period. It is worth mentioning that no RTS/CTS control is transferred in the sync period as in S-MAC. A CSMA/CA-based contention MAC mechanism is utilized to exchange a scheduling frame (SCH) among nodes in the data period to perform data communication in the sleep period of the duty cycle. The basic design of DW-MAC is illustrated in Fig. 2.3 where a sender and a receiver are synchronized in the sync period. In the data period, the transmitter sends SCH to the respective receiver. Then data communication is performed in the sleep period. The data communication cycle finishes with an ACK. The drawback of DW-MAC is that it requires to maintain synchronization in the network.

2.2 WuC Collisions in Synchronous TI-WuR Networks

Currently, an IEEE standard amendment, i.e., IEEE 802.11ba [16] is under standardization by the IEEE 802.11ba Task Group for ultra-low power WuR-enabled Wi-Fi networks. The target of this standard is to further reduce energy consumption and improve the battery lifetime of Wi-Fi stations based on the principle of WuR. Moreover, IEEE 802.11ah [17] is specifically designed to enlarge the coverage area to connect a massive amount of IoT devices. As a common feature, both IEEE 802.11ba and IEEE 802.11ah Task Groups investigate the potential of connecting IoT devices (non-3GGP connections) with existing Wi-Fi networks.

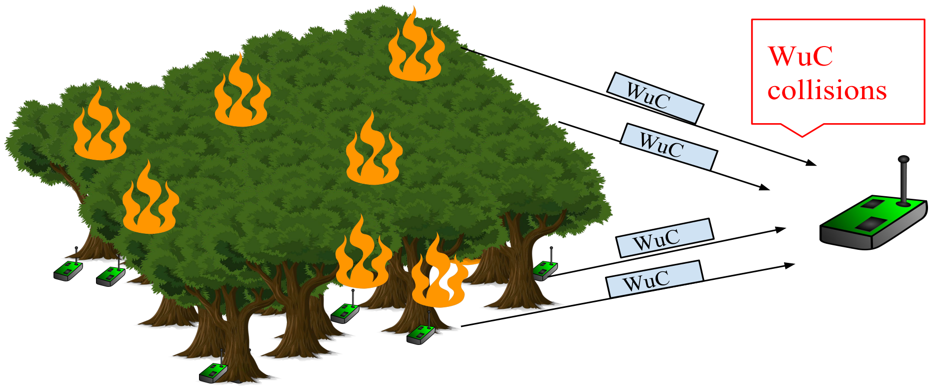


Figure 2.4: WuC collisions in a WuR network.

As briefly mentioned above, WuC collisions occur in a synchronous network (e.g., Wi-Fi) when multiple transmissions overlap with each other. Such a phenomenon is common under heavy traffic load conditions. When multiple nodes have packets to transmit, they wake up at the same time instant and send WuCs. These concurrent transmissions lead to WuC collision and reduce transmission reliability for data exchange in a network. On the other hand, reliable transmission is one of the most important performance metrics in many IoT applications. For instance, consider that a WuR-enabled IoT is deployed to monitor a specific area of a forest, as shown in Fig. 2.4. At the same time, multiple nodes detect a fire and start to send their sensed information to the sink. Due to WuC collisions in the network, the sink is unable to receive or decode the information. Thus the consequence of such WuC collisions has an adverse effect. It is also evident from our study that WuC alone may not perform better under heavy traffic conditions [18]. This observation triggered our motivation to propose a TI-WuR MAC protocol which can reduce WuC collisions in such a network.

2.3 BO-enabled WuR (BoWuR)

In this section, we outline the working principle of a proposed protocol, i.e., BoWuR which has been reproduced as Paper A in Part II of this thesis. The primary purpose of BoWuR is to reduce collisions among WuCs when a WuR-enabled network is operated in the TI mode. To do so, we adopt a similar procedure as used in CSMA/CA to WuCs. When a WuR node generates a packet, it senses the channel to check whether it is occupied or not by performing CCA. If the node finds channel being idle for a CCA period, it chooses a random BO slot from a fixed CW. Otherwise, it performs CCA again. If it is the only node which selects the smallest BO slot, it wins channel access competition and sends a WuC. The intended receiver validates the received WuC and triggers its MCU to switch on MR. When the MRs of both sending and receiving nodes are active, a data communication occurs immediately. The data communication cycle finishes with an acknowledgment from the target receiver. If no ACK is received within the timeout period, the node follows the steps explained above to retransmit the WuC packet. Note that no ACK is needed when a WuC is received in order to reduce latency. A successful data transmission cycle is

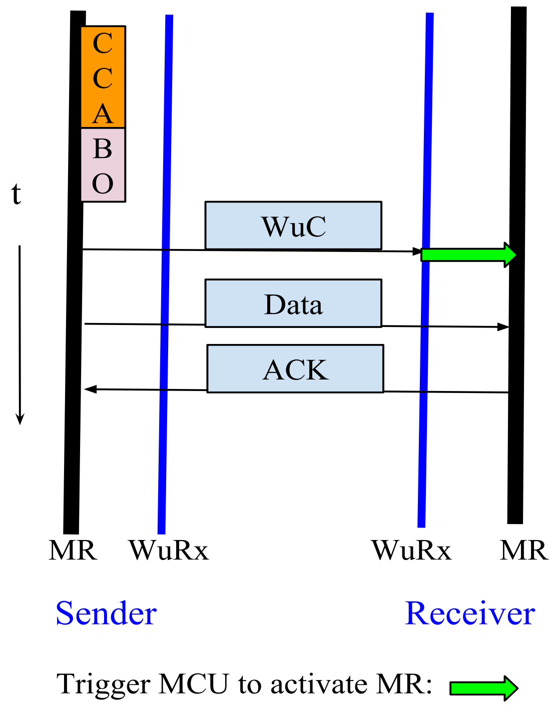


Figure 2.5: The operating principle of BoWuR.

illustrated in Fig. 2.5. The neighboring nodes which do not win the channel access competition freeze their BO counter for the duration specified in the NAV field of the WuC frame. The length of a WuC MAC frame is considered to be 48 bits. In a

WuC frame, 1 byte, i.e., bits 24-31, is allocated for the NAV field. The time needed for both WuC and data transmissions is specified in the first 7 bits of the NAV field, and the last bit is reserved for error check which is known as a parity bit. Bits 32-47 are reserved for the WuR address.

2.4 Performance Evaluation of BoWuR

To evaluate the performance of BoWuR, a DTMC is used as a mathematical tool for our performance analysis. A DTMC is a stochastic process with a finite state space which has discrete time and memory-less property. The developed model is inspired by [19] and [20] which are two popular analytical models for performance evaluation of a legacy CSMA/CA protocol under saturated and unsaturated traffic conditions respectively. However, the BO procedure applies prior to WuC transmission instead of data transmission.

We define four metrics, i.e, collision probability, system throughput, latency, and energy-efficiency for performance evaluation of BoWuR. Based on the developed analytical model, we first calculate the collision probability as network traffic load varies. The other performance metrics are derived from the obtained collision probability.

2.5 Chapter Summary

In this chapter, we have discussed several existing MAC protocols for synchronous networks. We explain the impact of WuC collisions in a WuR enabled IoT network. Furthermore, the main idea of the proposed BoWuR MAC protocol is outlined. Such a scheme provides higher network throughput, lower collision probability, shorter delay, and higher energy-efficiency. For more details on BoWuR, refer to Paper A in Part II of this dissertation.

Chapter 3

WuR MAC Protocols for Asynchronous Networks

In this chapter, we first present an overview of asynchronous MAC protocols. We also discuss why existing WuR protocols are not efficient when traffic load increases in a network. Furthermore, we propose three TI-WuR MAC protocols which can support various traffic loads in WuR-enabled networks.

3.1 A Few Popular Asynchronous MAC Protocols

In an asynchronous network, a node does not need to know the active or sleep status of other neighboring nodes. Moreover, a node can wake up at any instant of time, i.e., state transition does not occur at the mini-slot boundary. A typical example of such a network is asynchronous WSNs. In this section, a few popular asynchronous MAC protocols are summarized.

3.1.1 Unslotted CSMA-CA

The WSN/IoT standard amendment, i.e., IEEE 802.15.4/802.15.4e, adopts contention based MAC, i.e., unslotted CSMA-CA for an asynchronous network. It is worth noting that unslotted CSMA-CA is different from CSMA/CA (Wi-Fi standard IEEE 802.11). The difference is the order of performing BO and CCA. *In an unslotted CSMA-CA adopted network, a node follows BO before performing CCA whereas a node performs CCA first and then BO in a CSMA/CA based network.* Moreover, the BO freezing mechanism of CSMA/CA cannot be directly adopted in WSNs since no NAV field is defined in the MAC frame format of IEEE 802.15.4/IEEE 802.15.4e. Such BO freezing, i.e., transmission deferral for other nodes' transmis-

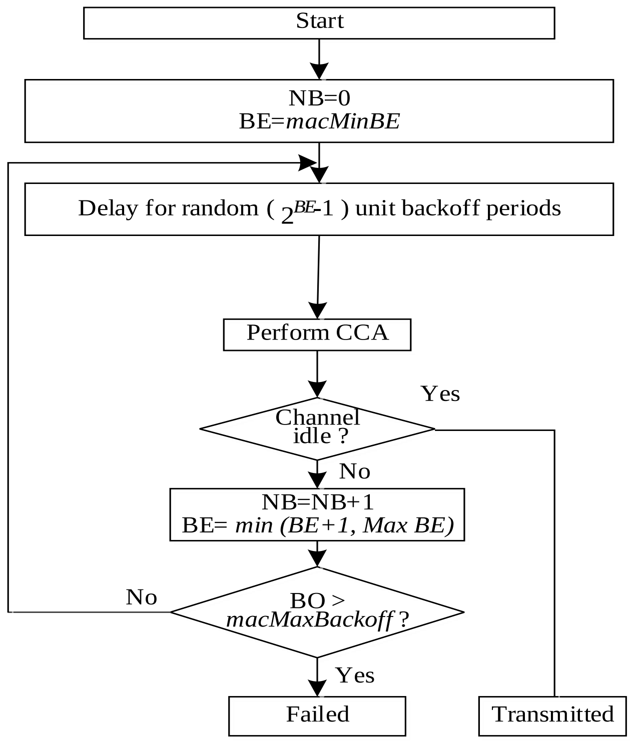


Figure 3.1: The working principle of unslotted CSMA-CA [1].

sions often increases queuing delay, leading to a longer packet delay especially for low-power devices which transmit at a lower data rate (e.g., 20 kbps, 250 kbps).

Considering the resource constraints (e.g., battery-powered, small buffer size, and limited computational capability) of sensor nodes, employing CSMA-CA instead of CSMA/CA is a more delay- and energy-friendly option. This is because BO consumes much lower energy than CCA. For instance, MR consumes 20.28 mA and 5.16 mA current during CCA and BO respectively [21]. Furthermore, neither BO freezing of other nodes' transmissions nor network synchronization is needed in CSMA-CA.

The flowchart of unslotted CSMA-CA is shown in Fig. 3.1. In unslotted CSMA-CA, a node which has a packet to transmit waits for a random number of BO slots selected uniformly in the range from 0 to $2^{BE}-1$, i.e., window ($W_{min} = 2^{BE_{min}}$) where BE indicates the BO exponent. BE_{min} is set to the minimum value of the BO exponent (*macMinBE*). When the BO counter reaches 0, the node senses the channel by performing CCA. If the channel is idle for the duration of CCA, the node transmits data. Otherwise, the node follows the BO procedure again, however, BE will be increased by one. At the i^{th} stage, $W_j = min(2^i W_{min}, W_{max})$ where $W_{max} = 2^{BE_{max}}$. BE_{max} denotes the maximum

BO exponent *macMaxBE*. A node will continue its attempt to transmit until the maximum BO limit ($macMaxCSMABackoffs$) is reached. If the data transmission is not successful within the maximum BO limit, it declares a transmission failure. After a transmission failure, the node retries to send the data packet until the maximum number of retries ($macMaxFrameRetries$) exceeds.

3.1.2 X-MAC

X-MAC [22] is one of the popular asynchronous DC-MAC protocols in which nodes in the network follow an individual sleep and wake-up schedule. This means that nodes in the network do not need to exchange control messages to maintain synchronization. A strobed preamble mechanism is adopted in X-MAC, allowing a node to send short preambles to the target receiver when it has any data to send. The interval between two short preambles allows an intended receiver to send an ACK to inform the transmitter that it is ready for data reception. As soon as the sender receives an ACK from the target receiver, the communication cycle is finished. When no incoming preamble is detected by a node during its active state, it goes to sleep after a specific period.

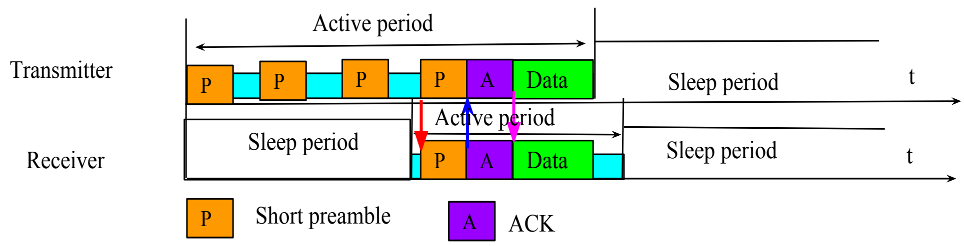


Figure 3.2: The working principle of X-MAC.

The operation of X-MAC is shown in Fig. 3.2 where node A sends consecutively a sequence of short preambles to node B after following its sleep schedule. When B wakes up, it sends an early ACK message to start a data communication cycle. After data communication, A goes to sleep if no more packet is queued in its buffer. B remains active for a sample period before going to sleep by checking whether any queued packet from A is transmitted or not. However, X-MAC still suffers from overhearing and long latency. Since the sending node needs to wait until the target node wakes up for receiving a short preamble, this procedure increases data communication latency.

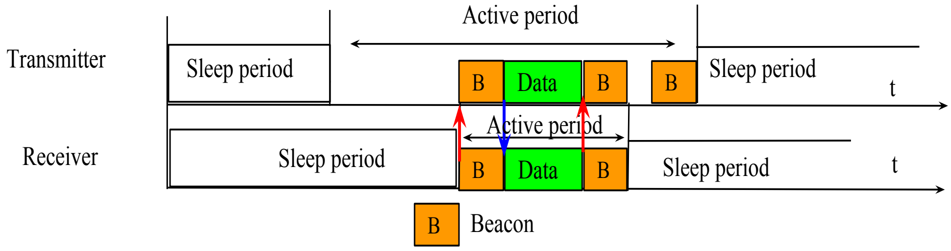


Figure 3.3: The working principle of RI-MAC.

3.1.3 RI-MAC

RI-MAC [23] is another asynchronous MAC protocol in which data transmissions occur in a receiver-initiated manner, meaning that it is the intended receiver which always triggers a packet transmission. Fig. 3.3 shows the basic operation of RI-MAC. In RI-MAC, each node wakes up according to its own wake-up schedule and broadcasts beacon messages, advertising that it is prepared to receive packets.

In contrast, a sending node which has a packet to transmit wakes up according to its schedule and silently scans for beacon messages from the target receiver. Upon detecting a beacon message from the target receiver, the sender transmits the data packet immediately. Then the intended receiver responds with another beacon message. The beacon carries twofold information, i.e., it acts as an ACK for the previous packet transmission and the announcement to receive packets intended for it. If the receiver does not receive any data frame after sending the beacon, it goes to sleep. The sender waits a bit more time and then it goes to sleep. In this way, RI-MAC reduces channel occupancy and improves network throughput. However, like other DC protocols, RI-MAC also suffers from its latency performance. The reason is that a sending node needs to wait until the target receiver wakes up and it needs time to advertise its readiness for packet reception.

3.2 Existing Asynchronous TI-MAC WuR Protocols

Sparse topology and energy management (STEM) [24] is one of the representative TI-WuR MAC protocols for asynchronous networks. It uses separate channels for sending WuC and data messages. Two variations of STEM exist in the literature, i.e., STEM-T and STEM-B. In STEM-T, a node sends a broadcast WuC to wake up its neighboring nodes. In STEM-B, a node transmits a unicast WuC to wake up an intended node. STEM adopted a DC mode for WuRxs while keeping MRs in the sleep mode unless it is required to switch on. Similar to STEM, pipelined tone wake-up (PTW) presented in [25] uses different channels for data and WuC transmissions. In PTW, the WuRxs operate in a DC manner. A sender broadcasts a WuC over the wake-up channel and sends the notification frame on the data channel to explicitly specify the intended node. Such a scheme is suitable for opportunistic networking. An energy harvesting TI-WuR MAC protocol, referred to as WUR-transmitter initiated cycled receiver (TICER), is proposed in [26] in which embedded battery is charged by an ambient environment. WUR-TICER utilizes the same channel for data and WuC exchange. Whenever a sender generates a packet, it broadcasts a WuC to inform other receivers that it has a packet to transmit.

However, a major problem of existing TI-MAC WuR protocols is that WuCs collisions may happen frequently when two or more nodes wake up and send WuCs at the same time.

3.3 TI-WuR MAC protocols

In this section, we outline the three TI-WuR MAC protocols which have been proposed in Paper B of this thesis.

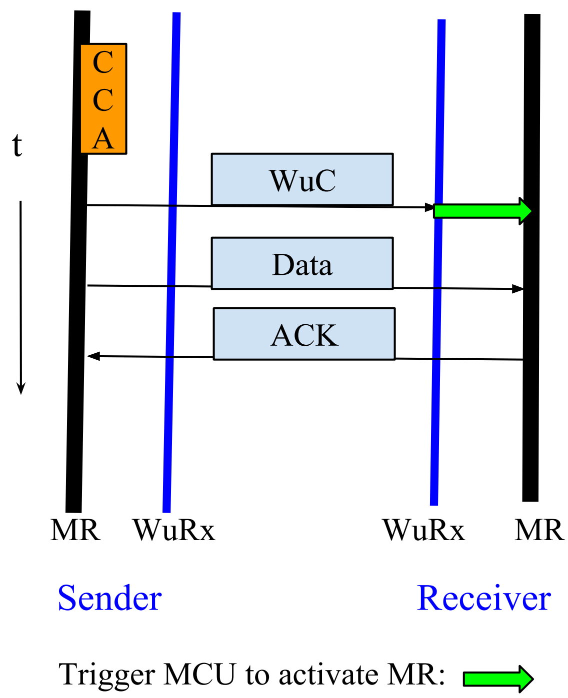


Figure 3.4: The working procedure of the CCA-WuR protocol.

3.3.1 CCA Enabled WuR (CCA-WuR)

As mentioned earlier, WuRs are not efficient when traffic load gets higher in a network. To ensure reliable transmission in the network, we propose the CCA-WuR protocol. The primary purpose of CCA-WuR is to reduce WuC collision. In a CCA-WuR adopted network, a node performs channel sensing by CCA to check the channel status, i.e., whether the channel is occupied by other transmission or not. A node sends a WuC if the channel is found to be idle for a CCA period. Otherwise, it senses the channel again and this procedure continues until the maximum retry limit is reached. When the WuC transmission is not successful within the retry limit, a node will discard the packet or inform the higher layer to take actions. Such a CCA enabled scheme fits well when traffic is light. The working principle of CCA-WuR is shown in Fig. 3.4.

3.3.2 CSMA-CA Enabled WuR (CSMA-WuR)

When traffic load becomes heavier, performing channel sensing using CCA prior to WuC transmission only is not enough to avoid concurrent transmissions.

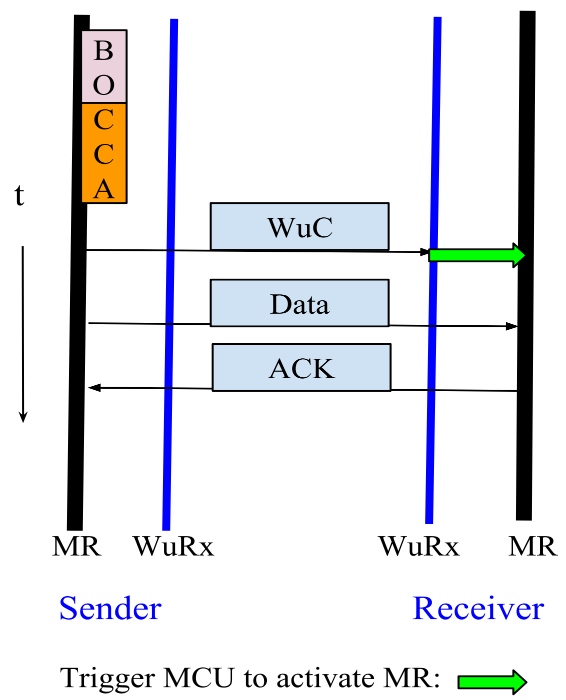


Figure 3.5: The working procedure of the CSMA-WuR protocol.

To support high traffic load in WuR-based IoT networks and overcome the constraint of CCA-WuR, we further propose the CSMA-WuR protocol. Although the principle of CSMA-WuR is similar to CSMA-CA used in IEEE 802.15.4/IEEE 802.15.4e, it is tailored to WuC transmissions. Moreover, CSMA-WuR adopts a fixed CW size whereas CSMA-CA embraces an exponential CW size.

In a CSMA-WuR adopted network, a node which generates a packet performs BO first *without sensing the channel*. When the BO counter reaches 0, the node performs CCA to check the channel status. If the channel is unoccupied, the node transmits a WuC. Otherwise, this process (i.e., BO and CCA) is repeated. A node discards the WuC packet if the transmission is not successful within the retry limit and informs the higher layer about the unsuccessful status of the WuC transmission.

As mentioned earlier, the energy consumption of BO much is less than CCA, i.e., $P(BO) < P(CCA)$. It is worth mentioning that under heavy traffic conditions multiple CCAs may be needed for CCA-WuR while CSMA-WuR requires merely one or few CCAs. The rest of the operation of CSMA-CA is similar to CCA-WuR. The operating principle of CSMA-WuR is presented in Fig. 3.5.

3.3.3 Adaptive WuR (ADP-WuR)

CCA-WuR and CSMA-WuR are suitable for light and high traffic respectively. However, traffic load in a network is not always static. Thus, employing CCA-WuR or CSMA-WuR *alone* will not be beneficial to all traffic load episodes. To handle traffic variation in a network, we further propose a learning-based MAC protocol, referred to as ADP-WuR. ADP-WuR can ensure reliable transmission for various traffic load conditions.

In an ADP-WuR adopted network, a node adaptively changes its MAC procedure, i.e., it activates CCA-WuR or CSMA-WuR based on the observed traffic load condition. To know the traffic load condition, a node keeps track on its WuC transmission attempts and measures it with the predefined threshold value for MAC mechanism alteration.

Fig. 3.6 describes the operating principle of ADP-WuR. In the beginning, a node adapts CCA-WuR. At each attempt stage, the node compares the attempt counter with a predefined threshold (NB) value. When the attempt counter exceeds the threshold, it alters its MAC, i.e., it adapts CSMA-WuR. The threshold value of a node can be configured dynamically. In that case, each node has to keep the statistics of the previous transmissions (e.g., WuC collisions, latency, number of attempts needed for the previous successful packet). Such a dynamic threshold adaptation is resilient to the MAC alteration.

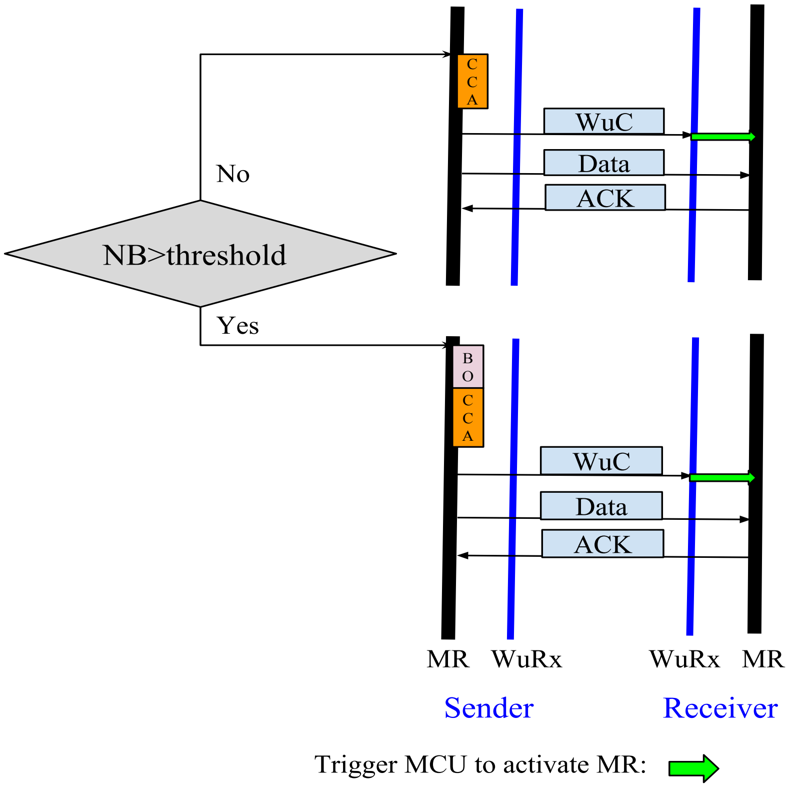


Figure 3.6: The working procedure of the ADP-WuR protocol.

3.4 A Brief of the Analytical Framework

To model the proposed protocols, we develop a novel analytical framework. There exists a framework in [27] which deals with a similar scenario. In [27], the authors modeled unslotted CSMA-CA based on a busy period of a regenerative cycle in an M/G/1 queue. The authors assume that the queue size is infinite. However, a finite queue size assumption is more realistic than the infinite queue assumption, specifically for WuRs since these devices are equipped with very limited storage capacity. Their model can be adopted only if both BO and CCA are present in the MAC mechanism. Since our purpose is to compare the performance of different MAC protocols (WuR, CCA-WuR, CSMA-WuR, and ADP-WuR), we cannot adopt the analytical model presented therein.

Furthermore, a state transition in most of the existing CSMA-CA frameworks including [28–36] is based on the approximation of mini-slots/symbol durations. That is, the occurrence of each state transition (e.g., from sleep state to active state and CCA to BO) is aligned to mini-slot boundaries where each state of a protocol operation consists of multiple mini-slots. However, due to the on-demand nature of WuR, a node can wake up (a sleep to active state transition) at any instant of time whenever it generates a packet. In other words, a state transition in WuR operation does not necessarily occur at the mini-slot boundary. This observation triggered our motivation to develop *a new framework which is based on finite queue size and does not dependent on the assumption that state transitions happen only at mini-slot boundaries*.

Different from M/G/1, our framework is based on an M/G/1/2 queue. Accordingly, we define a busy period which can be interpreted as the period that starts from a packet arrival in a vacant queue and finishes when the queue is vacant again. When the queue is not empty, the newly arrived packet will stay in the queue until it is either successfully transmitted or discarded from the queue in the same busy period. The interval between two consecutive busy periods is regarded as an idle period. During the idle period, no activity occurs but WuRxs continue to monitor the channel.

3.5 Chapter Summary

In summary, this chapter discusses several asynchronous MAC protocols. It highlights the WuC collision problem in a WuR network when traffic load is medium or high. Furthermore, it explains briefly how CCA-WuR, CSMA-WuR, and ADP-WuR reduce WuC collisions in an asynchronous network. CCA-WuR, CSMA-WuR, and ADP-WuR are proposed and investigated in-depth in Paper B.

Chapter 4

Further Reducing Overhearing and Latency in WuR Networks

This chapter provides an overview of the state-of-the-art WuR implementations. It highlights the impact of overhearing and latency in WuR networks. Moreover, we propose two schemes that can further reduce overhearing and latency for data transmission in WuR networks. Lastly, it briefly describes how these two schemes are validated through a WuR prototype.

4.1 State-of-the-art WuR Implementations

Since WuR represents a promising tendency for WSN evolution, a rapidly growing number of WuR implementations have been reported in the literature. A few notable implementations are outlined in this section. Two comprehensive surveys of the state-of-the-art WuRs can be found in [37] [38].

A sub-GHz WuR reported in [13] consumes merely 1.27 μW for channel monitoring. Its WuRx uses the OOK modulation scheme and consists of four major blocks, i.e., a matching network, an envelope detector, a comparator and a preamble detector. The output from the preamble detector triggers PIC12LF1552 MCU to validate address and wake up the MR if the address is matched. This sub-GHz WuR provides sensitivity and a data rate at -55 dBm and 100 kbps respectively. Its coverage reaches up to 45 m. This design was further improved by reducing the power consumption of WuRx in the listening mode to 0.152 μW with a coverage range of 50 m [39]. Another notable WuR design that integrates AS3933 for Wi-Fi device, i.e., IEEE 802.11, was proposed in [40]. The designed prototype achieves a sensitivity level of -52 dBm and has the capability of self-address decoding. It consumes 10.8 μW in the listening mode and 24 μW in the active mode. An off-the-shelf WuR transceiver which was developed based on ZL70103 for healthcare applications was presented [41]. The WuRx operates on out-of-band at 2.45 GHz with nW power consumption. This implementation is mainly suitable for body area networks. In most of the WuR implementations, OOK modulation appears as a popular modulation scheme for WuRxs since it offers better noise resistance capability and a simpler demodulation scheme. Nonetheless, frequency shift keying (FSK) based WuR design has been proposed in [42] where WuRx achieves a sensitivity level of -89 dBm with a bit rate of 45 kbps.

However, almost all the implementations highlighted above suffer from overhearing in the WuRx mode. Furthermore, these implementations require a WuC transmission prior to actual data transmission even it is designated to send small IoT data.

4.2 Effects of Overhearing and Latency

As mentioned earlier, overhearing occurs when a node overhears a packet which is not intended to it. In a WuR network, each WuRx always keeps listening to the channel. It decodes and validates all WuC packets from its neighbors in order to know whether the incoming WuC is targeted to it or not. It is worth mentioning that WuRxs consume much higher energy for decoding and validating a WuC packet than for actively monitoring channel. For instance, the WuR implementation in [2] operates on three energy consumption levels, i.e., deep sleep (0.39 μA), light sleep (1.9 μA), and active (4.1 mA) for channel monitoring, address validation, and data communication respectively. Such overhearing consumes a significant portion of energy, especially when the node density is high and traffic load is heavy in the network. That means, *overhearing energy wastage has an adverse impact on the battery life of WuR nodes*. Thus, it is of great interest to further reduce overhearing to prolong the lifetime of WuR nodes.

On the other hand, latency is another essential metric that has an impact in any form of communication as it represents the time taken by a packet to successfully reach the receiver. Many IoT/MTC applications such as remote monitoring, healthcare, and industrial automation require low latency communication.

In a WuR network, a WuC transmission is required prior to actual data transmission. *Such a WuC exchange prior to data transmission adds extra latency for packet transmission*.

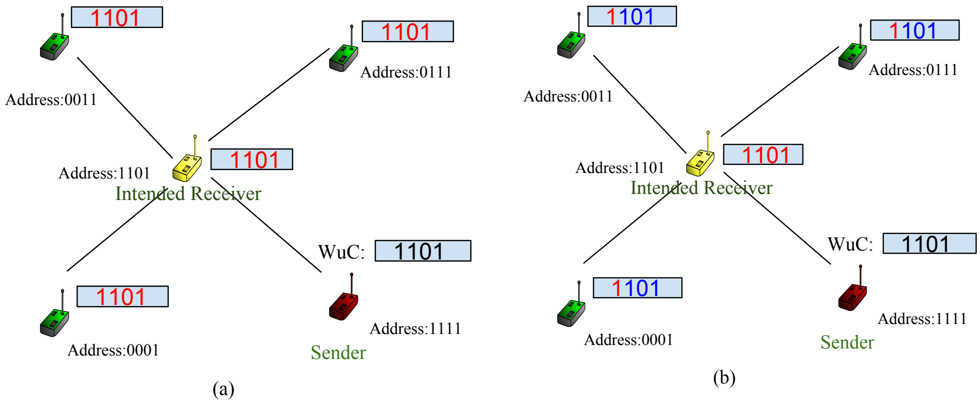


Figure 4.1: Full address decoding (a) versus early sleeping (b).

4.3 WuC Overhearing Avoidance

In a WuR network, a WuR node decodes a WuC packet, i.e., full address of a WuC to know whether the WuC packet is targeted to it or not [43]. If the WuC address matches with own address, then the WuRx triggers the MCU to wake its MR up. Otherwise, the MR keeps sleeping while the WuRx remains active to monitor the channel. In this section, we propose an early overhearing avoidance mechanism referred to as early sleeping. Furthermore, we explain how it is validated through a testbed developed at UiA [2].

4.3.1 Early Sleeping

The disadvantage of full address decoding is that unintended nodes need to decode and validate a complete address to identify an unintended WuC. Such full address decoding is not an energy-friendly option in WuR networks.

In early sleeping adopted WuR networks, a node decodes and validates a WuC address bit-by-bit to improve the performance of full address decoding. Unlike full address decoding, early sleeping allows a WuRx to decode and match the incoming WuC in a bit-by-bit manner. In other words, a WuRx decodes the first bit of a WuC address with its corresponding address bit. If the first bit matches, then it will decode and validate the next bit. For a targeted WuR node, this decoding procedure will continue until the complete address is matched with its own address. On the other hand, an unintended node in the network stops decoding and validating whenever a bit of the incoming WuC address mismatches with its own.

The benefit of early sleeping is that it minimizes overhearing energy consumption in WuR networks by letting unintended nodes sleep at an earlier phase before the full address validation is complete.

As an example, the difference between full address decoding and early sleeping is shown in Fig. 4.1. In both sub-figures, 4.1(a) and 4.1(b), the sender (address: 1111) sends a WuC to an intended receiver (address: 1101). Since the WuC is sent through a shared medium, the other three unintended nodes (addresses: 0011, 0111, and 0001) in the network receive the same WuC. With full address decoding, the unintended nodes also need to decode and validate the complete address sequence as shown Fig. 4.1(a). An unintended node with early sleeping enabled stops decoding the address whenever a bit of the received address mismatches with its corresponding address bit. Accordingly, it is clearly shown in Fig. 4.1 (b) that all the unintended nodes go to sleep after validating the first bit of the received address since it does not match with their first address bit.

Clearly, the number of sleeping nodes after the first bit validation in the early sleeping mechanism is increased in comparison with full address decoding. In this way, early sleeping reduces overhearing in the network.

4.4 Early Data Transmission

Massive IoT will have significant impact on network traffic [44]. Statistics have shown that 50% of IoT packets are small in size, i.e., less than 100 bytes [45]- [46]. We observe that some IoT packets are tiny, i.e., few bytes in size. Examples of such IoT packets are pressure, temperature, humidity, and light intensity.

Furthermore, in many IoT applications, sensor nodes infrequently send a small amount of information to the data center, e.g., one packet per day or every other hour [6]. A handshake simplification mechanism, referred to as early data transmission, is currently under standardization by 3GPP to support infrequent small data transmissions in IoT networks [47]. Early data transmission allows a node to transmit data during the random access procedure. Such a mechanism increases the battery lifetime of a node and minimizes latency of packet transmission.

To the best of our knowledge, no prior study has investigated early data transmission for WuR networks. In this section, we propose to enable early data transmission in WuR networks. Accordingly, we develop two variations of early data transmission , i.e., with and without ACK after early data transmission, and explain how these schemes are validated.

4.4.1 Early Data Transmission with ACK

The main idea of early data transmission is to send tiny data during the WuC transmission phase. In early data transmission with ACK, an ACK message is required from the receiver side to acknowledge the successfully delivery of a transmitted packet. To enable early data transmission with ACK, a sending WuR node first protects its tiny data using an error-correcting code, e.g., cyclic redundancy check (CRC). Then it encodes the destination WuRx address with the CRC appended data using the XOR operation. The output bits of the XOR operation are sent as a WuC from the MR of the sender.

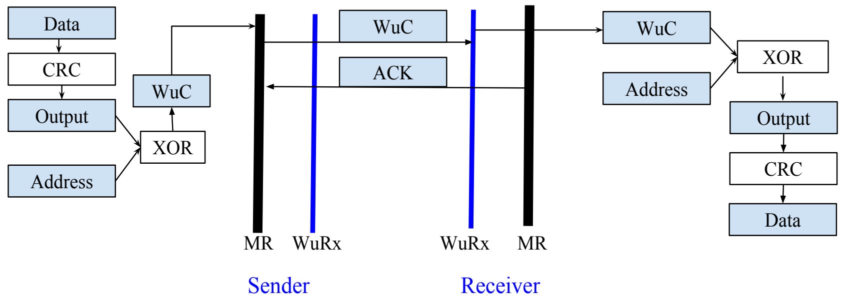


Figure 4.2: A successful transmission cycle of early data transmission with ACK.

A receiver follows a reverse operation to extract the information. That is, the receiver performs the XOR operation between its own address and the received WuC packet. The results of the XOR operation will be treated by the CRC polynomial as used at the sender. If the CRC value matches, then the WuRx interrupts the MCU to switch on its MR to send an ACK to inform the sender that the WuC transmission has been successful.

Early data transmission with ACK reduces the latency it takes to transmit small data and improves the energy efficiency of WuRxs. A successful transmission cycle of early data transmission with ACK is shown in Fig. 4.2.

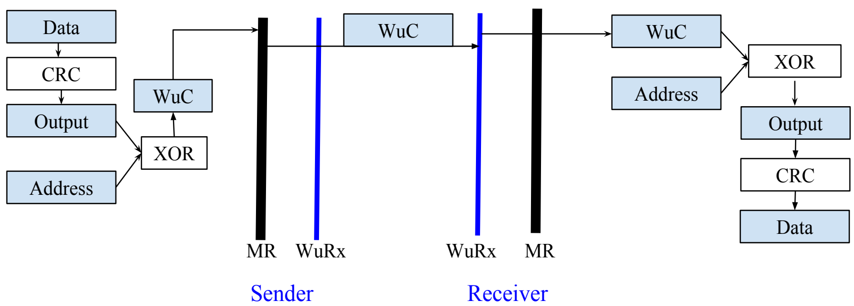


Figure 4.3: A successful transmission cycle of early data transmission without ACK.

4.4.2 Early Data Transmission without ACK

Early data transmission with ACK fits better for IoT applications where reliable data transmission is required along with low latency communication. However, achieving reliable data transmission via ACK might be costly in terms of energy consumption. From this observation, we propose another version of early data transmission referred to as early data transmission without ACK. In this scheme, no ACK is required from the MR of the receiving node to inform the sender about the successful delivery of a data transmission.

This scheme further reduces the latency and improves energy-efficiency since a data transmission cycle is finished without even waking up the MR. A successful transmission cycle of early data transmission without ACK is shown in Fig. 4.3.

4.4.3 Testbed Validation

Based on a testbed developed at UiA [2], we validate early sleeping and early data transmission. The current consumption of the developed WuRx is at the nanoampere level (390 nA). It adopts the OOK modulation and operates at a carrier frequency of 433 MHz. nRF52832 is used as the MR of a WuR node. The WuR node has three power consumption modes, i.e., deep sleep (for channel monitoring to detect WuCs), light sleep (for WuC address decoding), and active (for data transmission). The current consumption levels of these three modes are 0.39 μA, 1.9 μA, and 4.1 mA respectively. It is worth mentioning that WuCs can be decoded in the light sleep mode but it is not possible to send and receive data packets in this mode. To send and receive a data packet, the MR has to be in the active mode.

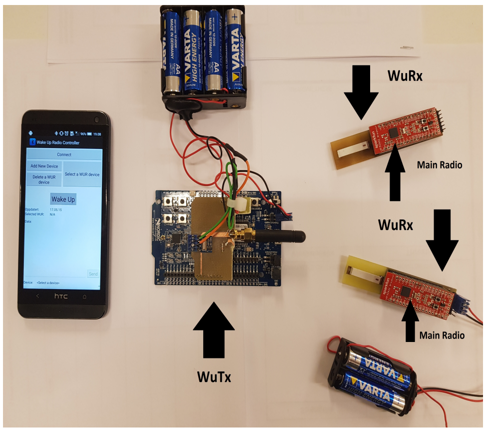


Figure 4.4: A WuR testbed with one gateway, one WuTx, and two WuRxs [2]

Whenever a WuRx detects a WuC, it activates the partial function of MCU to decode and validate the address. If the WuC address matches with its own, it switches on its MR for data communication over Bluetooth low energy (BLE). Fig. 4.4 presents the developed prototype for functionality validation which includes one gateway, one WuTx, and two WuRxs.

4.4.4 Validation of Early Sleeping

Based on the developed testbed, we have performed various experiments to measure the time taken by a WuRx to decode a WuC address when early sleeping or full address decoding is adopted respectively. A WuRx decodes and validates a complete address if full address decoding is adopted. When early sleeping is employed, the WuRx activates partial functions of MCU to check the first clock flank of the WuC address. Then it immediately starts the timeout timer to verify every bit of the received WuC address to avoid the MCU from accumulating a whole sequence of the address. The WuRx fully actives MCU, i.e., all functions of the MCU when it finds the received WuC address matches with its address entirely. The WuRx goes to sleep, i.e., it discards the decoding process whenever it finds a mismatch between the received address and its address. We find that the WuRx takes 10 ms to decode one bit. Accordingly, a WuRx needs 95 ms to decode a full 8-bit address which includes the time for preamble (13 ms) detection and switch-on time of MCU (2 ms). When early sleeping is deployed, a WuRx stops address decoding when it finds that the first bit of the received address does not match with its corresponding address bits after 23 ms. This is obvious since a WuRx requires 13 ms for the preamble and $1 \times 10 =$10 ms for decoding and validating one bit of an 8-bit received address.

For instance, when full address decoding is deployed (as shown in Fig. 4.1(a)), nodes consume (1.9 μA $\times$ 90 ms) = 0.513 μW more energy than when early early sleeping is adopted (as shown in Fig. 4.1(b)).

4.4.5 Validation of Early Data Transmission

We validate the functionality of early data transmission through the same testbed as used for validating the early sleeping mechanism. For early data transmission, a WuRx adopts a similar principle as full address decoding, i.e., a WuRx decodes and validates the entire sequence of the received address by activating the partial functions of the MCU. In case of early data transmission with ACK, the WuRx activates the full function of the MCU and switches on the MR to the active mode for sending an ACK. The WuRx goes to monitor the channel without switching on its MR since no ACK is needed in early data transmission without ACK.

The benefit of introducing early data transmission is that, when a data frame is tiny, it eliminates the necessity of separate data transmission after WuC transmission. Thus it saves energy cost for a packet transmission.

4.5 Chapter Summary

In summary, this chapter points out the overhearing and latency drawbacks caused by full address decoding and WuC handshake respectively. It proposes two schemes, i.e., early sleeping and early data transmission for WuR networks, which are presented in details in Paper C. Moreover, it explains how the early sleeping and early data transmission schemes are validated through a WuR testbed.

Chapter 5

Energy-Efficient Relay Selection in Multi-hop WuR Networks

5.1 Introduction

Having investigated various aspects of single-hop WuR networks, this chapter studies relay selection for multi-hop WuR enabled WSNs and highlights their relative strengths and weaknesses. Furthermore, a relay selection scheme which improves energy-efficiency by ensuring load-balancing in such networks is proposed.

5.2 Relay Selection Schemes for WSNs

This section focuses on traditional WSNs. There exist a vast number of relay selection schemes/protocols but they are designed for WSNs and cannot be directly applied to WuR enabled IoT networks. A few notable relay selection schemes for WSNs are briefly explained in this section.

Since WSNs consist of power-constrained IoT devices, efficient relay selection schemes/routing protocols play a major role to improve network lifetime. According to energy-efficient relay selection proposed in [48], the source node broadcasts a message and a set of next-hop relay nodes overhear that packet. A subset of relay nodes forward that packet to the sink or fusion center. This scheme reduces the number of relay nodes at each hop. However, such a scheme increases end-to-end energy consumption for data transmission since the sink receives the same data from multiple relay nodes. A multiple relay selection mechanism which was proposed in [49] selects the best relays by maximizing signal-to-noise ratio (SNR). The relays are sorted based on the received SNR in order to cooperate in data transmission. Like [48], this scheme is not energy friendly either since multiple copies of the same data packet get transmitted from several relays towards the destination.

A cooperative communication with a relay selection scheme for provisioning quality of service (QoS) in WSNs, referred to as QoS-RSCC, was proposed in [50]. QoS-RSCC was developed using a reinforcement learning framework to select optimal relays based on outage probability and channel efficiency. It can achieve near-optimal performance in terms of diversity gain and channel efficiency and fits well in dynamic networks. However, it is computationally expensive considering the low-computing capabilities of IoT devices. Relay selection based cooperative routing (RBCR) [51] calculates routing paths based on the channel state information and energy consumption of intermediate nodes. RBCR exhibits superior performance than ad hoc on-demand distance vector (AODV) routing in terms successful packet transmission rate and power consumption. However, RBCR requires to establish a path from source to destination prior to actual data transmission, thus increasing data communication latency.

5.3 Existing Relay Selection Schemes for WuRs

Existing relay selection protocols for WuR enabled WSNs can be classified into five categories: topology-based; energy-aware; semantic addressing based; load-balancing based; and flooding based schemes.

5.3.1 Topology-based Relay Selection

In this category, each node maintains the information about its next hop relay nodes along the end-to-end route towards the sink. This information is collected by the sink through control message dissemination. The authors in [52] proposed a topology control scheme to maintain the end-to-end routes for dual-radio enabled WSNs. Each node uses its WuR to establish an end-to-end route to the sink. This protocol operates on an out-of-band channel and reduces WuC collisions in the network. However, it increases latency for data communication.

5.3.2 Energy-aware Relay Selection

The primary objective of energy-aware relay selection protocols is to prolong network lifetime by selecting optimal relays. These relays are typically selected based on residual energy. Opportunistic wake-up MAC (OPWUM) is an energy-aware relay selection scheme proposed in [53] which relies on timer-based contention. OP-

WUM selects the best relay node among a set of neighbor nodes based on the residual energy metric. The sender does not require any prior knowledge about its neighbor nodes for selecting the best relay candidate. However, it utilizes RTS/CTS/about to send (ATS) handshake messages for selecting the best relay nodes, resulting in longer latency.

5.3.3 Multicast-based Relay Selection

In multicast-based relay selection, a sender selects a set of relay nodes from its neighborhood to forward data packets. The authors in [13] proposed a multicast based cross-layer protocol for WuR networks, known as ALBA-WUR. In ALBA-WUR, each node is a member of a specific group and a multi-hop WSN consists of multiple groups. When a node of a specific group wakes up, it multicasts its WuC. Then the next-hop nodes in the same multicast group will wake up. Multicast groups are selected based on region/neighborhood. After this phase, the sender transmits RTS and the awakened nodes reply with CTS. In the CTS packet, nodes include their queue priority index and geographic priority index. Based on this information, the sender selects the relay candidate. Nevertheless, this scheme still suffers from overhead in the relay selection process. Moreover, it is computationally expensive and not beneficial since a node is required to select a relay node based on a given metric, e.g., energy consumption.

5.3.4 Load-Balancing based Relay Selection

A few relay selection schemes are designed for load balancing in a network. Such schemes do not choose the shortest path metric towards the destination but consider the residual energy of the nodes to prolong network lifetime. WuRx-optimised routing and topology optimization approach (WRTA) [54] is a load balancing relay selection scheme which combines path calculations and topology optimization for on-demand communication. WRTA selects intermediate relay nodes towards destination based on the energy level, QoS parameters, and bandwidth towards the nodes. WRTA suits well for light traffic load. Nevertheless, it experiences high packet loss when traffic load is high in a network.

5.3.5 Flooding based Relay Selection

In flooding based protocols, a node which has data to transmit broadcasts a WuC. All the neighboring nodes which are in the broadcast zone forward the same received packet. The same procedure is repeated until the packet reaches the destination successfully. This type of protocols will introduce lots of traffic (broadcast storm). Zippy [55] is a cross-layer relay selection algorithm which adopts on-demand network flooding for multi-hop WuR networks. The Zippy protocol features asynchronous wake-up, bit-level data dissemination, and carrier frequency randomization, leveraging low complexity for WuRs. This protocol does not require costly topology maintenance and route or intermediate relay or route discovery process. However, it generates a large amount of redundant traffic.

To summarize, the state-of-the-art relay selection mechanisms for WuR enabled IoT networks suffer from long latency or energy wastage due to control message overhead or route establishment prior to data transmissions. Thus it is imperative to further study relation selection and develop new relay selection schemes for WuR enabled multi-hop WSNs.

5.4 A Lightweight Relay Selection Scheme for WuR

In this section, we present a lightweight relay (LR) selection scheme proposed for WuR enabled IoT networks, referred to as LR-WuR. LR-WuR is a cross-layer protocol which minimizes overhead by allowing a WuR node to choose a relay node from a set of relay nodes to forward data towards the destination without requiring a route establishment phase.

In a nutshell, LR-WuR maintains a lookup table for relay selection and ensures load balancing among relay nodes in the network. To do so, a sender maintains an address lookup database of its next level nodes to choose the next hop forwarder. As such, each node keeps track of the transmissions of its upper-level nodes which are within its coverage. When a sender has a packet to transmit/forward, it checks its database to select a relay which did not perform data packet exchange in the previous round(s) of transmission. When all the addresses of the connected next level nodes are marked, the lookup database will be reset. The neighbor discovery cost can be subsequently reduced by introducing such a lookup table in the node for forwarding packets.

The working principle of LR-WuR is presented in Fig. 5.1 where N1 and N2 are associated with two higher level nodes N3 and N4. All the nodes can generate packets and act as a relay for lower level nodes. When N1 generates a packet, it wakes up N3 from the sleep mode for forwarding its packet. Upon receiving the data packet, N3 sends an ACK packet to node N1. Then, N1 marks N3 as used in

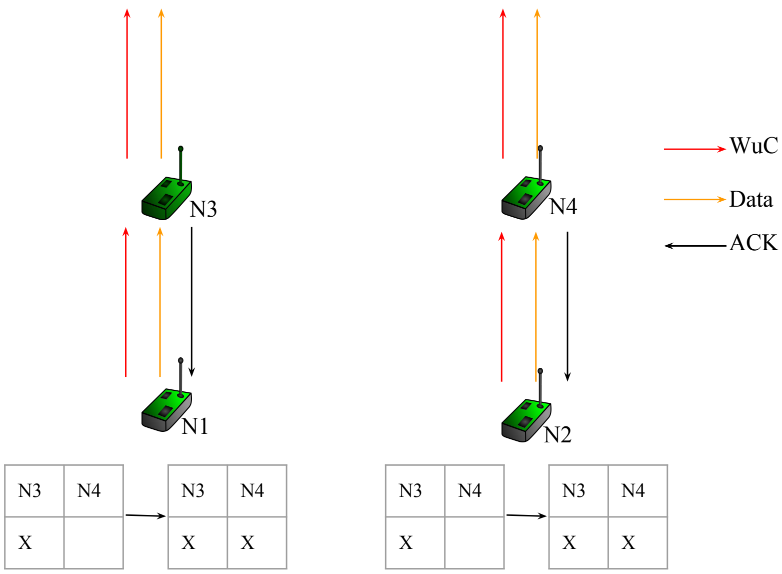


Figure 5.1: The operations of LR-WuR.

the database by inserting an 'X' in the lookup table. When N3 wakes up a next level node, N1 and N2 overhear the WuC packet. N2 utilizes this overheard transmission to update its database. N1 does not take any action since it already knows that N3 has been utilized.

Afterwards, when N2 generates a packet, it will wake N4 up instead of N3 since it knows from its lookup table that N3 transmitted/forwarded a packet in the previous round. Then N4 will forward the packet to the next level node towards the sink/destination. N1 and N2 will overhear the WuC transmission of N4. Accordingly, N1 will update its lookup table. When a node finds that all the connected next level nodes are exploited, it will reset its lookup table by deleting 'X' from the row. The same procedure will be repeated at each hop until the packet has reached the destination. It is worth mentioning that LR-WuR allows retransmission, i.e., the sending node/forwarding node retransmits the WuC and data packet if no ACK is received within an ACK timeout period. A node discards the packet if the preconfigured retransmission limit is exceeded.

LR-WuR requires neither a path establishment phase prior to actual data communication nor a control message dissemination protocol for relay selection. Therefore, LR-WuR ensures energy-efficient communication in WuR networks.

5.5 An Introduction to the Analytical Framework

In this section, we briefly introduce the analytical model to assess the performance of multi-hop LR-WuR enabled IoT networks.

To analyze the performance of LR-WuR in WuR enabled multi-hop networks, we develop an M/G/1 queuing framework. We calculate the collision probability of WuC transmissions using the developed model and compute packet error probability based on channel quality, i.e., SNR. We assume that no packet loss occurs due to buffer overflow. In a multi-hop WuR network, a WuR node sends a WuC to the targeted relay node in an on-demand manner whenever it generates a data packet or receives a data packet from a lower level node. Collisions among WuC transmissions may happen due to the randomness of packet generations in a WuR enabled IoT network. Within a hop, the probability that a packet loss occurs due to collision and channel error, denoted by P_l, can be computed as

$$P_l = P_c + (1 - P_c)P_e \tag{5.1}$$

where P_c and P_e are the collision probability and packet error probability (packet error occurs due to channel impairment) respectively. We further consider that the packet loss probability of a hop is independent of other hops towards the destination. Thus, based on the obtained P_l for a single hop, we compute the end-to-end packet loss probability by taking the product of the packet loss probability at each hop over N hops. Accordingly other performance metrics, i.e., end-to-end packet loss probability, latency, and energy consumption, are calculated.

5.6 Chapter Summary

This chapter provides a clue on relay selection and proposes a relay selection scheme for multi-hop WuR enabled IoT networks. The LR-WuR scheme with its details presented in Paper D ensures load balancing in the network and eliminates the necessity of route establishment prior to actual data communication. Thus it conserves energy and reduces latency of end-to-end data communication.

Chapter 6

Conclusions and Future Work

This final chapter is organized into three sections. It first summarizes the dissertation work. Then, the contributions of this dissertation are highlighted. Finally, it points out a few potential future directions to continue the research work relevant to the issues addressed in this thesis for WuR enabled IoT networks.

6.1 Conclusions

WuR is one of the key enabling techniques as it prominently improves the energy-efficiency of IoT networks/WSNs and prolongs substantially the lifetime of a node. After an extensive review on the state-of-the-art WuR protocols and mechanisms in the literature, a few fundamental problems are identified for this thesis work, i.e., WuC collisions, overhearing of WuC transmissions, latency, and overhead of data transmission.

In Chapter 2 and Chapter 3, the WuC collision problems are investigated for synchronous and asynchronous WuR networks respectively. Accordingly, we presented several MAC protocols to combat WuC collisions. The proposed protocols along with their performance evaluation are illustrated in details in Paper A and Paper B.

Chapter 4 discusses the overhearing energy wastage and latency issues in WuR networks. To reduce overhearing and latency, it introduces two transmission schemes which are proposed in Paper C.

For a multi-hop WuR network, route establishment before data transmission is a burden considering the energy constraint of WuR nodes. In Chapter 5, existing relay selection schemes for WuR networks are outlined and an energy-efficient relay selection scheme is presented. The proposed scheme is presented in Paper D.

6.2 Contributions

This dissertation takes forward the state-of-the-art research efforts on WuR enabled IoT networks from four directions: MAC mechanism design, enabling early sleeping and early data transmission, relay selection, and mathematical modeling for performance evaluation of WuR enabled IoT networks. In particular, the main contributions are summarized as follows.

- To improve the performance of WuR enabled IoT networks, four protocols, i.e., BoWuR, CCA-WuR, CSMA-WuR, and ADP-WuR, have been proposed and the performance of the proposed protocols is compared with a few popular WuR protocols. BoWuR is dedicated when traffic load is heavy in *synchronous* WuR networks. On the other hand, CCA-WuR, CSMA-WuR, and ADP-WuR protocols are suitable for *asynchronous* WuR networks under low, high, and variable traffic load conditions respectively.

- A bit-by-bit address decoding scheme, i.e., the early sleeping scheme, is proposed to avoid overhearing energy wastage of unintended WuR nodes. The performance of the bit-by-bit address decoding scheme is compared with the full address decoding scheme. It is evident that early sleeping prolongs the lifetime of WuR nodes thanks to the early stage sleeping property of bit-by-bit address decoding.

- An early data transmission scheme, i.e., EDT, has been developed to reduce delay and minimize energy consumption of data communications. EDT encodes tiny data with a WuC address and sends data together with the WuC as an encoded WuC frame to the target node. The target node decodes the encoded WuC packets in the light sleep mode of the MCU. No additional transmission by the MR is needed to send such tiny data packets.

- LR-WuR has been proposed and it is shown as an energy-efficient relay selection scheme for multi-hop WuR networks since it does not require a route establishment phase prior to actual data transmission. Moreover, LR-WuR does not require control messages for relay selection. Therefore, it reduces latency of data communication and improves energy-efficiency of WuR networks.

- The performance of the proposed MAC protocols, relay selection scheme, early sleeping and early data transmission mechanisms is theoretically assessed by the developed queuing frameworks and DTMC models. Discrete-event based simulations are performed to validate the accuracy of the developed models. Moreover, early sleeping and early data transmission are implemented through a WuR testbed.

6.3 Future Work

This dissertation focused mainly on MAC layer protocol design when WuR nodes are operated in the transmitter-initiated mode. Accordingly, a few schemes and protocols have been developed, and their performance has been investigated through mathematical analysis and discrete-event simulations. Beyond the research questions investigated in this thesis, other topics relevant to the focused research area can also be explored. For instance, RI-WuR, i.e., another branch of WuR transmission mode, WuR enabled massive IoT networks with dynamic traffic, edge computing for IoT devices are emerging areas where a communication paradigm shift is expected to meet the demands for emerging massive IoT applications. Relevant to these potential research themes, the following topics can be taken into consideration for future research.

- In many cases, a communication procedure is initiated by a data collector or receiver, i.e., RI-WuR. In RI-WuR, a sending/child node needs to be waken up from sleep after receiving a WuC even if it does not have any packet to transmit. Such an occurrence has a negative impact on the lifetime of the node. Thus more efficient MAC protocols need to be developed to avoid unnecessary wake-up of sending nodes. Moreover, investigating multi-hop communication in RI-WuR will enrich this research direction.

- Recently a research trend is moving from IoT to massive IoT. In massive IoT, one of the main challenges is to connect a huge amount of IoT devices (e.g., WuR nodes, 802.15.4e enabled sensor nodes) to the gateway/sink. Therefore, it is essential to develop novel access control mechanisms which are able to handle a vast volume of IoT connections as well as heterogenous types of IoT traffic efficiently.

- In our study, we considered a static network topology for performance evaluation of WuR. In real-life, network topology is not always static due to node activities (some nodes may become inactive and the deployed network topology may change due to reconfiguration). Therefore, one potential topic could be performance evaluation of MAC or cross-layer protocols for WuR enabled dynamic IoT networks.

- In many WSN/IoT applications, low latency for data collection is one of the key performance metrics. Thus one potential research topic could be implementing edge computing at the sink/gateway to enable IoT device's perceptions using lightweight machine learning algorithms.

- Ultra-reliable low latency communication (URLLC) is one of the three key features of 5G. However, it is very challenging to ensure URLLC when it comes to massive IoT. So far, how to combine massive IoT and URLLC for non-3GPP connections has not been well studied in the research community. Thus, developing novel data transmission protocols to ensure URLLC for massive IoT with non-3GPP connections could be an important research task.

REFERENCES

[1] IEEE Std., "IEEE 802.15.4: Low-rate wireless personal area networks (LR-WPANs) and physical layer (PHY) specifications," *IEEE-SA*, pp. 1–252, Apr. 2012.

[2] A. Frøytlog, T. Foss, O. Bakker, G. Jevne, M. A. Haglund, F. Y. Li, J. Oller, and G. Ye Li, "Ultra-low power wake-up radio for 5G IoT," *IEEE Commun. Mag.*, to appear, 2019.

[3] R. Minerva, A. Biru, and D. Rotondi, "Towards a definition of the Internet of things (IoT)," *IEEE Internet Initiative*, vol. 1, pp. 1–86, May 2015.

[4] Qualcomm, "LTE IoT is starting to connect the massive IoT today, thanks to eMTC and NB-IoT," Jun. 2017, [Online] Available: https://www.qualcomm.com/news/onq/2017/06/15/lte-iot-starting-connect-massive-iot-today-thanks-emtc-and-nb-iot, Accessed on 27 Dec. 2018.

[5] Ericsson, "Ericsson mobility report: Internet of things forecast," 2018, [Online] Available: https://www.ericsson.com/en/mobility-report/internet-of-things-forecast, Accessed on 27 Dec. 2018.

[6] 3GPP TR 45.820, "Cellular system support for ultra-low complexity and low throughput Internet of things (CIoT)," v13.1.0, Release 13, Dec. 2015.

[7] 3GPP TR 22.861, "Feasibility study on new services and markets technology enablers for massive Internet of things," v14.1.0, Release 14, Sep. 2016.

[8] 3GPP TR 38.913, "Study on scenarios and requirements for next generation access technologies," v14.2.0, Release 14, Jun. 2017.

[9] Y.-P. E. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, J. Bergman, and H. S. Razaghi, "A primer on 3GPP narrowband Internet of things," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 117–123, Mar. 2017.

[10] P. Huang, L. Xiao, S. Soltani, M. W. Mutka, and N. Xi, "The evolution of MAC protocols in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 101–120, Feb. 2013.

[11] F. Alfayez, M. Hammoudeh, and A. Abuarqoub, "A survey on MAC protocols for duty-cycled wireless sensor networks," in *Proc. Int. Conf. on Advanced Wireless Information, Data, and Commun. Tech.*, Jan. 2015, pp. 482–489.

[12] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674–687, Apr. 2016.

[13] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," in *Proc. IEEE INFOCOM*, Apr. 2015, pp. 522–530.

[14] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proc. IEEE INFOCOM*, Jun. 2002, pp. 1567–1576.

[15] Y. Sun, S. Du, O. Gurewitz, and D. B. Johnson, "DW-MAC: A low latency, energy efficient demand-wakeup MAC protocol for wireless sensor networks," in *Proc. ACM MobiHoc*, May 2008, pp. 53–62.

[16] D. K. McCormick, "IEEE technology report on wake-up radio: An application, market, and technology impact analysis of low-power/low-latency 802.11 wireless LAN interfaces," *802.11ba Battery Life Improvement: IEEE Technology Report on Wake-Up Radio*, pp. 1–56, Nov. 2017, [Online] Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8055459&isnumber=8055458, Accessed on 27 Dec. 2018.

[17] IEEE Std., "IEEE 802.11: Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE-SA*, pp. 1–3534, Dec. 2016.

[18] M. Zhang, D. Ghose, and F. Y. Li, "Does wake-up radio always consume lower energy than duty-cycled protocols?" in *Proc. IEEE VTC-Fall*, Sep. 2017, pp. 1–5.

[19] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[20] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions," *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 159–172, Feb. 2007.

[21] N. Semiconductors, "JN-AN-1001 JN516x power consumption," *Application Note, Eindhoven, The Netherlands*, 2016.

[22] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proc. ACM SenSys*, Oct. 2006, pp. 307–320.

[23] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC: A receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proc. ACM SenSys*, Nov. 2008, pp. 1–14.

[24] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Optimizing sensor networks in the energy-latency-density design space," *IEEE Trans. Mobile Comput.*, vol. 99, no. 1, pp. 70–80, Jan. 2002.

[25] X. Yang and N. H. Vaidya, "A wakeup scheme for sensor networks: Achieving balance between energy saving and end-to-end delay," in *Proc. IEEE RTAS*, May 2004, pp. 19–26.

[26] T. N. Le, M. Magno, A. Pegatoquet, O. Berder, O. Sentieys, and E. Popovici, "Ultra low power asynchronous MAC protocol using wake-up radio for energy neutral WSN," in *Proc. ACM ENSSys*, Nov. 2013, pp. 10:1–10:6.

[27] T. O. Kim, J. S. Park, H. J. Chong, K. J. Kim, and B. D. Choi, "Performance analysis of IEEE 802.15.4 non-beacon mode with the unslotted CSMA/CA," *IEEE Commun. Lett.*, vol. 12, no. 4, pp. 238–240, Apr. 2008.

[28] C. Buratti and R. Verdone, "Performance analysis of IEEE 802.15.4 non beacon-enabled mode," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, p. 3480, Sep. 2009.

[29] X. Cao, J. Chen, Y. Cheng, X. S. Shen, and Y. Sun, "An analytical MAC model for IEEE 802.15.4 enabled wireless networks with periodic traffic," *IEEE Trans. Wireless Commun.*, vol. 14, no. 10, pp. 5261–5273, Oct. 2015.

[30] A. Faridi, M. R. Palattella, A. Lozano, M. Dohler, G. Boggia, L. A. Grieco, and P. Camarda, "Comprehensive evaluation of the IEEE 802.15.4 MAC layer performance with retransmissions," *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, pp. 3917–3932, Oct. 2010.

[31] J. Mišić and V. B. Mišić, "Access delay for nodes with finite buffers in IEEE 802.15.4 beacon enabled PAN with uplink transmissions," *Comp. Commun., Elsevier*, vol. 28, no. 10, pp. 1152–1166, Jun. 2005.

[32] P. K. Sahoo and J.-P. Sheu, “Modeling IEEE 802.15.4 based wireless sensor network with packet retry limits,” in *Proc. ACM MSWiM*, Oct. 2008, pp. 63–70.

[33] R. K. Patro, M. Raina, V. Ganapathy, M. Shamaiah, and C. Thejaswi, “Analysis and improvement of contention access protocol in IEEE 802.15.4 star network,” in *Proc. IEEE MASS*, Oct. 2007, pp. 1–8.

[34] H. Wen, C. Lin, Z.-J. Chen, H. Yin, T. He, and E. Dutkiewicz, “An improved Markov model for IEEE 802.15.4 slotted CSMA/CA mechanism,” *J. of Comp. Sci. and Tech.*, vol. 24, no. 3, pp. 495–504, May 2009.

[35] S. Pollin, M. Ergen, S. C. Ergen, B. Bougard, F. Catthoor, A. Bahai, and P. Varaiya, “Performance analysis of slotted carrier sense IEEE 802.15.4 acknowledged uplink transmissions,” in *Proc. IEEE WCNC*, Mar. 2008, pp. 1559–1564.

[36] S. Pollin, M. Ergen, S. C. Ergen, B. Bougard, L. Van der Perre, I. Moerman, A. Bahai, P. Varaiya, and F. Catthoor, “Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 9, Sep. 2008.

[37] F. Z. Djiroun and D. Djenouri, “MAC protocols with wake-up radio for wireless sensor networks: A review,” *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 587–618, Q1 2017.

[38] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, “Ultra low power wake-up radios: A hardware and networking survey,” *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2117–2157, Q4 2017.

[39] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. Popovici, and L. Benini, “Design, implementation, and performance evaluation of a flexible low-latency nanowatt wake-up radio receiver,” *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 633–644, Apr. 2016.

[40] J. Oller, E. Garcia, E. Lopez, I. Demirkol, J. Casademont, J. Paradells, U. Gamm, and L. Reindl, “IEEE 802.11-enabled wake-up radio system: Design and performance evaluation,” *Electronics Letters*, vol. 50, no. 20, pp. 1484–1486, Sep. 2014.

[41] Microsemi, “Wireless for implantable medical devices ZL70103 MICS-band RF transceiver,” 2015, [Online] Available: https://www.microsemi.com/

product-directory/implantable-medical-transceivers/3915-zl70103, Accessed on 27 Dec. 2018.

[42] R. Van Langevelde, M. van Elzakker, D. van Goor, H. Termeer, J. Moss, and A. Davie, "An ultra-low-power 868/915 MHz RF transceiver for wireless sensor network applications," in *Proc. IEEE RFIC*, Jun. 2009, pp. 113–116.

[43] J. Lebreton, N. Murad, S. Kandukuri, and R. Lorion, "An energy-efficient addressing mechanism for wake-up radio systems," in *Proc. IEEE WTS*, Apr. 2016, pp. 1–6.

[44] Ericsson, "Ericsson mobility report: On the pulse of the networked society," Jun. 2015, [Online] Available: https://www.ericsson.com/assets/local/mobility-report/documents/2015/ericsson-mobility-report-june-2015.pdf, Accessed on 27 Dec. 2018.

[45] R. Sinha, C. Papadopoulos, and J. Heidemann, "Internet packet size distributions: Some observations," USC/Information Sciences Institute, Tech. Rep., May 2007, [Online] Available: https://www.isi.edu/~johnh/PAPERS/Sinha07a.pdf, Accessed on 27 Dec. 2018.

[46] T.-Y. Chan, Y. Ren, Y.-C. Tseng, and J.-C. Chen, "eHint: An efficient protocol for uploading small-size IoT data," in *Proc. IEEE WCNC*, Mar. 2017, pp. 1–6.

[47] A. Hoglund, D. P. Van, T. Tirronen, O. Liberg, Y. Sui, and E. A. Yavuz, "3GPP Release 15 early data transmission," *Communications Standards: A Supplement to IEEE Commun. Mag.*, vol. 2, no. 2, pp. 90–96, 2018.

[48] W. Alameddine, W. Hamouda, and J. Haghighat, "Energy efficient relay selection scheme for cooperative uniformly distributed wireless sensor networks," in *Proc. IEEE ICC*, Jun. 2014, pp. 184–189.

[49] A. Piltan, S. Salari, S. M. Khodaei, M. Mirza, and J.-P. Cances, "An efficient multiple relay selection scheme for wireless sensor networks under global power constraint," in *Proc. IEEE FITCE*, Aug. 2011, pp. 1–5.

[50] X. Liang, I. Balasingham, and V. C. Leung, "Cooperative communications with relay selection for QoS provisioning in wireless sensor networks," in *Proc. IEEE GLOBECOM*, Nov. 2009, pp. 1–8.

[51] A. B. Nacef, S.-M. Senouci, Y. Ghamri-Doudane, and A.-L. Beylot, "A combined relay-selection and routing protocol for cooperative wireless sensor networks," in *Proc. IEEE IWCMC*, Aug. 2012, pp. 293–298.

[52] T. Stathopoulos, M. Lukac, D. McIntire, J. Heidemann, D. Estrin, and W. J. Kaiser, “End-to-end routing for dual-radio sensor networks,” in *Proc. IEEE INFOCOM*, May 2007, pp. 2252–2260.

[53] F. Ait Aoudia, M. Gautier, and O. Berder, “OPWUM: Opportunistic MAC protocol leveraging wake-up receivers in WSNs,” *J. of Sensors*, vol. 2016, Sep. 2016.

[54] M. Vodel, M. Lippmann, and W. Hardt, “WRTA-wake-up-receiver optimised routing and topology optimisation approach,” in *Proc. IEEE ITST*, Nov. 2012, pp. 329–334.

[55] F. Sutton, B. Buchli, J. Beutel, and L. Thiele, “Zippy: On-demand network flooding,” in *Proc. ACM SenSys*, Nov. 2015, pp. 45–58.

PART II

Paper A

A

Title:	Enabling Backoff for SCM Wake-Up Radio: Protocol and Modeling
Authors:	Debasish Ghose and Frank Y. Li
Affiliation:	Dept. of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway
Journal:	*IEEE Communications Letters*, vol. 21, no. 5, pp. 1031-1034, May 2017.
DOI:	10.1109/LCOMM.2017.2653779.
Copyright ©:	IEEE

Enabling Backoff for SCM Wake-Up Radio: Protocol and Modeling

Debasish Ghose and Frank Y. Li

***Abstract* — In sub-carrier modulation (SCM) wake-up radio (WuR) enabled wireless sensor networks, a node can initiate data transmission at any instant of time. In this letter, we propose to activate a backoff procedure before sending wake-up calls (WuCs) in order to avoid potential collisions among WuCs. Consequently, no backoff is needed for the main radio after a WuC is received. A discrete-time Markov chain model is developed to evaluate the performance. Numerical results on network throughput, energy efficiency, average delay, and collision probability reveal the benefits of enabling backoff for SCM-WuRs, especially under heavy traffic loads or saturated traffic conditions.**

Keywords—WSNs, wake-up radio, wake-up call collision, Markov modeling, performance evaluation.

I. Introduction

Recently, a paradigm shift from duty-cycled (DC) medium access control (MAC) to wake-up radio (WuR) based wireless sensor networks (WSNs) is envisaged. With a reception power consumption level in the magnitude of 1000 times lower than the main radio (MR) [1] [2], WuR appears as a promising solution for achieving much longer lifetime by eliminating overhearing and idle listening in DC WSNs [1]– [4]. Among various platforms and implementations of WuRs, the sub-carrier modulation (SCM) WuR is probably the most popular approach due to its long coverage and in-depth studies in realistic scenarios [1] [2]. In a SCM-WuR based WSN, an auxiliary transceiver is introduced for the purpose of awakening up adjacent sensor nodes in an on-demand manner. A wake-up receiver (WuRx) is attached to the micro-controller unit (MCU) of a sensor node to detect the wake-up call (WuC) sent by a wake-up transmitter (WuTx). Upon the detection of such a WuC, the WuRx interrupts the MCU of the sensor node to switch on the MR from the sleep mode. When the MR is turned on, the active nodes can communicate with each other for data exchange. If multiple nodes are awoken at the same time, collisions among data frames are handled by adopting a contention based MAC protocol *among the MRs*.

Manuscript received December 16, 2016; accepted January 4, 2017. Date of publication January 12, 2017.

The benefits of WuR over DC MAC were discussed in [1]. Communications among the WuR enabled sensor nodes may be performed either in a transmitter-initiated (TI) or in a receiver-initiated (RI) mode. Which mode suits better is scenario specific, depending on the application requirements of a WuR based WSN. In general, RI suits better for *data collections* whereas TI is more appropriate for event-triggered *data reporting* in WSNs. Even though both RI and TI were presented in [1], the authors focused mainly on the RI mode when demonstrating the superiority of SCM-WuRs over a few popular DC MAC protocols. In the TI mode, however, *collisions may occur among WuCs when multiple SCM-WuRs detect an event at the same time*. The depiction of how to tackle such a collision problem among WuCs is not found in [1], nor is the performance analyzed mathematically therein.

The observation that collision avoidance was not addressed in [1]- [4] triggered our motivation to further investigate the performance of single-radio WuRs. In this letter, we propose a fixed-size contention window (CW) based backoff (BO) procedure together with clear channel assessment (CCA) in order to avoid collision among WuCs and reduce delay. Furthermore, we develop a discrete time Markov chain (DTMC) model to evaluate the performance of the BO-enabled transmission protocol. Apart from reducing the number of collisions among WuCs, such a backoff mechanism for WuCs would also help to achieve higher network throughput and energy efficiency.

II. Network Scenario and Protocol Design

In this section we describe the network scenario and WuR principle, and propose a BO-enabled SCM-WuR protocol.

A. Network Scenario and WuR Principle

Consider a distributed WuR-based WSN consisting of n number of sensor nodes. All sensor nodes encompass an integrated WuR and can communicate with each other over a single hop. A node can send a WuC to initiate a communication at any instant of time on an on-demand basis. The WuR platform is considered in this study is the SCM-WuR mentioned above. Its detailed design and performance assessment can be found in [1] [2]. In an SCM-WuR, *there is only one channel through which both data and WuC are transmitted* by switching the antenna configuration dynamically. A WuR may operate as either a WuRx or a WuTx, controlled by a programmable MCU. In what follows, we focus on a WuR-based WSN operated in the TI mode.

When operated in the TI mode, a node which has a packet to send transmits a WuC to the targeted node to initiate a data transmission by the MR. After detecting the WuC, the receiving node also switches on its MR for data communication. It is worth mentioning that *no ACK is needed for acknowledging the successful reception of the WuC* [1] to avoid frequent switches between WuR and MR. When both MRs are turned on, the node which sent the WuC follows the procedure of the adopted MAC protocol for data transmission. Depending on the type of MAC protocol being used, the MR may or may not perform a BO procedure before data transmission.

B. Backoff enabled SCM-WuR (BoWuR)

The main idea of the proposed fixed-size BO enabled SCM-WuR is to avoid collision among WuCs. The BoWuR works similar to the CSMA/CA MAC protocol but applies to WuCs. When an event is detected by a WuR, it performs CCA first to check whether the channel is idle or not. If the WuR finds the channel being idle then it selects a random backoff time from a fixed CW size, otherwise it performs CCA again. The node which selects the shortest backoff time sends its WuC and switches on its MR.

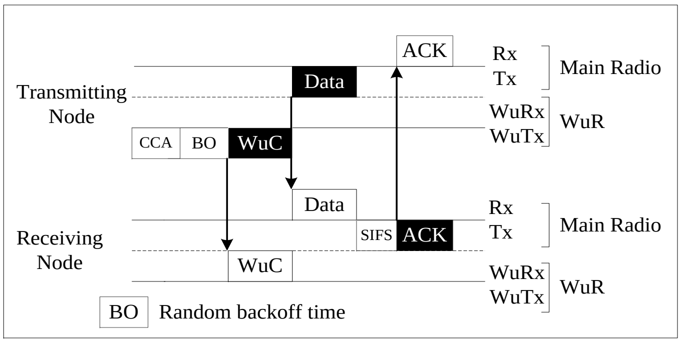


Figure A.1: Illustration of the working principle of the BoWuR protocol.

Since the WuC is sent through a wireless medium, all neighboring WuRs receive the same WuC. After decoding the WuC frame, the targeted WuR switches on its MR for communication. The other WuRs which have packets to send but lost WuC access contention, freeze their remaining backoff time and use it for next round access competition. When both transmitting and receiving MRs are switched on, data exchange starts immediately. The transmission cycle ends up with an ACK from the receiving MR to the transmitting MR. The principle of BoWuR is shown in Fig. A.1 for one data frame transmission.

However, *WuC collision happens if two or more nodes select the same smallest backoff time after CCA*. Since no ACK is needed after the WuC, the sending node will simply transmit its data frame from its MR. If no ACK is received at the MR after the ACK timeout period has elapsed, the sending node will retransmit its WuC following the same procedure.

The BoWuR WuC frame structure is extended from the one designed in [8] and it is shown in Fig. A.2. A WuC frame is of 48 bits long in which one byte is reserved for the network allocation vector (NAV) field and two bytes are dedicated to allocating WuR addresses.

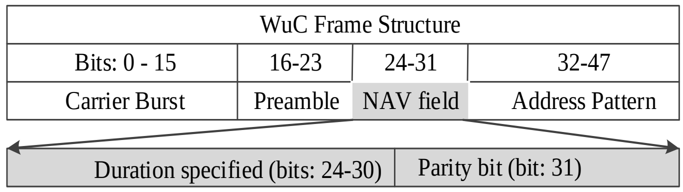


Figure A.2: The designed BoWuC frame format for SCM-WuR.

III. DTMC Modeling of BoWuR

Let $b(t)$ be a stochastic process which represents the BO counter for the node of interest. A discrete and integer time scale is adapted to illustrate the BO process. In the time scale, t and $t+1$ correspond to the initial instants of two successive BO slot durations after CCA. Each node reduces its BO time counter at the starting instant of each slot. The BO time decrement halts when the communication channel is sensed as busy. Thus, *the slot duration between* t *and* $t+1$ *may be longer than one specified slot time*, as it may be comprised of a packet transmission or collision.

In a BoWuR enabled WSN, each node selects its random BO time after CCA from the fixed-size CW, regardless its transmission history. That is, no matter whether the to-be-transmitted packet is a fresh or collided one, the BO time is always selected from $\{0, CW-1\}$. Since the backoff time selection depends only on the fixed CW size, the retry limit does not affect the DTMC model. In the DTMC, the one-step transition probabilities are given as

$$\begin{cases} P\{b(t+1)=k|b(t)=k+1\}=1, & k\epsilon\{0,CW-2\}, \\ P\{b(t+1)=k|b(t)=0\}=1/CW, & k\epsilon\{0,CW-1\}, \end{cases}$$

where the first and second parts of the expression indicate the BO time decrement at the beginning of each slot time and that a WuR has selected its random BO time from the uniformly distributed fixed-size CW, $\{0, CW-1\}$, respectively.

Let $b_k = \lim_{t\to\infty} P\{b(t)=k\}, \quad k \in \{0, CW-1\}$ be the stationary distribution of the Markov chain. We have

$$b_k = \frac{CW - k}{CW} b_0, \quad k \in \{0, CW - 1\}. \tag{A.1}$$

In (1), b_k is presented in terms of b_0. Thus it is easy to obtain a solution since the sum of b_k is equal to 1. That is,

$$\begin{aligned} 1 &= \sum_{k=0}^{CW-1} b_k = \sum_{k=0}^{CW-1} b_0 \cdot \frac{CW - k}{CW} = b_0 \sum_{k=0}^{CW-1} (1 - \frac{k}{CW}) \\ &= b_0 \cdot \frac{CW + 1}{2}. \end{aligned} \tag{A.2}$$

When the backoff counter reaches zero, a transmission occurs, i.e., a node wins the competition and transmits its WuC. However, the WuC transmission may not be successful since collision may occur. Accordingly, the transmission probability, τ, can be derived from (2) as $\tau = b_0 = \frac{2}{CW+1}$. It is clear that the value of τ depends only on the CW size and it is independent of the transmission history. That is, τ is the same for a fresh WuC or a retransmitted WuC.

IV. PERFORMANCE ANALYSIS

No hidden terminal existence is a common assumption for DTMC modeling of CSMA/CA, e.g., in [5]. Assume also that no packet loss occurs due to buffer overflow or channel error.

A. Network Throughput

Denote by S the network throughput, defined as the size of the MR data frame transmitted successfully over the length of a slot duration. The network throughput depends on the probability of a node being in the idle state P_i, probability of transmission P_t, probability of successful transmission P_s, and probability of collision P_c. P_t is the probability that at least one node transmits its WuC in the observed slot duration. Assuming that all n nodes have packets to send and they compete for channel access to transmit their WuCs, each node with a transmission probability τ, then $P_i = (1 - \tau)^n$. From P_i, it is easy to compute P_t, i.e., $P_t = 1 - (1 - \tau)^n$. The successful transmission probability that a data frame from the MR has been received at the destination correctly is

$$P_s = \frac{n\tau(1 - \tau)^{n-1}}{P_t}. \tag{A.3}$$

Furthermore, P_c is the probability that a collision occurred among the WuCs in the observed slot duration. A collision occurs when two or more nodes selected the same minimal backoff time and obtained channel access to transmit their WuCs at the same time, expressed as $P_c = P_t \cdot (1 - P_s)$.

Now, S can be formulated as [5]

$$\begin{aligned} S &= \frac{E[Data\ sent\ by\ the\ MR\ in\ a\ slot\ duration]}{E[length\ of\ a\ slot\ duration]} \\ &= \frac{L \cdot P_s \cdot P_t}{T_s \cdot P_s \cdot P_t + T_c \cdot P_c + \sigma \cdot P_i}, \end{aligned} \tag{A.4}$$

where T_s and T_c are the expected slot duration for a successful (no collision) and an unsuccessful transmission (since a collision happened) respectively. σ is an idle slot time, i.e., the duration between two idle slots, and L is the average size of the transmitted packet from the MR. To obtain the throughput in BoWuR, we need to elaborate the time needed for CCA, BO and WuC transmission respectively when calculating T_s and T_c, as shown below

$$\begin{cases} T_s = t_{cca} + t_{wuc} + t_h + t_l + SIFS + t_{ack} \\ T_c = t_{cca} + t_{wuc} + t_h + t_l. \end{cases} \tag{A.5}$$

In (A.5), t_{cca}, t_{wuc}, t_h, t_l, $SIFS$, and t_{ack} are the duration of CCA, WuC duration, time needed to transmit the MAC layer header and payload from the MR, short inter-frame space time (SIFS), and the ACK frame time respectively. As mentioned earlier, the MR in BoWuR transmits a data packet immediately after the WuC is sent no matter it is collided or not. Due to this reason, t_h and t_L are also included in the T_c computation.

B. Average Delay

The average delay is defined as the time duration needed to complete a packet transmission, from the starting instant of the CCA and until the instant when a positive ACK is received at the MR. The average packet delay, $E[D]$, is given by [6]

$$E[D] = E[Y] \cdot E[length\ of\ a\ slot\ duration] \tag{A.6}$$

where $E[Y]$ is the expected number of slot durations required for successfully transmitting a packet from the MR considering retransmissions, and $E[length\ of\ a\ slot\ duration]$ is the expected duration of a slot duration.

For calculating $E[Y]$, the conditional collision probability of a WuC, p_c, and the WuC drop probability, p_{drop}, need to be calculated. Conditioned on that a WuC is being transmitted, we have $p_c = 1 - (1 - \tau)^{n-1}$. p_{drop} is defined as the probability that a WuC has been retransmitted for a maximum number of times (i.e., the retry limit m has been reached) and it is dropped. In other words, p_{drop} is the probability that a WuC encounters $m + 1$ number of collisions. Since the CW size in BoWuR is fixed for both fresh and retransmitted packets, the packet drop probability can be simply expressed as $p_{drop} = p_c^{m+1}$. Correspondingly, we can express $E[Y]$ by:

$$E[Y] = \sum_{i=0}^{m} \left[\frac{(p_c^i - p_c^{m+1}) \cdot \frac{CW+1}{2}}{1 - p_c^{m+1}} \right], \qquad \text{(A.7)}$$

where $1 - p_c^{m+1}$ is the probability that a WuC is in the queue even though it exceeds the maximum retry limit and $(p_c^i - p_c^{m+1}) \cdot \frac{CW+1}{2}/1 - p_c^{m+1}$ is the probability that a WuC is in the queue when it is in the i^{th} retransmission stage. By substituting the values of $E[length\ of\ a\ slot\ duration]$ and $E[Y]$ into (6), the average delay $E[D]$ can be computed.

C. Energy Consumption and Efficiency

Let E_n be the average energy consumed by a reference node l in one slot duration. The reference node is a node randomly selected among those n nodes in the network. E_n can be computed as $E_n = \sum_{i \in \omega} E(i)\pi(i)$ where ω is the set of events (transmission, reception or collision) that can take place in a single slot duration, while $E(i)$ and $\pi(i)$ are the energy consumed during event i and its probability respectively [7].

Consider an identical transmission probability τ among n nodes. The six possible events and their probabilities are obtained as follows. 1) Idle slot, $\pi_i = (1-\tau)^n$; 2) Successful reception of a packet destined to node l, $\pi_s^r = \tau(1-\tau)^{(n-1)}$; 3) Successful reception of a packet not destined to node l, i.e., overhearing, $\pi_{oh}^r = (n-2)\tau(1-\tau)^{(n-1)}$; 4) Reception of a collided packet, $\pi_c^r = (1-\tau)[1-(1-\tau)^{n-1}-(n-1)\tau(1-\tau)^{n-2}]$; 5) Successful transmission of a packet by node l, $\pi_s^t = \tau(1-\tau)^{(n-1)}$; and 6) A packet transmitted by node l collided, $\pi_c^t = \tau[1-(1-\tau)^{n-1}]$. Note that the sum of these probabilities is equal to 1.

Table A.1: Parameter Configuration [1] [2] [8] [9]

Approach	Parameter	Value	Unit
Common	Supply voltage	3	V
Main radio	Data rate	20	kbps
	Transmission current	17.4	mA
	Reception current	18.8	mA
	Idle current	20	μA
	SIFS duration	100	μs
	Payload size	30-120	bytes
	ACK frame size	5	bytes
Wake-up radio	WuC duration	6	ms
	Data rate	8192	bps
	Transmission current (WuTx)	152	mA
	Reception current (WuRx)	8	μA
	Sleep current	3.5	μA
	Backoff current	5.16	mA
	CCA current	20.28	mA
	CCA duration	128	μs
	Slot time	320	μs
	CW size	16, 32	slots
	WuC packet size	6	bytes

Moreover, the energy consumed during each event is computed respectively as follows. 1) $E_i = \rho_{bo}\sigma$; 2) $E_s^r = \rho_{cca}t_{cca} + \rho_w^{rx}t_{wuc} + \rho_m^{rx}(t_h + t_l) + \rho_m^i SIFS + \rho_m^{tx}t_{ack}$; 3) $E_{oh}^r = \rho_{cca}t_{cca} + \rho_w^{rx}t_{wuc} + \rho_w^{sl}(t_h + t_l + SIFS + t_{ack})$; 4) $E_c^r = \rho_{cca}t_{cca} + \rho_w^{rx}t_{wuc} + \rho_w^{sl}(t_h + t_l)$; 5) $E_s^t = \rho_{cca}t_{cca} + \rho_w^{tx}t_{wuc} + \rho_m^{tx}(t_h + t_l) + \rho_m^i SIFS + \rho_m^{rx}t_{ack}$; and 6) $E_c^t = \rho_{cca}t_{cca} + \rho_w^{tx}t_{wuc} + \rho_m^{tx}(t_h + t_l)$, where ρ_{bo} and ρ_{cca} represent the power consumption during the BO and CCA duration. ρ_w^{sl}, ρ_w^{rx}, and ρ_w^{tx} denote the sleep, reception, and transmission power of the WuR, ρ_m^i, ρ_m^{tx}, and ρ_m^{rx} are idle, reception, and transmission power of the MR respectively. Accordingly, E_n can be expressed as

$$E_n = \pi_i E_i + \pi_s^r E_s^r + \pi_{oh}^r E_{oh}^r + \pi_c^r E_c^r + \pi_s^t E_s^t + \pi_c^t E_c^t. \tag{A.8}$$

Finally, energy efficiency η is calculated as the number of successfully transmitted bits per Joule, i.e., $\eta = \pi_s^t L / E_n$.

V. Numerical Results and Discussions

The obtained results are based on the parameters specified in Table A.1 unless oth-

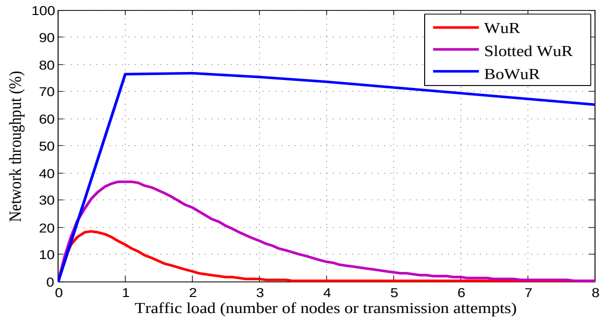


Figure A.3: Throughput comparison of WuR, slotted WuR, and BoWuR.

erwise stated. The propagation delay is not considered. For throughput calculation, we consider that each node always has a WuC to transmit, i.e., the network operates under the saturation traffic condition.

Fig. A.3 shows the network throughput comparison in percentage between BoWuR and two flavors of WuR MAC protocols. From this figure, it is evident that the maximum throughput of BoWuR is around four or two times higher than that of WuR or slotted WuR. This is because the latter two protocols work in an Aloha or slotted Aloha manner. In Aloha or slotted Aloha, it is well understood [10] that the maximum throughput is 18.4% and 36.8% respectively. Thanks to the CCA and BO mechanisms introduced in BoWuR, collision happens much rarer under the same network configuration, leading to much higher throughput.

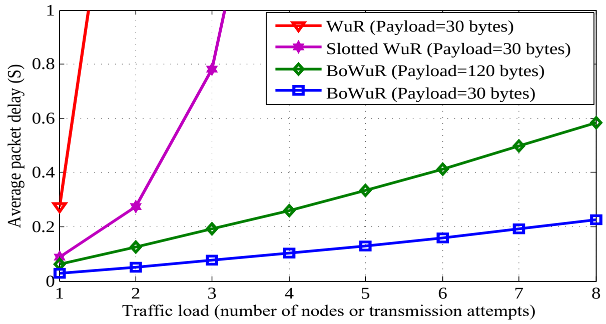


Figure A.4: Average delay as network traffic load varies.

Fig. A.4 depicts the delay performance as the injected traffic load increases. As expected, a longer delay is experienced with a higher traffic load for all protocols. This is because a packet experiences a higher number of retransmissions before it is successfully received, when more nodes are competing with each other. For Aloha and slotted Aloha, packets experience very long delays once the network is close to congestion or becomes saturated. For average delay calculation of BoWuR, the retry limit is configured as $m = 7$. In this case, $E[Y]$ becomes larger with a higher traffic load whereas $E[length\ of\ a\ slot\ duration]$ does not vary significantly for

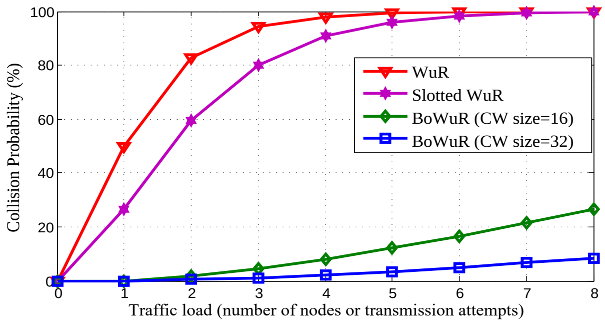


Figure A.5: Collision probability as network traffic load varies.

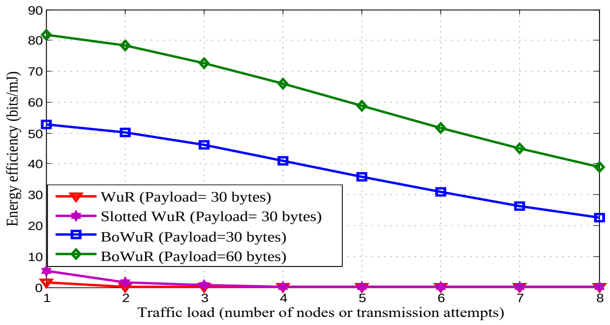


Figure A.6: Energy efficiency comparison of WuR, slotted WuR, and BoWuR.

each transmission. Consequently, $E[D]$ becomes longer. With the same traffic load, the larger the packet size, the longer the delay since it takes longer time to transmit a larger packet.

Furthermore, the collision probabilities are compared in Fig. A.5. For WuR or slotted WuR, a collision occurs when two or more nodes access the channel within a period of two slot durations or one slot duration respectively, and this collision probability increases rapidly with a higher traffic load. Consequently, the saturation points appear very early as traffic load grows, meaning that WuR and slotted WuR are inefficient to handle a large amount of traffic. On the other hand, BoWuR performs better than its two counterparts since it adopts a BO mechanism to avoid collision. In BoWuR, P_c depends not only on the number of competing nodes but also on the CW size. The larger the CW size, the lower the P_c.

Finally, Fig. A.6 illustrates the obtained energy efficiency for the reference node as the traffic load varies. For all studied protocols, η decreases with a heavier traffic load. As a node experiences a higher collision probability in a larger network, it consumes higher energy to transmit a packet successfully compared with in a small network. From this figure, it is clear that the energy efficiency of BoWuR is much higher than that of its counterparts thanks to its much lower collision probability. Furthermore, higher η is achieved with a larger size data packet. This is due to the fact that when a node obtains a transmission opportunity, more bits can be transmitted with a larger size packet, resulting in higher energy efficiency.

VI. Conclusions

In this letter, we advocate to activate a backoff procedure with a fixed contention window size before initiating WuCs for SCM-WuR enabled WSNs. Through DTMC modeling, we demonstrate the benefits of adopting such a BO procedure for WuCs, i.e., higher network throughput, lower collision probability, shorter delay, and higher energy efficiency. Although BoWuR is targeted at SCM-WuR, the same principle applies to other WuR implementations as well.

Paper A: References

[1] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE Trans. Netw.*, vol. 24, no. 2, pp. 674-687, Apr. 2016.

[2] ———, "Performance evaluation and comparative analysis of subcarrier modulation wake-up radio systems for energy-efficient wireless sensor networks," *Sensors*, vol. 14, no. 1, pp. 22-51, 2014.

[3] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," *in Proc. IEEE INFOCOM, 2015*, pp. 522-530.

[4] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. Popovici, and L. Benini, "Design, implementation, and performance evaluation of a flexible low-Latency nanowatt wake-up radio receiver," *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 633-644, Apr. 2016.

[5] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Select. Areas Commun.*, vol. 18, no. 3, pp. 535-547, Mar. 2000.

[6] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas, "IEEE 802.11 packet delay - A finite retry limit analysis," *in Proc. IEEE GLOBECOM, 2003*, pp. 950-954.

[7] M. Ergen and P. Varaiya, "Decomposition of energy consumption in IEEE 802.11," *in Proc. IEEE ICC, 2007*, pp. 403-408.

[8] NXP Semiconductors "JN-AN-1001 JN516x power consumption," Application note, Jun. 2016.

[9] AMS "AS3932" Data sheet, Mar. 2015.

[10] A. S. Tanenbaum and D. J. Wetherall, *Computer Networks*, 5th Edition, Boston, MA, USA: Pearson, 2011.

Paper B

Title:	MAC Protocols for Wake-up Radio: Principles, Modeling and Performance Analysis
Authors:	Debasish Ghose[†], Frank Y. Li[†], and Vicent Pla[‡]
Affiliation:	[†]Dept. of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway [‡]Dept. of Communications, Universitat Politècnica de València (UPV), 46022 València, Spain
Journal:	*IEEE Transactions on Industrial Informatics*, vol. 14, no. 5, pp. 2294-2306, May 2018.
DOI:	10.1109/TII.2018.2805321.
Copyright ©:	IEEE

B

B

MAC Protocols for Wake-up Radio: Principles, Modeling and Performance Analysis

Debasish Ghose, Frank Y. Li, and Vicent Pla

Abstract — **In wake-up radio (WuR) enabled wireless sensor networks (WSNs), a node triggers a data communication at any time instant by sending a wake-up call (WuC) in an on-demand manner. Such wake-up operations eliminate idle listening and overhearing burden for energy consumption in duty-cycled WSNs. Although WuR exhibits its superiority for light traffic, it is inefficient to handle high traffic load in a network. This paper makes an effort towards improving the performance of WuR under diverse load conditions with a twofold contribution. We first propose three protocols that support variable traffic loads by enabling respectively clear channel assessment (CCA), backoff plus CCA, and adaptive WuC transmissions. These protocols provide various options for achieving reliable data transmission, low latency, and energy efficiency for ultra-low power consumption applications. Then, we develop an analytical framework based on an M/G/1/2 queue to evaluate the performance of these WuR protocols. Discrete-event simulations validate the accuracy of the analytical models.**

Keywords—IoT/WSNs, energy-efficient communication, WuR, MAC protocol, modeling and performance evaluation.

B

I. Introduction

The global Internet is shifting rapidly from connected computers to connected small devices, i.e., the Internet of Things (IoT). As one of the key enabling technologies, wireless sensor networks (WSNs) form an integral part of the IoT thanks to their low-power consumption and rapid deployment features. In many IoT applications, the energy-efficiency of IoT devices, e.g., battery-powered sensor nodes, is of paramount importance. Traditionally, duty-cycle medium access control (MAC) has been adopted in WSNs to reduce energy consumption by letting nodes sleep and wake up cyclically. However, duty-cycle MAC mechanisms suffer from idle listening and overhearing during their on-states. While idle listening occurs when a

Manuscript received September 28, 2017; revised December 14, 2017; accepted January 21, 2018. Date of publication February 12, 2018; date of current version May 2, 2018. Paper no. TII-17-2251. The research of V. Pla was supported in part by grant number TIN2013-47272-C2-1-R from the Spanish Ministry of Economy, Industry and Competitiveness.

node listens to the channel for receiving control messages, overhearing occurs when a node overhears the control messages which are not intended to it. Although some of the proposed duty-cycle MAC protocols (e.g., PW-MAC [1] and RI-MAC [2]) are able to mitigate these problems from the transmitter end, idle listening cannot be completely eliminated from the receiver end since a receiver has to listen to the medium for receiving beacon messages.

In recent years, a paradigm shift from duty-cycled WSNs to wake-up radio (WuR) has been envisaged. A WuR provides energy-efficient communication by diminishing overhearing and idle listening thanks to its superior energy performance. In a WuR enabled sensor node, an additional wake-up receiver (WuRx) is attached to the micro-controller unit (MCU) of the node to detect wake-up calls (WuCs). As demonstrated in [5] [6], the power consumption of a WuRx is 1000 times lower than that of the main radio, i.e., at the μW level versus mW for the main radio. Upon detection/reception of such a WuC sent by a transmitter, the WuRx triggers the MCU of the sensor node to wake up its main radio from the sleep mode. When the main radio is activated, data communication is performed according to the adopted MAC protocol. A WuR may function either in a transmitter-initiated or a receiver-initiated mode. Which mode suits better depends on the application requirements of a WuR based WSN. In general, receiver-initiated suits better for *data collection* whereas transmitter-initiated is more appropriate for event-triggered *data reporting*. The superiority of WuR over duty-cycle MACs was demonstrated in [5]. One of the primary advantages of WuR is that it works in an on-demand manner. Such an on-demand communication does not only increase energy efficiency but also reduces latency of data transmissions [5] [6]. In addition, the address decoding and matching abilities of WuRx exhibit its proficiency to process the received data in an energy-efficient way [7] [8]. That means, a WuRx wakes up its associated MCU only when the received WuC address has been validated, thus diminishing potential false wake-up and reducing energy consumption due to overhearing in the network.

In its initial form, WuR is designed for low traffic WSNs regardless of application scenarios. A typical application of WSNs is environment surveillance. Over time, applications have been expanded to more diverse areas, such as health monitoring, smart home and smart city, industrial automation, and potentially in WiFi. In certain scenarios, the offered traffic load in a WuR WSN could become very high due to suddenly increased packet generation rates. Despite enormous energy saving, timely reporting and low packet loss are essential if an abnormal event/behavior is observed. As a motivating example, consider a distributed WuR WSN deployed to monitor an industrial environment, such as a goods storage warehouse or a shipping harbor, and it is operated in the transmitter-initiated mode. If multiple nodes detect an abnormality, such as a fire, and report it at the same time, collisions may happen. Then heavy packet loss occurs since there is no mechanism adopted in WuR for collision avoidance among WuCs. On the other hand, the traffic pattern of a WuR WSN may vary from time to time [10]. It is shown in [9] that WuR loses its superiority when traffic load is high. So far very few protocols which deal with WuC collisions can be found in the literature. Hence, it is of essential interest to reconsider WuR design and propose MAC protocols by taking traffic load into account. The aforementioned observations triggered our motivation to propose new WuR protocols in order to improve the performance of WuR under various traffic load conditions.

One essential reason for WuC collisions is that no clear channel assessment (CCA) before WuC transmission is performed for on-demand WuRs. In this paper, we propose three CCA enabled WuR protocols to improve the performance of WuRs. Furthermore, we develop a queuing model to evaluate the performance of WuR protocols and validate the accuracy of the model through extensive discrete-event simulations.

The main contributions of this work are as follows:

- Three MAC protocols are proposed for WuR-enabled WSNs: CCA-WuR, carrier sense multiple access (CSMA)-WuR and adaptive (ADP)-WuR.
- To model the behavior of the CCA-WuR, CSMA-WuR and ADP-WuR protocols, we develop a generic analytical framework based on an M/G/1/2 queue. The accuracy of the model is validated through discrete-event simulations.
- Closed-form expressions for calculating WuC loss probability, energy consumption, and latency are obtained based on the proposed analytical framework.
- To further improve the performance of WuR, two techniques are proposed. These techniques are targeted at increasing WuC transmission data rate and shortening WuRx addresses.

The remainder of this paper is structured as follows. In Section II, we summarize the related work and highlight the differences between our work and existing work. Section III presents the network scenario and assumptions. In Section IV, we present the three proposed protocols: CCA-WuR, CSMA-WuR and ADP-WuR. Section V presents the generic analytical framework for WuR protocols in details. Then the performance metrics are analyzed in Section VI. Section VII introduces the proposed techniques to improve the performance of WuR. Numerical results are presented in Section VIII, before the paper is concluded in Section IX.

II. Related work

The number of MAC protocols for WuR is growing in the literature. Two comprehensive surveys on the state-of-the-art WuR hardware, networking and MAC protocols were presented in [15] [16]. In brief, existing work falls into three main categories: 1) WuR circuit design; 2) WuR protocol design; and 3) performance evaluation of WuR. However, many studies cover two categories, e.g., circuit or protocol design plus performance evaluation.

A. WuR Circuit Design

The subcarrier modulation based correlator WuR [4] is one of the most popular inband WuR circuitry designs which uses a single channel to transmit both WuC and data. One of the key features of correlator WuR is that it can reach up to 100 meters which is the highest WuR transmission range among the designs that have been reported in the literature. ALBA-WUR [6] is another WuR which focuses on low-power consumption of WuRx. The authors presented the circuit design of ALBA-WUR and showed the effect of the data rate of WuC on the achieved transmission range. An ultra-low power, lower than 1 μW, WuR was reported in [7]. Its implementation can be adapted to different frequencies in the ISM band.

B. WuR Protocol Design

DoRa/DC-DoRa [21] is a WuR protocol which operates over two radios. For data transmission, the sink periodically sends a WuC addressed to each node, and the subsequent data transmission is performed on a separate channel. However, such a periodic polling mechanism increases transmission cost at the sink node. CMAC [22] is another multi-channel WuR protocol that uses a separate channel to send WuC and follows backoff (BO) before WuC transmission. In [20] the authors studied multi-hop WuR networks and proposed a protocol known as OPWUM. A node in OPWUM opportunistically selects the best relay node among its neighbors based on a given metric to resolve undesired neighborhood wake-up. For a similar reason, ZeroMAC [18] utilizes a radio frequency (RF) watchdog to wake up only the nodes on the communication path by sending unaddressed WuC in a hop-by-hop manner. RTM [19] and BoWuR [17] are two CSMA/collision avoidance (CA) alike WuR schemes which enable CCA plus BO before a WuC transmission.

Table B.1: Comparative Analysis of Our Protocols with the State-of-the-art WuR Protocols

Metric/Features	CCA-WuR [Ours]	CSMA-WuR [Ours]	ADPWuR [Ours]	DoRa/DC-DoRa [21]	OPWUM [20]	ALBA [6]	BoWuR [17]	Cor-WuR [5]
Network scenario	Single-hop	Single-hop	Single-hop	Single-hop	Multi-hop	Single-hop	Single-hop	Single- and multi-hop
Netw. sync. before WuC trans.	No	No	No	Yes	Yes	No	Yes	No
Multi-node competitions	Yes	Yes	Yes	No	No	No	Yes	No
Theoretical analysis	Yes	Yes	Yes	No	No	No	Yes	No
Simulation/testbed	Simulation	Simulation	Simulation	Simulation	Simulation	Testbed	No	Simulation
Single/dual radio	Single	Single	Single	Dual	Single	Single	Single	Single
Communication mode	TI	TI	TI	RI	TI and RI	—	TI	TI and RI
BO for WuC	No	Yes	Adaptive	No	Yes	No	Yes	No

B

C. Performance Evaluation of WuR

The superiority of Cor-WuR over duty-cycle MACs was demonstrated based on discrete-event simulations in [5]. However, the authors did not consider potential collisions for their performance evaluation. OPWUM [20] was validated using computer simulations. Extensive simulations were performed to validate the DoRa/DC-DoRa protocols [21], showing their performance improvement over IEEE 802.15.4 and duty-cycle MAC protocols. Furthermore, ALBA-WUR was validated using a testbed. Its benefit over duty-cycle MAC was demonstrated based on a point-to-point topology using selective awakening. Both an analytical model and a testbed validation of WuR were presented in [8]. For their performance evaluation, the authors adopted a model based on an absorbing Markov chain. However, how to deal with WuC collisions is not reflected in that paper. The BoWuR [17] protocol was analyzed using a discrete time Markov chain and its performance improvement over WuR was shown under saturated traffic conditions.

In Table B.1, we summarize the major differences between our protocols and a few other state-of-the-art WuR protocols. None of the WuR protocols that have been reported in the literature considered purely asynchronous networks for tackling WuC collisions. Rather, the existing WuR protocols either perform network synchronization before WuC transmissions, adopt a synchronous mode, or employ CSMA/CA (IEEE 802.11) MAC to handle WuC collisions. For instance, multiple nodes in a BoWuR based network count down their BO counters from the same time instant, the same as what is used in WiFi where multiple stations start counting down after a distributed coordination function (DCF) interframe space. Recall that WuR works in an on-demand and purely asynchronous manner and the MCU of a WuR-enabled node only wakes up when it generates a packet. The clusterhead does not adopt any mechanism to keep track of such an on-demand wake-up pattern. Therefore, it is infeasible to exchange periodic/aperiodic beacons for maintaining network synchronization, especially in transmitter-initiated mode which is the focus of this work. In this paper, we propose three MAC protocols which are tailored to WuR-enabled WSNs/IoT networks operated in an asynchronous mode and develop an analytical framework to evaluate the performance of these protocols.

III. NETWORK SCENARIO AND WUR DESCRIPTION

In this section, we first describe the network scenario and assumptions and then present the design of a reference WuR.

A. Network Scenario and Assumptions

Star and tree topologies are two popular network topologies that are used for environmental surveillance in typical battery-powered WSN applications. In such a network, sensor nodes monitor the environment in a specific sensing area and transmit their measured/monitored data towards one common destination, the sink or clusterhead. In this study, we consider an *event-triggered data reporting* WSN with a star topology since such a scenario is more prone to transmission collisions. The sensor nodes are WuR-enabled and are operated in the transmitter-initiated mode. Collisions occur if the transmissions of more than one node overlap with each other.

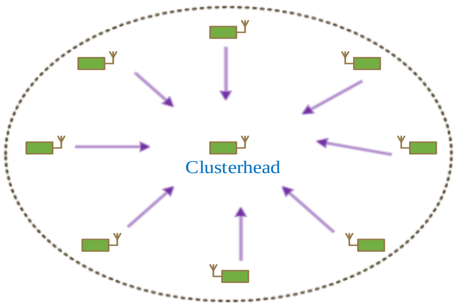


Figure B.1: A WSN with multiple contending nodes and a sink.

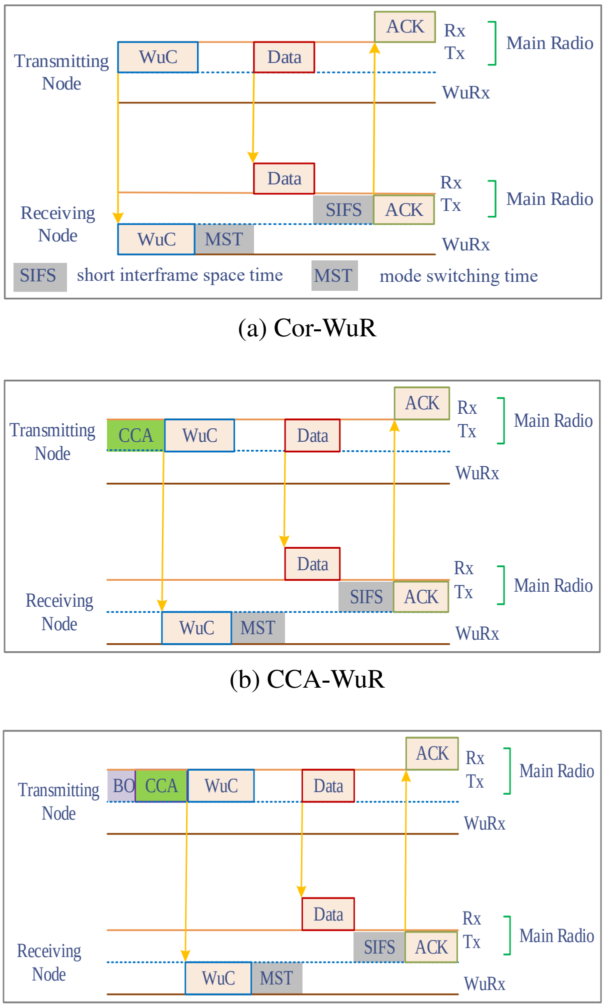
Under such a scenario, consider a network cluster consisting of $N + 1$ sensor nodes including one clusterhead and N member nodes, as shown in Fig. B.1. The N member nodes compete with each other *in an asynchronous mode* for data reporting towards the clusterhead over a single hop. Each node has a finite queue capacity and is equipped with a WuR transceiver in addition to its main radio.

Assume that at each node (except the clusterhead) packets are generated based on a Poisson process with an arrival rate of λ. The channel is considered to be error-free and no hidden terminal exists in this cluster.

B. Description of the Reference WuR Prototype

The reference WuR design considered in this study is based on the subcarrier modulation WuR introduced in [3]. Its design details can be found in [3] [4]. A CC1101 [25] chip, which operates in the 868 MHz ISM band and supports different modulation techniques including on-off keying, is used as the main radio of a node. To extend its transmission range, an *additional* CC1190 RF front-end [4] is integrated at the transmitter, reaching a transmission power level of +20 dBm. A 125 kHz wake-up signal is modulated using on-off keying modulation on the 868

B



(a) Cor-WuR

(b) CCA-WuR

(c) CSMA-WuR

Figure B.2: Illustration of the Cor-WuR, CCA-WuR, and CSMA-WuR protocols.

MHz carrier frequency at the transmitter. At the transmitter side, there is only one channel through which both data and WuC are transmitted from the main radio by switching the antenna alternatively.

At the receiver side, a WuRx is built based on an off-the-shelf low-frequency integrated circuit, AS3932 [27], that works at 125 kHz with address decoding and matching capabilities. An on-off keying modulated WuC is demodulated at the receiver using a Schottky diode, followed by a low-pass filter. Afterward only the envelope signal is processed at the WuRx. Upon detecting a wake-up signal, it interrupts the MCU to switch from the sleep to the active mode. The WuC frame structure is based on AS3932 [27], consisting of a carrier burst, a preamble (0101010.... ON/OFF modulated carrier), and a 16-bit address.

IV. Transmission Protocols for WuR

In this section, we present the design and working principle of a reference WuR and then propose three MAC protocols with a focus on collision avoidance among WuCs.

A. Correlator-WuR (Cor-WuR): A Benchmark Protocol

The reference WuR protocol considered in this paper is Cor-WuR, also refereed to as SCM-WuR in [5], [11] and [17]. It can operate either in the transmitter-initiated or the receiver-initiated mode. In what follows, we focus on a Cor-WuR based WSN operated in the transmitter-initiated mode. A data transmission cycle of the transmitter-initiated WuR lasts from the instant of a WuC initiation to the instant when an acknowledgment (ACK) is received, as illustrated in Fig. B.2(a). In the transmitter-initiated mode, a node which has a packet to transmit sends a WuC to the targeted node using its main radio. After receiving the WuC and decoding the address correctly, the targeted node switches on its main radio from the sleep mode for data communication. Right after a data exchange finishes, both transmitter and receiver switch off their main radios and go to sleep. However, both nodes keep their WuRxs active and continuously listen to the channel. It is worth mentioning that in Cor-WuR no ACK is sent upon the successful reception of the WuC [5] [11]. Neither does it exist any MAC mechanism for WuC transmissions. For the subsequent data transmission after receiving a WuC correctly, a MAC protocol, for instance CSMA/CA, could be adopted. Depending on the MAC protocol being adopted, the main radio of the sending node may or may not perform CCA or/and a BO procedure before a packet transmission.

In the rest of the section, we propose three MAC protocols for WuR to avoid WuC collisions and explain their principles.

B. CCA Enabled WuR (CCA-WuR)

As mentioned earlier, the existing WuR solutions do not perform carrier sensing prior to a WuC transmission. The main idea of a CCA enabled WuR is to make sure that no other node is transmitting a WuC before it starts sending its own WuC. Such a CCA mechanism reduces collision among WuCs. When an event is detected by a WuR integrated sensor node, it performs CCA first to check whether the channel is idle or not. If the WuR finds the channel idle for a CCA duration, it transmits the WuC; otherwise it performs CCA again. This procedure will continue until the maximum number of attempts is reached. If the maximum number of attempts is reached, the data frame will be discarded from the queue.

Since the WuC is sent through a wireless medium, all neighboring WuRxs receive the same WuC. After decoding the WuC frame, the targeted WuR, i.e., the clusterhead switches on its main radio for communication and the main radio of the other nodes will continue to sleep. When both transmitting and receiving main radios are switched on, data exchange starts immediately. The transmission cycle ends up with an ACK from the receiving main radio to the transmitting main radio. At the end of each transmission cycle, the main radios of both nodes go to sleep but their WuRxs are still actively listening to the channel. The principle of CCA-WuR is shown in Fig. B.2(b) for one data communication cycle.

C. CSMA-CA Enabled WuR (CSMA-WuR)

Although CCA-WuR is capable of handling light traffic load in WuR-based WSNs, performing CCA alone is not sufficient to eliminate collisions among WuCs during a high traffic load episode. To overcome the limitation of CCA-WuR, we further propose CSMA-WuR, a CSMA-CA enabled WuR MAC protocol. CSMA-WuR works similarly to the unslotted CSMA-CA MAC protocol of IEEE 802.15.4 but it is tailored to WuCs. It is worth mentioning that the main difference between the CSMA/CA and the CSMA-CA protocol is the order of performing CCA and BO. In CSMA/CA, *a node performs CCA first and then BO afterward, whereas the order is reversed in CSMA-CA*.

In CSMA-WuR, upon detecting an event by a node, it first performs a BO procedure without checking whether the channel is idle or not. As soon as the BO waiting time ends, it checks the channel status by performing a CCA. If it finds the channel idle for a duration of CCA, it sends a WuC. Otherwise it repeats the BO and CCA procedure again. Similar to CCA-WuR, this procedure will be repeated until the attempt limit has been reached. The rest of the CSMA-WuR operation is the same as in CCA-WuR. The whole operation procedure of the CSMA-WuR protocol is presented in Fig. B.2(c).

D. Adaptive WuR (ADP-WuR)

From the design principles of CCA-WuR and CSMA-WuR, it is clear that CCA-WuR and CSMA-WuR are well suited for light and heavy traffic respectively. However, traffic load varies over time and it might not be beneficial to always employ

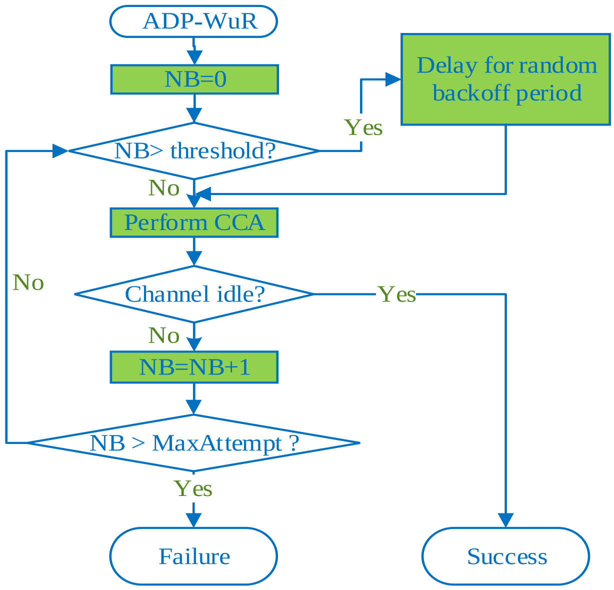


Figure B.3: The working procedure of the ADP-WuR protocol.

either CCA-WuR or CSMA-WuR *alone* under all traffic conditions. Therefore, we further propose an adaptive protocol which enables either CCA-WuR or CSMA-WuR adaptively based on traffic conditions. To do so, each node needs to keep track on its WuC transmission attempt counter and compares it with a pre-configured attempt threshold for MAC mechanism adaptation.

Fig. B.3 illustrates the working procedure of ADP-WuR, where NB stands for the BO stage. Considering a static network cluster, we set a pre-configured attempt threshold in this study. Initially, a node follows the CCA-WuR mechanism for a WuC transmission. If the transmission is unsuccessful up to the pre-configured threshold (i.e., NB), it switches to CSMA-WuR for its next attempts of the ongoing WuC transmission. Let us consider the attempt threshold as two. Whenever a node detects an event, it follows CCA-WuR first. After two unsuccessful attempts, the node adopts CSMA-WuR from its third attempt onward until the limit is reached.

Furthermore, the attempt threshold of a node may also be configured dynamically based on traffic conditions or/and transmission status. To do so, each node has to keep the statistics of its previous transmissions (e.g., WuC loss, successful packet delay, number of attempts required for previous successful WuC transmissions). Such a dynamic threshold provides more flexibility for MAC operations at the cost of high complexity. On the other hand, for event-triggered data reporting, congestion may arise abruptly and releases after a short period. In this case, the threshold adaptation scheme may not be able to keep up with the instantaneous traffic conditions and thus may not achieve the expected performance improvement.

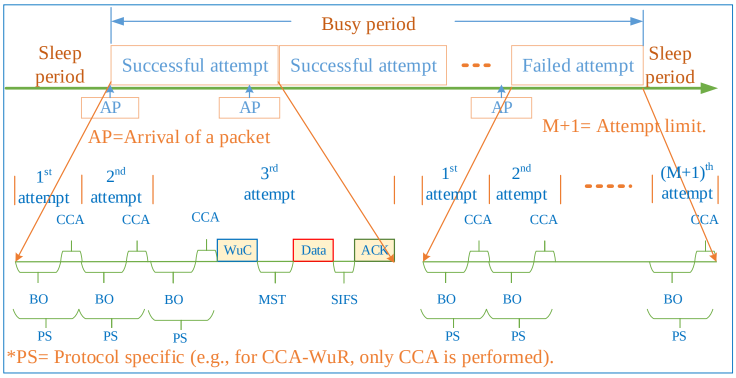


Figure B.4: Illustration of a busy period based on the proposed WuR protocols.

Fig. B.4 illustrates a generic busy period of CCA-WuR, CSMA-WuR and ADP-WuR. A busy period is defined as the period that begins with the arrival of a packet into an empty queue and ends when the queue becomes empty again. Note that all packets generated during a busy period will be processed (either successfully transmitted or discarded) in that busy period. In other words, a busy period may consist of multiple packets, with one or more transmission attempts per packet as shown in this figure. A successful attempt means that a packet is successfully delivered with an ACK received at the transmitter, whereas a failed attempt means that a packet is discarded after the retry limit has been reached. The lower half of this figure illustrates the behavior of the transmitter for a successful and a failed attempt (note that the detailed behavior within a transmission attempt is protocol dependent. As an example, CSMA-WuR which requires BO plus CCA is shown in Fig. B.4 for illustration purposes).

V. Modeling WuR protocols

The analytical framework for modeling WuR and the proposed protocols is presented in this section. This framework was initially inspired by the approach presented in [12], which was based on the analysis of a regenerative cycle of the M/G/1 queue. In [12], the authors considered BO first and CCA afterward and accordingly calculated packet loss probability based on a similar busy period definition presented therein. However, their model was based on an *infinite size queue* and assumed the same packet processing rate regardless of the BO period structure of the adopted MAC protocol. A consequence of their assumption is that the key performance metrics have the same value irrespectively of the BO periods of the protocol under study. Furthermore, Cor-WuR, which is the benchmark for our protocol design, does not perform BO or CCA prior to WuC transmissions. Hence, the presented model in [12] is not directly applicable for performance evaluation of Cor-WuR.

Considering a *finite size queue*, we develop below a generic framework for performance evaluation of both Cor-WuR and the proposed protocols based on an M/G/1/2 queuing model which captures the on-demand nature of the WuR operation. Unlike the M/G/1 model, our model considers that there can be at most two packets in the queue, i.e., the one at the head of the queue (HoL), whose transmission is underway, plus another one. This choice is motivated by the fact that sensors usually have small buffers, and also because a reduced buffer size model (e.g., buffer-less; or equivalent formulations) is a technique that has been adopted to extend Bianchi's model [13] to a non-saturated scenario (see, e.g., [14]). The validity and accuracy of this modeling approach will be verified through extensive computer simulations (see Sec. VIII). The notations of the analytical model are summarized in Table B.2.

A. Models for CCA-WuR, CSMA-WuR and ADP-WuR

In our model, we focus on a single representative device, referred to as the *tagged device*. The tagged device is modeled as an M/G/1/2 queue in which the service time of a packet represents the duration from the epoch that the packet arrived at the HoL to the epoch it is transmitted successfully with an ACK or it is discarded.

The analysis of the model revolves around the calculation of the probability of an unsuccessful attempt, i.e., the probability that the channel is sensed as busy right after a CCA. Denote this probability as α and assume that it is constant regardless of the attempt stage. A similar assumption is also adopted in many MAC protocol studies, including [13]. Based on this assumption, it follows that the service time is independent and identically distributed.

Using the same reasoning as in [12] we can write

$$\alpha = \frac{(N-1)(1-P_L)\,\mathrm{E}[\Gamma](T_{\mathrm{CCA}}+T_{\mathrm{TA}})}{1/\lambda + \mathrm{E}[\Gamma]\,\mathrm{E}[D_{\mathrm{HoL}}]}, \tag{B.1}$$

where Γ is the number of packets served (note that served includes both successfully transmitted and acknowledged, and discarded after the maximum number of attempts, $M+1$) in a busy period; P_L is the probability of a packet being discarded after $M+1$ unsuccessful attempts; $T_{\mathrm{TA}} = T_{\mathrm{wuc}} + T_{\mathrm{MST}} + T_{\mathrm{data}} + T_{\mathrm{SIFS}} + T_{\mathrm{ack}}$ is the total duration of a transmission attempt, including the reception of ACK; and D_{HoL} is the duration from the time a packet arrives at the HoL until right after the

Table B.2: Notations Used in the Analytical Framework

Notation	Description
N	Number of nodes in the network/network cluster
Q	Queue capacity of a node (in packets)
$M+1$	Maximum number of attempts incl. 1 initial transmission and M attempts
W_i	Contention window size in i^{th} contention stage
λ	Packet generation rate to a node
α	Probability that the channel is sensed as busy right after each CCA
$\mathrm{E}[\Gamma]$	Expected number of packets served in a busy period
P_S	Successful packet transmission probability
P_L	WuC loss probability
a_0	Probability that no packet arrives during the time a packet is at the HoL
w_k	Mean accumulated duration including BO and CCA for the k-th attempt
$\mathrm{E}[D_{\mathrm{HoL}}]$	Mean sojourn time of a packet at the HoL
T_{FA}	Duration of a failed transmission attempt
T_{SA}	Duration of a successful transmission attempt
$\mathrm{E}[T_{\mathrm{BP}}]$	Mean duration of a busy period
$\mathrm{E}[A]$	Expected number of transmission attempts per frame

last CCA for that packet (after the last CCA the packet will be either transmitted or discarded).

In what follows, we derive expressions for P_L, $\mathrm{E}[D_{\mathrm{HoL}}]$, and $\mathrm{E}[\Gamma]$, which in turn depend on α. Thus, when substituted into (B.1) a non-linear equation will be obtained from which the value of α can be solved numerically. Clearly,

$$P_L = \alpha^{M+1}. \tag{B.2}$$

Let W_i and σ denote, respectively, the contention window size at the i-th BO stage and the duration of a BO slot. The mean accumulated duration of the BO plus CCA until the k-th transmission attempt, w_k, is given by

$$w_k = \sum_{i=0}^{k-1} \frac{W_i - 1}{2}\sigma + kT_{\mathrm{CCA}}. \tag{B.3}$$

Then, the expected HoL delay can be obtained as

$$\mathrm{E}[D_{\mathrm{HoL}}] = \sum_{v=0}^{M} \alpha^v (1-\alpha) w_{v+1} + \alpha^{M+1} w_{M+1}. \tag{B.4}$$

The first part of (B.4) corresponds to a situation where the transmission, including WuC and data, is successful, whereas the second part corresponds to a situation where a WuC is discarded after $M+1$ unsuccessful CCA attempts.

At the time instant a packet reaches the HoL, it will be the only packet in the queue. Thus, if no packet arrives during its sojourn at the HoL, that packet will be the last one in a busy period (see Fig. B.4). Let us denote by a_0 the probability of this event, i.e, that no packet arrives during the time a packet is at the HoL. Then, the number of packets served in a busy period, Γ, follows a geometric distribution

$$\Pr(\Gamma = k) = (1-a_0)^{k-1}a_0, \qquad k = 1, 2, \dots, \tag{B.5}$$

and

$$\mathrm{E}[\Gamma] = \frac{1}{a_0}. \tag{B.6}$$

Let us now represent by S the sojourn time of a packet at the HoL, i.e., the time from its arrival at the HoL until it is successfully transmitted (including the reception of ACK) or discarded. Note that $S = D_{\text{HoL}}$ if the packet is discarded, and $S = D_{\text{HoL}} + T_{\text{TA}}$ if it is transmitted. Furthermore, let $f_S(t)$ denote the probability density function of S. Then, recalling that the packet generation process is Poisson, we can write

$$a_0 = \int_0^\infty f_S(t)e^{-\lambda t}\,dt = \mathcal{L}\{f_S\}(\lambda), \tag{B.7}$$

where $\mathcal{L}\{f_S\}(\lambda)$ denotes the Laplace transform of f_S evaluated at λ.

Using Laplace transforms it can be shown that (the details of this deduction are omitted due to the page limit)

$$a_0 = \sum_{v=0}^{M} \alpha^v(1-\alpha)H_{v+1}(\lambda)e^{-T_{\text{TA}}\lambda} + \alpha^{M+1}H_{M+1}(\lambda), \tag{B.8}$$

where

$$H_n(s) = \frac{e^{-nT_{\text{CCA}}s}}{(1-e^{-\sigma s})^n}\prod_{i=0}^{n-1}\frac{1-e^{-W_i\sigma s}}{W_i}. \tag{B.9}$$

Although the numerical evaluation of (B.8) does not pose a problem, we propose herein an alternative approach to obtain a_0, based on an approximation, that leads to a simpler derivation and expression. The approximation consists of substituting the random duration of each BO stage by its mean value. Using this approximation, it easily follows that

$$a_0 \approx \sum_{v=0}^{M} \alpha^v(1-\alpha)e^{-(w_{v+1}+T_{\text{TA}})\lambda} + \alpha^{M+1}e^{-w_{M+1}\lambda}. \tag{B.10}$$

As noted, substituting (B.2), (B.4), (B.6) (along with (B.8) or (B.10)) into (B.1) yields a non-linear equation from which the value of α can be solved numerically.

Next we particularize the coefficients w_k (defined in (B.3)) for each of the considered protocols and introduce a minor adaptation of the analysis for the case of WuR.

B. CCA-WuR

CCA-WuR performs CCA prior to a WuC transmission but not BO. This is equivalent to considering a contention window of size equal to 1 regardless of the attempt stage, i.e., $W_i = 2^0 = 1$ for $i = 0, 1, \ldots, M$. Then,

$$w_k = kT_{\text{CCA}}. \tag{B.11}$$

C. CSMA-WuR

Prior to WuC transmission, the CSMA-WuR protocol performs BO and then CCA. In all attempt stages the random number of slots is selected uniformly from $\{0, 1, \ldots, W-1\}$, i.e., the size of the contention window is kept constant: $W_i = W$ for $i = 0, 1, \ldots, M$.

B

Hence, w_k can be calculated as

$$w_k = k\left(\frac{W-1}{2}\sigma + T_{\text{CCA}}\right). \tag{B.12}$$

D. ADP-WuR

The ADP-WuR protocol is a combination of the two previous ones. In the first t attempts it behaves as CCA-WuR, and as CSMA-WuR after that. That is,

$$W_i = \begin{cases} 1 & \text{if } i = 0, \ldots, t-1 \\ W & \text{if } i = t, \ldots, M+1, \end{cases} \tag{B.13}$$

and then

$$w_k = \begin{cases} kT_{\text{CCA}} & \text{if } k = 0, \ldots, t-1 \\ (k-t+1)\frac{W-1}{2}\sigma + kT_{\text{CCA}} & \text{if } i = t, \ldots, M+1. \end{cases} \tag{B.14}$$

E. Model for Cor-WuR

According to Cor-WuR, a node sends a WuC to the intended WuRx whenever it generates a data packet. Unlike in the above analyzed protocols, collisions with the transmission of other devices are more likely to occur since no CCA is performed. Thus, it is essential to investigate the collision probability.

Let us denote by α the collision probability. As before, α represents the probability of an unsuccessful transmission attempt, but now the transmission (of the WuC and the data frame that follows) actually occurs and the failure is detected when the ACK timer times out.

After a failed attempt, a device starts over the transmission process until the maximum number of attempts, $M+1$, has been reached. If the transmission could not be completed successfully after the maximum number of attempts, the device simply discards that data packet. The illustration of a frame transmission is presented in Fig. B.5, where $T_{\text{FA}} = T_{\text{wuc}} + T_{\text{MST}} + T_{\text{data}} + T_{\text{SIFS}}$ (respectively, $T_{\text{SA}} = T_{\text{TA}}$) is the duration of a failed (respectively, successful) transmission attempt.

A transmission attempt of the tagged device will be successful if no busy periods of the other $N-1$ devices overlap with its. For this to occur, none of the other $N-1$ devices should start a busy period during the vulnerable period of the transmission of the tagged device. For simplicity, we assume that all busy periods have a constant duration equal to its mean, $\mathrm{E}[T_{\text{BP}}]$. With this assumption, an approximation for the collision probability can be obtained as follows

$$1-\alpha \approx \left(e^{-\frac{\lambda}{\mathrm{E}[\Gamma]}(\mathrm{E}[\Gamma]\,\mathrm{E}[A]T_{\text{FA}}+T_{\text{SA}})}\right)^{N-1}. \tag{B.15}$$

In this approximation, it is also assumed that busy periods occur according to a Poisson process with rate $\lambda/\,\mathrm{E}[\Gamma]$.

The mean duration of a busy period is given as

$$\mathrm{E}[T_{\text{BP}}] = \mathrm{E}[\Gamma]\,\mathrm{E}[A]T_{\text{FA}}, \tag{B.16}$$

where A is the number of transmission attempts per frame, and its mean value is obtained as follows

$$\mathrm{E}[A] = \sum_{i=0}^{M}\alpha^i(1-\alpha)(i+1)+(M+1)\alpha^M = \frac{1-\alpha^{M+2}}{1-\alpha}+(M+1)\alpha^M-(M+2)\alpha^{M+1}. \tag{B.17}$$

Since the WuC is sent without a prior BO or CCA, the probability that no packet arrives during a packet sojourn at the HoL can be approximated as

$$a_0 \approx \sum_{v=0}^{M}\alpha^v(1-\alpha)e^{-(vT_{\text{FA}}+T_{\text{SA}})\lambda} + \alpha^{M+1}e^{-(M+1)T_{\text{FA}}\lambda}. \tag{B.18}$$

As no retransmission (after a failed attempt) is considered in Cor-WuR, (B.18)

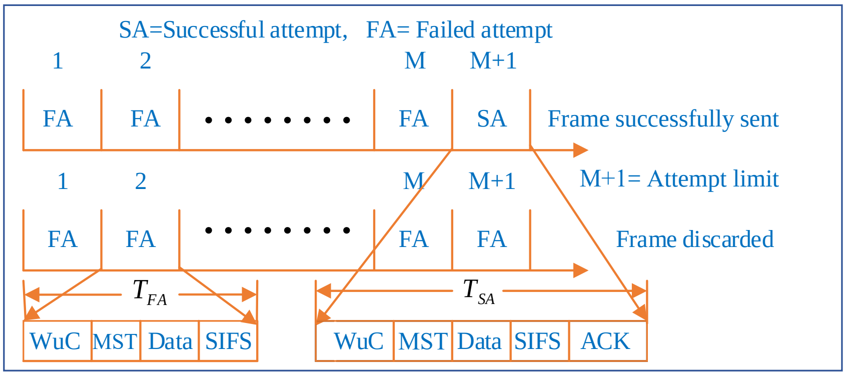


Figure B.5: A transmission cycle of Cor-WuR.

is simplified by setting $M = 0$

$$a_0 \approx (1-\alpha)e^{-T_{\text{SA}}\lambda} + \alpha e^{-T_{\text{FA}}\lambda}, \tag{B.19}$$

which can be further simplified if we assume that $T_{\text{TA}} = T_{\text{SA}} \approx T_{\text{FA}}$:

$$a_0 \approx e^{-T_{\text{TA}}\lambda}. \tag{B.20}$$

Similarly, $\text{E}[A] = 1$. Substituting these simplified equations into (B.15) yields

$$\alpha \approx 1 - e^{-(N-1)\lambda T_{\text{SA}}(1+e^{-T_{\text{TA}}\lambda})}. \tag{B.21}$$

VI. Performance Analysis

To evaluate the performance of the WuR MAC protocols, we define three performance metrics and derive their expressions based on the obtained unsuccessful attempt probability.

A. WuC Loss Probability

The WuC loss probability, denoted by P_L, is defined as the probability of a packet being discarded after $M+1$ unsuccessful attempts. The WuC loss probability has already been introduced in Sec. V.A and its general expression is given in (B.2). Note, however, that the mean accumulated duration including BO and CCA for the k-th attempt, w_k, which is needed for P_L calculation, is protocol specific.

Since Cor-WuR discards a packet after one unsuccessful attempt, P_L can be obtained directly using (B.21), i.e., $P_L = \alpha$.

B. Average Packet Delay

The average delay, denoted by T_S, is defined as the duration from the time a packet arrives at the HoL until it is successfully transmitted or discarded. This means that the delay experienced by an unsuccessful packet before it is discarded has been counted in the expressions derived below. Accordingly, T_S can be calculated as

$$T_S = (1 - P_L)T_t + P_L T_L, \tag{B.22}$$

where T_t and T_L are the mean delays from the time a packet arrives at the HoL until it is successfully transmitted (including the reception of ACK) and lost respectively. T_L is given as

$$T_L = \sum_{i=0}^{M} \frac{W_i - 1}{2}\sigma + (M+1)T_{\text{CCA}}.$$

The mean HoL delay can be written as $\mathrm{E}[D_{\text{HoL}}] = P_L T_L + (1-P_L)(T_t - T_{\text{TA}})$, where the $(T_t - T_{\text{TA}})$ term corresponds to the HoL delay for a successfully transmitted packet. Now, T_t can be obtained as

$$T_t = \frac{\mathrm{E}[D_{\text{HoL}}] - P_L T_L}{1 - P_L} + T_{\text{TA}}.$$

For a given protocol, $\mathrm{E}[D_{\text{HoL}}]$ and T_L can be formulated based on the protocol behavior (e.g., for CCA-WuR, we have $T_L = (M+1)T_{\text{CCA}}$).

The Cor-WuR protocol does not perform CCA or BO. Correspondingly, we have $T_S = P_L T_{\text{FA}} + (1 - P_L)T_{\text{SA}}$.

C. Energy Consumption

The energy consumption is quantified by E_S (in Joule), which represents the average energy consumed by the tagged device when it attempts to transmit a packet. This attempt can result in a successful transmission or a failed one. Let E_{wuc}, E_{data}, E_{ack}, E_{CCA}, E_{SIFS}, E_{idle}, E_{bo}, and E_{MST} be respectively the energy consumptions for WuC transmission, data packet transmission, ACK reception, CCA, SIFS, idle slot, BO, and switching on the MCU. Then E_S is given as

$$E_S = (1 - P_L)E_t + P_L E_L, \tag{B.23}$$

where E_t and E_L are the energy consumed for a successful and lost packet respec-

B

tively, and E_L can be calculated as

$$E_L = \sum_{i=0}^{M} \frac{W_i - 1}{2} E_{\text{bo}} + (M+1) E_{\text{CCA}}.$$

The average energy consumed by a packet during BO and CCA, denoted as E_{HoL}, can be formulated as

$$E_{\text{HoL}} = P_L E_L + (1 - P_L)(E_t - E_{\text{TA}}), \tag{B.24}$$

where $E_{\text{TA}} = E_{\text{wuc}} + E_{\text{MST}} + E_{\text{data}} + E_{\text{SIFS}} + E_{\text{ack}}$ and $(E_t - E_{\text{TA}})$ is the energy consumed by a successful packet during BO and CCA. Now, E_t can be calculated as

$$E_t = \frac{E_{\text{HoL}} - P_L E_L}{1 - P_L} + E_{\text{TA}}, \tag{B.25}$$

where

$$E_{\text{HoL}} = \sum_{v=0}^{M} \alpha^v (1-\alpha) \sum_{i=0}^{M} \frac{W_i - 1}{2} E_{\text{bo}} + (M+1) E_{\text{CCA}} + \alpha^{M+1} E_L.$$

Note that E_L and E_{HoL} depend on the behavior of each specific protocol (e.g., for CCA-WuR, $E_L = (M+1)E_{\text{CCA}}$ and $E_{\text{HoL}} = (M+1)E_{\text{CCA}} + \alpha^{M+1} E_L$).

As mentioned earlier, Cor-WuR does not perform CCA or BO. Therefore, $E_S = P_L E_{\text{FA}} + (1 - P_L) E_{\text{SA}}$ where $E_{\text{SA}} = E_{\text{TA}}$ and $E_{\text{FA}} = E_{\text{wuc}} + E_{\text{MST}} + E_{\text{data}} + E_{\text{SIFS}}$.

VII. Effect of Data Rate and Address Scheme

In this section, we present two techniques for performance improvement of WuR protocols obtained from our MAC protocol design experience.

A WuC consists of a carrier burst, preamble, and 16-bit address scheme. Since a WuC is modulated using on-off keying and transmitted at a lower data rate, the duration of a WuC is indeed longer than that of a data packet. For example, the time needed to transmit a data packet with a size of 35 bytes at a data rate of 250 kbps is 1.12 ms [12], whereas the WuC transmission time is 12.2 ms [5]. Due to such a long WuC transmission, the busy period increases, resulting in a reduced successful packet transmission probability. One of the effective ways to shorten WuC duration is to increase the date rate for WuC transmissions. It is worth mentioning that the benchmark Cor-WuR [5] adopted 2730 bps for its WuC transmission. We recommend therefore to increase the data rate of WuC transmissions, to, e.g., 5460 bps.

Table B.3: Parameter Configuration [5] [11] [17] [26] [27]

Radio type	Parameter	Value	Unit
Common	Supply voltage	3	V
Main radio	Data rate	250	kbps
	Transmission current	17.4	mA
	Reception current	18.8	mA
	Idle current	20	μA
	SIFS duration	192	μs
	Payload size	35	bytes
	ACK frame size	11	bytes
Wake-up radio	WuC duration	6, 12.2	ms
	WuC Transmission current	152	mA
	Reception current (WuRx)	8	μA
	Sleep current	3.5	μA
	BO current	5.16	mA
	CCA current	20.28	mA
	MCU switching current	2.7	μA
	Time to switch on MCU	1.79	ms
	CCA duration	1.92	ms
	Slot time	320	μs
	Contention window size	32, 64	slots
	WuC packet size	4, 2	bytes
	Maximum WuC attempts	7	times

However, a tradeoff between WuC duration and coverage needs to be considered before deciding which data rate to adopt.

Another point for performance improvement is to apply a shorter address scheme. The IEEE 802.15.4 standard adopts a 16-bit address scheme to cover up to 2^{16} nodes. However, such a network size is not common for small or medium size WuR enabled WSNs since it increases WuC delay and energy cost significantly. With a shorter address, the busy period shrinks and the successful WuC transmission probability increases. For a network cluster composed of up to 256 nodes, we would configure the address length to 8 bits.

VIII. Numerical results and discussions

In this section, we provide numerical results to evaluate the performance of the WuR protocols. The accuracy of the analytical model has been verified by discrete-event simulations. In all cases, the discrepancy between the analytical results and those obtained by simulations was below 2%. Note that *ana* and *sim* in these figures indicate the results obtained from analytical expressions and simulations respectively.

Consider a WuR WSN as shown in Fig. 1, with an average packet arrival rate of $\lambda = 10$ packets/s and variable traffic load represented by different number of nodes, $N \in \{10, 15, ..., 30\}$, in the network. The remaining parameters are configured based on the specifications listed in Table B.3.

To perform computer simulations, we made a custom-built discrete-event simulator. The developed simulator mimics the behavior of the studied WuR MAC protocols according to the principle of each protocol presented above. That is, a WuR-enabled node wakes up upon the arrival of a packet into the queue and maintains a timestamp to keep track the time duration of every single state transition. The sending node follows the procedure of the adopted MAC protocol, i.e., Cor-WuR, CCA-WuR, CSMA-WuR, or ADP-WuR prior to each WuC transmission. Upon receiving a WuC, the clusterhead, which is the only targeted receiver in our scenario, decodes and validates the address of the WuC. If the decoded address matches its own, it switches on its main radio for data communication. The timestamp for each packet is maintained until the packet has been successfully delivered or is discarded. Note that the simulation results are completely independent of the analytical model and expressions presented earlier in Section V.

A. WuC Loss Probability: A Comparison of Four Protocols

Fig. B.6 presents the P_L variation as the network size, N, varies. From this figure, it is evident that the analytical results match precisely the simulation results. With a large margin, all three proposed protocols outperform the Cor-WuR protocol in terms of WuC loss probability, P_L. The reason is that Cor-WuR does not have a defer mechanism for WuC transmissions. On the contrary, our proposed protocols enable CCA, or BO plus CCA, or adaptive mechanism prior to WuC transmissions, thus eliminating collisions. Therefore, data transmissions with higher reliability can be achieved based on the proposed protocols.

On the other hand, P_L increases with the number of nodes. When traffic load increases in the network, the duration of busy periods increases, leading to a lower successful WuC transmission probability. A similar trend applies to all the studied protocols as the network size grows. Among CCA-WuR, CSMA-WuR, and ADP-WuR, the latter two perform slightly better than CCA-WuR, thanks to the additional BO procedure. The CSMA-WuR and ADP-WuR show nearly similar performance all the studied network size. This is due to the fact that the ADP-WuR performs BO

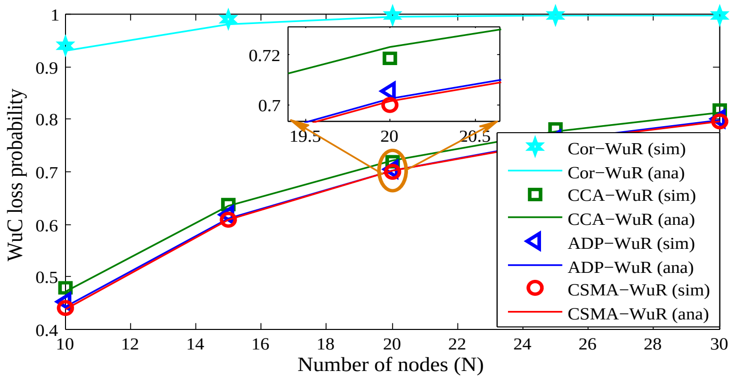


Figure B.6: WuC loss probability comparison of Cor-WuR, CCA-WuR, CSMA-WuR, and ADP-WuR with the WuC duration as 12.2 ms.

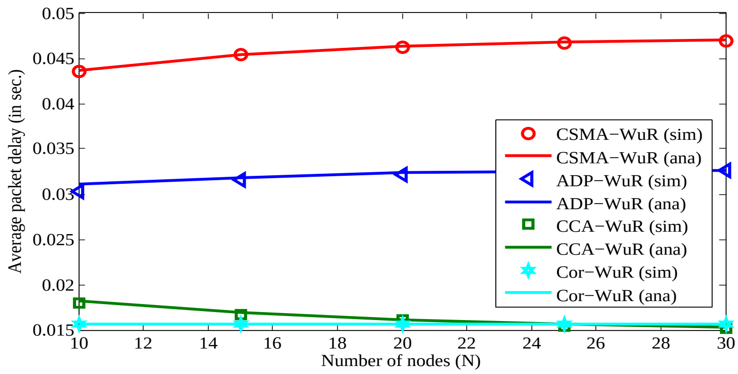


Figure B.7: Average packet (successful/discarded) delay comparison of Cor-WuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

in addition to CCA when the transmission when pre-configured threshold exceeds.

B. Average Packet Delay

Fig. B.7 illustrates how the average delay for a successful or discarded packet varies with different network sizes. From the figure it is clear that T_S of Cor-WuR is the shortest one among all four protocols and it is constant regardless of the network size. This behavior is mainly due to the fact that Cor-WuR neither performs CCA or BO prior to a WuC transmission nor does it allow any retransmissions. On the contrary, our proposed protocols perform CCA or CCA plus BO before WuC transmissions for both initial and retry attempts. Such CCA or BO plus CCA and retry stages introduce extra delay for WuC transmissions. Among the proposed three protocols, the average packet delay of CCA-WuR is the shortest since it needs only CCA before a WuC transmission. Between CSMA-WuR and ADP-WuR, the latter one performs better since it performs BO (prior to CCA) only after the pre-configured attempt threshold is reached. On the other hand, the average packet

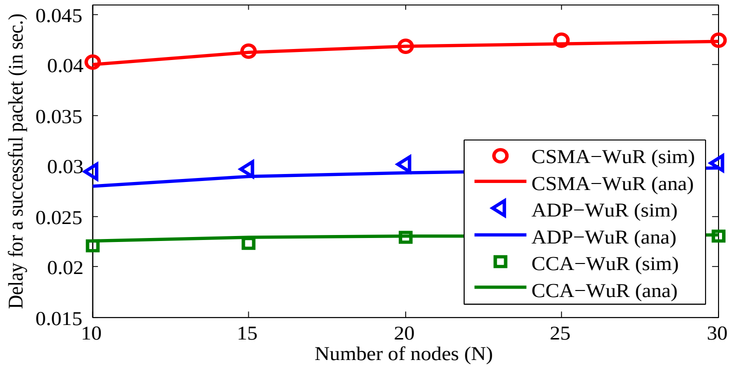


Figure B.8: Comparison of proposed protocols in terms of average delay for a successful packet when the WuC duration is 12.2 ms.

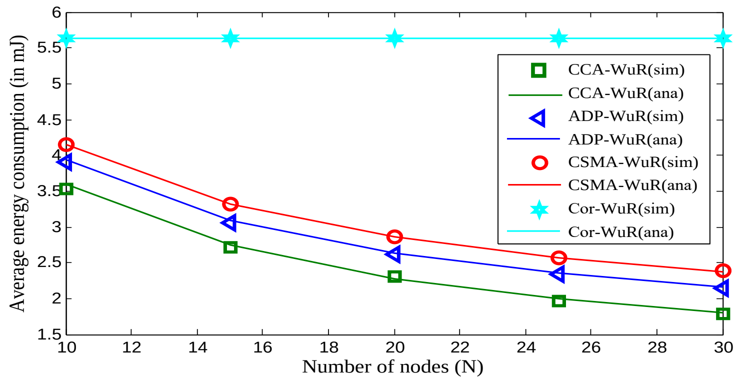


Figure B.9: Energy consumption comparison of Cor-WuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

B

delay, T_s, of CSMA-WuR and ADP-WuR becomes longer with a larger network size, whereas an opposite trend is observed for the delay of CCA-WuR. The reason is as follows. The CCA-WuR protocol keeps a packet in the queue for only up to 7 CCA intervals, (i.e., $7 \times T_{\text{CCA}} = 13.44$ ms) before discarding the packet and this duration is shorter than the duration of a successful packet transmission. More specifically, a successful transmission (even if it occurs after the first CCA, i.e., $T_{\text{CCA}} + T_{\text{TA}}$ = 17.382 ms where T_{CCA}= 1.92 ms and T_{TA}= 12.2 ms + 1.79 ms + (35+11)×8 bits/250 kbps = 15.462 ms) takes more time than discarding the packet after seven unsuccessful CCA attempts. As observed in Fig. 6, the proportion of packets that are discarded, P_L, increases with network size. As a consequence, the average packet delay decreases.

Fig. B.8 represents the average delay for transmitting a packet successfully based on the proposed protocols. It is observed that CCA-WuR performs better than the other two protocols since no BO is performed in CCA-WuR. When comparing the delay and loss probability performance of Cor-WuR versus the proposed protocols as shown in Fig. B.6 - Fig. B.8, we need to keep in mind that Cor-WuR, which achieves the shortest delay, does not allow retransmissions. This means that each generated packet will be transmitted only once, regardless of its transmission history. Therefore, employing Cor-WuR directly could lead to a disadvantage that no packets will be transmitted successfully when the network size is large.

C. Average Energy Consumption

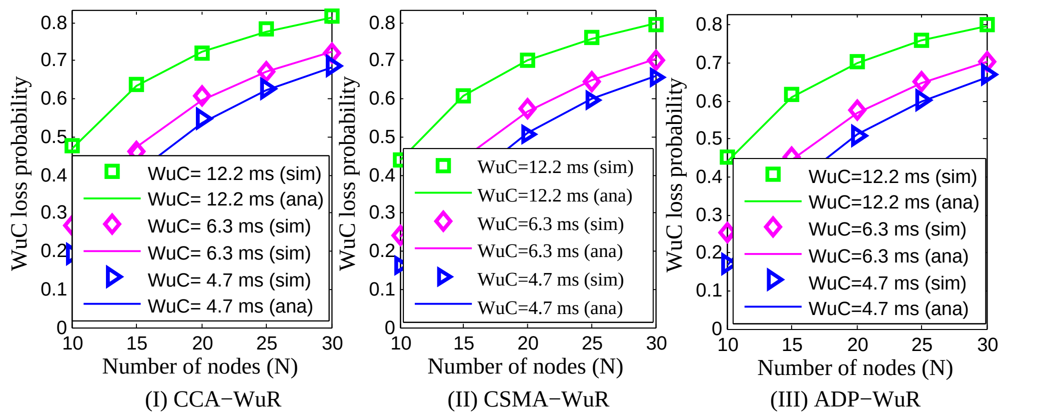
The average energy consumption for each WuC transmission (successful/unsuccessful) with various network sizes is illustrated in Fig. B.9 for these four protocols. As shown in the figure, the energy cost of all three proposed protocols is lower than that of Cor-WuR. This is obvious because a node needs to consume nearly an equal amount of energy for a lost or a successful WuC transmission when Cor-WuR is employed. On the other hand, a node that operates on one of our protocols consumes lower energy for the lost WuC transmission since it simply discards data packet after performing CCA or BO plus CCA when the attempt threshold has been reached. Among these three protocols, CCA-WuR consumes lowest energy since it performs merely CCA prior to WuC transmissions.

D. Tradeoff of Higher Bit Rate and Short Address

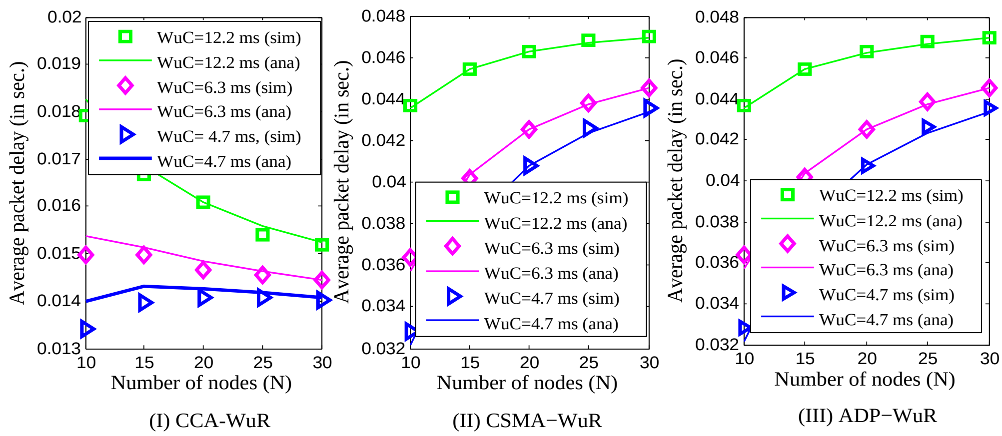
We now demonstrate the benefits of applying a shorter address and a higher data rate for WuC, the two techniques presented in Section VII. Although the numerical results are obtained based on an 8-bit address and a 5460 kbps WuC transmission data rate, these parameters are application dependent and are re-configurable.

Fig. B.10 illustrates the effects of these two techniques on the performance the three proposed protocols in terms of WuC loss probability, average packet delay and average energy consumption respectively. From Fig. B.10(a), it is evident that the WuC loss probability decreases with a higher WuC transmission data rate. This is because the WuC duration becomes shorter with a higher data rate. Similarly, the performance with an 8-bit address is better than that of a 16-bit address for all three protocols since shorter time is needed to transmit a WuC with a smaller frame size, leading to a lower P_L.

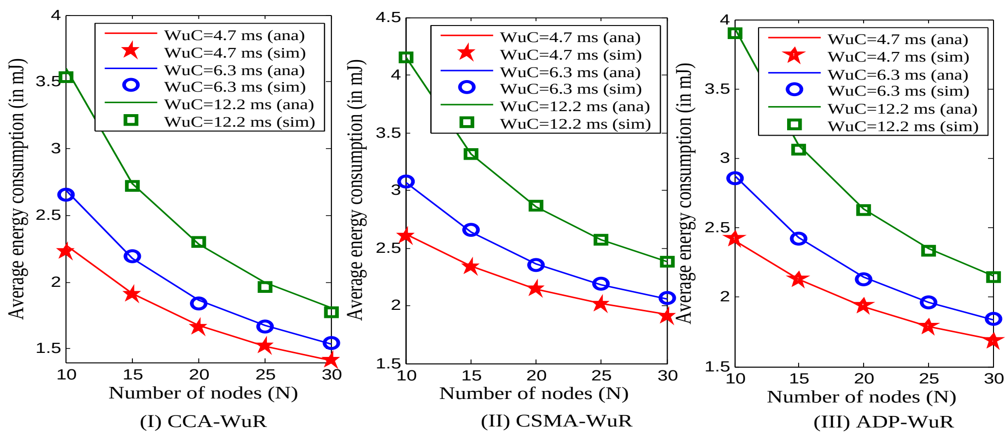
The effects on average packet delay are presented in Fig. B.10(b). All the proposed protocols exhibit similar behavior, i.e., T_S with a higher data rate (at 5460 bps), is shorter than the one that is obtained from its counterpart (at 2730 bps). When a shorter address is adopted, the average packet delay of all the protocols further reduces. For the delay behavior with respect to traffic load or network size, the same trend as shown in Fig. B.7 has been observed. That is, the average packet delay increases with a larger network size for CSMA-WuR and ADP-WuR but decreases for CCA-WuR. The reason is the same as explained in Subsection VIII.B.



(a) WuC loss probability.



(b) Average packet delay.



(c) Average energy consumption.

Figure B.10: Effects of data rate and address length for WuC transmissions.

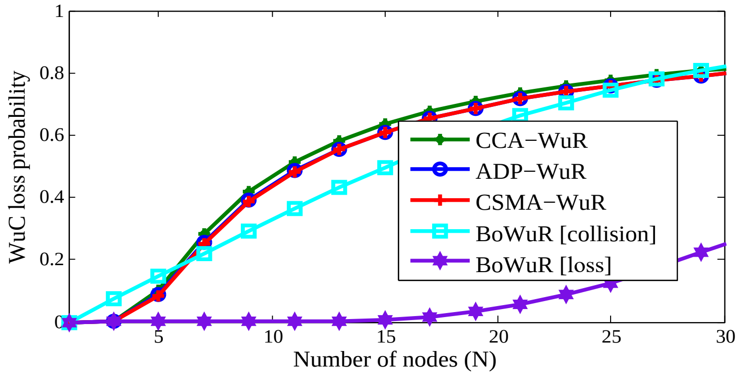


Figure B.11: WuC loss probability comparison of BoWuR, CCA-WuR, CSMA-WuR, and ADP-WuR with the WuC duration as 12.2 ms.

As can be observed in Fig. B.10(c), lower energy consumption is achieved for all three protocols with a higher data rate and a shorter address length. This is because the higher the data rate, the shorter the WuC duration. So does the effect from a shorter WuC address. With a shorter WuC duration, lower energy is consumed. Since in our protocols a discarded packet consumes lower energy than a successfully transmitted packet does and the number of discarded packets increases with a larger N, the downward trend of E with N is self-evident.

Furthermore, it is expected that the performance of all these protocols deteriorates under an error-prone channel. However, the performance curves will still exhibit a similar trend as shown above under an error-free channel assumption since WuC losses due to protocol behavior is statistically independent of losses caused by channel impairments [23]. Accordingly, we can claim that Cor-WuR will reach the non-deliverable stage for packet transmissions earlier due to both channel failures and transmission collisions in an error-prone channel, whereas our proposed protocols would still be operational over a wider range of network size and traffic load conditions.

E. Performance Comparison with BoWuR

In this subsection, we compare the performance of our proposed protocols with BoWuR [17], which is a representative WuR protocol based on CSMA/CA. The BoWuR protocol performs CCA first and then BO (i.e., as in IEEE 802.11 DCF but prior to WuC transmissions). BoWuR assumes that all nodes in a network are synchronized, and can freeze their BO and defer transmissions upon overhearing a WuC transmission. It allows a node to retransmit a WuC if a collision happened in the previous transmission attempt. A node discards a WuC if the retransmission limit attempt is exceeded. The performance of BoWuR is evaluated under saturated

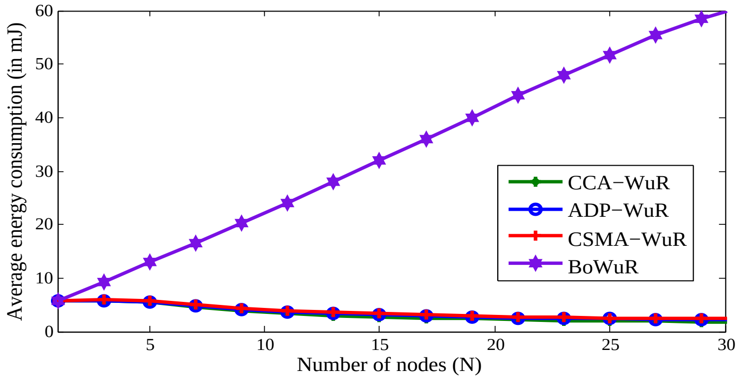


Figure B.12: Average energy consumption comparison of BoWuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

B

traffic in [17]. To make a fair comparison, we have studied and analyzed the performance of BoWuR versus our protocols under unsaturated traffic condition based on [23]. For the performance comparison results presented below, we considered $N \in \{1, 2, ..., 30\}$ and the remaining parameters are configured according to Table B.3.

The performance of the WuC loss probability as the number of nodes varies is shown in Fig. B.11. The WuC loss probabilities for these four protocols, i.e., BoWuR (marked as BoWuR [loss] in the figure), CCA-WuR, CSMA-WuR and ADP-WuR, are calculated based on the discarded packets when the attempt limit is exceeded. From the figure, it is observed that BoWuR outperforms the proposed protocols with respect to this probability. *This benefit is however achieved at an assumption on network synchronization which is not realistic for on-demand based WuR operations*. In the same figure, we show another curve for BoWuR (marked as BoWuR [collision]), illustrating the collision probability for BoWuR which will be used for delay and energy comparison in the following paragraphs.

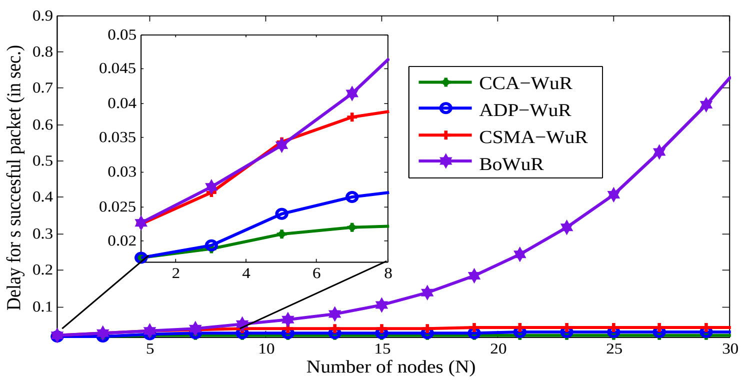


Figure B.13: Average successful packet delay comparison of BoWuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

The average energy consumption comparison of these four protocols is illustrated in Fig. B.12. As expected, the proposed protocols require lower energy than BoWuR under all configured network sizes. This is because a BoWuR node still retransmits its WuC even if the previous WuC transmission is collided. With a higher collision probability (the BoWuR [collision] curve shown in Fig. B.11) at a larger network size, the number of retransmissions for a successful WuC transmission will be higher, leading to higher energy consumption. On the contrary, the proposed protocols perform CCA after BO to check the channel occupancy status and continue the BO procedure if the channel is not idle, i.e., the energy needed for WuC transmissions is saved. The same procedure will be repeated until a WuC transmission is successful or the attempt limit has been reached.

Furthermore, Fig. B.13 presents a comparison of the average packet delay for successfully transmitted packets among BoWuR and our proposed protocols. It is observed that the average successful packet delay of all the proposed protocols is shorter than that of BoWuR for a large network. This is because a BoWuR node freezes its BO counter if it does not win the channel access competition and it has to retransmit the packet if the previous WuC transmission collides. The delay of BoWuR increases significantly with the number of nodes since the WuC collision probability (the BoWuR [collision] curve shown in Fig. B.11) grows with a larger network size.

IX. Conclusions

In this paper, we proposed three WuR protocols known as CCA-WuR, CSMA-WuR, and ADP-WuR respectively with a focus on collision avoidance of WuCs and applying WuR to various traffic conditions. An M/G/1/2 model is developed to evaluate the performance of the proposed protocols as well as of a benchmark protocol, Cor-WuR. The obtained analytical and simulation results coincide with each other and demonstrate that all three proposed protocols outperform Cor-WuR in terms of WuC loss probability and average energy consumptions, at the cost of a longer packet delay. Among these three protocols, CCA-WuR suits best for applications where both short delay and high energy-efficient communication are required while a moderate packet loss probability is tolerable. Meanwhile, CSMA-WuR and ADP-WuR are preferable when packet delivery reliability is of more importance. Between CSMA-WuR and ADP-WuR, the latter one performs better in terms of packet delay and energy consumption. As our future work, we will further study quantitatively the impact of error rate in error-prone channels on the performance of the studied protocols, investigate the feasibility of using pseudo-orthogonal sequences for wake-up signal generation, and implement the proposed protocols in a test-bed for real-life experiment based performance evaluation. With such implementations and deployments, the applicability of WuR-based WSNs/IoT networks to industrial as well as other environments is expected to boom in years to come.

Paper B: References

[1] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks," *in Proc. IEEE INFOCOM, 2011*, pp. 1305-1313.

[2] S. Yanjun, O. Gurewitz, and D. B. Johnson, "RI-MAC: A receiver initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," *in Proc. SenSys, 2008*, pp. 1-14.

[3] G. U. Gamm, M. Sippel, M. Kostic and L. M. Reindl, "Low power wake-up receiver for wireless sensor nodes," *in Proc. ISSNIP, 2010*, pp. 121-126.

[4] G. U. Gamm and L. M. Reindl, "Range extension for wireless wake-up receivers," *in Proc. IEEE SSD, 2012*, pp. 1-4.

[5] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674-687, Apr. 2016.

[6] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," *in Proc. IEEE INFOCOM, 2015*, pp. 522-530.

[7] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. Popovici, and L. Benini, "Design, implementation, and performance evaluation of a flexible low-latency nanowatt wake-up radio receiver," *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 633-644, Apr. 2016.

[8] F. A. Aoudia, M. Gautier, M. Magno, O. Berder, and L. Benini, "Analytical and experimental evaluation of wake-up receivers based protocols," *in Proc. IEEE GLOBECOM, 2016*, pp. 1-7.

[9] M. Zhang, D. Ghose, and F. Y. Li, "Does wake-up radio always consume lower energy than duty-cycled protocols?" *in Proc. IEEE VTC-Fall, 2017*, pp. 1-5.

[10] V. Jelicic, M. Magno, D. Brunelli, V. Bilas, and L. Benini, "Benefits of wake-up radio in energy-efficient multimodal surveillance wireless sensor network," *IEEE Sensors J.*, vol. 14, no. 9, pp. 3210-3220, Sep. 2014.

[11] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Performance evaluation and comparative analysis of subcarrier modulation wake-up radio systems for energy-efficient wireless sensor networks," *Sensors*, vol. 14, no. 1, pp. 22-51, 2014.

[12] T. O. Kim, J. S. Park, H. J. Chong, K. J. Kim, and B. D. Choi, "Performance analysis of IEEE 802.15.4 non-beacon mode with the unslotted CSMA/CA," *IEEE Commun. Lett.*, vol. 12, no. 4, pp. 238-240, Apr. 2008.

[13] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535-547, Mar. 2000.

[14] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions," *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 159-172, Feb. 2007.

[15] F. Z. Djiroun and D. Djenouri, "MAC protocols with wake-up radio for wireless sensor networks: A review," *IEEE Commun. Surveys & Tuts.*, vol. 19, no. 1, pp. 587-618, Q1, 2017.

[16] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra low power wake-up radios: A hardware and networking survey," *IEEE Commun. Surveys & Tuts.*, vol. 19, no. 4, pp. 2117-2157, Q4, 2017.

[17] D. Ghose and F. Y. Li, "Enabling backoff for SCM wake-up radio: protocol and modeling," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1031-1034, May 2017.

[18] S. H. Lee, Y. S. Bae, and L. Choi, "On-demand radio wave sensor for wireless sensor networks: Towards a zero idle listening and zero sleep delay MAC protocol," *in Proc. IEEE GLOBECOM, 2012*, pp. 560-566.

[19] P. Sthapit and J.-Y. Pyun, "Effects of radio triggered sensor MAC protocol over wireless sensor network," *in Proc. IEEE CIT, 2011*, pp. 546-551.

[20] F. A. Aoudia, M. Gautier, and O. Berder, "OPWUM: Opportunistic MAC protocol leveraging wake-up receivers in WSNs," *Journal of Sensors*, vol. 2016, Article ID 6263719, 2016.

[21] J. M. Lebreton, S. Kandukuri, N. Murad, and R. Lorion, “An energy-efficient duty-cycled wake-up radio protocol for avoiding overhearing in wireless sensor networks,” *Wireless Sensor Network*, vol. 8, pp. 176-190, Aug. 2016.

[22] K. R. Chowdhury, N. Nandiraju, D. Cavalcanti, and D. P. Agrawal, “CMAC-A multi-channel energy efficient MAC for wireless sensor networks,” *in Proc. IEEE WCNC, 2006*, pp. 1172-1177.

[23] F. Daneshgaran, M. Laddomada, F. Mesiti, and M. Mondin, “Unsaturated throughput analysis of IEEE 802.11 in presence of non ideal transmission channel and capture effects,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1276-1286, April 2008.

[24] IEEE Standard, 802.15.4e, “Part 15.4: Low-Rate Wireless Area Network (LR-WPANs) Amendment 1: MAC sublayer”, New York: IEEE Standard for Information Technology, 2012.

[25] Texas Instruments, *CC1101 low-power sub-1 GHz RF transceiver datasheet* , USA, Nov. 2013.

[26] NXP Semiconductors, *JN-AN-1001 JN516x Power Consumption*, Application Note, Eindhoven, The Netherlands, Jun. 2016.

[27] *AMS AS3932 Data sheet*, Austria, Mar. 2015.

Paper C

Title:	Enabling Early Sleeping and Early Data Transmission in Wake-up Radio-enabled IoT Networks
Authors:	Debasish Ghose, Anders Frøytlog, and Frank Y. Li
Affiliation:	Dept. of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway
Journal:	*Computer Networks*, vol. 153, pp 132-144, April 2019.
DOI:	10.1016/j.comnet.2019.03.002.
Copyright ©:	Elsevier

C

Enabling Early Sleeping and Early Data Transmission in Wake-up Radio-enabled IoT Networks

Debasish Ghose, Anders Frøytlog, and Frank Y. Li

Abstract — **Wireless sensor networks (WSNs) are one of the key enabling technologies for the Internet of things (IoT). In such networks, wake-up radio (WuR) is gaining its popularity thanks to its on-demand transmission feature and overwhelming energy consumption superiority. Despite this advantage, overhearing still occurs when a wake-up receiver decodes the address of a wake-up call (WuC) which is not intended to it, causing a certain amount of extra energy waste in the network. Moreover, long latency may occur due to WuC address decoding since WuCs are transmitted at a very low data rate. In this paper, we propose two schemes, i.e., early sleeping (ES) and early data transmission (EDT), to further reduce energy consumption and latency in WuR-enabled IoT/WSNs. The ES scheme decodes and validates an address bit-by-bit, allowing those non-destined devices go to sleep at an earlier stage. The EDT scheme enables a sender to transmit small IoT data together with WuC packets so that the main radio does not have to be in full operation for data reception. We implement both schemes through a WuR testbed. Furthermore, we present a framework based on M/G/1 and assess the performance of the schemes through both theoretical analysis and simulations.**

Keywords—WSNs, IoT, WuR, early sleeping, early data transmission, implementation and testbed, modeling and performance evaluation.

I. Introduction

The 5th generation (5G) wireless network aims to facilitate various unprecedented capacities such as enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), as well as ultra-reliable low latency communications (URLLC) [2]. Furthermore, a heterogeneous network integrating 5G with the Internet of things (IoT) paves the way for massive IoT applications [3] [4]. With the inter-network connectivity 5G IoT provides to small-size, low-cost, and often battery-powered devices, a variety of applications are envisaged, ranging from mission-critical services, smart home and smart city, to industrial automation and smart farming [5] [6] [7]. In such massive IoT and wireless sensor network (WSN)

*Manuscript submitted October 02, 2018. This work was presented in part at the IEEE International Conference on Communications (ICC), Kansas City, MO, USA, May 2018 [1].

applications, performing energy-efficient communication for battery powered IoT/WSN devices is of vital importance [8] [9].

Traditionally, duty-cycled (DC) medium access control (MAC) mechanisms, which allow WSN devices sleep and wake up periodically or aperiodically, have been adopted to reduce energy consumption. However, idle listening and overhearing occur in DC-MAC mechanisms during their active periods when a node senses the channel for receiving control messages and when an unintended node overhears the transmission of other nodes respectively. In recent years, wake-up radio (WuR) has emerged as a convincing solution to replace DC-MAC for providing energy-efficient communication in IoT/WSN networks [10] [11].

In a WuR-enabled IoT/WSN node, an auxiliary wake-up receiver (WuRx) is attached to the micro-controller unit (MCU) of a main radio (MR). While the MR, which is responsible for data transmission, is active only when necessary, the WuRx is always on, waiting for detecting wake-up calls (WuCs) at any time. The power consumption of a WuRx is 1000 times lower than that of the MR, i.e., the reception power of WuRx is in μW whereas it is in mW when a main radio is in full operation [10]- [16]. Upon the detection/reception of such a WuC sent by a transmitter, the destined WuRx triggers its MR to wake up from the sleep mode to perform data communication afterwards. In addition to this overwhelming energy saving, another advantage of WuR is that it works in a purely asynchronous manner. Such an on-demand communication operation remarkably reduces latency in comparison with DC-based MAC operations.

C

WuR was initially developed for energy-efficient data collection and reporting in WSNs. Over time, the application scenarios of WuRs have been expanded to diverse wireless networks including IoT, Wi-Fi, and mMTC. Despite the great superiority on energy consumption that WuR provides, overhearing is not eliminated in WuR-enabled IoT networks. Indeed, overhearing occurs when a node decodes and validates an address of a WuC which is not intended to it. Although a WuRx operates in μW, the effect of overhearing in WuR-IoT cannot be simply ignored considering the effect of energy-hungry (decoding and matching) components in WuRxs, long WuC duration, and network size. On the other hand, low latency communication is an important performance indicator in many IoT applications. Very recently, the third generation partnership project (3GPP) has started to work on standardizing early data transmissions (EDT) as one of the 5G new radio techniques to further support energy-efficient and low latency communication for mMTC applications [17] [18]. So far, little work can be found in the literature with respect to reducing overhearing and latency in WuR-enabled IoT considering WuC decoding and EDT. This paper makes an effort towards this direction.

In this paper, we propose two schemes, referred to as early sleeping (ES) and EDT respectively, tailored for eliminating overhearing and shortening latency in such networks. More specifically, ES decodes and matches the address of a WuC bit-by-bit so that the non-destined nodes can go to sleep at an earlier stage. EDT enables a transmitter to send IoT small data encoded with a WuC so that a data transmission may be completed without fully waking up the MR. Both schemes are implemented through a WuR testbed. Furthermore, we develop a queuing model to evaluate the performance of the proposed schemes in WuR-IoT. Extensive simulations are performed to validate the accuracy of the analytical model. In brief, the main contributions of this work are as follows:

- Two energy-efficient schemes are proposed for WuR-enabled IoT, i.e., ES and EDT. While both schemes reduce the energy consumption of such a network, the latter one also minimizes the latency of data transmission.
- For proof of concept, the schemes are implemented in a small-scale WuR testbed and the functionalities of these schemes are demonstrated via the testbed.
- To evaluate the performance of the ES and EDT schemes for a larger network, we present a generic framework based on an M/G/1 queuing model. The accuracy of the model is validated through discrete-event simulations.
- Analytical expressions for three performance parameters including packet delivery ratio (PDR), latency, and energy consumption are derived based on the proposed analytical framework.

The remainder of this paper is structured as follows. In Section 2, we summarize the related work and highlight the qualitative differences between our work and the existing ones. The network scenario and the WuR principle are presented in Section 3. In Sections 4 and 5, we first propose the ES and EDT schemes and then develop a generic M/G/1 queuing framework for performance evaluation of these schemes. The performance metrics are defined and derived in Section 6, followed by testbed implementation and experimental validation presented in Section 7. The simulation results are explained in Section 8, before the paper is concluded in Section 9.

II. RELATED WORK

In this section, we summarize briefly the related work relevant to this study from four perspectives, i.e, WuR prototype implementation, WuR protocol design, theoretical frameworks for WuR-enabled IoT networks, and early data transmission for

5G new radio. For more detailed surveys on the state-of-the-art WuR techniques, please refer to [19] [20].

A. WuR Prototype Implementation

The nanowatt wake-up radio [11] is a WuR prototype which focuses on ultra-low power consumption for WuRxs, with a WuRx power consumption level around 1 μW when listening to the channel. The prototype can operate in different frequencies of the industrial, scientific and medical (ISM) band through on-off keying (OOK) modulation. Its minimum sensitivity is -35 dBm and the response time for an interrupt is 100 μs. Subcarrier modulation WuR [13] is another popular inband WuR circuitry design and its WuC can reach up to 100 meters. It also employs OOK modulation for sending addressable WuC with a sensitivity level of around -53 dBm. A near-zero power WuRx design with -69 dBm sensitivity was proposed in [21], and it was implemented based on an insulator complementary metal-oxide-semiconductor (CMOS) process using metal-oxide-semiconductor field-effect transistor (MOSFET) devices. The wake-up signal implemented therein is also OOK-modulated, but it is operated at 113.5 MHz with a power consumption level of 4.5 nW. Furthermore, the authors in [22] proposed a Bluetooth low energy (BLE)-compliant WuRx achieving a sensitivity level of -80 dBm. It consumes power at 240 nW and maintains latency at 200 μs. Such ultra-low power consumption is achieved by employing a low power bandpass filter-based frequency-shift keying (FSK) demodulator and a correlator. Furthermore, a DC scheme and a packet structure were constructed to maintain the tradeoff between latency and power consumption while ensuring low false alarm rates.

C

B. WuR Protocol Design

Many WuR protocols have been proposed to show the benefits of WuR over traditional DC-MAC based WSNs or to improve the performance of existing WuR schemes. SCM-WuR [10] is a WuR scheme which relies on a single channel for WuC and data packet transmissions. No acknowledgment (ACK) is sent after the successful delivery of a WuC but it is needed when a data packet is received. The performance of SCM-WuR was evaluated using simulations, showing its superior performance over DC-MAC protocols. Another energy-aware cross-layer scheme is OPWUM [14] which opportunistically selects the best relay nodes for forwarding packets based on neighbors' energy. Moreover, ALBA-WuR [15] employs semantic addressing for relay selection, providing a geographic cross-layer solution

Table C.1: A comparative analysis of our schemes with the recent state-of-the-art WuR schemes

Considerations/Features	[10]	[15]	[16]	[27]	[25]	[23]	[24]	ES [1]	EDT
Collision probability	No	No	No	No	Yes	Yes	Yes	Yes	Yes
Testbed implementation	Yes	Yes	No	Yes	No	No	No	Yes	Yes
Theoretical analysis	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Discrete-event simulations	Yes	Yes	Yes	No	No	No	Yes	Yes	Yes
Overhearing	No	No	No	No	No	No	No	Yes	Yes
Address scheme for WuR	No	No	No	No	Yes	No	No	Yes	Yes

for WuR-enabled WSNs. Considering that multiple wake-up transmitters (WuTxs) may send WuCs at the same time, a backoff-enabled WuR scheme, BoWuR, was proposed in our earlier work in order to reduce collision probability for transmitter-initiated (TI) data reporting [23]. For receiver-initiated (RI) data collection, a multiple packet transmission MAC scheme has been proposed which reserves the channel for a node to send multiple packets consecutively [24]. Furthermore, the authors of [25] proposed to adopt a short local address to reduce the latency of WuC transmission. Their addressing scheme utilized partial functions of MCU to decode and match a full address. Improved performance over the correlator-based address decoding approach is shown therein. As part of this study, we proposed in [1] a bit-by-bit WuC address decoding scheme which is known as ES in this paper.

C. Theoretical Frameworks for WuR-enabled IoT

Although a number of theoretical frameworks for modeling IoT or massive IoT applications exist, e.g., [4], there are few frameworks which deal with WuR-enabled IoT networks. Among them, an analytical model which is based on an absorbing Markov chain was presented in [27] to assess the performance of WuR-enabled IoT networks. In [23], the performance of the BoWuR protocol was analyzed using a discrete time Markov chain for a single hop SCM-WuR enabled IoT network under saturated traffic conditions. Based on duty-cycled received-initiated WuR IoT networks, an analytical model was developed in [24] to evaluate the performance of aggregated packet transmissions. However, none of these frameworks considered an ES or/and EDT scheme for their performance evaluation.

D. EDT for 5G New Radio

To enhance the capability of LTE towards 5G new radio, 3GPP is currently investigating EDT techniques for the purpose of further reducing power consumption and latency for mMTC in cellular networks [17] [18]. More specifically, EDT applies to both uplink and downlink data transmission and it is performed during the random access procedure (after the physical random access channel (PRACH) transmission and before the radio resource control (RRC) connection setup is completed). With EDT, a small amount of data exchange can be achieved during the random access procedure *before* the data link is formally established.

However, none of the aforementioned WuR prototypes and protocols considered eliminating overhearing for WuC transmissions. In Table C.1, we make a qualitative comparison of our schemes with a few state-of-the-art WuR protocols. To the best of our knowledge, this work is first effort which applies EDT to WuR-enabled IoT networks.

III. NETWORK TOPOLOGY AND WUR PRINCIPLE

In this section, we illustrate the network scenario as well as the design and operating principle of WuR.

A. Network Scenario and Assumptions

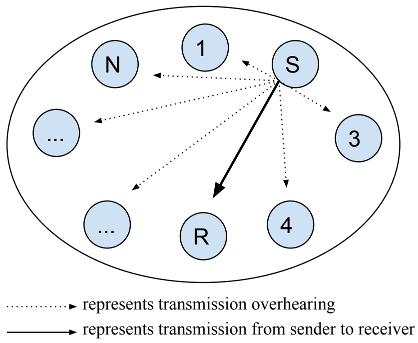


Figure C.1: A single-hop WuR-IoT/WSN with N contending nodes where S and R stand for the sender and the receiver, and the solid and dashed arrows represent intended transmission and other nodes' overhearing, respectively.

Consider a WuR-enabled IoT/WSN consisting of N number of nodes as shown in Fig. C.1. All IoT/WSN devices are integrated with a WuR and can communicate with each other over a single-hop in a full mesh topology. Packet arrivals follow a Poisson process with rate λ for each device. Assume further that the channel is error-free and no packet loss occurs due to buffer overflow. However, packet loss occurs due to collisions.

B. Operating Principle of WuR

In a WuR based IoT/WSN, a WuRx is embedded with the sensor node for awakening up the MCU of its MR in an on-demand manner. A data communication cycle in a WuR network starts with a WuC and ends with an ACK after the data packet is correctly received. WuR can operate either in the receiver-initiated or transmitter-initiated mode. While RI-WuR is suitable for data collection, TI-WuR fits better for event-triggered data reporting. In what follows, we focus on TI-WuR for performance evaluation of WuR-IoT. In TI-WuR, a node sends a WuC to the intended node once it has a packet to transmit. After receiving the WuC, the intended node turns on its main radio for data communication unless EDT without ACK, which will be presented in the next section, is employed. When both MRs are active, data communication is performed. A transmission cycle finishes with an ACK from the targeted receiver. It is worth mentioning that no ACK is necessary to acknowledge the successful delivery of a WuC transmission to avoid frequent switching of radio operation mode [10] [23]. The working principle of TI-WuR is shown in Fig. C.2.

IV. Enabling ES and EDT for WuR

In this section, we propose ES and EDT tailored for WuR-enabled IoT/WSNs and explain their working principles.

A. Early Sleeping for WuC Decoding

Typically a WuRx needs to decode the full address of a WuC before it decides whether to wake its MR up or not [25]. This procedure is referred to as full address decoding (FAD), which is a typical scheme for WuC address decoding. In this paper, we propose to decode an address bit-by-bit in order to improve the performance of FAD.

More specifically, the ES scheme defines an address decoding and validating rule that is able to diminish WuC overhearing energy cost for unintended WuRxs. It uses *partial MCU functions* to decode and validate the address. Different from FAD, ES allows a node to decode and match the received address bit-by-bit instead of

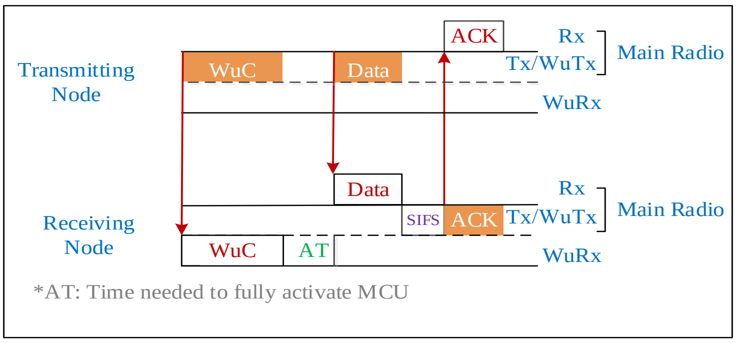


Figure C.2: Illustration of a data transmission cycle of WuR.

C

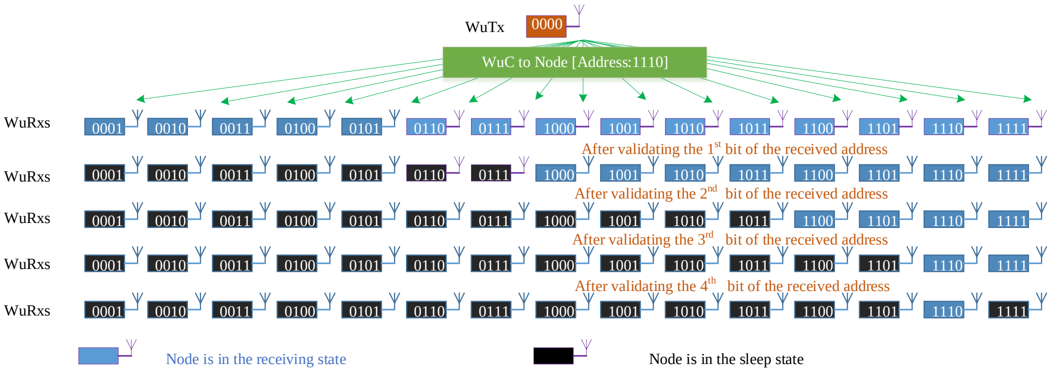


Figure C.3: Operation of ES with a network which consists of 16 nodes and uses a 4-bit address.

decoding and validating a complete address after all bits are received. That means, the MCU decodes the first bit of the received address sequence and matches it with its corresponding address bit. If the first bit of the received address matches, then it will decode and validate the next bit. For an intended receiver, this process will continue until the whole address matches with its own address. For a non-intended receiver, whenever a bit of the received address mismatches with its own, the node stops the decoding and matching process and goes to sleep immediately. In this way, ES reduces energy consumption in a network by letting overhearing nodes sleep at an earlier stage before the whole WuC decoding procedure is complete.

As an example, consider a network cluster consisting of 16 nodes and let us use a 4-bit local address for illustration. All nodes in this cluster hear each other. Assume now that a sender (address: “0000”) transmits a WuC to wake up a targeted node (address: “1110”) for performing data communication. All unintended nodes overhear the WuC transmission. If ES is employed, 7 nodes with addresses “0XXX” will go to sleep right after decoding and validating the first bit. The remaining 8 nodes will decode the second bit. Among these 8 nodes, 4 nodes with addresses “10XX” will go to sleep right after decoding the second bit. This process will continue as more overhearing nodes go to sleep, thus reducing energy consumption for the whole network. In the end, only the intended node will wake up after decoding and validating all 4 bits. The hierarchical structure of the address decoding and matching stage of ES is illustrated in Fig. C.3. Clearly, the number of sleeping nodes in a WuR with ES will be increased in comparison with WuR with FAD.

B. Early Data Transmission Together with WuC

In this subsection, we propose an early data transmission scheme which is tailored to WuR-enabled IoT/WSNs. The scheme is still referred to as EDT. The proposed EDT scheme defines a data transmission procedure in which a small-size data packet is jointly encoded with the WuRx address and transmitted through a WuC *before the MR is fully waken up*. That is, a WuR transmitter performs data transmission simultaneously while sending a WuC with the support of *partial operation* * of the main radio of an intended node. Such a scheme reduces the latency it takes to transmit small data and improves the energy efficiency of WuRxs. Depending on whether an ACK upon the successful reception of a small data is needed or not, EDT can be operated in one of the following two modes.

C. EDT with ACK

In this mode, the to-be-transmitted small data is encoded via an error-detecting code, e.g., cyclic redundancy check (CRC), at the transmitter side. The checksum for a given polynomial of CRC is appended to the small data. After that, the appended data will be encoded with the intended WuRx address. Then the output bits of the XOR operation will be transmitted as the WuC frame. After receiving a WuC, the MCU of the MR performs partial functions to decode and validate the received the WuC.

More specifically, a reverse operation will be performed at the receiver side. That is, the receiver will perform the XOR operation between its address and the received packet. The output bits of the XOR operation will be divided by the same polynomial used at the transmitter. If the result of this division is zero, then the WuRx will issue an interrupt to fully wake up the MR to the active mode for sending an ACK after each successful data transmission. If the result of the division is not equal to zero, it will treat the WuC as overhearing and the MCU will go to the deep sleep mode as soon as the decoding process is finished. The data transmission cycle of EDT with ACK is illustrated in Fig. C.4(a) and its workflow chart is shown in Fig. C.5 respectively.

As an example, consider a TI-WuR scenario where a sender intends to send a small data, e.g., humidity 53$\%$ (data “110101”) to an intended receiver (WuRx address “11111100”). For a given polynomial, $g(x) = x^2 + 1$ (101), the data packet with CRC will be “11010111”. Then, the CRC appended data packet will be en-

*The hardware of the MR used in our implementation is nRF52832. It has three modes: deep sleep, light sleep, and active with corresponding current consumption at each mode as 0.3 μA, 1.9 μA, and 4.1 mA respectively (refer to Footnote 4). In the light sleep mode, the MCU is able to perform partial functions for WuC decoding.

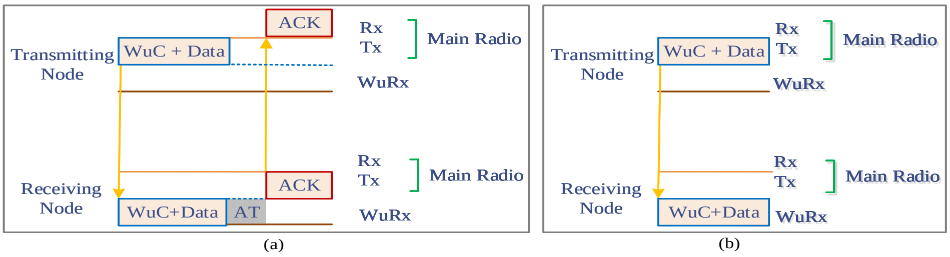


Figure C.4: The data transmission cycle of EDT: (a) with and (b) without ACK.

coded with the wake-up receiver address "11111100" using the XOR operation. The output bits of XOR, i.e., "00101011", will be transmitted as an encoded WuC. At the receiver side, a reverse operation will be performed to extract the actual data message. That is, the intended receiver will perform XOR operation between its address "11111100" and the received WuC "00101011". The output of this operation, "11010111", will be checked with the same CRC polynomial using division. If the remainder is zero, the intended node removes the CRC bits to extract the actual data. The detailed procedures for this data encoding and decoding process at the transmitter and the receiver are presented in Fig. C.6 respectively.

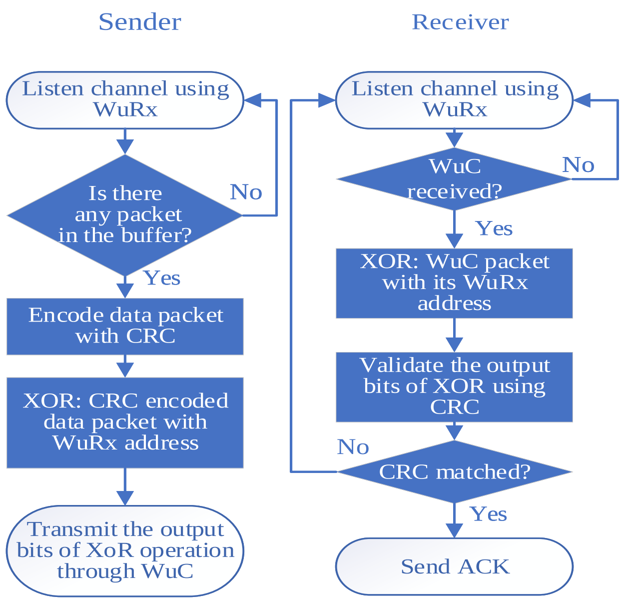


Figure C.5: Operations of EDT with ACK at the transmitter and the receiver.

D. EDT without ACK

From the design principle of EDT with ACK, it is clear that EDT with ACK is well suited for IoT/WSN applications where ACK for a successful data recep-

Transmitter	**Receiver**
Step 1. Encode data with CRC Data : 1 1 0 1 0 1, CRC polynomial:1 0 1 1 1 0 1 0 1 0 0 ÷ 1 0 1 Quotient:1 1 1 0 1, Remainder: 1 1 Message with CRC: 1 1 0 1 0 1 1 1	**Step 1.** WuC packet XOR WuRx address 0 0 1 0 1 0 1 1 XOR 1 1 1 1 1 1 0 0 Output: 1 1 0 1 0 1 1 1
Step 2. Message with CRC XOR WuRx address 1 1 0 1 0 1 1 1 XOR 1 1 1 1 1 1 0 0 Output: 0 0 1 0 1 0 1 1	**Step 2.** Check message for CRC error Data : 1 1 0 1 0 1 1 1, CRC polynomial:1 0 1 1 1 0 1 0 1 1 1 ÷ 1 0 1 Quotient:1 1 1 0 1, Remainder: 0 0 No error. Wake-up the MR
Step 3. Transmit output bits as WuC	**Step 3.** Send an ACK message

Figure C.6: Illustration of EDT with ACK: An example.

tion is required. However, ACK might not always be beneficial if retransmission is not necessary in a network. Accordingly, we propose another variant of the EDT scheme, i.e., EDT without ACK, which disables ACK. In this case, no ACK is transmitted from the MR of the intended receiver back to the sender upon a successful data reception. As both data and ACK transmissions are performed when the MR is operated in the active mode, EDT without ACK further decreases latency and reduces power consumption since such a small data exchange cycle is accomplished without fully waking up the MR, as shown in Fig. C.4(b).

Furthermore, both versions of EDT can be used to transmit a data packet based on a given WuC address length. In our prototype implementation to be presented in Section 6.3, we have designed the WuC address length as 16 bits, allocating 10 bits to represent data values. However, a shorter or longer address may apply. With a shorter WuC address, one value could represent a larger range for the parameter of interest.

V. Modeling Data Transmission in WuR

In this section, we present a generic framework in order to model data transmission in WuR-enabled IoT/WSNs with ES and/or EDT capability. This framework is based on the analysis of a regenerative cycle of M/G/1 and it applies to both ES and EDT based data transmissions†. The notations used in our analysis are summarized in Table C.2.

In most of the existing unslotted carrier sense multiple access with collision avoidance (CSMA-CA) based models, e.g., [29], a state transition is based on the approximation of mini-slot/symbol duration. However, due to the on-demand nature

†The same model applies to both ES and EDT since the number of transmitting nodes is irrelevant to whether the address is decoded bit-by-bit (by ES) or as a whole (by FAD), or the WuC is encoded with small data (by EDT) or not.

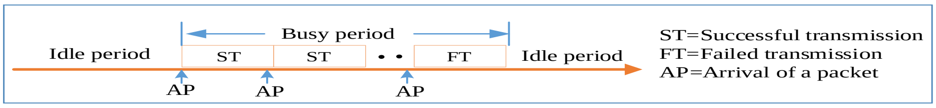


Figure C.7: Illustration of a regenerative cycle.

of data transmission in WuR-enabled IoT/WSNs, a node can wake up (a sleep-to-active state transition) at any instant of time whenever it generates a packet. In other words, a state transition in a WuR operation does not necessarily occur at the mini-slot boundary. Moreover, the standard analytical framework of ALOHA-alike schemes does not consider the buffer size of the nodes and assumes that the number of nodes in the network is very large ($n \to \infty$). Herein we develop a model based on M/G/1 which considers *small buffer size, limited number of nodes, and does not dependent on the assumption that state transitions occur only at mini-slot boundaries*.

In a WuR-enabled IoT/WSN, a node sends a WuC to the intended WuRx whenever it generates a data packet. A collision occurs when the transmissions of two or multiple nodes overlap with each other. Collisions cannot be avoided since no collision avoidance is adopted before sending a WuC.

Denote by P_c the probability of occurring a collision. A successful transmission of a device prevails when none of the other $(N-1)$ devices starts a busy period during the vulnerable period of the device. The busy period is defined as the period in which a device is continuously busy [30]. All packets arrived in a busy period will be served (successfully or unsuccessfully) in that busy period itself. The interval between two busy periods is an idle period. The regenerative cycle in our model is shown in Fig. C.7.

A regenerative cycle consists of a busy and an idle period. For analysis simplicity, we consider that all busy periods have a constant duration which is equal to its mean value, T_{B}. For a given λ, this assumption is reasonable since no retransmission is allowed, the size of IoT small data packets (e.g., temperature, humidity, etc.) is identical (e.g., 2 bytes), and the duration of a successful transmission, T_{ST}, as well as that of an unsuccessful transmission, T_{FT}, have the same length. Even though T_{ST} and T_{FT} are scheme dependent, for a given scheme, i.e., ES or EDT, we have $T_{\mathrm{ST}} \approx T_{\mathrm{FT}}$ since the ACKtimeout duration, $T_{\mathrm{ack}}^{\mathrm{tout}} \approx T_{\mathrm{SIFS}} + T_{\mathrm{ack}}$ where T_{ack} and T_{SIFS} are the duration of ACK and short inter-frame space (SIFS) respectively, is needed for identifying an unsuccessful transmission whereas a successful transmission is confirmed when an ACK is received[‡]. With this simplification, $\mathrm{E}[T_{\mathrm{B}}]$ is

[‡]Note that $T_{\mathrm{ack}}^{\mathrm{tout}} = 0$ for EDT without ACK.

Table C.2: Major Notations Used in the Analysis

Notation	Description
N	Number of nodes in the network/network cluster
λ	Packet arrival rate to a node
$\mathrm{E}[\Gamma]$	Expected number of packets served in a busy period
P_s	Successful packet transmission probability
P_c	Collision probability
b_0	Probability that no packet arrives during the time a packet is at the head of the line
T_{FT}	Duration of an unsuccessful transmission
T_{ST}	Duration of a successful transmission
$\mathrm{E}[T_{\mathrm{B}}]$	Mean duration of a busy period

obtained as

$$\mathrm{E}[T_{\mathrm{B}}] = T_{\mathrm{ST}}\mathrm{E}[\Gamma], \tag{C.1}$$

where Γ is the number of packets served in a busy period of the M/G/1 queuing system with traffic intensity $\alpha = \lambda T_{\mathrm{S.}}$ $\mathrm{E}[\Gamma]$ can be computed as

$$\mathrm{E}[\Gamma] = \frac{1}{1-\alpha}. \tag{C.2}$$

Thus, P_c can be approximated as

$$P_c \approx 1 - \left[e^{-\frac{\lambda}{\mathrm{E}[\Gamma]}(\mathrm{E}[T_{\mathrm{B}}]+T_{\mathrm{ST}})}\right]^{N-1}. \tag{C.3}$$

By inserting the value of $\mathrm{E}[\Gamma]$, $\mathrm{E}[T_{\mathrm{B}}]$ for a given λ, the value of P_c can be obtained.

C

VI. Performance Metrics

In this section, we define three metrics for performance evaluation, i.e., packet delivery ratio, latency, and energy consumption and derive the their expressions.

A. Packet Delivery Ratio

Denote by P_s the PDR. It is defined as the ratio between the number of successful transmissions and the total number of transmission attempts during a regenerative cycle. Since there exists only one reason for packet loss in this study, i.e., packet loss due to collisions, P_s can be computed from (C.3) as follows

$$P_s = (1 - P_c). \tag{C.4}$$

B. Latency

The average latency of a successfully delivered packet, denoted by T_d, is defined as the duration from the time instant that a packet arrives at the transmission queue of the data generation node until the moment the packet is successfully transmitted to the intended node. Since no retransmission is allowed, T_d for an unsuccessful transmission is almost the same and it is obtained by

$$T_d = T_{\mathrm{ST}} \approx T_{\mathrm{FT}} \tag{C.5}$$

where T_{ST} and T_{FT} are the duration of a successful transmission and a failed transmission respectively.

Furthermore, the obtained value of T_d depends on the scheme employed in the network. For ES, $T_{\mathrm{ST}} = T_{\mathrm{wuc}} + T_{\mathrm{AT}} + T_{\mathrm{data}} + 2\,T_{\mathrm{SIFS}} + T_{\mathrm{ack}}$ where T_{wuc}, T_{AT}, and T_{data} are the duration for WuC transmission, fully active MCU, and data transmission respectively. Since no separate transmission is required for sending data, the duration of a successful transmission for EDT with ACK or without ACK is $T_{\mathrm{ST}} = T_{\mathrm{wuc}} + T_{\mathrm{AT}} + 2T_{\mathrm{SIFS}} + T_{\mathrm{ack}}$ or $T_{\mathrm{ST}} = T_{\mathrm{wuc}} + T_{\mathrm{SIFS}}$ respectively.

C. Energy Consumption

Denote by E the energy consumption *for the whole network* consisting of one transmitting node, one destined node and $(N-2)$ unintended nodes. We have

$$E = E_T + E_R, \tag{C.6}$$

C

where E_T is the energy consumption of the sending node for a packet transmission (successful/unsuccessful). It is given by

$$E_T = P_c E_c^T + P_s E_{Tx}, \tag{C.7}$$

where E_c^T and E_{Tx} are the energy consumed by the sender for a successful and unsuccessful packet respectively. E_R is the total reception energy consumed by the nodes in the network with N nodes for one packet transmission. It includes the energy consumed by the destined node for receiving a packet and the energy consumed by the other $(N-2)$ unintended, i.e., overhearing nodes. E_R is given by

$$E_R = P_c E_c^R + P_s E_{\mathrm{Rx}}, \tag{C.8}$$

where E_c^R represents the total reception energy consumed by a network for an unsuccessful transmission. Assuming that collisions happen due to the simultaneous transmissions of two nodes, then E_c^R can be estimated as $E_c^R = (N-2)E_{\text{idle}}$. E_{idle} is the energy consumed by a node when it is actively monitoring the channel using WuRx. For a collided packet, a node (intended or unintended receiver) consumes the same energy as needed for idle listening since it cannot detect and decode the packet. The total reception energy consumed by the network for a successful packet, E_{Rx}, can be calculated as

$$E_{\text{Rx}} = E_{\text{IN}} + E_{\text{UN}}, \tag{C.9}$$

where E_{IN} and E_{UN} are the energy consumption of the destined node and the unintended nodes respectively. Lastly, the average reception energy consumed by a node for a packet transmission, denoted by E_{R}^{avg}, is obtained as

$$E_{\text{R}}^{avg} = \frac{E_R}{N-1}. \tag{C.10}$$

Furthermore, the duration of a successful or an unsuccessful transmission will be scheme dependent since ES and EDT adopt different address decoding rules. Thus, E_c^T, E_{Tx}, E_{IN}, and E_{UN} need to be calculated specifically for each scheme. For the ES scheme, we have

$$E_c^T = E_{\text{wuc}}^T + E_{\text{AT}} + E_{\text{data}}^T + E_{\text{SIFS}} + E_{\text{ack}}^{\text{tout}},$$

$$E_{Tx} = E_{\text{wuc}}^T + E_{\text{AT}} + E_{\text{data}}^T + 2E_{\text{SIFS}} + E_{\text{ack}}^R,$$

where E_{wuc}^T, E_{AT}, E_{data}^T, E_{SIFS}, E_{ack}^R, and $E_{\text{ack}}^{\text{tout}}$ are the energy consumption for WuC transmission, fully activated MCU, data packet transmission, SIFS, ACK reception, and ACK timeout respectively. E_{IN} can be obtained as $E_{\text{IN}} = E_{\text{wuc}}^R + E_{\text{AT}} + E_{\text{data}}^R + E_{\text{SIFS}} + E_{\text{ack}}^T$ where E_{wuc}^R, E_{data}^R and E_{ack}^T are the energy consumption respectively for WuC reception, data packet reception and ACK transmission. With an assumption that N is exponential to the power of 2, E_{UN} for ES can be approximated as

$$E_{\text{UN}} \approx \sum_{i=1}^{B} i E_{\text{vwuc}}^R \lceil \frac{N-2}{2^i} \rceil + (N-2)E_{\text{prm}}^R, \tag{C.11}$$

$$E_{\text{vwuc}}^R = P_{\text{wuc}}^R T_b, \tag{C.12}$$

where P_{wuc}^R, T_b, E_{prm}^R, and B represent the reception power of WuC, the time needed to decode and validate 1 bit of an address, the energy needed for preamble detec-

C

tion and partially switching on MCU, and the total number of bits in the received address, respectively.

In EDT, no separate transmission is required for sending a small data packet. Thus E_c^T, E_{Tx}, and E_{IN} for EDT with ACK can be obtained as

$$E_c^T = E_{\mathrm{wuc}}^T + E_{\mathrm{AT}} + E_{\mathrm{SIFS}} + E_{\mathrm{ack}}^{\mathrm{tout}},$$

$$E_{Tx} = E_{\mathrm{wuc}}^T + E_{\mathrm{AT}} + 2E_{\mathrm{SIFS}} + E_{\mathrm{ack}}^R,$$

$$E_{\mathrm{IN}} = E_{\mathrm{wuc}}^R + E_{\mathrm{AT}} + E_{\mathrm{SIFS}} + E_{\mathrm{ack}}^T.$$

Furthermore, since no ACK is required for EDT without ACK, E_c^T, E_{Tx}, and E_{IN} are obtained as follows

$$E_c^T = E_{\mathrm{wuc}}^T + E_{\mathrm{SIFS}},$$

$$E_{Tx} = E_{\mathrm{wuc}}^T + E_{\mathrm{SIFS}},$$

$$E_{\mathrm{IN}} = E_{\mathrm{wuc}}^R.$$

Lastly, E_{UN} is the same for both EDT with and without ACK and it is given by

$$E_{\mathrm{UN}} = (N-2)(BE_{\mathrm{vwuc}}^R + E_{\mathrm{prm}}^R). \tag{C.13}$$

VII. Testbed Implementation and Validation

C

In this section, we first give a brief overview regarding the prototype implementation of our WuR testbed and then present the experiments which validate the functionalities of the ES and EDT schemes.

A. Implementation Overview

Based on the principles presented above, we have implemented a small-scale WuR-enabled IoT/WSN testbed which supports both ES and EDT. As shown in Fig. C.8, each IoT/WSN device is composed of two components, a self-designed circuit as the WuRx and a main radio built based on an nRF52832 system-on-chip (SoC) from Nordic Semiconductor[§]. nRF52832 is an ultra-low power SoC supporting multi-protocols especially suitable for BLE applications. It is built based on a 32-bit

[§]Nordic Semiconductor, nRF52832 data sheet, [Online]. Available: `http://infocenter.nordicsemi.com/pdf/nRF52832_PS_v1.3.pdf`.

ARM Cortex-M4F CPU with a 512 kB + 64 kB RAM and an embedded 2.4 GHz transceiver. Without being interrupted, the MCU retains in the deep sleep mode consuming merely 0.3 μA current.

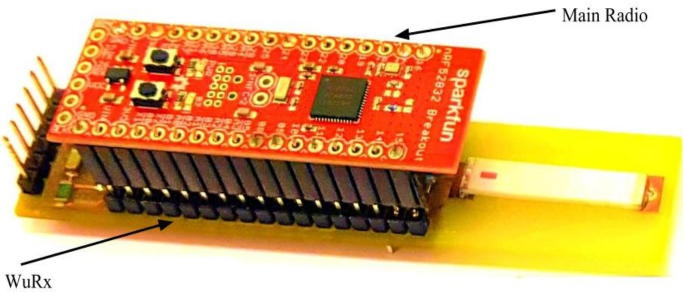


Figure C.8: A prototype WuR device which consists of a WuRx and an MR.

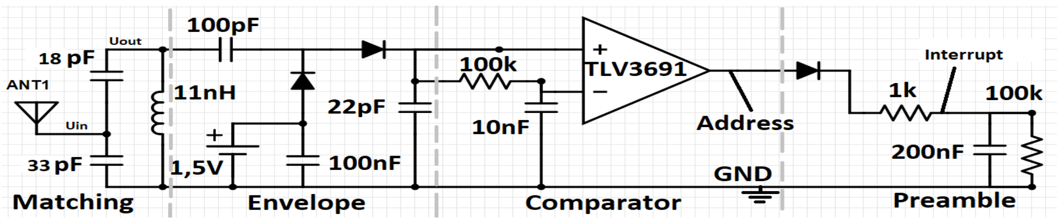


Figure C.9: Illustration of the implemented WuRx: Schematic

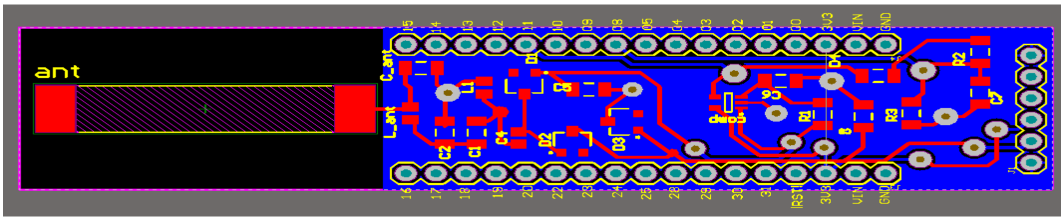


Figure C.10: Illustration of the implemented WuRx: PCB

As illustrated in Figures C.9 and C.10 , the WuRx consists of four blocks: a matching network, an envelope detector, a comparator, and a preamble detector. The matching network provides impedance matching to the envelope detector. The envelope detector demodulates the received OOK signal and generates pulses which represent the ‘1’s and ‘0’s of the address sequence. Then, the comparator generates pulses based on the signal from the envelope detector so that it is readable by the MCU. The preamble detector provides an interrupt signal to the reset pin of the MCU to wake it up from the *deep sleep* mode to the *light sleep* mode (with 1.9 μA current consumption) under which the MCU can decode and validate the received address. When all bits of a received address match with its own, it activates the

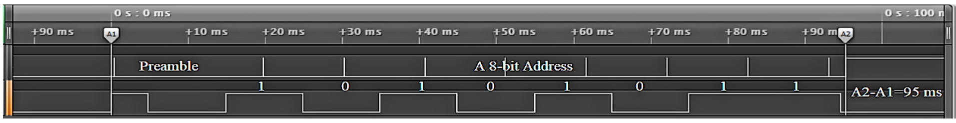


Figure C.11: A WuC duration with 8-bit address in the Saleae logic analyzer.

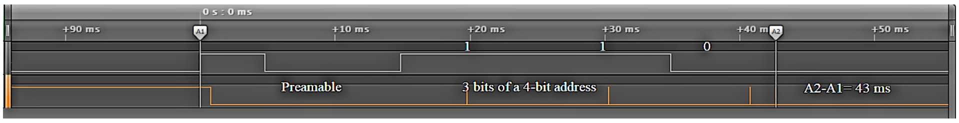


Figure C.12: Time needed to decode and validate 3 bits of a 4-bit address.

MCU fully to the *active* mode to perform data communication over BLE. Otherwise, it goes to deep sleep but continues listening to the channel. For address decoding, both the FAD and ES schemes are implemented in our testbed.

The WuRx implemented in our prototype consumes merely 90 nA current and it is always active, listening to the channel continuously. Since the current consumption at the deep sleep mode of the MCU is 0.3 μA, the current consumption level is 0.39 μA when no event happens. To decode an address, the current consumption is 1.909 μA when the MCU is operated at the light sleep mode. For data transmission when the MR is active, it is performed over BLE at a much higher data rate than the one used for WuC transmissions, at a current consumption level of 4.1 mA.

B. Experimental Validation of the ES Functionality

Based on the developed WuR prototype, we have performed various experiments to measure the duration of a WuC address decoding procedure for both FAD and ES. In both cases, whenever a WuRx detects a WuC signal, it starts to process it. As soon as the preamble detector detects a preamble, it provides an interrupt signal to the MCU to initialize a low frequency clock and general-purpose input/output so that address decoding can start, with the MCU in partial operation.

We have measured the duration to decode and validate one bit of a received WuC address and find out that it is 10 ms. As shown in Fig. C.11, a WuC duration with an 8-bit address for FAD is measured using a Saleae logic analyzer. The whole procedure lasts for 95 ms including preamble 13 ms, $8 \times 10 = 80$ ms for decoding and matching, and 2 ms to switch on the MCU to the active mode.

When ES is employed, the procedure is somewhat different. After a WuRx observes the first clock flank of the address, it starts immediately the timeout timer to check each bit of the received address in order to prevent the MCU from acquiring for a complete sequence of the address. It fully activates the MCU only if the received address matches completely with its own. Whenever a bit of the received address mismatches, the WuR goes to the deep sleep mode instantly. Continue the example shown in Fig. C.3 where a 4-bit address is used. It would take 55 ms to decode this address if FAD is used. Fig. C.12 reveals that the duration for a WuRx stops processing a WuC address after decoding and validating the first three bits of a 4-bit address. It is observed that the WuRx needs 43 ms (between A2 and A1) before it goes to sleep after decoding and validating the first three bits of a 4-bit WuC address "1110" since the fourth bit mismatches with its own.

C. Experimental Validation of the EDT Functionality

Based on the same prototype, experiments to validate the functionality of EDT are also carried out. For EDT, a WuRx follows a similar procedure as needed for FAD for WuC address decoding and data processing. After demodulating the WuC packet, the XOR operation is performed between WuRx's address and the received packet. The output bits of XOR will be treated by the CRC polynomial for error check. In case of EDT with ACK, the WuRx will trigger the MR to wake it up from *light sleep* to *active* for sending an ACK upon each successful data reception. When EDT without ACK is employed, however, the WuRx will not trigger the MR for ACK transmission. It is worth mentioning that the measured duration to decode and validate one bit of a received WuC packet in EDT is also 10 ms, indicating that EDT does not add any extra latency for data processing. Instead, shorter latency can be achieved since no data transmission is performed by the MR, in addition to energy saving.

Based on the nRF UART 2.0 mobile APP development kit [¶], we have developed an Android APP programmed in Android Studio. Fig. C.13 illustrates a screenshot for data sniffing of an EDT experiment when EDT without ACK was employed. In this example, a pair of WuR devices exchanged data and the developed APP is used as a sniffer to capture data transmission and reception. As mentioned earlier, the receiving node uses its WuRx and MCU in the light sleep mode for data reception but the MR is not fully activated. The address of the WuRx has 16 bits and the CRC polynomial is 101. For a data packet which has also 16 bits after CRC, we use 3 bits to represent up to 8 different data types and 10 bits to represent data values.

For example, data types 000, 001, and 010 represent temperature, humidity, and light intensity respectively. To report a temperature of 24 degrees Celsius, i.e., binary '0000011000', data type 000 will be appended. Then the data packet will be '0000000011000' before CRC. Assume that the address of the WuRx is configured

[¶] Nordic Semiconductor UART APP source code, [Online]. Available: `https://github.com/NordicSemiconductor/Android-nRF-UART`.

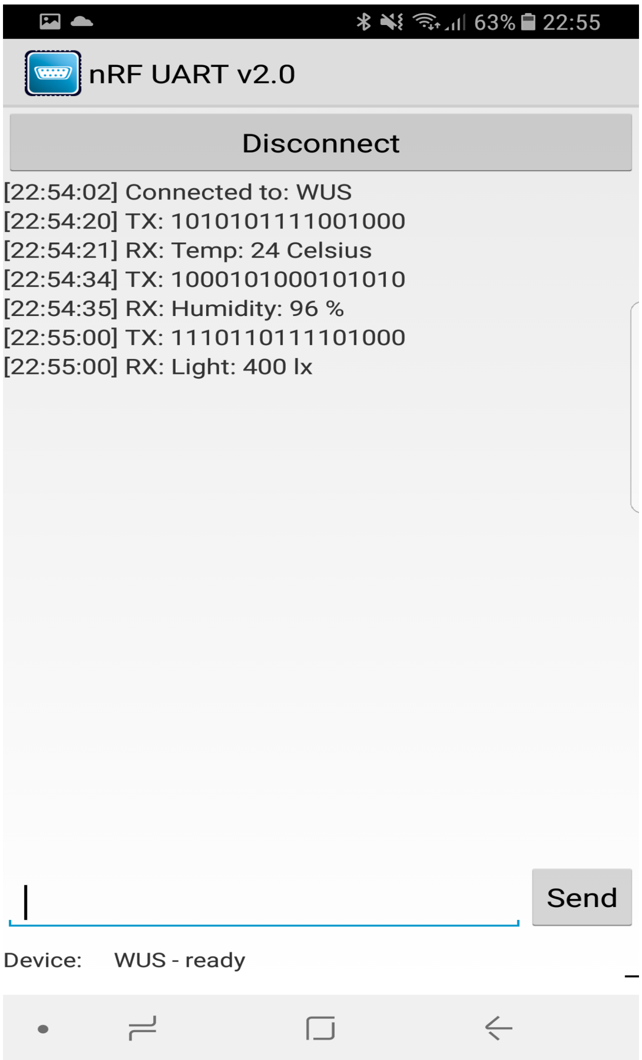


Figure C.13: A screenshot for EDT data snipping for EDT without ACK.

as ‘1010101110101011’. After the CRC and XOR operations, the to-be-transmitted WuC packet will be ‘1010101111001000’. At the receiver side, the WuRx performs reverse operation and extract the temperature value, as shown on the second and third lines of the screenshot. In a similar way, other parameters like humidity and light intensity can also be extracted from the received WuC packet which is encoded with data and the WuRx address.

VIII. Numerical Results and Discussion

In this section, we first validate the theoretical framework by comparing the analytical results with the simulation results, and then evaluate the performance of the proposed schemes with respect to the defined metrics.

A. Simulation Configurations and Theoretical Framework Verification

Consider a WuR-enabled IoT network as presented in Fig. 1. To verify the preciseness of the theoretical framework, we perform extensive computer simulations based on this network with the following parameter configurations. The numbers of devices is configured as $N = 32$ and the packet arrival rate varies in the range of $\lambda \in \{0.1, 0.2, ..., 0.5\}$ packet/s. The remaining parameters are configured based on Table C.3 unless otherwise stated.

To perform simulations, we constructed a custom-built discrete-event simulator in MATLAB which is similar to the one we built in our early work [26]. The developed simulator mimics the behavior of the studied WuR-IoT with the ES or EDT capabilities implemented respectively. That is, a node wakes up as soon as a packet arrives in the queue and maintains a timestamp to track the time duration of every single state. The sending node works in an on-demand manner, like in ALOHA, for sending its WuCs. Upon receiving a WuC, the targeted node decodes the WuC and switches on its MR for data communication (for FAD, ES, and EDT with ACK), or keeps its MR in the light sleep mode (for EDT without ACK) for small data transmission. On the other hand, the unintended nodes go to sleep earlier according to the ES principle described earlier.

When comparing FAD with ES in a network with 32 devices, we denote FAD and ES as FAD-5 and ES-5 respectively as a 5-bit address length would be sufficient. Since a 16-bit address is designed in our EDT implementation, these two schemes are denoted as FAD-16 and ES-16 respectively when comparing FAD/ES with EDT. Note however that in our simulations the number of devices in the network is always configured as 32 regardless of the adopted address length.

C

The analytical results which were obtained based on the M/G/1 model and the simulation results that were obtained from simulations are presented jointly in Figs. 13−19. In these figures, "analytical" and "simulation" represent the analytical and simulation results respectively. Furthermore, it is worth mentioning that *the obtained results from simulations are independent of the ones obtained from analytical expressions*. From Figs. 13−19, it is evident that the analytical and simulation results coincide with each other. Thus, the accuracy of the analytical model is verified.

B. Simulation Configurations and Theoretical Framework Verification

In this subsection, we first evaluate the performance of the proposed schemes with respect to three parameters, e.g., packet delivery ratio, average latency, and energy consumption, and then estimate node lifetime.

Table C.3: Parameter Configurationfor Performance Evaluation [1] and Footnote 4

Radio type	**Parameter**	**Value**	**Unit**
Common	Supply voltage	3	V
	Battery Capacity	220	mAh
	Packet arrival rate	1-5	packets/s
Main radio	Data rate	125	kbps
	Transmission current	5.3	mA
	Reception current	5.4	mA
	Idle current	1.4	μA
	SIFS duration	192	μs
	Payload size	25	bytes
	ACK frame size	10	bytes
Wake-up radio	WuC duration (5 bit address for a network with 32 nodes)	55	ms
	Reception current (WuRx)	1.9	μA
	Idle current	0.39	μA
	Time to switch on MCU	2	ms
	Duration of preamble and partially switch on MCU	13	ms
	Time needed to decode and validate 1 bit address	10	ms

C

B.1. Packet Delivery Ratio

Fig. C.14 depicts the obtained PDR for FAD and ES as traffic load λ varies, for a network with 32 devices. As expected, a WuR-IoT network with both FAD and ES experiences lower PDR with a higher traffic load. This is due to the fact that collision probability increases with the injected traffic. A higher collision probability leads to a lower number of successful transmissions thus a lower PDR. Since ES does not have any impact on WuC duration for TI-WuR transmissions, both schemes show an equal amount of collisions under the same traffic load, hence achieving the same PDR.

Furthermore, the duration of WuCs, which is decided by the adopted address length and WuC data rate, has an impact on collision probability. The longer the WuC duration, the higher the P_c, and vice-versa. This is because a longer WuC occupies the channel for a longer period of time than a shorter WuC. When the channel is occupied over a longer period, the possibility of occurring overlapped

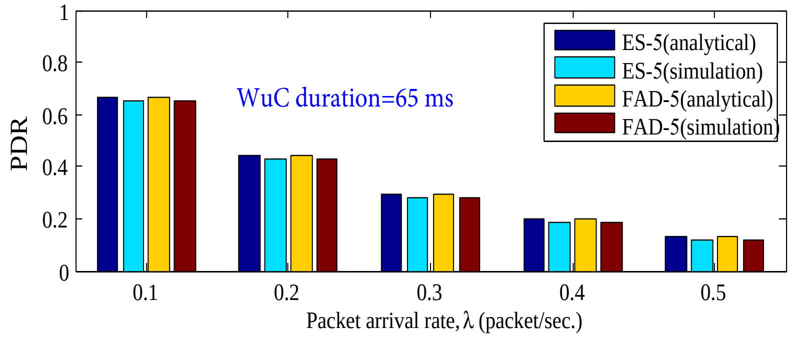


Figure C.14: Packet delivery ratio: FAD-5 versus ES-5.

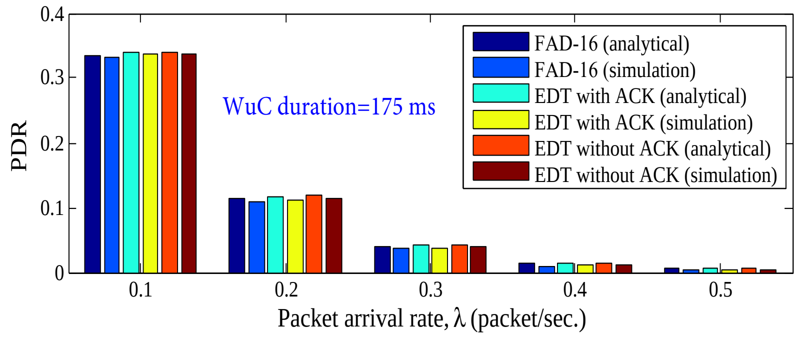


Figure C.15: Packet delivery ratio: FAD-16 versus EDT with and without ACK.

transmissions increases, and consequently P_c rises. The difference between the WuC values used in Figs. C.14 and C.15, i.e., WuC duration = 65 versus 175 ms, explains why the achieved PDR in the upper figure is much higher than the ones achieved in the lower figure.

From Fig. C.15, a similar PDR trend is observed when EDT with and without ACK is compared with the FAD-16 scheme, i.e., PDR decreases with the traffic load. This is due to the fact mentioned earlier, i.e., collision probability increases with the traffic load and a higher collision probability leads to a lower PDR. Moreover, the achieved PDRs for FAD-16, EDT with or without ACK, are nearly equal since the WuC duration in all three cases are the same, as 175 ms.

B.2. Average Latency

Fig. C.16 illustrates the achieved latency for FAD-5 versus ES-5. From this figure, it is clear that both address decoding schemes achieve the same delay performance and the resulted latency does not vary with traffic load. This is because the latency defined in this study is based on successfully transmitted packets. To receive a packet successfully, both FAD and ES have to decode all address bits in a

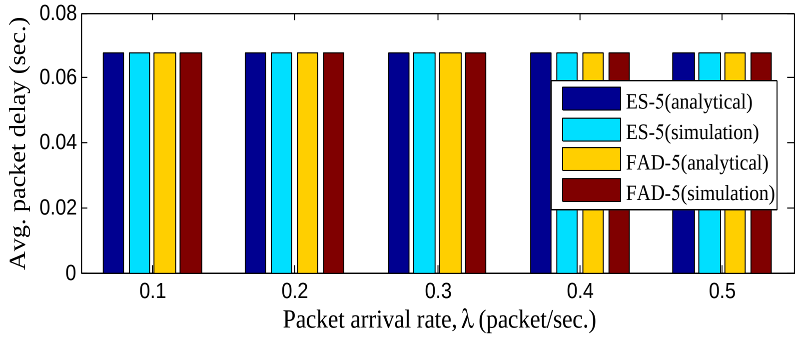


Figure C.16: Latency: FAD-5 versus ES-5.

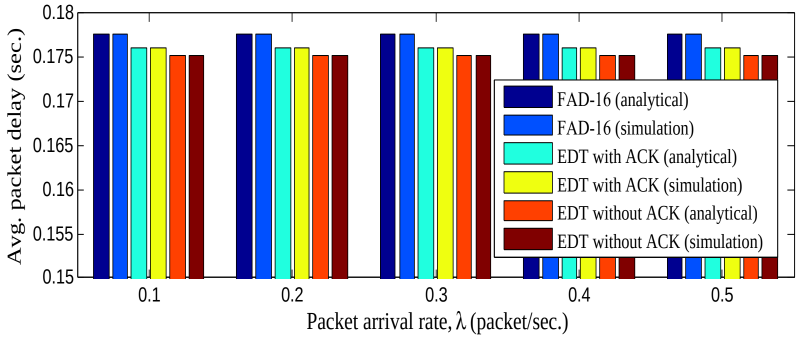


Figure C.17: Latency: FAD-16 versus EDT with and without ACK.

WuC. Thus, a same data transmission procedure applies to both schemes.

Compare now the latency performance among FAD-16, EDT with and without ACK, in Fig. C.17. It is clear that EDT without ACK achieves the best performance. This is due to the fact that EDT without ACK does not need to wake up the MR to transmit an ACK upon a successful data reception. Based on the parameter configuration mentioned above, approximately 2 and 1.7 ms less time is achieved when comparing the latency obtained based on FAD-16 and EDT with ACK respectively[‖]. It is worth mentioning herein that, although these two figures report the results for the same metric, they are based on two different address lengths which correspond to different network sizes. While Fig. C.16 represents the obtained latency when the WuC address length is 5 bits (WuC duration = 65 ms), the results presented in Fig. C.17 are based on a 16 bit address (WuC duration = 175 ms). Clearly, the latency from the latter case will be longer than the one obtained from the former case.

[‖]The reason that the achieved latency reduction is not very significant is as follows. In our testbed implementation, the WuC is sent at 100 bps, whereas the data rate is 125 kbps. With a higher WuC data rate, more significant benefits for EDT will be achieved.

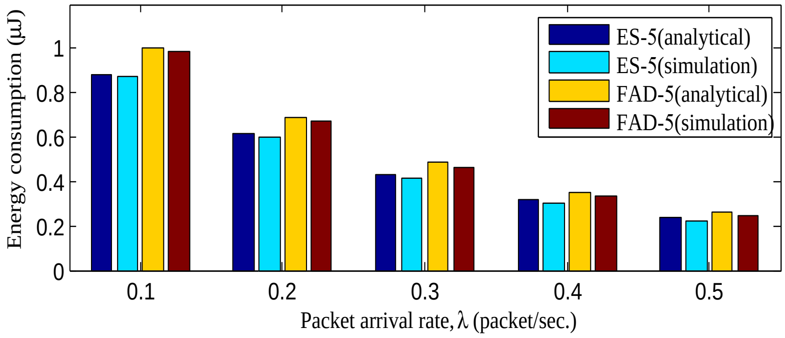


Figure C.18: Average reception energy: FAD-5 versus ES-5.

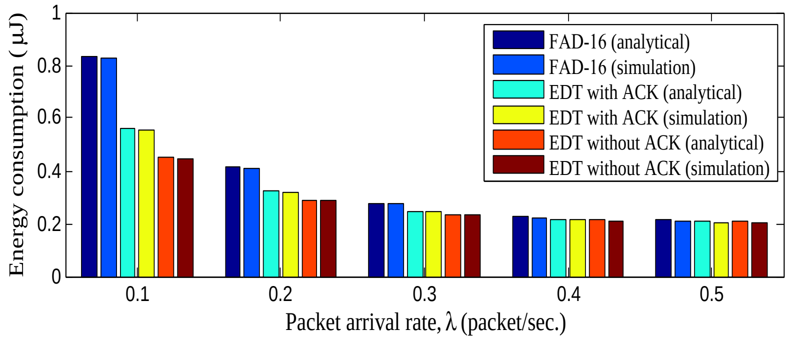


Figure C.19: Average reception energy: FAD-16 versus EDT with/without EDT.

B.3. Average Reception Energy Per Node

The average reception energy consumption per node for the FAD-5 and ES-5 schemes is shown in Fig. C.18. For each data transmission, the overall reception energy for the whole network includes the energy consumptions by the targeted node as well as by the unintended nodes. The targeted node consumes energy for receiving WuC, turning on MCU, data packet reception, radio switching, and transmitting ACK, whereas the non-destined nodes need energy merely for overhearing WuC transmissions. The theoretical average reception energy consumption per node is obtained based on (C.10). We observe that FAD consumes higher energy than ES under all configured traffic loads. This behavior is mainly due to the efficient address decoding mechanism of ES in which non-destined nodes go to sleep at an earlier stage. When traffic load increases, both schemes exhibit a downward trend, as a result of higher collision probabilities which are shown in Fig. C.14.

Fig. C.19 depicts the average reception energy consumption for FAD-16, EDT with and without ACK respectively. When EDT is employed, no matter with or without ACK, lower energy consumption is achieved in comparison with FAD-16, for all studied traffic loads. This is because the EDT schemes transmit data along

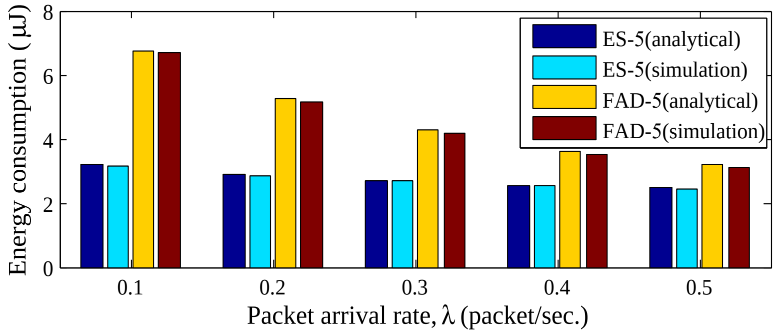


Figure C.20: Overhearing energy consumption: FAD-5 versus ES-5.

with WuCs, saving energy for data transmission and reception. Among FAD-16 and the two variants of EDT, EDT without ACK performs the best since it takes full advantage of low power WuRx, i.e., the MR does not have to wake up. When traffic load becomes heavier, however, the advantage of EDT diminishes. This is because EDT itself does not have any mechanism to avoid overhearing. With a heavy traffic load, the energy consumption due to overhearing becomes a dominant component for total device energy consumption, compromising the benefit brought by EDT in comparison with ES.

Furthermore, when comparing the results from Figs. C.18 and C.19, we observe that the EDT schemes perform better than the ES scheme despite that non-destined nodes can sleep earlier using ES. This is because no separate data packet transmission is required for data transmission in EDT. Note however that EDT applies only to small data, e.g., with 10 bits for data as mentioned earlier in our testbed experiments.

C

B.4. Overhearing Energy under Various Traffic Loads

Fig. C.20 illustrates the variation of overhearing energy consumption of a WuR-IoT as traffic load varies, for ES versus FAD. In both cases, the overhearing energy consumption decreases with the traffic load. The reason is that the number of unsuccessful transmissions increases at a higher traffic load but the unintended nodes are not able to detect those collided packets. When comparing these two schemes, it is evident that ES performs better than FAD since ES reduces the number of overhearing nodes at each step of address decoding and matching. More specifically, ES reduces approximately one half of the number of overhearing nodes after decoding the first bit of a WuC address, and this trend continues bit-by-bit in the address decoding procedure. In the end, only one overhearing node, or none, decodes and validates a whole address in ES, whereas all $(N-2)$ overhearing nodes decode and validate an address in FAD. As observed in this figure, ES reduces overhearing energy wastage by approximately $30\% - 50\%$ as compared to FAD for all the traffic

Table C.4: Estimated Lifetime (in years)

FAD-16	ES-16	EDT with ACK	EDT without ACK
142.6392	142.7832	107.9190	108.2350

loads.

As mentioned in the previous subsection, EDT does not avoid overhearing. To activate EDT, all nodes in the network need to decode and validate the full range of the WuC address. Thus, the overhearing energy consumptions for FAD, EDT with and without ACK are the same. When ES is employed, however, it will perform best in terms of minimizing overhearing since an unintended ES node needs only partial address decoding before going to sleep.

B.5. Lifetime Estimation

Let us now calculate the expected lifetime of the studied network. Assume an ideal homogeneous network in which all nodes behave identically under a collision-free condition. Then the network lifetime is the same as the lifetime of a randomly selected node, assuming that all nodes have the same amount of initial energy, deplete their energy at the same rate, and have therefore the same node lifetime. To perform our lifetime estimation, we follow the packet arrival pattern for typical IoT applications considered in [2, 3], i.e., one packet per every other hour. We further assume that each device is powered by a 3V button/coin cell battery with capacity 220 mAh. A decreasing rate of 2% for battery self-discharging and circuit disintegration is included in our calculations.

Table C.4 depicts the network lifetime for the studies schemes, i.e., FAD, ES, and EDT with two flavors. From Table C.4, we observe that ES performs slightly better than FAD. Among without EDT, EDT with ACK, and EDT without ACK, the EDT scheme without ACK is the best in terms of lifetime. The lifetime of ES is longer than all other schemes since it uses short local WuC address (e.g. 5 bits for the network with N=32) and bit-by-bit address decoding scheme.

The reason for this insignificant lifetime extension of ES over FAD or EDT over without EDT is the fact that the energy consumed for one successful packet transmission is higher than the energy consumed for reception in a WuR-IoT device, i.e., WuC transmission energy is dominating the total energy consumption of a network. We argue, however, the proposed schemes is still meaningful since a real life WSN/IoT application for environment surveillance may experience much lower traffic load as configured in this study.

C. Further Discussions

Finally, it is worth mentioning that in this study we regard concurrent transmissions of two or more devices as collisions regardless of the interference level. When capture effect is taken into account, a transmission which is much stronger than other transmission(s) to the same receiver may survive. In such a case, the achieved packet delivery ratio would be slightly higher but the obtained delay and energy consumption would be basically the same since our schemes do not allow retransmission and a collided packet consumes the same amount of energy. To precisely analyze the performance under more realistic conditions, one may consider that packet losses due to capture effect and channel impairments are statistically independent of losses due to protocol behavior [28].

IX. Conclusions and Future Work

In this paper, we have proposed two schemes, i.e., early sleep and early data transmission, for the purpose of further reducing overhearing and latency in WuR-enabled IoT/WSNs. The proposed schemes are implemented in a small-scale WuR testbed with self-designed wake-up receivers supporting both ES and EDT. The functionalities of these two schemes are demonstrated via real-life experiments. To evaluate the performance of the proposed schemes in larger networks, we developed an M/G/1 model and derived expressions for three parameters. Through analysis and discrete-event simulations, we demonstrate that ES performs better than full address decoding in terms of average energy consumption. Moreover, EDT shows its performance superiority in terms of latency.

A definite direction for our future work is analytically exploring the performance of the proposed schemes by investigating the effect of interference levels among concurrent transmissions, especially under realistic channel conditions. Another direction is to expand our testbed to a larger scale and perform more real-life experiments for metric based performance evaluation.

Paper C: References

[1] D. Ghose, A. Frøytlog, and F. Y. Li, “Reducing overhearing energy in wake-up radio-enabled WPANs: Scheme and performance,” *in Proc. IEEE ICC, 2018*, pp. 1-6.

[2] 3GPP TR 38.913, “Study on scenarios and requirements for next generation access technologies,” v14.3.0, Jun. 2017.

[3] Sierra Wireless, Ericsson, Altair, Sony, Virtuosys, AT&T, Verizon, Sequans, Orange, KDDI, Nokia, NTT Docomo, KT, SoftBank, SK Telcom, Telenor, and

KPN, “Evaluation of LTE-M towards 5G IoT requirements,” *White Paper*, v1.0, Dec. 2017.

[4] R. Casadei, G. Fortino, D. Pianini, W. Russo, C. Savaglio, M. Viroli, “Modelling and simulation of opportunistic IoT services with aggregate computing,” *Future Generation Computer Systems*, vol. 91, pp. 252-262, Feb. 2019.

[5] M. R. Palattella, M. Dohler, Al. Grieco, G. Rizzo, J. Torsner, T. Engel, and L. Ladid, “Internet of things in the 5G era: Enablers, architecture, and business models,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 3, pp. 510-527, Feb. 2016,

[6] B. Holfeld, D. Wieruch, T. Wirth, L. Thiele, S. A. Ashraf, J. Huschke, I. Aktas, and J. Ansari, “Wireless communication for factory automation: An opportunity for LTE and 5G systems,” *IEEE Commun. Mag.*, vol. 54, no. 6, pp. 36-43, Jun. 2016.

[7] G. Fortino, W. Russo, C. Savaglio, W. Shen, and M. Zhou, “Agent-oriented cooperative smart objects: From IoT system design to implementation,” *IEEE Trans. Syst. Man. Cybern. Syst.*, vol. 48, no. 11, pp. 1939-1956, Dec. 2018.

[8] A. Čolaković and M. Hadžialić, “Internet of things (IoT): A review of enabling technologies, challenges, and open research issues,” *Computer Networks*, vol. 144, pp. 17-39, 2018.

[9] G. Fortino, W. Russo, and C. Savaglio, “Agent-oriented modeling and simulation of IoT networks,” *in Proc. IEEE FedCSIS, 2016*, pp. 1-4.

[10] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, “Has time come to switch from duty-cycled MAC schemes to wake-up radio for wireless sensor networks?” *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674-687, Apr. 2016.

[11] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. Popovici, and L. Benini, “Design, implementation, and performance evaluation of a flexible low-latency nanowatt wake-up radio receiver,” *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 633-644, Apr. 2016.

[12] D. Ghose, F. Y. Li, and V. Pla, “MAC Protocols for wake-up radio: Principles, modeling and performance analysis,” *IEEE Trans. Ind. Informat.*, vol. 14, no.5, pp. 2294-2306, May 2018.

[13] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Performance evaluation and comparative analysis of subcarrier modulation wake-up radio systems for energy-efficient wireless sensor networks," *Sensors*, vol. 14, no. 1, pp. 22-51, 2014.

[14] F. A. Aoudia, M. Gautier, and O. Berder, "OPWUM: Opportunistic MAC scheme leveraging wake-up receivers in WSNs," *Journal of Sensors*, vol. 2016, Article ID 6263719, 2016.

[15] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," *in Proc. IEEE INFOCOM, 2015*, pp. 522-530.

[16] V. Jelicic, M. Magno, D. Brunelli, V. Bilas, and L. Benini, "Benefits of wake-up radio in energy-efficient multimodal surveillance wireless sensor network," *IEEE Sensors J.*, vol. 14, no. 9, pp. 3210-3220, Sep. 2014.

[17] Qualcomm Inc., "Physical layer aspects of early data transmission-R1-1807103," *3GPP TSG RAN WG1 Meeting #93*, Busan, Korea, May, 2018.

[18] Ericsson, "Early data transmission for MTC-R1-1805850," *3GPP TSG RAN WG1 Meeting #93*, Busan, Korea, May, 2018.

[19] F. Z. Djiroun and D. Djenouri, "MAC schemes with wake-up radio for wireless sensor networks: A review," *IEEE Commun. Surveys & Tuts.*, vol. 19, no. 1, pp. 587-618, Q1, 2017.

[20] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra low power wake-up radios: A hardware and networking survey," *IEEE Commun. Surveys & Tuts.*, vol. 19, no. 4, pp. 2117-2157, Q4, 2017.

[21] P.-H. P. Wang, H. Jiang, L. Gao, P. Sen, Y.-H. Kim, G. M. Rebeiz, P. P. Mercier, and D. A. Hall, "A near-zero-power wake-up receiver achieving -69 dBm sensitivity," *IEEE J. Solid-State Circuits* , vol. 53, no. 6, pp. 1640-1652, Jun. 2018.

[22] M. R. Abdelhamid, A. Paidimarri and A. P. Chandrakasan, "A -80 dBm BLE-compliant, FSK wake-up receiver with system and within-bit dutycycling for scalable power and latency," *in Proc. IEEE Custom Integrated Circuits Conference (CICC), 2018*, pp. 1-4.

[23] D. Ghose and F. Y. Li, "Enabling backoff for SCM wake-up radio: Scheme and modeling," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1031-1034, May 2017.

[24] L. Guntupalli, D. Ghose, F. Y. Li, and M. Gidlund, “Energy efficient consecutive packet transmissions in receiver-initiated wake-Up radio enabled WSNs,” *IEEE Sensors J.*, vol. 18, no. 11, pp. 4733-4745, Jun. 2018.

[25] J. Lebreton, N. Murad, S. Kandukuri, and R. Lorion, “An energy-efficient addressing mechanism for wake-up radio systems,” *in Proc. IEEE Wireless Telecommunications Symposium (WTS), 2016*, pp. 1-6.

[26] L. Jiao, F. Y. Li, and V. Pla, “Modeling and performance analysis of channel assembling in multichannel cognitive radio networks with spectrum adaptation,” *IEEE Trans. Veh. Technol.*, vol. 61, no. 6, pp. 2686-2697, Jul. 2012.

[27] F. A. Aoudia, M. Magno, M. Gautier, O. Berder, and L. Benini, “Analytical and experimental evaluation of wake-up receivers based schemes,” *in Proc. IEEE GLOBECOM, 2016*, pp. 1-7.

[28] D. Malone, K. Duffy, and D. Leith, “Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions,” *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 159-172, Feb. 2007.

[29] C. Buratti and R. Verdone, “Performance analysis of IEEE 802.15.4 non beacon-mode,” *IEEE Trans. Veh. Tech.*, vol. 58, no. 7, pp. 3480-3493, Sep. 2009.

[30] J. Virtamo, Class Lecture, Topic: “M/G/1 queue,” S-38.3143, College of Engineering and Science, Helsinki University of Technology, Helsinki, Oct., 28, 2005.

Paper D

Title:	Lightweight Relay Selection in Multi-hop Wake-up Radio Enabled IoT Networks
Authors:	Debasish Ghose[†], Luis Tello-Oquendo[‡], Frank Y. Li[†], and Vicent Pla[‡]
Affiliation:	[†]Dept. of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway [‡]Dept. of Communications, Universitat Politècnica de València (UPV), 46022 València, Spain
Conference:	*IEEE Global Communications Conference (GLOBECOM)*, Abu Dhabi, UAE, December 2018.
DOI:	10.1109/GLOCOM.2018.8647449.
Copyright ©:	

D

Lightweight Relay Selection in Multi-hop Wake-up Radio Enabled IoT Networks

Debasish Ghose, Luis Tello-Oquendo, Frank Y. Li, and Vicent Pla

Abstract —**Wake-up Radios (WuRs) are becoming more popular in Internet of Things (IoT) networks owing to their overwhelming advantages such as low latency, high energy saving, and on-demand communication. In a multi-hop WuR-IoT network, route establishment between source and destination prior to actual data communications consumes a significant portion of energy. To reduce energy consumption and protocol overhead for route establishment, we propose a lightweight relay (LR) selection scheme, referred to as LR-WuR, where a lookup table and acknowledgment are used for next hop relay selection. We develop an analytical model to evaluate the performance of LR-WuR. Extensive simulations are performed to validate the accuracy of the analytical model. Furthermore, we present a framework for deriving optimal network parameters.**

I. Introduction

Energy-efficient communication is a key requirement in many Internet of Things (IoT) applications [1]. To reduce energy consumption of IoT networks, communication protocols play an important role. For instance, duty-cycling (DC) medium access control (MAC) mechanisms in IoT networks improve the energy-efficiency of IoT devices by letting nodes be active or sleep cyclically. However, nodes operated in DC-MAC suffer from overhearing and idle listening when they overhear transmissions of neighboring nodes and listen to the channel for control messages. Recently, a paradigm shift has been envisaged from DC-MAC to wake-up radio (WuR) [2]. The key features of WuR are on-demand communication and ultra-low energy consumption [2] [3]. In a WuR-enabled IoT device, a wake-up receiver (WuRx) is connected to a main radio (MR) equipped with a micro-controller unit (MCU). The power consumption of a WuRx is in the order of μW or nW whereas it is in mW for an MR [2] [3]. The MR of a node sleeps most of the time, however, it keeps WuRx always active for receiving wake-up calls (WuCs). Upon receiving a WuC from a sender, the WuRx wakes up the MCU by means of issuing an interrupt. When the MRs of both sending and receiving nodes are awake, data communication occurs. As soon as the data communication cycle ends up with an acknowledgment

(ACK) from the receiver end, both MRs go to sleep but keep their WuRxs active to listen to the channel.

Several cross-layer schemes have been proposed to minimize energy consumption for route establishment. Existing schemes can be divided into three categories: energy-aware; geographic; and flooding based schemes [4]. One of the popular energy-aware cross-layer schemes is OPWUM [5] which opportunistically selects the best relay nodes for forwarding packets based on neighbors' energy. To avoid collisions, backoff (BO) [6] [7], channel sensing (CS), request to send (RTS), clear to send (CTS) and about to send (ATS) handshakes are performed before a WuC transmission. One of the representative flooding based cross-layer WuR schemes is ZIPPY [8]. It allows on-demand flooding through WuR using unaddressed WuCs. ALBA-WuR [3] is a geographic cross-layer solution for WuR-IoT networks. It employs semantic addressing for relay selection. GREENROUTE [9] is another scheme that also uses semantic addressing. The main difference between ALBA-WuR and GREENROUTE lies in the handshake procedure, i.e., ALBA-WuR uses only RTS whereas GREENROUTE performs both RTS and CTS.

One common assumption of ALBA-WuR, GREENROUTE, and OPWUM is that a node can defer their transmission for the duration specified in the network allocation vector (NAV) in the MAC header of the RTS/CTS/ATS packet. Note, however, that such a NAV field is not defined in the MAC frame format of wireless sensor networks (WSNs)/IoT standard IEEE 802.15.4/IEEE 802.15.4e. Moreover, ALBA-WuR and GREENROUTE require a complex relay selection procedure which is not feasible for WuR-IoT networks. The CS and RTS/CTS/ATS procedure increases latency and energy consumption in a network. Flooding-based protocols increase unnecessary traffic load in the network. Moreover, none of the aforementioned studies considered error-prone channels for performance evaluation even though channel status has a significant impact on the performance of the scheme.

The above observations triggered our motivation to propose an energy-efficient relay selection scheme to further reduce the energy consumption of multi-hop WuR-IoT networks. The proposed cross-layer lightweight relay (LR) selection scheme, referred to as LR-WuR, introduces a lookup table and uses ACK for efficient relay selection. We develop an M/G/1 queuing model to assess the performance of LR-WuR and validate the accuracy of the analytical model based on discrete-event simulations. Furthermore, we present a framework to derive optimal network parameters.

The remainder of this paper is structured as follows. In Sec. II, we present the network scenario. Sec. III introduces LR-WuR. In Sec. IV, we develop the

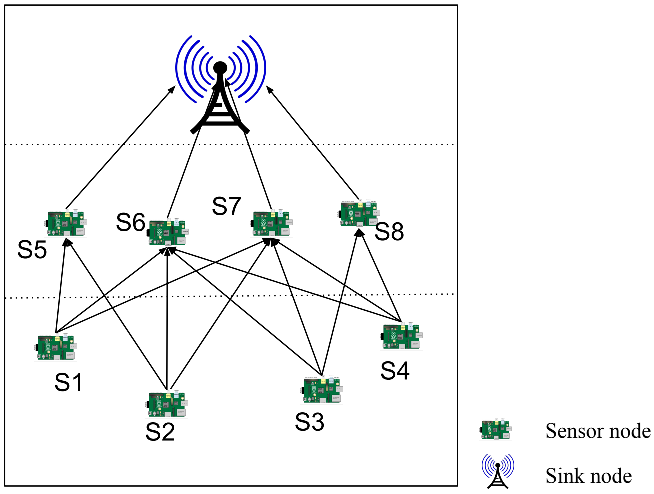


Figure D.1: Illustration of a multi-tier WuR-IoT network.

analytical model. Afterwards, performance metrics are defined in Sec. V. In Sec. VI, we present an optimization framework to derive network parameters. Numerical and simulation results are discussed in Sec. VII, before the paper is concluded in Sec. VIII.

II. Network Scenario and Assumptions

Consider a multi-tier event-triggered data reporting scenario, where sensor nodes generate packets and send them to the sink node through intermediate nodes, as shown in Fig. D.1. All sensor nodes are integrated with a WuRx and can only communicate with its one-hop neighboring nodes within the coverage range. A sensor node sends a unicast WuC to a relay node as soon as it has a packet to transmit. After receiving the WuC, the relay node turns on its MR for data communication. When both MRs are active, data communication will be performed. The same procedure is repeated hop-by-hop until the packet reaches the sink.

We consider that the network is operated under light traffic, which is a reasonable assumption for many smart home, smart farm, and industrial IoT applications [4]. We further assume that no packet loss occurs due to buffer overflow.

III. Protocol Design

In this section, we present the proposed LR-WuR cross-layer relay selection scheme. This scheme reduces overhead by allowing a node to select its next relay node to forward a packet toward the destination *without* performing a route establishment handshake. In a nutshell, LR-WuR uses a lookup table for relay selection. It allows a node to poll a next tier node among a set of potential relays for

D

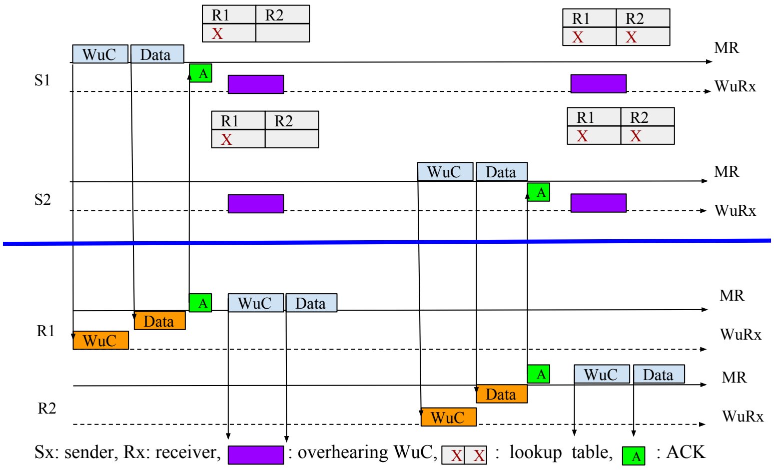


Figure D.2: Operations of LR-WuR.

forwarding a packet and ensures load balancing among relay nodes in the network. To do so, LR-WuR enables a sender to maintain an address lookup table of its next tier nodes to select the next hop forwarder. Each node in the network keeps track of the transmissions of upper tier nodes with whom it is connected by overhearing their transmissions and accordingly marking its address lookup table. When a node has a packet to forward, it checks the address lookup table to select an upper hop node which did not transmit or forward a packet in the previous round(s) of transmission. If all the addresses of next hop nodes are marked, the lookup table will be reset.

The operation of LR-WuR is illustrated in Fig. D.2, where nodes S1 and S2 are connected with two upper tier nodes, i.e., R1 and R2. Initially both R1 and R2 have the same status. When S1 has a packet to send, it wakes up R1 for forwarding its packet. First, it sends a WuC and then transmits the data frame to R1. Upon receiving the packet, R1 sends an ACK to S1. S1 marks R1 as utilized in the lookup table, switches off its MR and keeps active its WuRx for monitoring the channel. When R1 sends a WuC to its next hop node for forwarding the received packet, both S1 and S2 overhear that WuC transmission. Since S1 already marked the forwarder as utilized in the lookup table, it will not make any changes in this table. In the mean time, S2 uses this overheard WuC transmission to update its lookup table. Accordingly, when S2 has a packet to transmit, it will forward its packet to R2 instead since it knows that R1 has already transmitted/forwarded a packet in the previous round of packet transmission. Afterwards, S2 will update its lookup table. When R2 forwards the received packet from S2, S1 will update its lookup table after overhearing the WuC transmission of R2. When all the next hop forwarders are exploited, S1 and S2 will reset their lookup table. Note that if no ACK is received from the re-

D

ceiver, the sending node retransmits the WuC and data after an ACK timeout. It discards the packet if the maximum number of retransmission is exceeded. The same procedure applies to each hop until the final destination is reached.

IV. Analytical Model

In this section, we first develop an M/G/1 queue model to calculate the collision probability of WuC transmissions and then calculate packet error probability based on signal-to-noise ratio (SNR). In a multi-hop WuR-IoT network, a node transmits a WuC to the targeted WuRx in an on-demand manner, i.e., whenever it generates a packet or receives a packet to forward. Collisions among WuCs may occur due to the randomness of packet arrivals in the network. Denote by P_c the probability that a collision happened during the WuC or data transmission. Since no separate ACK is required upon receiving a WuC, a collided WuC can only be detected if no ACK is received from the intended receiver after transmitting the *data frame*. If no ACK is received within the ACK timeout period, the sending node will retransmit its WuC. A node transmits the same WuC up to $(K+1)$ times including the initial transmission and K retransmissions. If the WuC retry limit, K, has been reached, the sending node simply discards the packet from the queue. A successful transmission of the reference node, which is a node of interest randomly selected among neighboring nodes, occurs when none of the other $\sum_{i\epsilon\mathcal{A}} N_i$ nodes starts a busy period during the vulnerable period. Here, $\mathcal{A}$ is the set of neighbors of the reference node and its hidden nodes. A busy period of a reference node is defined as the period in which a node is uninterruptedly busy. A busy period starts with a packet arrival into an empty buffer and finishes when the buffer becomes empty again. A busy period of the reference node is shown in Fig. D.3 where T_{F} ($T_{\mathrm{F}} = T_{\mathrm{SIFS}} + T_{\mathrm{wuc}} + T_{\mathrm{AT}} + T_{\mathrm{data}} + T_{\mathrm{ack}}^{\mathrm{tout}}$) is the duration of an unsuccessful attempt and T_{S} ($T_{\mathrm{S}} = T_{\mathrm{wuc}} + T_{\mathrm{AT}} + T_{\mathrm{data}} + 2\,T_{\mathrm{SIFS}} + T_{\mathrm{ack}}$) is the duration of a successful attempt. T_{wuc}, T_{AT}, T_{data}, T_{SIFS}, T_{ack} and $T_{\mathrm{ack}}^{\mathrm{tout}}$ are the duration of WuC, fully active MCU, data, short inter-frame space (SIFS), acknowledgment, and ACK timeout, respectively.

We assume that $T_{\mathrm{S}} \approx T_{\mathrm{F}}$ since $T_{\mathrm{ack}}^{\mathrm{tout}} \approx T_{\mathrm{SIFS}} + T_{\mathrm{ack}}$. With this assumption, the mean duration of a busy period, $\mathrm{E}[T_{\mathrm{B}}]$, can be obtained as

$$\mathrm{E}[T_{\mathrm{B}}] = T_{\mathrm{S}}\,\mathrm{E}[\Gamma]\,\mathrm{E}[n], \tag{D.1}$$

where Γ is the number of packets served in a busy period of the M/G/1 queuing system with traffic intensity $\alpha = \lambda\,T_{\mathrm{S}}\,\mathrm{E}[n]$ and $\mathrm{E}[\Gamma] = \frac{1}{1-\alpha}$; and $\mathrm{E}[n]$ is the expected

D

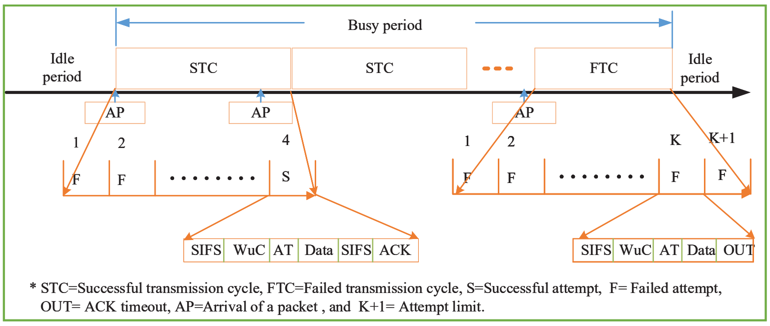


Figure D.3: Illustration of a busy period.

number of transmission attempts for each packet.

$$\mathrm{E}[n] = \sum_{i=0}^{K} P_c^i(1-P_c)(i+1) + (K+1)P_c^{K+1}, \tag{D.2}$$

and P_c is the collision probability of the reference node. For analysis simplicity, we consider that all busy periods have a constant duration equal to its mean, T_B. With this simplification, P_c can be approximated as

$$P_c \approx 1 - \left[e^{-\frac{\lambda}{\mathrm{E}[\Gamma]}(\mathrm{E}[T_\mathrm{B}]+T_\mathrm{S})}\right]^{\sum\limits_{i\epsilon\mathcal{A}} N_i}. \tag{D.3}$$

By solving the non-linear equations (D.1), (D.2), and (D.3), the value of P_c can be obtained.

Furthermore, a packet loss may also occur due to imperfect channel conditions. Denote by ρ the SNR at the receiver. Thus, packet error rate due to channel impairment, denoted by $P_e(\rho)$, can be formulated as

$$P_e(\rho) = 1 - [1 - P_e^{\mathrm{WuC}}(\rho)][1 - P_e^{\mathrm{DATA}}(\rho)], \tag{D.4}$$

where $P_e^{\mathrm{WuC}}(\rho) = 1 - [1 - P_b^{\mathrm{OOK}}(\rho)]^{L_{\mathrm{WuC}}}$ is the WuC packet error rate (PER) due to channel impairment, in which L_{WuC} is the WuC packet size [in bits] and P_b^{OOK} is the bit error rate (BER) of the on-off keying (OOK) modulation; $P_e^{\mathrm{DATA}}(\rho) = 1 - [1 - P_b^{\mathrm{BPSK}}(\rho)]^{L_{\mathrm{DATA}}}$ is the data PER due to channel impairment, in which L_{DATA} is the data packet length [in bits] and P_b^{BPSK} is the BER of the binary phase shift keying (BPSK) modulation.

For a given link, $P_b^{\mathrm{OOK}}(\rho)$ and $P_b^{\mathrm{BPSK}}(\rho)$ can be determined based on the adopted noise and propagation model of the channel. Furthermore, it is a common assumption that ACKs are sent as error-free packets [10].

Since packet loss occurs due to both collision and transmission error because of channel impairment, the probability of packet loss, denoted by P_l, is given by

$$P_l = P_c + (1 - P_c)P_e. \tag{D.5}$$

V. PERFORMANCE ANALYSIS

In this section, we define three performance metrics and derive their expressions based on the obtained P_l.

A. Packet Delivery Ratio

Packet delivery ratio (PDR), denoted by Φ^{e2e}, is defined as the ratio between the number of packets successfully delivered to the final destination/sink node and the number of packet generated by a source, e.g., the reference node. The PDR in a multi-hop WuR network can be obtained from packet loss probability considering packet losses due to both collision and channel impairments. Accordingly, Φ^{e2e} is computed as

$$\Phi^{\text{e2e}} = \prod_{i=1}^{H} (1 - P_{l_i}^{K+1}), \tag{D.6}$$

where P_{l_i} is the packet loss probability at the i^{th} hop, K is the retry limit and H is the total number of hops between source and sink.

B. Average End-to-End Delay

The average delay of a successful received packet, denoted by T^{e2e}, is defined as the time from the moment a packet arrives at a data generation node until it is successfully transmitted to the sink. As mentioned earlier, transmissions may fail due to both collisions and channel impairment. Thus T^{e2e} can be obtained as

$$T^{\text{e2e}} = T_{\text{S}} \sum_{i=1}^{H} \frac{\sum_{v=0}^{K} P_{l_i}^{v}(1 - P_{l_i})(v + 1)}{1 - P_{l_i}^{K+1}}, \tag{D.7}$$

where T_{S} is the duration of a successful transmission. Propagation delay is ignored in the calculation as it is very small.

C. Energy Consumption

The end-to-end energy consumption is quantified by E^{e2e} (in Joule), which represents the average energy consumed by the nodes along the path from the reference node to the sink to deliver a packet successfully. E^{e2e} depends on the PDR in a multi-hop network, and it is calculated as

$$E^{\text{e2e}} = E_{\text{T}} \sum_{i=1}^{H} \frac{\sum_{v=0}^{K} P_{l_i}^{v}(1 - P_{l_i})(v+1)}{1 - P_{l_i}^{K+1}}, \tag{D.8}$$

where E_{T} is the energy consumed by a sender and a receiver for a successful transmission computed as $E_{\text{T}} = E^{T}_{\text{wuc}} + E_{\text{AT}} + E^{T}_{\text{data}} + E^{R}_{\text{wuc}} + E^{R}_{\text{data}} + 2\,E_{\text{SIFS}} + E^{T}_{\text{ack}} + E^{R}_{\text{ack}}$. Subsequently, E^{T}_{wuc}, E^{R}_{wuc}, E_{AT}, E^{T}_{data}, E^{R}_{data}, E_{SIFS}, E^{T}_{ack}, and E^{R}_{ack} are the energy consumption for WuC transmission, WuC reception, fully activation of MCU, data packet transmission, data packet reception, SIFS, ACK transmission and ACK reception, respectively. The average energy consumed by a node along the transmission path from the reference node to the sink for each packet it handles, denoted by E_{avg}, can be computed as

$$E_{\text{avg}} = E_{\text{S}} \sum_{v=0}^{K} P_{l}^{v}(1 - P_{l})(v+1) + E_{\text{F}}(K+1)P_{l}^{K+1}, \tag{D.9}$$

where $E_{\text{S}} = E^{T}_{\text{wuc}} + E_{\text{AT}} + E^{T}_{\text{data}} + 2\,E_{\text{SIFS}} + E^{T}_{\text{ack}}$ and $E_{\text{F}} = E^{T}_{\text{wuc}} + E_{\text{AT}} + E^{T}_{\text{data}} + E_{\text{SIFS}}$. $\sum_{v=0}^{K} P_{l}^{v}(1 - P_{l})(v+1)$ represents a condition where a transmission is successful (including WuC and data), whereas $(K+1)P_{l}^{K+1}$ represents another condition where a packet is discarded after $(K+1)$ transmission attempts.

VI. Parameter Optimization Framework

In this section, we develop a parameter optimization framework for the studied WuR-enabled multi-hop IoT network.

Different IoT applications have different quality of service (QoS) requirements ranging from reliable communication or higher energy-efficiency to delay-sensitive applications or any combination of them. We consider three application dependent weights: w_E, w_T, and w_Φ for the average energy consumption, average latency, and PDR objectives, respectively. Then, different degrees of QoS requirements for various IoT applications can be achieved by adjusting the specific weight value according to a specific IoT application where $0 \leq w_{\text{E}}, w_{\text{T}}, w_{\Phi} \leq 1$ and $w_{\text{E}} + w_{\text{T}} + w_{\Phi} = 1$.

Then, aiming at simultaneously optimizing network parameters based on the application requirements in terms of T^{e2e}, E^{e2e}, and Φ^{e2e}, we consider a multi-objective optimization function which is quantified by a weighted linear combination of each objective. As these three objective terms differ in units in which these performance metrics are measured, we normalize each metric and optimize their deviations concerning certain pre-defined utopia values which are used to provide non-dimensional objective functions [11]. Accordingly, the overall objective

Table D.1: Cross-Layer Optimization Framework

Inputs:
Φ^*, E^*, T^*, Φ^{TH}, E^{TH}, T^{TH}, $K^{\max}$, w_{E}, w_{T}, w_{Φ}
Objective:
$\underset{\theta\epsilon\mathbb{R}^3}{\text{minimize}} \quad J(\theta) = w_{\text{E}} J_E(\theta) + w_{\text{T}} J_T(\theta) + w_{\Phi} J_{\Phi}(\theta)$
Subject to:
(D.11), (D.12), (D.13).

function for the envisaged WuR-enabled multi-hop IoT network becomes

$$J(\theta) = w_{\text{E}} J_E(\theta) + w_{\text{T}} J_T(\theta) + w_{\Phi} J_{\Phi}(\theta), \quad J(\theta) \in \mathbb{R}^3 \tag{D.10}$$

where $\theta = (\text{WuC}_{TR}, K, \rho) \in \mathbb{R}^3$ is the vector of decision variables: WuC transmission rate, number of retransmissions, and minimum SNR, respectively; $J_E(\theta) = \left(\frac{E^{\text{e2e}}}{E^*} - 1\right)$, $J_T(\theta) = \left(\frac{T^{\text{e2e}}}{T^*} - 1\right)$, $J_{\Phi}(\theta) = \left(1 - \frac{\Phi^{\text{e2e}}}{\Phi^*}\right)$; E^*, T^*, and Φ^* are, respectively, the end-to-end energy consumption, latency, and PDR utopia values for normalizing purposes.

Moreover, we consider three performance requirements in terms of energy consumption, maximum E^{TH}, maximum latency, T^{TH}, or minimum end-to-end PDR, Φ^{TH}. These requirements are set as constraints of the optimization problem. The corresponding system constraints are given by

$$E^{\text{e2e}} \leq E^{\text{TH}}, \tag{D.11}$$

$$T^{\text{e2e}} \leq T^{\text{TH}}, \tag{D.12}$$

$$\Phi^{\text{e2e}} \geq \Phi^{\text{TH}}. \tag{D.13}$$

Based on the input parameters, i.e., E^{TH}, T^{TH}, Φ^{TH}, E^*, T^*, Φ^*, most suitable network parameters in the set θ are found such that $J(\theta)$ is optimized. The decision variables take values from a discrete set that leads the optimization framework to an integer programming problem. The size of this problem is small, allowing MCUs to yield a solution in few seconds through an exhaustive search. The optimization framework is summarized in Table D.1.

VII. Numerical Results and Discussions

In this section, we present the numerical results obtained from the analytical model and discrete-event simulations. Consider a connected network where 15 nodes are distributed as a 5×3 grid over an area of 250 m $\times$ 100 m and these nodes send measured information to the sink, which is located on top of the deployment area.

Table D.2: Parameter Configuration [2] [6] [7]

Radio type	Parameter	Value	Unit
Common	Supply voltage	3	V
	Coverage range	100	m
	Number of hops	3	
	Packet arrival rate	0.001, 0.1, 1	packet/sec.
	SNR	3 to 10	dB
Main radio	Data rate	250	kbps
	Transmission current	17.4	mA
	Reception current	18.8	mA
	Idle current	20	μA
	SIFS duration	192	μs
	Payload size	35	bytes
	ACK frame size	11	bytes
Wake-up radio	WuC duration	12.2, 6.1	ms
	WuC Transmission current	152	mA
	Reception current (WuRx)	8	μA
	Sleep current	3.5	μA
	MCU switching current	2.7	μA
	Time to switch on MCU	1.79	ms
	WuC packet size	4	bytes
	Maximum WuC attempts	0 to 7	times
	WuC transmission rate	2730, 5460	bps

To perform simulations, we made a custom-built discrete-event simulator which mimics the behavior of the WuR-IoT network. That is, a node sends a WuC to its next tier relay node as soon as it has a packet to transmit. The targeted WuRx switches on its MR for performing data communication as soon as it receives a WuC. This procedure will continue until the packet successfully reaches the sink or is dropped after the retry limit has been exceeded. A timestamp is maintained for each packet that records from the instant of arrival until when it is successfully delivered to the sink or is dropped at any hop. We configure $\lambda = 0.1$ packet/sec to reflect the light traffic condition in the studied network. The remaining parameters that are configured in our analysis and simulations are based on Table D.2 unless otherwise stated.

A. PDR, Delay, and Energy Performance

Fig. D.4 depicts the obtained PDR as ρ varies. As expected, a WuR-IoT network experiences lower PDR when channel quality is poor, i.e., ρ is low. This is because the BER is higher with a low ρ, leading to a lower number of successful transmissions thus a lower PDR. Moreover, the PDR of the proposed LR-WuR scheme is higher when retransmissions are allowed, as illustrated in Fig. D.5 for different number of retransmissions when $\rho = 6$ dB. This behavior is expected since retransmission

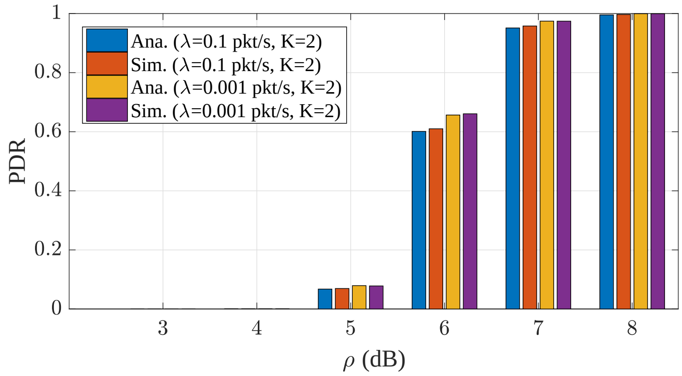


Figure D.4: PDR as a function of ρ when the WuC duration is 12.2 ms.

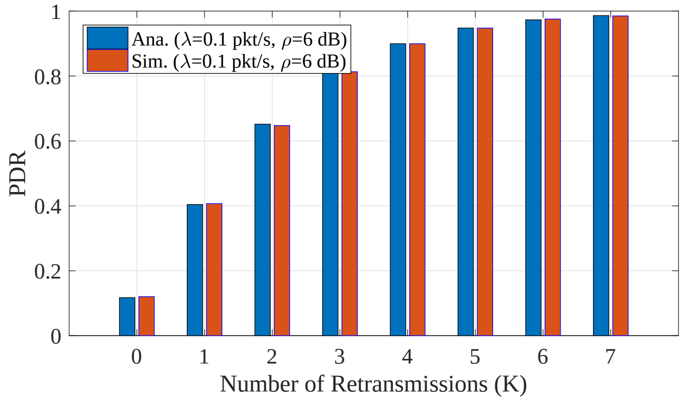


Figure D.5: PDR as a function of the number of retransmissions when the WuC duration is 12.2 ms.

increases the probability of successful transmission at a cost of higher delay and energy consumption as explained next.

Fig. D.6 shows the average end-to-end delay performance as ρ varies. The obtained end-to-end delay decreases with ρ. The reason is that channel quality improves when ρ increases, which reduces the average number of transmission attempts required to successfully deliver a packet to the sink. Comparing $K = 4$ and $K = 1$, the latter one performs better at 6 and 7 dB since more transmissions are allowed when $K = 4$. Furthermore, both $K = 4$ and $K = 1$ show the same performance when channel quality is excellent, i.e., $\rho = 8$ dB, because a lower number of retransmissions is required.

From Fig. D.7, a similar trend regarding the average energy consumed by a node for transmitting a packet (successfully or discarded) is observed. That is, energy consumption with $K = 4$ is higher than that with $K = 1$ at 6 and 7 dB. The reason is that channel errors are overcome through retransmissions at a cost of higher energy consumption. The energy performance with $K = 4$ and $K = 1$ is almost the same at 8 dB since a lower number of retransmissions is performed.

D

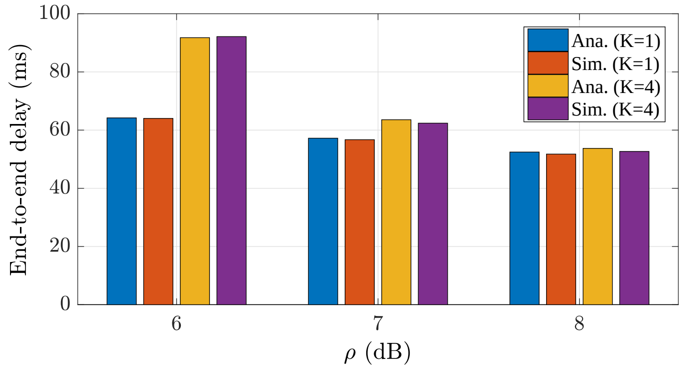


Figure D.6: Average end-to-end delay as a function of ρ when the WuC duration is 12.2 ms.

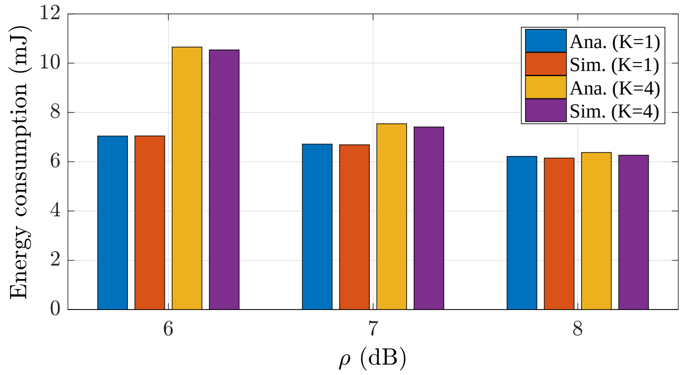


Figure D.7: Energy consumption of a node for each packet it handled (successfully or discarded) when the WuC duration is 12.2 ms.

It is worth mentioning that there often exists a tradeoff between PDR and delay/energy consumption when channel quality is poor. That is, a high PDR is achievable under error-prone channel conditions at the cost of a longer delay and higher energy consumption. This observation conforms with the PDR behavior illustrated in Fig. D.5.

B. Effect of Higher WuC Transmission Rate

Fig. D.8 presents the effects of WuC transmission rate on the performance of LR-WuR in terms of PDR and end-to-end latency under error-free channel conditions. From Fig. D.8 (a), it is evident that the PDR increases with WuC transmission rate. This is due to the fact that a node needs a shorter time to transmit a WuC when a higher data rate, i.e., 5460 bps is adopted in comparison with a lower transmission rate, i.e., 2730 bps. A shorter WuC duration leads to a reduced busy period, resulting in a lower collision probability. As a result, PDR increases. The impact of higher WuC transmission rate on average end-to-end packet delay is showed in Fig. D.8 (b). The end-to-end delay of LR-WuR with a higher WuC rate (at 5460 bps)

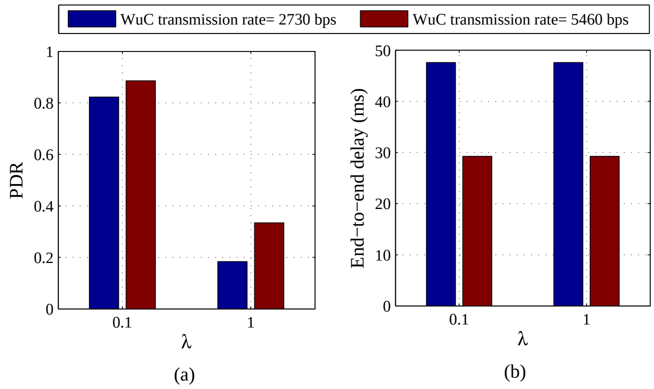


Figure D.8: Effect of WuC transmission rate when $K = 0$.

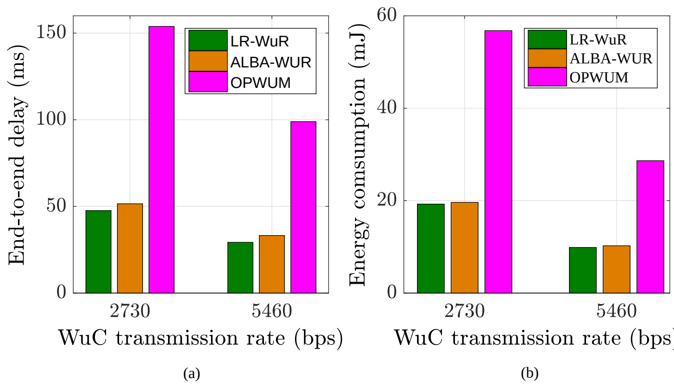


Figure D.9: Comparison of LR-WuR, ALBA-WUR, and OPWUM.

is shorter than at 2730 bps. This is because a higher WuC transmission data rate reduces WuC duration thus achieving a lower average end-to-end delay.

C. Performance Comparison with ALBA-WUR and OPWUM

In this subsection, we compare the performance of LR-WuR with two representative cross-layer WuR schemes mentioned earlier, i.e., OPWUM [5] and ALBA-WUR [3]. OPWUM performs BO/CS/RTS/CTS/ATS handshakes prior to data transmission. On the other hand, ALBA-WUR sends WuC and waits for RTS before data transmission. Since both ALBA-WUR and OPWUM were evaluated under an error-free channel condition without allowing retransmissions, we configure $\rho = 10$ dB and $K = 0$ so that LR-WuR is operated under the same conditions for a fair comparison. The results are illustrated in Fig. D.9 with a traffic intensity of $\lambda = 0.001$ packet/sec. From Fig. D.9(a), it is observed that LR-WuR outperforms ALBA-WuR and OPWUM in terms of average end-to-end delay. This is because LR-WuR only sends WuC and does not perform BO/CS/RTS/CTS/ATS or RTS like OPWUM or

Table D.3: Optimal Network Parameters for Different QoS Requirements

Threshold metric	WuC$_{TR}$	K	ρ
PDR ($\geq$ 0.9)	5460 bps	2	$\geq$ 7 dB
Delay ($<$ 30 ms)	5460 bps	0	$\geq$ 6 dB
Energy Consumption ($<$ 20 mJ)	5460 bps	0	$\geq$ 6 dB
Delay ($<$ 30 ms) and PDR ($\geq$ 0.9)	5460 bps	0	10 dB
Delay ($<$ 30 ms) and Energy Consumption ($<$ 20 mJ)	5460 bps	0	$\geq$ 6 dB

ALBA-WUR respectively, prior to a WuC transmission. A similar trend is also observed for end-to-end energy consumption in Fig. D.9(b). That is, LR-WuR consumes less energy than its counterpart schemes OPWUM and ALBA-WUR. This is due the reason that OPWUM and ALBA-WUR consume energy respectively for BO/CS/RTS/CTS/ATS/DATA/ACK and WuC/RTS/DATA/ACK, whereas LR-WuR only consumes energy for WuC/DATA/ACK. Furthermore, OPWUM requires the longest delay and consumes the highest end-to-end energy among these three schemes since it requires more control messages than the other two schemes.

D. Optimal Network Parameter Configuration

Aiming at providing differentiated services for applications with different QoS requirements, we analyze finally the combination of QoS requirements and provide optimal configuration values for the studied WuR-IoT network so that the selected requirement is satisfied. We first analyze each QoS requirement (PDR, delay, or energy consumption) and provide the network parameters (WuC$_{TR}$, K, ρ) that satisfy them individually. After that, we analyze the combination of QoS requirements: $[(\Phi^{\text{TH}} > 0.9)\&(D^{\text{TH}} < 30\,\text{ms})]$ and $[(D^{\text{TH}} < 30\,\text{ms})\&(E^{\text{TH}} < 20\,\text{mJ})]$. The tested values for the WuC transmission rate are WuC$_{TR}$ $= \{2730, 5460\}$ bps, the packet arrival rate is set to be 0.1 packet/sec, the number of retransmissions allowed ranges from 0 to 7, and the SNR ranges from 3 dB to 10 dB.

As can be observed in Table D.3, most of the QoS requirements are satisfied when the number of retransmissions is kept at a lower value, as $K = 2$, WuC$_{TR}$ $= 5460$ bps, and the channel quality is high ($\rho \geq 7$ dB). More specifically, if the aim is to reduce delay and energy consumption in the WuR network, retransmissions should be disabled and the WuC rate should be kept at a high value, such as WuC$_{TR}$ $= 5460$ bps.

D

VIII. CONCLUSIONS

In this paper, we proposed a lightweight relay selection scheme, LR-WuR. LR-

WuR efficiently selects a relay node from the set of next hop forwarders by utilizing a lookup table and ACK. An M/G/1 model is developed to assess the performance of LR-WuR. The obtained analytical and simulation results match with each other. We also presented an optimization framework. Based on the application requirement, optimal network parameters can be derived by applying this optimization framework. Furthermore, LR-WuR is compared with two state-of-the-art WuR schemes, i.e., ALBA-WUR and OPWUM. The obtained results demonstrate that LR-WuR outperforms these two schemes in terms of average end-to-end delay and end-to-end energy consumptions.

Paper D: References

[1] Z. Zhou, F. Xiong, C. Xu, Y. He, and S. Mumtaz, "Energy-efficient vehicular heterogeneous networks for green cities," *IEEE Trans. Ind. Informat.*, vol. 14, no. 4, pp. 1522-1531, Apr. 2018.

[2] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674-687, Apr. 2016.

[3] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," *in Proc. IEEE INFOCOM, 2015*, pp. 522-530.

[4] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra low power wake-up radios: A hardware and networking survey," *IEEE Commun. Surveys & Tuts.*, vol. 19, no. 4, pp. 2117-2157, Q4 2017.

[5] F. A. Aoudia, M. Gautier, and O. Berder, "OPWUM: Opportunistic MAC protocol leveraging wake-up receivers in WSNs," *Journal of Sensors*, vol. 2016, Article ID 6263719, 2016.

[6] D. Ghose, F. Y. Li, and V. Pla, "MAC protocols for wake-up radio: Principles, modeling and performance analysis," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2294-2306, May 2018.

[7] D. Ghose and F. Y. Li, "Enabling backoff for SCM wake-up radio: Protocol and modeling," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1031-1034, May 2017.

[8] S. Felix, B. Bernhard, B. Jan, and T. Lothar, "Zippy: On-demand network flooding," *in Proc. ACM SenSys, 2015*, pp. 45-58.

[9] S. Basagni, V. Di Valerio, G. Koutsandria, and C. Petrioli, "Wake-up radio-enabled routing for green wireless sensor networks," *in Proc. IEEE VTC-Fall, 2017*, pp. 1-6.

[10] F. Borko and S. A. Ahson, *Long Term Evolution: 3GPP LTE Radio and Cellular Technology*, New York, CRC Press, 2016.

[11] R. Marler and J. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369–395, 2004.