U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Improving QoS for IEEE 802.15.4e DSME Networks

John Harrison Kurunathan

Supervisor: Prof. Eduardo Manuel Medicis Tovar

Co-Supervisor: Dr. Ricardo Augusto Rodrigues da Silva Severino

Co-Supervisor: Prof. Anis Koubaa

Co-Supervisor: Prof.Luis Miguel Pinho de Almeida

Programa Doutoral em Engenharia Electrotécnica e de Computadores

January, 2021

Faculdade de Engenharia da Universidade do Porto

Improving QoS for IEEE 802.15.4e DSME Networks

John Harrison Kurunathan

Dissertation submitted to Faculdade de Engenharia da Universidade do Porto
to obtain the degree of

Doctor Philosophiae in Electronic & Computer Engineering

President: Dr. José Alfredo Ribeiro da Silva Matos

External Referee: Dr. Ye-Qiong Song

External Referee: Dr. Zdenek Hanzalek

Internal Referee: Dr. Paulo Portugal

Internal Referee: Dr. Pedro Souto
Supervisor: Prof. Eduardo Manuel Medicis Tovar

January, 2021

Abstract

Wireless Sensor Networks have been enabling a large bandwagon of applications and usage in the industrial, domestic and commercial areas. Advancements in microelectronics and communication technologies during the past decade has been fueling the increasing pervasiveness and ubiquity of these infrastructures, eventually paving a way to support the paradigm of Internet of Things. Specifically in the industrial domain, there are applications which put-forth strict requirements in terms of timeliness and reliability. Other QoS properties such as scalability, energy efficiency and robustness must also be taken into concern when pushing these infrastructures towards a possible reality.

This thesis explores the functionalities of the IEEE 802.15.4e MAC behaviors, keeping the the Deterministic Synchronous Multi-channel Extension (DSME) as the major focus. The legacy IEEE 802.15.4 standard was enhanced by the IEEE 802.15.4e with the augmentation of functionalities such as frequency hopping, dedicated and shared timeslots and several techniques to improve scalability and the energy efficiency of the application. In this thesis we present some architectural solutions (mechanisms, algorithms, protocol add-ons) that can address some of the most prominent Quality of Service (QoS) challenges in terms of timeliness, scalability, robustness and energy-efficiency

In order to efficiently address the network demands, in terms of QoS aspects such as latency, resources, and reliability, it is mandatory to carry out a thorough network planning. Modeling the fundamental performance limits of the networks is of paramount importance to understand their behavior under the worst-case conditions and to make the appropriate design choices. In this thesis, we accurately compute the worst case bounds of a network and provide a deep insight towards applications which these MAC behaviors would suit based on their respective network architectural properties.

In regards with timeliness, the IEEE 802.15.4 was one the legacy standards that introduced the concept of guaranteed timeslots to ensure time bounded communication. The DSME and TSCH MAC behaviors of the IEEE 802.15.4e improved more in this front by proving multi-channel access and increasing the available bandwidth, and concurrently increasing scalability. However, there lies a challenge in scheduling the guaranteed timeslots stringently and using the capabilities of the standard to its fullest extent. This thesis provides scheduling mechanisms and cross-layer architectures to reduce the overall latency and improve the reliability and reduce the power consumption of the network.

As we now move towards the paradigm of increased pervasiveness in the field of Wireless Sensor Networks, there is a need for architectures that have the capability to adapt on run-time to suit the prerequisites of the underlying application. Despite the capability to support complex networks, there is a need for adaptable infrastructures to provide good QoS without any dire trade-offs. In this thesis, we provide one such mechanism that can alter the network infrastructure based on its demands.

i

As the DSME protocol has the capability to facilitate communication with bounded time, it opens possibilities of adaption into several time-critical application domains. In this thesis, we explore the possibility of the adaption of the DSME protocol for time and safety critical Advanced driving assistance systems (ADAS) systems. To effectively test and validate these systems, there is a need for tools that can support the simulation of these complex communication infrastructures from the control and the networking perspective. This thesis introduces a co-simulation framework that enables the simulation of an ADAS application scenario in these two fronts, analyzing the relationship between different vehicle dynamics and the delay required for the system to operate safely, exploring the performance limits of different wireless network configurations.

Resumo

As redes de sensores sem fio (WSN) têm possibilitado uma grande variedade de aplicações e uso nas áreas industrial, doméstica e comercial. Os avanços nas tecnologias de microeletrónica e comunicação durante a última década resultaram no aumento da difusão e da omnipresença dessas infraestruturas, eventualmente abrindo um caminho para apoiar o paradigma da Internet das Coisas (IoT). Especificamente no domínio industrial, existem aplicações que apresentam requisitos rigorosos em termos de precisão e confiabilidade. Outras propriedades de QoS (Qualidade de Serviço), como escalabilidade, eficiência energética e robustez, também devem ser levadas em consideração ao empurrar essas infraestruturas para uma realidade possível.

Esta tese explora as funcionalidades dos comportamentos MAC IEEE 802.15.4e, mantendo o DSME (Extensão Multicanal Síncrona Determinística) como o foco principal. O padrão IEEE 802.15.4 foi aprimorado pelo IEEE 802.15.4e com o aumento de funcionalidades como salto de frequência, intervalos de tempo dedicados e compartilhados e várias técnicas para melhorar a escalabilidade e a eficiência energética do aplicativo. Nesta tese, apresentamos algumas soluções arquitetónicas (mecanismos, algoritmos, complementos de protocolo) que podem abordar alguns dos desafios mais importantes de Qualidade de Serviço (QoS) em termos de pontualidade, escalabilidade, robustez e eficiência energética

Para atender com eficiência às demandas da rede, em termos de aspectos de QoS, como latência, recursos e confiabilidade, é obrigatório realizar um planeamento completo da rede. A modelagem dos limites fundamentais de desempenho das redes é de suma importância para entender seu comportamento nas piores condições e fazer as escolhas de "design" apropriadas. Nesta tese, calculamos com precisão os limites de pior caso de uma rede e fornecemos uma visão profunda dos aplicativos que esses comportamentos MAC atenderiam com base em suas respectivas propriedades arquiteturais da rede.

Por precisão, o IEEE 802.15.4 foi um dos padrões que introduziu o conceito de intervalos de tempo garantidos (GTS) para garantir a comunicação com limite de tempo. O DSME e o TSCH MAC do IEEE 802.15.4e melhoraram mais nessa frente, provando acesso multicanal e aumentando a largura de banda disponível e aumentando simultaneamente a escalabilidade. No entanto, existe um desafio em agendar rigorosamente os intervalos de tempo garantidos e usar os recursos do padrão em toda a sua extensão. Esta tese fornece mecanismos de agendamento e arquiteturas de camada cruzada para reduzir a latência geral e melhorar a confiabilidade e o consumo de energia da rede.

À medida que avançamos agora em direção ao paradigma de maior penetração no campo das redes de sensores sem fio, há uma necessidade de arquiteturas que tenham a capacidade de se adaptar no tempo de execução para atender aos pré-requisitos do aplicativo subjacente. Apesar da capacidade de oferecer suporte a redes complexas, há necessidade de infra-estruturas adaptáveis para fornecer boa QoS sem quaisquer compromissos negativos. Nesta tese, fornecemos um desses mecanismos que podem alterar a infraestrutura de rede com base em suas exigências.

Como o protocolo DSME tem a capacidade de facilitar a comunicação com tempo limitado, ele abre possibilidades de adaptação em vários domínios de aplicativos críticos para o tempo. Nesta tese, exploramos a possibilidade de adaptação do protocolo DSME para sistemas críticos de tempo e segurança, sistemas avançados de assistência à direção (ADAS). Para testar e validar efetivamente esses sistemas, são necessárias ferramentas que possam suportar a simulação dessas infraestruturas de comunicação complexas do ponto de vista do controle e da rede. Esta tese apresenta uma estrutura de co-simulação que permite a simulação de um cenário de aplicativo ADAS nessas duas frentes, analisando a relação entre diferentes dinâmicas do veículo e o atraso necessário para o sistema operar com segurança, explorando os limites de desempenho de diferentes configurações de rede sem fio.

Acknowledgments

The thought of doing a doctorate in philosophy was considered a gargantuan task to me some years back. Despite being one of my most exciting endeavors in life, this was a task that required patience, focus, smart work, love for science and above all an unfettered faith from family and peers. Only tremendous support of several key people can help in succeeding such endeavors. First and foremost I would like to thank my advisors Prof. Eduardo Tovar and Dr. Ricardo Severino for guiding me through every step of this exciting journey. I had great learning experiences and tremendous motivation towards research which is something I aspire for.

Prof. Eduardo Tovar has been a key person in this journey by providing insightful ideas and an enthusiastic approach towards research. His constructive criticism and encouragement has helped me immensely to understand the importance in defining the opportunities in science and help building a better tomorrow. Despite his very busy schedule, he has always allocated time to review my research and provide valuable feedbacks. Additionally, I would also like to thank Dr. Eduardo Tovar for his indispensable role in directing the CISTER lab and upholding a great research environment.

Working under the supervision of Dr. Ricardo Severino has helped me to look into various paradigms of a research problem and taking a practical approach towards research. I would like to thank him for sitting with me through everyday-meetings and being insightful to help through the work process. I thank him for reviewing the publications and providing valuable comments for improvement. I will treasure the interactions with him and his technical acumen and teamwork helped through various obstacles

I would like to extend immense gratitude to Dr. Anis Koubaa for providing brilliant insights in this research work. Despite being countries apart, Dr. Anis Koubaa was kind and diligent to allocate time and provide valuable feedback in this work. His thought-process and inquisitiveness are some of his many qualities that I hope to imbibe in myself. His guidance towards providing attention to detail and perfection in work were very inspiring.

I would like to thank Prof. Luis Almeida and Prof. Jose Silva Matos for letting me to serve as a part of the Comissão de Acompanhamento and allowing me to be part in organizing the doctoral congress of FEUP. It helped me to understand the academic community better. I would like to thank Dr. Walter Tiberti, Bruno Vieira and Lukas my co-authors for being a part in exploring some of the important aspects of my research. At this moment, I would also like to remember some young minds such as Pedro Neto, Madhusudhan and Tiago Pinto to whom I had the fortunate opportunity to co-supervise their theses.

My fellow researchers and students at CISTER lab have been immensely helpful and been very supportive. I specially thank Geoffrey Nelissen, Patrick Yomsi, Ali Awan, Ragunathan Rajkumar, Kai Li, Claudio Maia, Ramiro Robles, Michele Albano, Borislav Nikolic, Joao Lourelio, Konstantiyn and Artem Burmyakov for providing several opportunities to learn and brainstorm on various research aspects. I thank Ines Almeida, Sandra Almeida and Cristiana Barros for handling bureaucracies and administrative issues. I specially thank Dr. Hazem Ismail Ali for

the long walks, tea and debates and Humberto Carvalho for making me feel home and i thank them for livening up my days during PhD and and hor helping to create a good work-life balance. I also thank Syed Aftab Rashid for being an amazing partner in crime and being the best travelling companion. I also extend my warm gratitude to Mubarak Ojewale,Jing jing Zheng, Shashank Gaur, Radha reddy, Enio Filho, Ghazi Amor, Sangeeth Nila, Jatin Arora, Amir Hossein, Giann, Ishfaq, Javier and Yillian who helped creating a educative working environment.

I specially would like to thank Dr. Jayaparvathy for her constant support, encouragement and guidance in my academics. I would like to thank James Mendelssohn, Emanuel Mesquita and Fatima Serro for allowing me a part of their choral groups which were really helpful to ease the mind and accelerate my thought process. I extend my thanks to my friends Antonio Ferreira, Renato Marinho and Guillherme Amaral, Jose Ramos Veiga who made me feel home during my years of stay in Porto. I also remember my brothers Samuel Solomon, Wagner Sam Fredrick, Kandavel, Jagannathan, Harish Swaminathan, Jean Daniel, Anand Krishnan and sister Janani Arivudayanmabi for being a motivation and moral support through all these years. I also thank my in-laws for their love and prayers.

All these would not have been possible without the remarkable words of my late grandfather Mr. Gurunathan who once told me that education is a limitless process and one must learn as much as he can. With these words as an inception, coupled with diligent efforts and prayers of my father John. G. Pandidurai who did immense efforts for my schooling which i can never repay and the love of my mother V. Juliet who fostered me with knowledge since I was a child has always been the driving force of my day-to-day life. The work ethic of my mother as a teacher for more than 25 years has inspired me to be a part and to contribute to the academic community. My family's faith in me has been the guiding light to all the footsteps I have taken so far. Finally, I do not have enough words to express my gratitude to my loving wife Monica Harrison who has constantly encouraged me and filled me with positive mindset that has helped me move more focused towards the goal. I specially thank her for reading the thesis and giving me a general third person perspective to my research work.

J. Harrison Kurunathan

Publications

Journals

1. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "IEEE 802.15. 4e in a nutshell: Survey and performance evaluation." IEEE Communications Surveys Tutorials 20.3 (2018): 1989-2010.

2. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "DynaMO—Dynamic Multisuperframe Tuning for Adaptive IEEE 802.15. 4e DSME Networks." IEEE Access 7 (2019): 122522-122535.

3. Kurunathan, John Harrison. "Study and overview on WBAN under IEEE 802.15. 6." U. Porto Journal of Engineering 1.1 (2015): 11-21.

Conference and Workshops:

1. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "Worst-case bound analysis for the time-critical MAC behaviors of IEEE 802.15. 4e." 2017 IEEE 13th International Workshop on Factory Communication Systems (WFCS). IEEE, 2017.

2. Kurunathan, John Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "Routing Aware DSME Networks." Proceedings of the 3rd Doctoral Congress in Engineering (DCE 2019). 2019.

3. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "DynaMO: dynamically tuning DSME networks." ACM Sigbed Review 16, no. 4 (2020): 8-13.

4. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "Symphony: routing aware scheduling for DSME networks." ACM Sigbed Review 16, no. 4 (2020): 26-31.

5. Kurunathan, Harrison, Ricardo Severino, and Eduardo Tovar. "WiCAR - Simulating towards the Wireless Car." 15th International Workshop on Dependable Smart Embedded Cyber-Physical Systems and Systems-of-Systems (DECSoS) (2020).

6. Li, Kai, Kurunathan, Harrison, Ricardo Severino, and Eduardo Tovar. "Cooperative key generation for data dissemination in cyber-physical systems." 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS). IEEE, 2018.

7. Tiberti, Walter, Bruno, Vieira, Kurunathan, Harrison, Ricardo Severino, and Eduardo Tovar. "Tightening up security in low power deterministic networks" 2020 IEEE 15th International Workshop on Factory Communication Systems (WFCS). IEEE, 2020.

Posters and Work in Progresses

1. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "Towards worst-case bounds analysis of the ieee 802.15. 4e." Work in Progress Session, 22nd IEEE Real-Time Embedded Technology Applications Symposium. Institute of Electrical and Electronics Engineers, 2016.

2. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. "An Efficient approach to Multisuperframe tuning for DSME networks." 2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN). IEEE, 2018.

Technical Reports

1. Kurunathan, Harrison, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. RPL over DSME: A Technical Report. CISTER-Research Centre in Realtime and Embedded Computing Systems, 2018.

"The best and safest method of philosophizing seems to be first to inquire diligently into the properties of things, and establishing those properties by experiments, and then to proceed more slowly to hypotheses for the explanation of them"

Sir. Issac Newton

x

Contents

List of Figures

List of Abbreviations

ADAS	Advanced Drive assist system
AMCA	Asynchronous multi channel Adaptation
ASN	Absolute Slot Number
BI	Beacon Interval
BLIS	Blind Spot Information System
BO	Beacon Order
CAP	Contention Access Period
CCA	Clear Channel Assessment
CFP	Contention Free Period
CoAP	Constrained Application Protocol
COTS	Commercially Off the Shelp Technology
CPS	Cyber-Physical Systems
CRC	Cyclic Redundancy Check
CSL	Coordinated Sampled Listening
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSP	Constraint Satisfaction Problem
DAG	Direct Acyclic Graph
DAO	Destination Advertisement Object
DIO	DODAG Information Object
DIS	DODAG Information Solicitation
DODAG	Destination Oriented Direct Acyclic Graph
DSME	Deterministic Synchronous Multi-channel Extension
DV	Distance Vector
EB	Enhanced Beacon
ECU	Electronic Control Unit
ETSI	European Telecommunications Standards Institute
ETX	Expected Transmission Count
EUI	Extended Unique Identifier
FFD	Fully Functional Device
GACK	Group Acknowledgment
GINA	Guidance and Inertial Navigation Assistant motes
GTS	Guaranteed Time Slots

IE	Information Elements
IFS	Inter Frame Spacing
ILP	Integer Linear Programming
IoT	Internet of Things
IR-UWB	Infra Red - Ultra Wide Band
ISA	International Society of Automation
ISM	industrial, scientific and medical
LE	Low Energy
LLDN	Low Latency Deterministic Network
LoRaWAN	Long Range Wide Area Network
LQI	Link Quality Indicator
LR-WPAN	Low Rate Wireless Personal Area Network
MAC	Medium Access control Layer
MD	Multi-superframe Duration
MDT	Maximum Dedicated Timeslot Algorithm
MLME	MAC layer management entity
MO	Multi-superframe Order
NC	Network Calculus
OCP	Objective Code Point
ODE	Open Dynamics Engine
OF	Objective Function
PAN	Personal Area Network
PDR	Packet Delivery Ratio
QoS	Quality of Service
RFD	Reduced Functional Device
RFID	Radio Frequency Identification Blink
RIT	Receiver Initiated Transmissions
ROS	Robotic Operating System
RPL	Routing Protocol
RSSI	Receiver Signal Strength Indicator
SD	Superframe Duration
SO	Superframe Order
TDMA	Time Division Multiplexing Access
TSCH	Time Slotted Channel Hopping
VLSI	Very Large Scale Integrated Systems
WBAN	Wireless Body Area Networks
WirelessHART	Wireless Highway Addressable Remote Transducer Protocol
WSN	Wireless Sensor Networks

Chapter 1

Introduction

1.1 Research Context

Every century had a dawn and a rule of a certain technology. From the day man used doves for communication to our modern days of ubiquitous mobile phones, the field of communication and networking has been developing exponentially. The humongous improvement in the field of microelectronics has expertly shaped human life in several aspects towards betterment.

We belong to an era of the technical-industrial revolution, where the world is moving towards the paradigm of the Internet of Things in which every device will be sensing, communicating, and even performing appropriate actions. The term Internet of Things abbreviated as "IoT" is coined for a system that communicates and actuates through the Internet in real-time. IoT systems are one step ahead of the Cyber-Physical Systems (CPS), by providing seamless interconnection between the computational and physical elements to implement a task. In any IoT system, a specific set of requirements or quality demands is embodied for a unique application. In such scenarios, these systems need a rethinking of the legacy computing and networking concepts to achieve a robust interaction between the computing devices and the environment.

Adoption of IoT is a stepping stone for the creation of smart homes, smart buildings, and, ultimately, smart cities. This paradigm shift is quite visible by the increasing number of smart devices from phones, assistants, cars, and even smart bulbs that we use in the day-to-day life. All this has been possible due to the gargantuan development in the field of microelectronics such as memories, batteries, energy scavenging, Very Large Scale Integration (VLSI), and hardware design. However, there is a constant need for Large scale - low power communication infrastructures. This is one of the significant facts that fueled the origins of " *Wireless Sensor Networks (WSN)*".

Wireless Sensor Networks enable a wide bandwagon of possibilities to be implemented in real-time applications, such as building automation, industrial automation, environmental control, disaster management, and even personal health care. All these applications are received and recognized by the end-users based on the quality aspects of the system, which can

also be abbreviated as Quality of Service (QoS). QoS is always associated with a non-functional property such as throughput, robustness, latency, end-to-end delay, and reliability. Several standards and protocols are released almost everyday to support the infrastructure of these QoS demanding applications.

Despite the induction of standards and protocols to support these applications, there is an utmost need to understand their behavior and enhance them to use it to their fullest capability. It is also essential to develop new mechanisms, algorithms, and network infrastructures to facilitate the QoS aspects that these applications may impose. Test and validation of these proposals in real WSN hardware will further help to foster the technology towards faster adoption. There is room for much research to be done to understand the bounds of the communication protocols and to determine the boundaries of the quality that the protocol can achieve.

There is also a constant need for developing more maintainable and scalable systems. The primary key here is to have a system that would be less power-hungry, and, at the same time, would not compromise on any QoS aspects. The challenge lies in determining the correct trade-off in these systems. The current state-of-technology reveals a strong immaturity and lack of solutions such as efficient protocols and network architectures concerning these trade off based QoS properties. Due to the lack of quality reliant mechanisms and proper QoS trade-off strategies, many protocols are left astray despite their capability to support prominent network infrastructures.

Researchers have established bounds for several standards to determine an application domain for the respective protocol. There is also work done to enhance these standards to suit the quality demands the application imposes. Despite these efforts, there is an undeniable gap between the ever-growing demands of the industry and what the standard offers. This gap has to be diminished through extensive research in implementing methodologies that can improve the overall QoS of the system without any dire compromise. As the technology matures, the gap between simulations and real WSN deployments can be abridged. Eventually, there will be a possibility to create de-facto standards for network communication, so that users can set up these networks without in-depth technical knowledge.

Background in WSN

The term "Wireless Sensor Network" means *"a network in which several nodes are sensing and communicating through a wireless medium to perform an application-specific task"*. The primary component of a Wireless Sensor Network is the *sensor*. It converts physical data into a readable digital context. The physical parameters that a sensor can measure can be temperature, humidity, sound, pressure, motion, vibration, atmospheric variables like humidity, pollutants, and even health related parameters such as glucose level, drug level, heartbeat or pulse.

A sensor node is made up of a sensor or sensors, a memory device, a micro-controller, a energy device, and a transmitter or sometimes a transceiver. A sensor node should be capable

of sensing, storing, processing, and receiving information. Wireless Sensor Networks can also alleviate the difficulty of manual data collection in large-scale deployments. Once deployed, a Wireless Sensor Network should have the capability to operate autonomously without any human intervention. Some of the various application fields where Wireless Sensor Networks are used are listed below:

- **Environmental sensing:** Wireless Sensor Networks have been used to monitor natural phenomena such as volcanoes [1], glacial environment [2], and even the soil-water content measurement [3]. One of the key advantages of a Wireless Sensor Network is to acquire data unobtrusively. Scalable interoperable networks are in demand for sensing the environment; they can provide a wide array of data such as seismic, acoustic, and even high-resolution images.

- **Structural health monitoring:** Sensor networks have been used to monitor the condition of building structures [4], civil infrastructures like bridges [5], coal mines [6] and maintenance of cultural assets like monuments [7]. The major advantage of using WSN in the field of structural health monitoring is the low cost and minimal effort needed to collect abundant information on the structural integrity of the buildings. Some of these systems do not have the luxury to operate with a steady power supply. Several low-energy architectures that support WSN make it a strong candidate for structural monitoring applications.

- **Body sensor networks:** The advent of sensor networks is a massive boon to the medical industry resulting in several health monitoring systems [8], [9]. It also has been used in physical rehabilitation [10] and pervasive health care monitoring [11]. Body sensor-based network infrastructures have the ability to provide determinism and strict security measures [12].

- **Disaster management:** Wireless Sensor Networks have been used to avert disasters and help controlling and recovering from the damage. They have been helpful in forest fire monitoring [13], earthquake monitoring systems [14] and robot-assisted search and rescue systems [15]. More recent research is being carried out where drones are used to control accident-prone traffic wirelessly [16].

- **Industrial sensing:** Sophisticated sensing and control is provided to industries through WSNs [17]. They are used to remotely monitor pipelines, gas leaks, corrosion, equipment condition, and the performance of the equipment [18], or they can play a crucial role in the safety concerns of an industry [19]. Industrial sensing network infrastructures must have the capability to maintain strict timeliness properties. As these systems are considered to be safety-critical (i.e., hazardous to humans), a large effort has to be taken to avoid problems like a single point of failure, which is very common in wired networks.

The Institute of Electrical and Electronics Engineers Standards Association (IEEE-SA) is an organization under IEEE which is responsible for publishing standards for communications, power, energy, robotics, avionics, information technology and many other areas of electrical and electronics engineering.

Several communication standards are published to support the QoS demands of a niche set of applications. For instance, in Wireless Body Area Networks (WBAN), the applications require the WBAN to provide communication with a very minimal delay and should not have any compromise on the integrity of data. IEEE 802.15.6 [20] is the standard that supports WBAN. This standard specifies a network infrastructure in such a way that deterministic transmissions of data can be carried out in a secure channel shielding it from any interference.

In this thesis, we keenly focus on the IEEE standard 802.15.4e. This standard was an evolution of the IEEE 802.15.4 standard to accommodate industrial sensing requirements. Several Quality of Service (QoS) aspects, such as latency, throughput, power consumption, reliability, or robustness are given the utmost importance in the field of industrial sensing. This standard provides various technologies and mechanisms that help the user define bounds for the network in terms of latency and scalability.

1.2   Motivation and challenges

The IEEE 802.15.4 standard [21] is one of the most notable standards specified to support Low Rate Wireless Personal Area Networks (LR-WPAN). The IEEE 802.15 working group was established in 2003 and was instrumental in the developement of several technologies like the ZigBee [22] or the WirelessHART [23]. The IEEE 802.15.4 defined the two lowest layers of the protocol stack, namely the physical layer and the data link layer.

Even though the IEEE 802.15.4 standard was suitable for low rate and low energy communications, its features were not suitable for the required determinism and reliability imposed by the industrial applications. Industrial sensing requires assurance for very high reliability, robustness, low latency, and even scalability. Therefore, in 2012, the IEEE standard 802.15.4-2011 was published with the intention to enhance and add functionalities to the IEEE 802.15.4 Medium Access Control (MAC) to make it suit the demands of the industrial domain. This amendment is called IEEE 802.15.4e, where 'e' is the abbreviation for *enhanced*. IEEE 802.15.4e provided five different MAC behaviors with the intention to fulfil the more stringent requirements of a variety of industrial applications.

Careful network planning is mandatory to address the Quality of Service demands in terms of latency and resources. Learning and modeling these networks helps us to understand the boundaries of the standard and to provide a surety for the industries to adopt them for usage. Research efforts on providing accurate bounds for these standards will help in bridging the gap between the academic and the industrial sectors. The IEEE 802.15.4e is now superseded with the IEEE 8021.5.4 - 2015, but still is supported by the crucial MAC behaviors introduced in the former standard.

Despite the several enhancements the IEEE 802.15.4e brought to the table, it had several drawbacks. For instance, there were no proper scheduling mechanisms that could suffice the demands of a large-scale WSN deployment. Unlike the classical IEEE 802.15.4, which accommodates both star and hierarchical topologies, the IEEE 802.15.4e standard specifies more complex topologies such as mesh. The major challenge here lies in integrating a viable network layer and providing a definitive architecture that can support fixed bounded transmissions (determinism) on the network.

Among the other challenges, a successful network must find it easy to allocate the necessary resources with high flexibility on the go. With the ever-growing IoT and industrial applications, there is a need for standards that can support dynamic networks. However, the main problem the dynamic networks have to encompass is to adjust the resources based on necessity. If the high road is provided by allocating maximum bandwidth, several QoS properties like delay and bandwidth will be compromised. Therefore, there is a need for efficient mechanisms to support the dynamic nature of IoT networks.

There is also a need to explore possible use-cases and scenarios in which this technology can fit in. A challenge was open for the research community to identify, validate, and provide necessary claims for a technology to be used in industrial applications. Providing solutions for use-case scenarios will help the industrialists to build reliable systems that are capable of shifting the QoS intelligence from the user on to the network infrastructure. This can further aid in enabling numerous real-world applications and pave the way towards a smarter, interconnected, QoS-efficient world.

1.3   Approach

The approach taken in this dissertation contribute to the practical use of the IEEE 802.15.4e protocol is threefold. First, we analyse the protocol by studying the MAC behaviors, their drawbacks and the applications these MAC behaviors support. Then we devise mechanisms using network simulators and mathematical modeling to improve the state-of-the-art QoS levels, these networks can provide. Finally, we explore use-cases where this standard can be adopted and learn its respective behavior.

In line with analysing the standard, it is crucial to know the worst-case bounds of the network to assure determinism in an industrial application. These industrial applications impose stringent requirements in terms of determinism, scalability, and robustness. Most of these Quality-oriented properties must be supported by the underlying communication layers to enable an efficient IoT application. At this point, we rely predominantly on the IEEE 802.15.4e protocol as a baseline, and in particular, at the MAC sublayer that supports deterministic, synchronous multi-channel communication. We analyse the time-critical MAC behaviors using mathematical modeling and analyse their respective non-functional properties.

In accordance with improving the network properties of the standard, this thesis will focus on the implementation of new QoS mechanisms and algorithms that will aim at further

improving the overall efficiency in terms of network throughput, delay, reliability, and energy-efficiency. The proposals we provide in this research work predominantly target in alleviating the crucial QoS demands, which in turn can help in improving the possibility of a wide-spread adoption of the IEEE 802.15.4e in the industrial field.

At the MAC sublayer, there have been several unique enhancements like the provision of multi-channel access and and multi-channel adaption. Such kind of mechanisms have the ability to greatly improve the available bandwidth. However, although there is a possibility of supporting deterministic large-scale real-time traffic, proper scheduling mechanisms to assure the Quality of Service in terms of latency and reliability are absent. Both scalability and reliability have to be equally addressed, by providing innovative network infrastructure ideas and scheduling mechanisms to support complex mesh networks. We tackled this problem by providing a novel multi-channel scheduling algorithm that helps the network to achieve maximum quality without any dire trade-offs.

The IEEE 802.15.4e provides several enhancements to the MAC sublayer, but there is a complete lack of tailored network layers to support the topologies the standard can offer. For instance, there is a lack of flexibility to adapt to the changes of a dynamic IoT network at run-time. This thesis provides solutions to networks that must have the capacity to dynamically accommodate and adapt to the changing traffic flows without varying of the network infrastructure.

Most importantly, to properly identify the QoS challenges and to validate the solutions to them, a real-life application must be devised. In this research work, we design the IEEE 802.15.4e network to support an intra-vehicle communication scenario. The intention of our proposal is to validate, test and foster the potential of our network infrastructures to be employed in real-world-IoT applications in the near future.

1.4   Thesis statement

The objective of this thesis is to devise architectural solutions (mechanisms, algorithms, protocol-add-ons) for supporting strict QoS requirements (throughput, scalability, timeliness and energy-efficiency) in large-scale WSN infrastructures to enable effective IoT communication. In this thesis, we predominantly rely on the IEEE 802.5.4e, and particularly on the MAC behavior called Deterministic Synchronous Multi-channel Extension (DSME). Thus, the thesis proposition can be stated as follows:

The IEEE 802.15.4e protocol is augmented with a set of QoS mechanisms that can effectively support the demands that emerging IoT and industrial IoT deployments may impose

1.5   Contributions

The contributions of this thesis are as follows:

C1: An in-detail network analysis for the time-critical MAC behaviors of the IEEE 802.15.4e by determining their worst-case bounds for various Quality of Service they provide.

C2: The design of a detailed network architecture to implement a QoS-aware routing protocol on top of the IEEE 802.15.4e MAC sublayer.

C3: The design of a novel time-critical scheduling mechanism that also provides traffic differentiation for the multi-channel access in IEEE 802.15.4e DSME networks.

C4: The design of an adaptive scheduling algorithm to support dynamically evolving large scale IoT networks.

C5: An implementation of the DSME MAC behavior for an intra-car communication.

With C1 we provide a clear network model for all the MAC behaviors of IEEE 802.15.4e. All the MAC behaviors of IEEE 802.15.4e support the network using different network infrastructures and properties. We modeled the worst-case bounds for all the MAC behaviors based on Network Calculus, a theoretical formulation that helps in defining and understanding the structural properties of an integrated network such as scheduling, delay dimensioning, and buffering of a network system for specific bounds of traffic. The results correspondent to this research were published [24], [25], and are detailed in the Chapter 4 of this thesis.

Regarding C2 and C3, we provide a detailed network architecture combining the capabilities of Routing Protocol for Lossy networks (RPL), a classic routing protocol, and DSME. Though DSME supports complex topologies like mesh, it lacks in providing a routing layer to support a reliable network infrastructure. In regards to this issue, we provide a methodology for integrating RPL on top of a DSME network. We also provide an algorithm to reduce the overall latency by scheduling transmissions of the network in an efficient manner. We provide a traffic differentiation technique in this scheduling algorithm to prioritize certain data transmissions. The results correspondent to this research were published [26] and are presented in the Chapter 5 of this thesis.

Regarding contribution C4, published in [27], [28], and presented in Chapter 6, we provide a solution to the problem of scalability in a dynamically evolving IoT network. One of the characteristics that IoT networks need to exhibit is the capacity to adapt to a change in the network without any significant compromise in the Quality of Services provided. DSME networks have the capability to provide a deterministic service, but it mainly relies on static settings that are defined at the beginning of the operation. We provide a tuning algorithm that can tune these settings on the go and provide better results in terms of throughput, latency, and bandwidth utilization.

Initially regarding contribution C5, published in [29], and presented in Chapter 7, we take a classic intra-car communication scenario and analyze the performance of DSME for its functioning and communication. We evaluate DSME to be a possible candidate for communication support in safety-critical applications. These applications demand low latency for communication and very high reliability. We define a specific path-loss model for intra-car communication and we also evaluate the QoS for this stringent application scenario. We learn the impact of DSME on the crash rate of the vehicle and provide methods to alleviate them. We also demonstrate the functionalities of the DSME network and advantages our proposed network infrastructures in a robotic-network co-simulator.

1.6   Outline of the thesis

The rest of this dissertation is organized as follows

In Chapter 2 we give a brief introduction to the IEEE 802.15.4e protocols and discuss their features as compared to the parent protocol. We provide in-depth technical details of the protocols and the enhancements, which will serve as a base-point for the mechanisms we propose and evaluate later in chapters 5 and 6 of the thesis.

In Chapter 3 we provide a literature survey on the the relevant related works that address the modelling and analysis of IEEE 802.15.4e. Limitations of these works are analysed and identified in that chapter. We also identify some open issues which serve as a baseline for the solutions proposed in this thesis.

In Chapter 4 we provide a complete network model of the time-critical MAC behaviors of the IEEE 802.15.4e, using Network Calculus (NC). We discuss concepts such as delay bounds, service curves, latency, and throughput analysis. We then provide a detailed performance analysis of all the MAC behaviors based on NC. That chapter gives also an insight into the worst-case bounds of the MAC behaviors and the parameters that have an impact on the overall Quality of Service levels.

In Chapter 5 we devise an architecture for implementing routing over the traditional IEEE 802.15.4e deterministic networks. We also propose a novel scheduling algorithm to schedule the routes provided by the routing layer in a time-efficient way.

In Chapter 6 we propose a novel tuning algorithm that has the capability to tune the parameters of the network in order to accommodate a dynamic number of nodes joining or leaving the network. We provide an in-depth network analysis for all the problems the proposed tuning algorithm will alleviate.

In Chapter 7 we discuss a use-case of an intra-car communication in which DSME is used as the MAC sublayer and analyze its respective QoS levels. We also provide an architecture towards the integration of a robotic simulator to emulate a real-life intra-vehicle scenario that communicates using DSME. We analyze several real-life problems like crashes based on the communication carried out in the DSME layer.

Finally, we conclude with Chapter 8, where we briefly discuss the contributions and outline some potential future research directions.

Chapter 2

IEEE 802.15.4e : An overview

Over the last decade there have been a few standards that have aimed at satisfying the demands of low-power wireless communications [30], [31], [32]. A paradigmatic example is the IEEE 802.15.4 [21], first published in 2003 for WPAN (Wireless Personal Area Networks). The protocol defines only the physical and data-link layers; thus, a few proposals such as the ZigBee [33] or the RPL [34] protocols followed to complement the communications stack.

2.1   IEEE 802.15.4 - LR-WPAN standard

The IEEE 802.15.4 standard was issued by Internet Engineering Task Force (IETF) in the year 2006, and it was aimed at facilitating communication for low rate wireless personal area networks. The components of the IEEE 802.15.4 are described briefly as follows:

- **Components of the WPAN:** In the IEEE 802.15.4 standard, devices can be classified into Fully Function Devices (FFD) and Reduced Function Devices (RFD). The Fully Function Devices (FFD) encompass all the capabilities such as routing, association and formation of a network. The PAN Coordinator is a FFD that acts as the main controller to which other devices may be associated. It is responsible for the time synchronization of the entire network. Sometimes, a FFD can also act as a Coordinator providing local synchronization services and routing to its neighbors. Every coordinator must be associated to a PAN Coordinator and it forms its own network if it does not find other networks in its vicinity. The Reduced Function Device (RFD) is typically the end node of an IEEE 802.15.4 network. A RFD is intended for applications that are extremely simple, such as a light switch or a passive infrared sensor, which are typically synced with a coordinator and are not capable of a routing functionality.

- **Physical layer:** As shown in Figure 2.1, the IEEE 801.15.4 operates in three different frequency bands: 2.4 GHz (with 16 channels); 915 MHz (with 10 Channels) and 868 MHz (with only one channel). The data rate also varies depending on the bands used. The 2.4 GHz band operates with a data rate of 250 Kbps. The 915 MHz and the 868 MHz bands

operate at 40 and 20 Kbps, respectively. The physical layer is responsible for the activation and deactivation of the radio transceiver, for measurement of the link quality, for clear channel assessment and for channel selection.
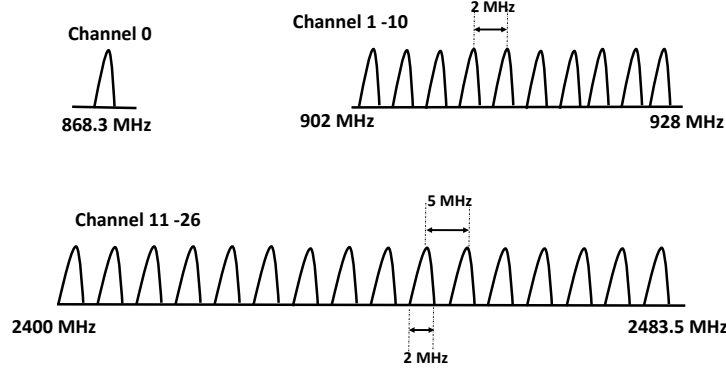
Figure 2.1: Operating frequency bands

- **MAC sublayer:** The MAC sublayer for the IEEE 802.15.4 is designed to work either on a beacon enabled or a non-beacon enabled mode. In the beacon-enabled mode, the entire network is synchronized using periodic beacons and is supported by a structure denoted as the superframe (Figure 2.2). The superframe consists of an active period, where data transmission occurs, and an inactive period during which the device enters a sleep state to save energy. The active period is further divided into the Contention Access Period (CAP) and the Contention Free Period (CFP). During the CAP, the nodes in the network contend to access the channel using slotted CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). The CFP is equipped with 7 Guaranteed Time Slots (GTS), which are used by the nodes that require guaranteed bandwidth.
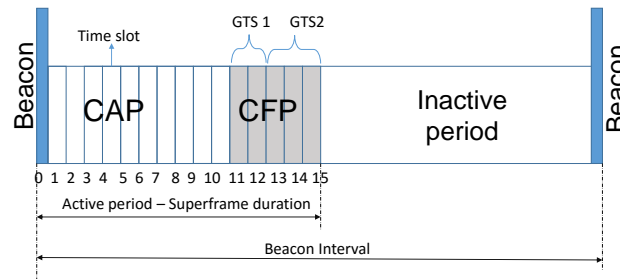
Figure 2.2: Superframe structure of 802.15.4

2.2 Enhancements provided by the IEEE 802.15.4e

The IEEE 802.15.4e [35] was proposed as an amendment of the legacy IEEE 802.15.4 standard, to satisfy the requirements of emerging IoT applications, particularly in the industrial domain.

The IEEE 802.15.4e only provides enhancements to the MAC sublayer, leaving the physical and security layers untouched. Akin to its predecessor, the IEEE 802.15.4e sets the encryption algorithm for cyphering the data for transmission, but the standard does not specify any particular authentication policies to be implemented.

The IEEE 802.15.4e defines five MAC behaviors, instead of following a more conservative "one-size-fits-all" strategy. Hence, it improves its flexibility in accommodating different kinds of application requirements. In general, these new MAC behaviors are quite different from the ones considered in the legacy IEEE 802.15.4-2011. From the proposed MAC behaviors, the Deterministic Synchronous Multi-channel Extension (DSME) is perhaps the closest to the legacy protocol, but nonetheless it brings significant enhancements to the IEEE 802.15.4 beacon-enabled mode by implementing multi-channel frequency hopping and Group Acknowledgments. The other MAC behaviors present far more substantial changes. The Time Slotted Channel Hopping (TSCH) uses fixed size Time Division Multiplexing Access (TDMA) timeslots and multi-channel hopping. The Asynchronous Multi-Channel Adaptation (AMCA) is perhaps the one that most resembles a non-beacon enabled mode in an IEEE 802.15.4 network. On the other hand, the Low Latency Deterministic Network (LLDN) uses a TDMA approach to support deterministic traffic.

The various enhancements provided by IEEE 802.15.4e are discussed next:

- **Multi-channel access:** One of the main disadvantages of the original IEEE 802.15.4-2011 was the lack of multi-channel access. Multi-channel access predominantly helps in mitigating performance degradation due to the interference in the network. The legacy standard only supports a single channel for communication, restricting the capability to accommodate a large number of nodes without contention, and therefore deteriorating its network performance, overall delay and throughput. In contrast, some MAC behaviors of IEEE 802.15.4e overcome these limitations by supporting multi-channel operation at the physical layer for some of its MAC behaviors, such as DSME and TSCH. The nodes are given the capability to access the channel either through *channel hopping* mechanisms or *channel adaptation* mechanisms. With channel hopping, the channel hopping sequence is statically predetermined in advance. Conversely, with channel adaptation, the PAN Coordinator has the ability to allocate different channels for data transmission, based on the respective channel quality.

- **Information Elements (IE):** Information Elements is a concept already defined in the original IEEE 802.15.4; however, it has been further extended in the IEEE 802.15.4e with additional functionalities. Apart from the header and management layer based Information Elements used in the IEEE 802.15.4, unique IE have been introduced to support the various MAC behaviors. For example the Information Element for a DSME-enabled network carries the superframe specification such as the number of superframes in a multi-superframe, number of channels, time synchronization specification, Group Acknowledgment and channel hopping specifications. The IE of TSCH does not incorporate

superframes, multi-superframes or Group Acknowledgment specifications but it carries relevant information such as timeslot length, timeslot ID or channel hopping sequence.

- **Low latency and low energy:** The IEEE 802.15.4e protocol provides an improved support for low latency communications, more suitable for industrial control applications, still providing a trade-off between latency and energy efficiency. The IEEE 802.15.4e allows devices to operate at a very low duty cycle and also provides deterministic latency, which is a main requirement for time-critical applications.

  To accommodate Low Energy requirements the non beacon-enabled MAC behaviors such as AMCA and the transmissions in the CAP region of the beacon-enabled MAC behaviors such as DSME are supported by low energy mechanisms. The amendment specifies two low energy mechanisms based on the latency requirements of the applications. The *Co-ordinated Sampled Listening* (CSL) is usually used for applications with very low latency requirements. In CSL-enabled receiving devices, the channel(s) are periodically sampled for incoming transmissions at low duty cycles. The receiving and the transmitting devices coordinate with each other to reduce the overall transmitting overhead. The *Receiver Initiated Transmissions* (RIT) mechanisms are used for latency-tolerant applications (i.e., tolerating latency of more than 10 seconds). The RIT mode supports applications that run on low duty cycles and low traffic load. It is also applicable for regions like Japan, where consecutive radio transmissions are limited by national regulations.

- **Multi-purpose frames:** The frame formats of all the MAC behaviors in the IEEE 802.15.4e are based on specific features of each MAC behavior and its targeted application. The DSME MAC behavior, for instance, supports applications where determinism and scalability are fundamental. The MAC frame format of DSME thus supports guaranteed timeslots with multi-channel capability. It also provides features such as Group Acknowledgment (GACK) to reduce the overall delay for several GTS based transmissions. The LLDN MAC behavior supports applications that require very high reliability. The frame format of LLDN has therefore provisions for retransmission of frames using separate uplink timeslots and GACK to acknowledge several frames using a single ACK frame, thus maintaining low latency and high reliability in a network.

- **Enhanced Beacons:** Enhanced Beacon (EB) is a revision of the standard beacon format that is used in IEEE 802.15.4-2011 networks. It provides greater flexibility and it is used to provide application-specific beacon content to the DSME and the TSCH MAC behaviors. An EB can be differentiated from the legacy beacon by the frame version information issued by the PAN Coordinator. EB carries information on whether TSCH/DSME and Low Energy (LE) are enabled, and information about the respective channel hopping sequences.

- **MAC performance metrics:** The IEEE 802.15.4e amendment supports a feedback to the upper layers on the network performance via *"MAC performance metrics"*. These met-

rics provide information on the link performance (quality of the channel), which may help the network layer to take efficient routing decisions, thereby reducing the overall power consumption and latency of the network. The feedback information includes: (i) the number of transmitted frames that required one or more retries before acknowledgment; (ii) the number of transmitted frames that did not result in an acknowledgment after a duration of *macMaxFrameRetries*; (iii) the number of transmitted frames that were acknowledged properly within the initial data frame transmission; and (iv) the number of received frames that were discarded due to security concerns.

- **Fast Association:** In the IEEE 802.15.4, the association procedure encompasses a significant delay. This is mostly due to the fact that in the IEEE 802.15.4, the device must wait till the end of *MAC response Wait Time* before requesting the association data (e.g., short address) from the PAN Coordinator. To address this issue, the IEEE 802.15.4e introduces a Fast Association (FA) mechanism, under which the device requests for association from the PAN Coordinator. If resources are available, the PAN Coordinator allocates a short address to the device. It also sends an association response that contains the assigned short address and the status indicator for a successful Fast Association.

- **Group Acknowledgment:** The DSME and the LLDN MAC behaviors of the IEEE 802.15.4e both support Group Acknowledgment (GACK). Several successful transmissions can be acknowledged using a single GACK either within the adjacent beacon interval or as a separate Group Acknowledgment frame. GACK can be issued by the PAN Coordinator only for a dedicated timeslot in the case of LLDN or a guaranteed timeslot in the case of DSME. This greatly improves the efficiency by reducing the overall waste of bandwidth with multiple acknowledgments. In addition, there is a provision to acknowledge all the retransmissions in the network, thus saving time and energy. However, failure of a GACK can result in losing the entire acknowledgment data. This can prolong the waiting for an alternative GACK to be issued in a new timeslot.

All of these enhancements apply to the time-critical MAC behaviors of IEEE 802.15.4e. They have been designed in such a way that they complement the application which the specific MAC behavior of the standard supports.

As already mentioned, the IEEE 802.15.4e provides five different MAC behaviors: Radio Frequency Identification Blink (RFID); Asynchronous multi-channel Adaptation (AMCA); Deterministic Synchronous Multi-channel Extension (DSME); Low Latency and Deterministic Networks (LLDN) and Time Synchronous Channel Hopping (TSCH). In what follows, we first start with a quick overview of the two non real-time MAC behaviors (the RFID BLINK and ACMA modes), and then we describe, in more, detail the time-critical MAC behaviors, namely DSME (in Section 2.3), TSCH (in Section 2.4), and LLDN (in Section 2.5).

- **RFID:** RFID [36] is one of the most popular technologies used for location tracking and item and people identification. RFID integration with wireless sensor networks have

been used in global commercial markets like Walmart [37] for tagging and identifying products. It has also been employed in the field of tele-medicine [38] for security and privacy purposes. In the Blink mode of a RFID-based IEEE 802.15.4e network, transmitters communicate with the receivers using their 64-bit address and an optional payload data.

The Blink mode in RFID allows the device to communicate its ID, an *Extended Unique Identifier* - EUI-64 source address, and an additional payload data to devices with which they communicate. Devices can connect with each other without any prior association or acknowledgment. The Blink frame can be used for *"transmit only"* purposes and it coexists with other devices in the network. The Blink frame can be rejected during the frame processing if the devices are not interested in Blink enabled communications. Blink uses a multipurpose *"minimal"* frame that consists only of the header fields necessary for their operation. This helps in reducing the overall power consumption in the network.

- **AMCA:** In dense sensor network applications such as structural health monitoring [4], a single channel approach does not have the capability to handle such densely populated networks. The variance of the channel quality is usually large in these dense networks [39], leading to link asymmetry and thus jeopardizing the application performance. Multi-channel adaptation is a possible method to overcome link asymmetry. The AMCA *(Asynchronous Multi-Channel Adaptation)* mode can be enabled for the non-beacon enabled mode of IEEE 802.15.4e networks. In the *synchronous multi-channel adaptation mechanism*, two devices cannot communicate using a common channel. In the case of AMCA, the device selects a mac designated channel based on the channel link quality. In order to switch channels for either listening or transmitting, a channel probe is requested by the coordinator of ACMA during an active scan. The channel probe always probes all the available channels in the network and switches the transmission to a better channel in case of a poor quality transmission.

In contrast to AMCA and RFID, the three other MAC behaviors are designed for time-critical applications, which provide deterministic guarantees and improved robustness. In what follows, we provide a comprehensive overview of these time-critical MAC behaviors.

2.3 Deterministic Synchronous Multi-channel Extension (DSME)

2.3.1 Application overview

The Deterministic Synchronous Multi-channel Extension (DSME) MAC behavior targets applications with QoS requirements such as bounded latency, high reliability and scalability. Industrial automation and process control applications are well known for being very sensitive to the loss of data, considering the criticality of exchanged information [40]. On the other hand, health monitoring systems present stringent time constraints, such as those resulting from the

need to guarantee very short end-to-end latency (e.g. < 10 ms) requirements [41]. In addition, several WSNs applications such as outdoor monitoring (as those reported in [42], [43]) have a need to be deployed in a high density, and thus scalability becomes a major concern. DSME aims at providing solutions for applications with this kind of stringent QoS requirements.

To address this objective, DSME provides several enhanced features to the native IEEE 802.15.4, namely: (i) multi-superframe; (ii) CAP reduction; (iii) Group Acknowledgment; (iv) distributed beacon scheduling and (v) Channel diversity modes to address the non-functional properties mentioned above. We present these features in detail in the next sections.

2.3.2 Multi-superframe

The PAN Coordinator of a DSME network defines a cycle of repeated superframes called the *multi-superframe structure* (Figure 2.3). Similar to IEEE 802.15.4, a superframe in the multi-superframe structure will have a Contention Access Period (CAP) and a Contention Free Period (CFP). In a Multi-superframe, a single common channel is utilized for a successful association, it is also used to transmit the EB frames and the frames transmitted during the CAP. The number of superframes that a Multi-superframe should accommodate is determined by the PAN Coordinator, based on the number of data packets meant to be transmitted within the time interval, and is conveyed to the nodes through an Enhanced Beacon (EB). Multi-superframe, due to its multi-channel features incorporated in it, helps in the formation of peer-to-peer topologies, like a mesh.
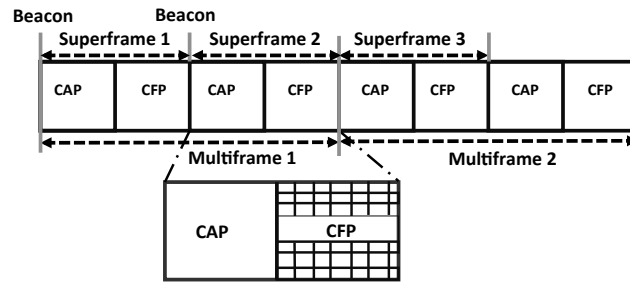
Figure 2.3: Multi-superframe structure of DSME

The standard defines the structure of the superframe by the values of Superframe Duration (SD), Multi-superframe Duration (MD) and the Beacon Interval (BI), which is the time span between every beacon. The Multi-superframe Duration is a new parameter introduced in DSME; it provides the length of all the individual superframes within the multi-superframe. These parameters are defined in the following equations:

$$MD = aBaseSuperframeDuration \times 2^{MO} symbols \qquad (2.1)$$

$$for\ 0 \le SO \le MO \le BO \le 14$$

$$BI = aBaseSuperframeDuration \times 2^{BO} symbols \qquad (2.2)$$

$$for\ 0 \leq BO \leq 14$$

$$SD = aBaseSuperframeDuration \times 2^{SO} symbols \qquad (2.3)$$

$$for\ 0 \leq SO \leq BO \leq 14$$

In the previous definitions, *BO* is the *MAC Beacon Order* and it defines the transmission interval of a beacon in a superframe. *MO* is the *MAC Multi-superframe Order* and it represents the beacon interval of a multi-superframe. *aBaseSuperframeDuration* is the minimum duration of a superframe corresponding to the initial order of the superframe (i.e, *SO* = 0). This duration is fixed to 960 symbols (a symbol represents 4 bits) corresponding to 15.36 ms, assuming a bit data rate of 250 Kbps in the 2.4 GHz frequency band. The total number of superframes and multi-superframes in a DSME network can be determined by $2^{(BO-SO)}$ and $2^{(BO-MO)}$, respectively.

As an example, let us consider the case where *BO* = 3, *MO* = 3 and *SO* = 2. In this case, there are two superframes that are combined in a single multi-superframe, as illustrated in Figure 2.4. The DSME GTSs in the available channels are shown as grids in the CFP region for the aforementioned parameters. The horizontal axis of the grid represents the time, and the vertical axis of the grid represents the frequency. This means that several GTSs can be allocated at a same time, but on different frequencies (i.e., channels).
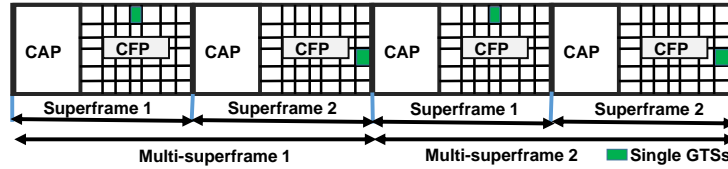
Figure 2.4: Example of a superframe structure with $BO = 3, MO = 3, SO = 2$

The CFP region of the IEEE 802.15.4-2011 enabled superframe comprises seven GTSs. DSME enhances the functionality of the traditional GTS by extending its number using the multi-superframe's multi-channel communication. This enables the protocol to select better channels based on link quality and to accommodate a higher number of transmissions, and thus to increase the overall reliability and scalability. Figure 2.5 gives an example where 4 transmissions have to be handled in the CFP region. DSME, with 3 channels, is taken for comparison with the legacy IEEE 802.15.4. It can be seen that the legacy IEEE 802.15.4 accommodates 4 timeslots for 4 transmissions, whereas DSME requires only 2. Additionally, it provides more timeslots to be occupied by other nodes in the network, thus increasing the overall scalability. In the original IEEE 802.15.4 standard there was a single point of failure problem, since beacon scheduling and slot allocation were carried out in a centralized fashion by the PAN Coordinator. When there is a problem in a root of a network, the entire network collapses due to the lack

of capability for selecting an alternative route. This has been mitigated in DSME by distributed beacon scheduling, as it provides the capability for peer-to-peer communications like mesh networks, which can utilize alternative routing paths to reach a destination.
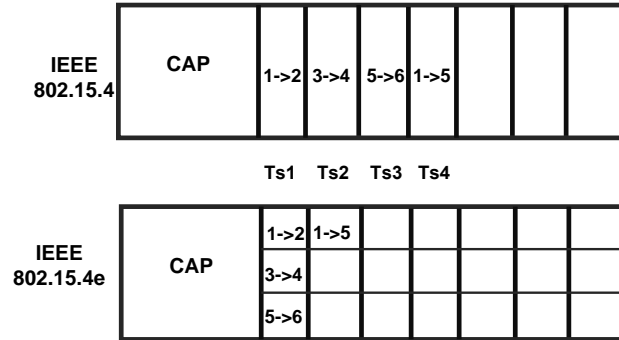
Figure 2.5: Comparison of the DSME GTS with the legacy GTS

Furthermore, DSME also addresses energy efficiency by providing new features such as CAP reduction and Group Acknowledgment (GACK), which will be presented next.

CAP reduction

It is possible for the PAN Coordinator to reduce the size of the CAP by enabling it only in the first superframe of a multi-superframe; this technique is called CAP reduction. In this way, the remaining superframes only present a longer CFP (Figure 2.6). It radically increases the number of DSME GTSs that are allocated to the neighboring nodes, while saving energy, since there is no need for a node to stay active during a CAP if no transmissions are expected to occur. CAP reduction is a very suitable add-on for highly dense networks with stringent QoS requirements in terms of delay and reliability.
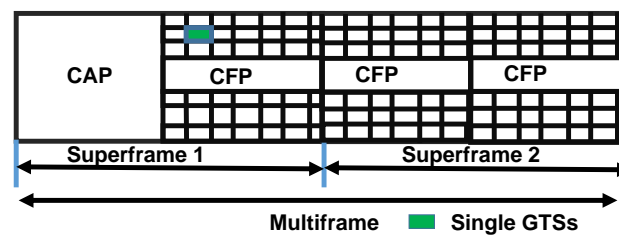
Figure 2.6: CAP Reduction technique

Group Acknowledgment

Another important functionality of the DSME GTS is its Group Acknowledgment (GACK) feature. This mechanism provides the capability of sending a single acknowledgment for all guaranteed transmissions within the same multi-superframe. The GACK reduces the latency and energy consumption by combining several acknowledgments into a single group acknowledgment.

The PAN Coordinator announces the GACK feature using an Enhanced Beacon with a *mac GACK Flag*; it is defined using a GACK element (Figure 2.7). In addition, a single Group Acknowledgment is sent by the PAN Coordinator to acknowledge every DSME GTS transmissions in the CFP region. GACK element indicates the reception status for the set of GTS data frames it acknowledges and new slot allocations (for the retransmission of failed GTS transmissions). The GACK element carries the *bitmap* which indicates the state of transmissions in the guaranteed timeslots. *GACK Device List* is exclusive for DSME, and it indicates the devices for which the guaranteed timeslots are allocated in their respective CFP region. The *GACK Index field* is also a DSME exclusive; it specifies the start of every GTS for the allocated nodes in an order in accordance to the *GACK device list.*
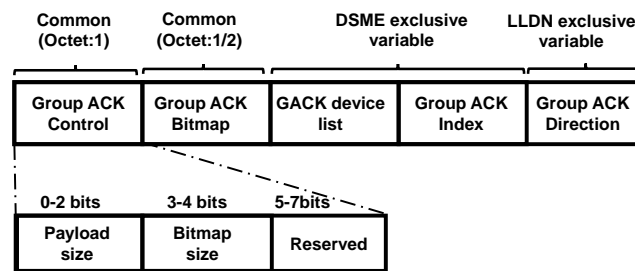
Figure 2.7: GTS element

In Figure 2.8, we consider that the shaded portion in the grid of the first CFP and the second timeslot of the adjacent superframe as the DSME GTSs allocated for retransmission. A single GACK (fourth timeslot of the second superframe) can be given for all these transmissions. Group Acknowledgment saves a lot of power and time that is spent for individual acknowledgments. In the case of a failed transmission, a new DSME GTS will be assigned to carryout the process.
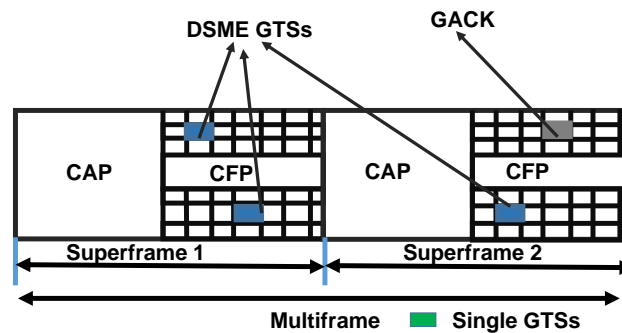
Figure 2.8: Group Acknowledgment in DSME

Beacon scheduling

In order to build more complex networks topologies, such as mesh, it is mandatory to carryout an efficient beacon scheduling to avoid interference and collisions. In a DSME network, all the devices are time-synchronized using the values of the *Timestamp* field of the received beacons

from the device they are associated with, thus maintaining global time synchronization in the PAN. When a node wants to join a network, it uses a `MLMESCAN.request` primitive to initiate scanning over all the available channels in the network. During this scanning process, the joining node searches for all coordinators transmitting Enhanced Beacon frames.

The neighboring devices send their beacon schedule information to the new joining device by transmitting an Enhanced Beacon. This beacon schedule is updated as a bitmap sequence. The new joining device searches for a vacant beacon slot and, if available, will claim it to use it for sending its own beacons.

In the CAP of DSME and the shared timeslots of LLDN (will be later in Section 2.4), due to the hidden-node problem there is a risk of a beacon collision as two or more devices are trying to compete for the same beacon slot number. The beacon scheduling procedure cannot completely eliminate this risk. In Figure 2.9, let us consider that device D and E are willing to join the network. These devices receive the beacon bitmap from their neighboring device A. Now there is a possibility of collision when both D and E want to use the same vacant beacon slot within the CAP. This is called the hidden node problem. This is because devices D and E are hidden from each other, and they cannot listen to each others transmissions.

This is solved in DSME via the `DSME-Beacon allocation notification` command which solves the contention by prolonging the received DSME-beacon allocation commands based on the MAC performance metrics. Every beacon frame has its own allocation superframe duration number (`SDIndex`) for identification. If both devices D and E send a `DSME-Beacon allocation notification` command with same `SDIndex` value, device A, which is common to both D and E determines which device should be given the higher priority based on MAC performance metrics. As such, the conflict is resolved.
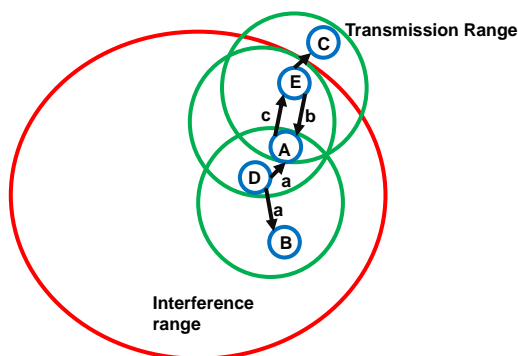
Figure 2.9: Hidden node problem in DSME

2.3.3 Channel diversity

When heterogeneous RF devices having the same RF spectrum are present in the same network, it causes significant interference and failure of transmissions [44], thus affecting the overall reliability of the network. Channel diversity is a mechanism that helps in overcoming this issue.

The DSME MAC protocol defines two channel diversity mechanisms: (1) *channel adaptation* and; (2) *channel hopping*. The kind of channel diversity mechanism under which the DSME operates will be conveyed by the PAN Coordinator, through a DSME PAN descriptor Information Element (IE) in its Enhanced Beacon.

In *channel adaptation*, the PAN Coordinator has the capability to allocate the DSME guaranteed timeslots either in a single channel or through different channels to an end device. This decision is influenced by the link quality of the current channel. The link quality of the channel is conveyed to the PAN Coordinator through the MAC performance metrics, which was discussed in Section 2.2. If the link quality of an allocated DSME GTS becomes degraded, the PAN Coordinator also possess the ability to deallocate a specific DSME GTS.

Channel adaptation in DSME-GTS is illustrated in Figure 2.10. Devices 0, 1 and 2 use channel 1 during the timeslots 1 and 2; then they later switch to channel 4 during timeslots 5 and 6. This switching of channels can be a result of link quality degradation or some performance metric specified by the PAN Coordinator.
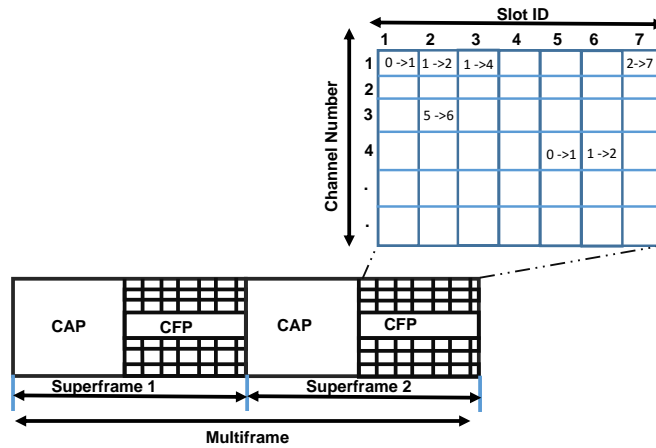
Figure 2.10: Channel adaptation in DSME

Channel hopping is a methodology by which several devices hop over different channels in a predefined channel order. Channel hopping is a well established technique that has been used in radio communication systems for decades. In radio systems, many receivers can select a channel from a predefined set to receive the required broadcast information [45].

In DSME, the guaranteed timeslots hop over a predefined series of channels called as the *hopping sequence*. The *channel hopping sequence* is defined by the the upper layers of the standard. This same hopping sequence will be carried out over the entire time frame of transmission. All the devices in the PAN must be time synchronized, so any form of interference is avoided by coordinating the channel usage among devices in the same interference range. In channel hopping, a channel offset is used to provide orthogonality among all devices employing the same channel hopping sequence list. In Figure 2.11, the shaded cells of the grid represent the DSME GTSs in a CFP, which has a *hopping sequence* that follows a channel offset of 0 and 1 which repeats synchronously for every channel in the CFP.
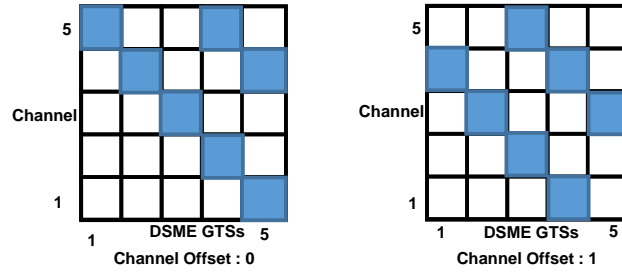
Figure 2.11: Channel hopping in DSME

2.4 Low Latency Deterministic Networks (LLDN)

2.4.1 Application overview

The Low Latency Deterministic Networks (LLDN) MAC behavior targets applications that typically demand robustness because of the critical nature of the data. For instance, LLDN is a suitable MAC behavior for terrain surveying [46], where large geographical areas are surveyed to capture their temporal dynamics. In the exemplified terrain surveying, more than 100 nodes communicate with a coordinator in a single hop. LLDN is a star topology exclusive MAC behavior, making it suitable for applications that demand a centralized control.

Several process control applications have a very small round-trip time and the communication has to be carried out in a periodic basis. Fuzzy logic Petri networks [47] have been used to define control in networking. LLDNs networking techniques provide more determinism in small round-trip communication, thus making them more suitable to support these centralized networks.

We present an overview of some the most important features of LLDN, such as the LLDN Superframe, network topology and its data transfer models.

2.4.2 LLDN superframe

The LLDN PAN Coordinator uses Low Latency superframes (LL frames, as shown in Figure 2.12) for transferring data. This superframe type is composed of four parts: *the beacon; uplink timeslots; management time slots and the bidirectional timeslots*. The LLDN MAC behavior exclusively supports the beacon-enabled communication of IEEE 802.15.4e.

The beacon occupies the start of every superframe and it provides time synchronization for the entire network. The beacon in the LLDN superframe is immediately followed by uplink and downlink *management timeslots*, which use a slotted CSMA/CA mechanism for channel access to send the management information. The next part of the superframe comprises *the uplink timeslots* that are used in a unidirectional transmission (to the PAN Coordinator). These timeslots are reserved for dedicated nodes assigned by the PAN Coordinator. A part of these timeslots are reserved as retransmission timeslots. Finally, the *bidirectional timeslots* are used for the communication from the PAN Coordinator to the nodes. They can also be configured as

uplink timeslots. The direction of these bidirectional slots is sent through an Enhanced Beacon, which is issued at the configuration setup phase of the network.
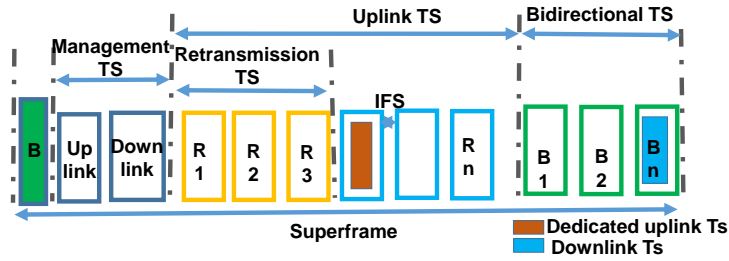
Figure 2.12: Superframe of LLDN

LLDN transmission states

The network setup for every node in a LLDN network follows three transmission states: Discovery; Configuration and Online (Figure 2.13).

The network setup begins in the Discovery state. In this phase, the superframe is composed of one beacon and two management timeslots (one downlink and one uplink). The device that wants to join the network scans the available channels for a LLDN PAN Coordinator, which will be broadcasting Enhanced Beacons indicating the Discovery state. If the scanning device wants to join the PAN Coordinator, it uses an uplink management timeslot to send a join request to the PAN Coordinator. In the uplink management timeslot, nodes rely on CSMA/CA to access the channel. This uplink is used to transmit the discovery response frame, which will have the current configuration of the device. When the PAN Coordinator receives a message from a node, it responds with an acknowledgment. This entire process should happen within *macLLDNdiscoveryModeTimeout*, which is 256 seconds. If successful, the PAN Coordinator will switch its state to Configuration.

The Configuration state can happen either after the discovery of a new device or the failure to discover a device. In the Configuration state, the superframe changes again, comprising one beacon, two management timeslots (one downlink and one uplink). When a device receives a beacon indicating the Configuration state, it will send a `Configuration Status frame` to the LLDN PAN Coordinator. The `Configuration Status frame` contains the current configuration of the device such as the MAC address, required timeslot duration, uplink/downlink data communication and the assigned timeslots. The device will keep on sending the `Configuration Status frame` till the end of the Configuration state or until the moment it receives a `Configuration Request frame`. The Configuration Request frame is sent by the PAN Coordinator and it contains the new configuration for the device. The `Configuration Request frame` specifies the existence, length of the management timeslots and the directions of the bidirectional frames. An acknowledgment is sent to the LLDN PAN Coordinator when the device receives the command of the `Configuration Request frame`. Finally, after the initial setup states, data transmission can occur between the PAN

Coordinator and the device. Once the Configuration Request frame has successfully been acknowledged by the new node, the Online state begins. The superframe configuration is changed to accommodate a beacon, and several timeslots (both uplink and downlink), depending on their respective configuration. LLDN facilitates retransmission in case of failure and collisions. It provides the Group Acknowledgment (GACK) feature, similar to the one in DSME. GACK can be used in the uplink timeslots to acknowledge several retransmission frames. When a new device wants to join the network after the Online state, it scans all the available channels until it finds a LLDN PAN Coordinator issuing EBs indicating the Discovery state.
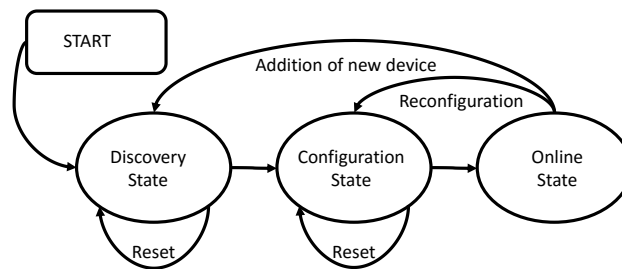
Figure 2.13: Configuration states of LLDN

Network topology

In LLDN, all the nodes are individually connected to a PAN Coordinator, thus forming a star topology. LLDN consists of two kinds of devices: (i) devices that can send data to the PAN Coordinator using an uplink; and (ii) devices that can send and receive data from the PAN Coordinator using bidirectional timeslots. The selection configuration and the number of timeslots are determined by the higher layer. Figure 2.14 shows a simple scenario of a LLDN-enabled sensor-actuator system connected to a single PAN Coordinator through a star topology. The sensors will be able to send data as an uplink to the PAN Coordinator. The nodes with the actuator, on the other hand, can perform an uplink as well as receive actuating signals from the PAN Coordinator as a downlink.

2.4.3 Data transfer modes

To a coordinator

When a device wants to send data to a PAN Coordinator, it waits till the reception of a beacon. At reception, the device synchronizes with respect to the configuration received. The device has the option to either use a dedicated timeslot or a shared timeslot. The data frame is transmitted without contention in the case of a dedicated timeslot. In some cases, more than one device can be assigned to a single timeslot. These are called *shared group timeslots*. If the device transmits its data frame in a shared group timeslot (not the slot owner), then the data frame is transmitted using slotted CSMA/CA.The dataflow from an uplink dedicated timeslot and uplink shared timeslot to an associated node is depicted in Figure 2.15.
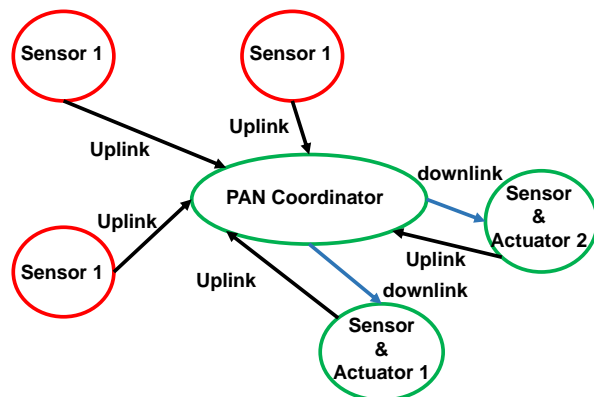
Figure 2.14: Example of a LLDN network topology
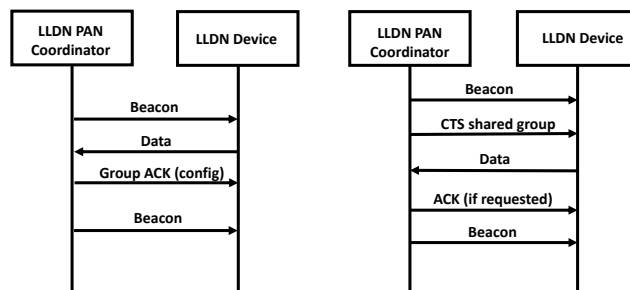
Figure 2.15: Data flow from dedicated timeslot and shared timeslot to an associated node

To a new device

When the LLDN PAN Coordinator wants to transfer data to a LLDN device, a bidirectional timeslot is assigned for transmission. The direction of the bidirectional timeslots (either uplink or downlink) is set during the Configuration phase of the setup. The LLDN PAN Coordinator can configure the bidirectional timeslots to downlink and use them entirely for data transmission to the nodes. Any data transmission from the LLDN PAN Coordinator is carried out without contention. The device sends an acknowledgment upon a successful data reception. Figure 2.16 illustrates the data flow diagram from the PAN Coordinator to a LLDN device.
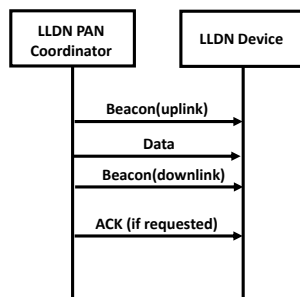
Figure 2.16: Data flow to a new device

2.5   Time Slotted Channel Hopping (TSCH)

2.5.1   Application overview

The Time Slotted Channel Hopping (TSCH) MAC behavior provides very high reliability and time critical assurances. TSCH is a suitable candidate for implementing sensor-actuator networks in oil and gas industries, which are defined as safety critical [48], as the prime concern can be human and environmental safety. These industries are prone to interferences that affect the functionality of wireless devices. TSCH supports the *frequency Hopping* mechanism, which greatly improves the reliability of the network by effectively mitigating the effects of interference and multi-path fading at a considerable scale.

Time-slotted communication links greatly reduce the unwanted collisions that can lead to catastrophic failures. Data centers monitoring [49] are also prone to collisions, because the network has to accommodate dense sensors and is tightly coupled with high network traffic increasing the chances for collision in the network. TSCH can help accommodating a dense network and, at the same time, cope with stringent time constraints using fixed length timeslots and multi-channel access. TSCH uses TDMA-based slotframes, thus facilitating collision-free transmissions.

In the coming sections, we present some of the most interesting features of TSCH, such as slotframes, channel hopping, fast association, time-node synchronization and TSCH CSMA/CA.

2.5.2   Slotframes

In TSCH, the superframe concept used in DSME, LLDN and its parent standard IEEE 802.15.4 has been replaced by the concept of *slotframes*. Every slotframe is a collection of timeslots. Communications in each timeslot can be either contention (i.e., using CSMA-CA) or non-contention based. Every timeslot accommodates a transmission and an eventual acknowledgment. The slotframe size is defined by the number of timeslots in the slotframe. Every slotframe repeats in cyclic periods, thus forming a communication schedule. For identification, a slotframe handle is associated at the start of every slotframe. The TSCH is topology independent, supporting a wide variety of topologies, from a star to a full mesh.

Figure 2.17 shows a three time-slotted slotframe. In the example, the slotframe repeats every three timeslots. Let us take 3 devices, A,B,C; in timeslot 0, device A transmits its data to B and during timeslot 1, B transmits to C and during timeslot 2 the devices remain in an idle state. This repeats in the same order for all the three timeslots. Every timeslot in a TSCH PAN has an Absolute Slot Number (ASN), which increases globally and is used to compute the channel in which a pair of nodes communicate.

TSCH provides concurrent slotframes to support concurrent transmissions; these are called *multiple slotframes*. These slotframes can have different communication schedules. A multiple slotframe is established by configuring different communication schedules and connectivity

ASN 1	ASN 2	ASN 3	ASN 4	ASN 5	ASN 6	ASN 7	ASN 8
Ts 0	Ts 1	Ts 2	Ts 0	Ts 1	Ts 2	Ts 0	Ts 1
A -> B	B -> C		A -> B	B -> C		A -> B	B -> C
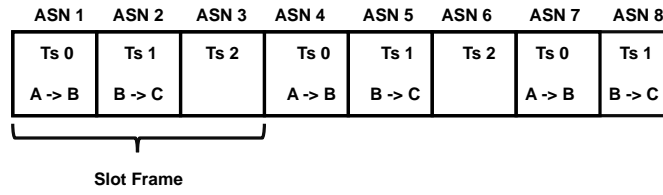
Slot Frame

Figure 2.17: Example illustrating the use of TSCH- slotframes

matrices to work in parallel. All slotframes are aligned to timeslot boundaries. For example, in Figure 2.18, it can be seen that timeslot 0 of every slotframe is projected back to ASN = 1. The PAN Coordinator holds the responsibility to align slotframes in a multiple slotframe.

ASN 1	ASN 2	ASN 3	ASN 4	ASN 5	ASN 6	ASN 7	ASN 8
Ts 0	Ts 1	Ts 2	Ts 0	Ts 1	Ts 2	Ts 0	Ts 1
		3 Time slot frame					
Ts 0	Ts 1	Ts 0	Ts 1	Ts 0	Ts 1	Ts 0	Ts 1
		2 Time slot frame					
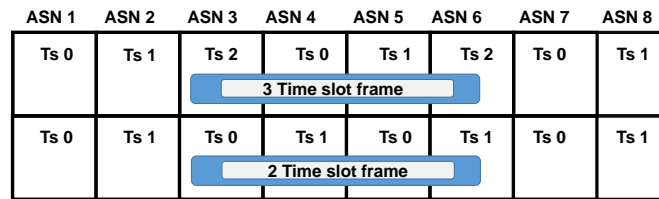
Figure 2.18: TSCH- multi-slotframes

Unlike DSME, which supports channel diversity mechanisms, the multi-channel communication of TSCH relies entirely on channel hopping. TSCH can utilize 16 channels for communication, which are defined by a channel offset, an integer value ranging in the interval of [0-15]. In TSCH, the link between two nodes is defined by n, channel offset. This is a pairwise assignment of the timeslot "n" where the two nodes communicate and their respective offset. The frequency used for TSCH communication can be defined by f as follows:

$$f = F[(ASN + \text{channel Offset})\%N_{channels}] \tag{2.4}$$

In Equation 2.4, $N_{channels}$ defines the number of channels under usage for the specific network as it is not mandatory to utilize all the 16 channels. Channels can be left out to improve energy efficiency or in case the channel quality is deteriorated. *ASN* helps in determining the number of timeslots elapsed since the beginning of the network duration. The function F can be defined as a lookup table. Also from Equation 2.4 it can be noted that a different channel ($N_{channel}$) can be implemented over for the same offset for an incremented *ASN*; i.e., the channel hopping mechanism can be utilized with a different frequency over the same link.

PAN formation

When a network is to be established, the PAN Coordinator starts broadcasting an Enhanced Beacon (EB) in response to a `MLME BEACON.request` from a higher layer. This action is termed as Advertising. The devices that want to connect with the PAN Coordinator should be in the broadcast range of the PAN Coordinator. The EB contains time information, channel hopping information, timeslot information and initial link information. Time information pro-

vides the specific time period at which the nodes should synchronize with the network. Time slot information describes the time at which data will be transmitted. Lastly, the Initial link information gives the time when to listen to an advertising device and transmit to the same device.

A device wishing to join the network either does active scanning or passive scanning after receiving an `MLME SCAN.request` from a higher layer. After an `MLME-BEACON-NOTIFY`, the higher layer, initializes the slotframe and Initial link information available in the Enhanced Beacon. When the device synchronizes with the network, the higher layer changes the device into TSCH mode by issuing a TSCH association primitive.

The devices may also request for additional slotframes and link resources before association. In order to get additional link resources (slotframes and links) the device should undergo a security handshake for authenticating the joining process. After getting associated in a network, the fully functional devices are completely capable of transmitting Enhanced Beacons for synchronizing and adding nodes to the network. The size of the network plays a crucial role in determining the advertising rate and the configuration by the higher layers. These configurations will have a direct impact on the non-functional properties such as scalability and power consumption of the network.

Fast Association

Fast Association (FastA) is a feature of the time critical MAC behaviors of IEEE 802.15.4e. To carryout a Fast Association, the higher layer of the device posts a `MLME ASSOCIATE.request primitive`, triggering the FastA procedure in the MAC sublayer. The request is sent to the PAN Coordinator which acknowledges its reception.

Fast Association removes the wait time duration `macResponseWaitTime`, which is used for the association process in the legacy IEEEE 802.15.4. This efficiently reduces the association delay. The association request command contains an acknowledgment request, and the PAN coordinator confirms it by sending an acknowledgment frame. If the PAN coordinator has sufficient resources, the higher layer allocates a 16 bit address to the device. The MAC sublayer then generates a status indicating `FastA successful`. The device can then use the `macShortAddress` for its association within the PAN.

2.5.3 Time and node synchronization

Time propagates outwards from the PAN Coordinator in a TSCH based network. A communicating device must synchronize its network time with another device in its vicinity at periodic intervals by using the neighbor device as a time source/reference, all synchronized devices should have a prior idea of where the timeslot begins and where it ends.

Node-to-node synchronization is also done to ensure connection with the neighboring nodes in a slotframe-based network. Time source neighbors keep track of the devices and if

they do not receive a request from the device, at least once per keep alive period, they will send an empty acknowledgment frame to perform an acknowledgment-based synchronization.

Let us take the example of the network containing one PAN Coordinator and two nodes, as shown in Figure 2.19. The PAN Coordinator acts as a time keeper of the entire network. Device 1 synchronizes only to the PAN Coordinator. The device 2 has no influence over the time synchronization of Device 1, whereas the synchronization of Device 2 is dependent both on the PAN Coordinator and Device 1.
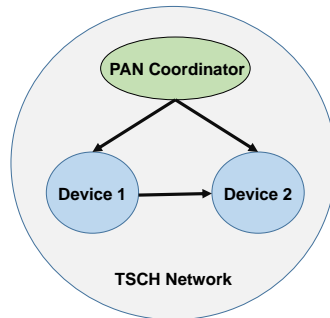
Figure 2.19: TSCH- Time node synchronization example

Synchronization happens whenever a device exchanges a frame with a time source neighbor. To support this, there are two methods: either through the time information that is received within an acknowledgment from the receiver; or from the arrival time of a frame from the time source neighbor. This is called *Acknowledgment-based* and *frame-based synchronization* in a TSCH network respectively.

Under the *acknowledgment-based* method, synchronization is carried out through the exchange of data frames and acknowledgments. The receiver calculates the difference between the expected arrival time and the actual arrival time. Then a timing correction information is sent to the transmitting node through an acknowledgment.

Under the *frame-based synchronization* method nodes synchronize to its own network clock, whenever they receive a data frame from a time source neighbor. The receiver calculates the time difference between the expected arrival time and the actual arrival time, and uses this information to adjust its own network clock.

TSCH CSMA/CA

Based on the *TSCH IE* issued at the network formation, TSCH either operates based on slotted CSMA/CA or on the dedicated timeslot-based TSCH mode. When the transmission link is established under slotted CSMA/CA, it performs a *Clear Channel Assessment* (CCA). If the channel is idle, transmission occurs, else the network waits for the transmission link to reach the destination.

Two kinds of links can be established in a TSCH CSMA/CA enabled system; they are the *shared link* and the *dedicated link*. Shared links are used to assign to more than one device during data transmission. When a packet is transmitted using a shared link, it awaits for an

acknowledgment. These links are prone to collisions and failures. To reduce collisions, re-transmission backoff algorithms are used. The backoff window keeps on increasing with every retransmission. At the point of a successful transmission the backoff window is set to the minimum value.

In dedicated links, transmission is more reliable since there is no contention between nodes to occupy the channel. In a dedicated link, the backoff window does not change when the transmission is successful, but when the transmit queue becomes empty, the backoff window is reset.

2.6 Summary

In this chapter, we provided an overview of the various MAC behaviors of the IEEE 802.15.4e. In particular, we stressed the ones with time-critical capabilities, which are employed in the methodologies proposed throughout this dissertation. We discussed the mechanisms through which the network infrastructure is supported and the methodologies by which the topologies are established.

In the following chapter, we provide an in-depth survey of the related works on the 3 time-critical MAC behaviors of IEEE 802.15.4e, and the methodologies used to analyse the models and their use.

Chapter 3

Related Research Work

The IETF (Internet Engineering Task Force) has been fostering several efforts aiming at providing a variety of communication and computing infrastructures to support the emerging IoT applications. The legacy IEEE 802.15.4 standard [21] paved the way to various paradigms in the area of Cyber-Physical Systems (CPS), but greatly suffered due to issues like lack of reliability, energy inefficiency, and reduced scalability. To tackle these issues several standards have been developed in the past few years to support IoT infrastructures.

The WirelessHART [50] is one of the pioneer standards to support wireless communication in control-oriented applications. The WirelessHART adopted the IEEE 802.15.4 PHY layer and it included a time-synchronized MAC sublayer based on TDMA with channel hopping. This standard opened the doors to the possibility of integrating complex topologies like mesh. Adding to channel access capabilities, the WirelessHART also provides self-organizing and healing techniques that can mitigate the presence of obstacles and improve the reliability of the network. However, WirelessHART operates on a centralized router scheme in which a network manager will be responsible for the overall scheduling and maintenance of the network. In some cases, this scheme can have a dire impact on robustness. In order to improve the robustness of the network, there have been some proposals for including blacklisting features [51]. By using blacklisting, degraded channels can be blocked from communicating, and thus concurrently increase the robustness of the network.

Other standards like the ISA-100.11a [52] also provide more secure approaches for communication in process and control applications. ISA-100.11a takes after the infrastructure of WirelessHART, but it adds new hopping modes such as slow, slotted and hybrid. Hopping modes of the ISA-100.11a like the slow hopping mode are highly energy consuming as the devices have to constantly listen to the channel for incoming packets. Akin to WirelessHART, this standard also uses both a centralized scheme and a Time Division Multiple Access (TDMA) scheme for scheduling.

Long Range Wide Area Network (LoRaWAN) [53] is a Medium Access Control (MAC) protocol similar to the IEEE 802.15.4. It supports wireless communications for low-powered devices over long ranges. It is implemented on top of spread spectrum modulation in industrial, sci-

entific and medical (ISM) radio bands. LoRa modulation is based on Chirp spread- spectrum technology, which makes it work well with channel noise, multipath fading and the Doppler effect [54].

There have also been other protocols, like ZigBee [55], [56], that complemented over the network and application layers on IEEE 802.15.4. In ZigBee, the nodes can sleep for an inactive period in order to drastically improve the battery life. It also provides rigid network infrastructures for the management of the network, routing and even security. In the recent years, ZigBee has gained a wide adoption in various application domains, such as home automation, smart energy or health care. Some of the major drawbacks of the IEEE 802.15.4 protocol and its variants were rectified by the time-critical MAC behaviors of IEEE 802.15.4e. The IEEE 802.15.4e supports multi channel access opening possibilities to implement complex topologies like mesh. It also has the ability to support more than 7 GTSs, which is a great feature to improve the overall scalability the network can support. Apart from these features, the standard has a lot of open challenges, and therefore room for enhancements.

The following summary of the related works will serve as a baseline for our contributions in the subsequent chapters. In what follows, we have grouped the problems that we address in this thesis and their respective related works.

3.1   Need for analyzing bounds and quality of the network

Several researchers have analyzed the MAC behaviors of IEEE 802.15.4e and studied the various quality aspects of the standard. Wun-Cheol Jeong *et al.* [57] carried out a performance evaluation of the DSME MAC behavior, finding that the throughput of the DSME MAC protocol is 12 times higher when compared to that of the IEEE 802.15.4 slotted CSMA/CA in a multi-hop, 5x5 square grid device network deployment (in a star topology). The authors also stated that the energy consumption under the DSME MAC remains constant through time, whereas under the CSMA-CA it exponentially increases. Using mathematical modeling it was verified that the throughput increases significantly when CAP reduction is applied to the system. Lacob Juc *et al.* [58] did an energy consumption and performance analysis comparing the DSME and the TSCH MAC behaviors. The authors found that DSME performs better in terms of the energy spent in data transfers. In their findings, it was also stated that DSME (because of its shorter timeslots) can save a lot more bandwidth when short packets are transmitted. These works predominantly focus towards analyzing the power consumption of DSME, but its impact resulting from the change in the Multi-superframe Order (MO) and DSME presets were not studied in their works.

Junhee Lee *et al.* [59] analyzed the performance of the IEEE 802.15.4e DSME MAC protocol under WLAN interference in terms of Frame Error Rate (FER) and aggregate throughput. It was observed that the FER of slotted CSMA/CA increases when the number of end devices increases. On the other hand, the FER of the DSME MAC behavior remains the same regardless of the number of endpoint devices, due to orthogonal DSME-GTS allocation. The authors

confirmed that the DSME-enabled WSN tolerates better heterogeneous interference. Further complementing their research, the authors also proved that DSME manages to maintain a high level of throughput when compared to the slotted CSMA/CA. The increase in the end devices can have an impact in the overall latency of the network. The impact of error rate over the multi-channel access schemes of the network, and the possible ways of error mitigation using channel adaptation, were not clarified in their research work.

Celia Ouanteur *et al.* [60] have evaluated the performance of the LLDN MAC behavior of the IEEE 802.15.4e, by using a three dimensional Markov chain model. They have derived theoretical expressions for reliability, energy consumption, throughput, delay and jitter in the LLDN mode, and compared it against the IEEE 802.15.4 slotted CSMA/CA to showcase its superiority. Additionally, they claim that LLDN has better reliability than CSMA/CA at any data packet length. They also have extended their results with Monte Carlo simulations to validate the accuracy of their theoretical analysis. LLDN provides better results than CSMA/CA due to its time division multiplexing properties. A comparison against a standard that provides an equal caliber of determinism like the DSME (that uses GTSs), was not explored in their research work.

Berger *et al.* [61] have found that the energy efficiency of LLDN can be increased by combining the use of relay nodes and combinatorial testing. Relay nodes can increase the power efficiency by retransmitting the data packet if a negative acknowledgment is indicated in the GACK, thus avoiding packet-transmission failures in the system. Combinatorial testing stores different erroneous copies of the same source packet and attempts to recover the original data of the source packet by CRC tests on variations of the received packets. A real-time implementation was made using the MSP430 microcontroller on an MSP430-EXP430F5438 Experimenter Board 2 and a CC2520 transceiver (from Texas Instruments) on the daughter board CC2520EMK. Retransmitting packets immediately can have a dire impact over the other QoS, such as latency and throughput. The trade-offs to acquire better power efficiency was not discussed in this work. The feasibility of the DSME-enabled MAC in a real-time environment was studied by T. Paso *et al.* [62]. The performance of the communication system utilizing IR-UWB PHY (Infra Red-Ultra Wide Band PHY) was analytically derived and simulated. It was concluded that, the usage of DSME GTSs slots in the network improves its performance in terms of throughput and end-to-end packet delivery.

Domenico *et al.*, in their paper [63], consider a random-based advertisement algorithm for TSCH, and evaluate its power efficiency through simulation. The radio has to be always on for the node to join the network contributing to the power dissipation. The authors remark that the standard does not have any method in issuing an Enhanced Beacon (EB). They conclude that the joining time is mainly influenced by the number of channel offsets used for advertising Enhanced Beacons.Though this work tries to create a co-relation between the issue of Enhanced Beacons and the energy efficiency, the impact due the dynamic number of nodes joining in the network was not discussed. Furthermore, this can be extrapolated to other QoS aspects, like latency, where the node has to wait for a longer time because of an offset to get accommodated in its respective timeslot.

Jian Wei *et al.* [64] compares TSCH with Coordinated Sample Listening (CSL). In CSL, the transmitter sends wake-up frames to the receiver. The receiver wakes-up periodically to get the timing information, and it then acknowledges the transmission. TSCH was compared with CSL using the Cooja simulator [65] for Contiki OS [66]. The star topology was put to test and it was proved that TSCH outperforms CSL in terms of power efficiency, but it trades-off with higher delay. Delay in TSCH can be easily mitigated by using static scheduling in a multi-channel access scheme. Multi-channel access in TSCH also opens possibilities to support complex topologies like mesh. The effect of multi-channel access where multiple transmissions have the capability to concurrently listen was not explored in this work.

Giuliana *et al.* [67] carried out a simulation-based performance analysis on DSME and TSCH MAC behaviors. The authors show that the DSME performs better than TSCH in terms of end-to-end latency when the number of nodes is higher than thirty. The group ACK feature of DSME helps nodes allocated to DSME GTSs to be acknowledged faster, thus reducing the end-to-end latency. In addition, the authors infer that TSCH has a more flexible slotframe structure than DSME. In TSCH, a single timeslot can be inserted or removed from the slotframe, but every timeslot has to accommodate an acknowledgment. They also stated that the unavailability of GACK in TSCH proves to be a disadvantage in the case of larger networks. This work uses statistical analysis to evaluate the performance of the DSME. In our dissertation, we use worst-case bound analysis to determine the limits of these MAC behaviors, and we also evaluate the impact of its features on the QoS of the network.

In this thesis, we carry out a in-depth performance evaluation of all the time-critical MAC behaviors, namely LLDN, TSCH and DSME. These evaluation are based on a mathematical tool called Network Calculus [68]. We state the advantages, drawbacks, possibilities of adaptation, challenges and open issues in all the aforementioned MAC behaviors. The IEEE 802.15.4e MAC behaviors lacked a proper tool for analyzing their worst case bounds. We built a tool based on Network Calculus to provide the worst-case bounds in terms of latency and throughput for all the MAC behaviors. We compare all the MAC behaviors against each other and found that DSME has the capability to be more flexible and to provide a better throughput with lower delay when the traffic is higher. Other research works, like in [57] and [59], have addressed the better performance of DSME with respect to energy consumption and throughput. In our work, we provide QoS analysis for the DSME networks in terms of throughput, delay, energy consumption and reliability. One of the related research works ([67]) addressed the advantage of DSME over TSCH networks when a large amount of guaranteed traffic is involved. In Chapter 4 of this thesis, we complement this finding and add additional results to explore the behavior of DSME networks in various traffic scenarios. We also additionally explore the impact on features like CAP reduction and variation in multi-superframe orders for variable traffic scenarios.

3.2   Need for adaptive and enhanced techniques

There are several research works that introduce new enhancements to the existing MAC behaviors of IEEE 802.15.4e. Peng du *et al.* [69] introduced an enhanced version of TSCH, called A-TSCH (Adaptive TSCH). The authors use a blacklisting algorithm in order to improve the performance of the existing TSCH mode in 802.15.4e. In this method, the algorithm blacklists channels based on their link quality. In A-TSCH, the transmitting nodes have a knowledge of the neighbors' blacklisted channels. Senders and receivers use the same hopping sequences for communicating, and the nodes insert this blacklist information in their broadcast. The same process is carried out to maintain the timing information as well. A-TSCH was coded on top of the Berkley OpenWSN stack [70], and was implemented in Guidance and Inertial Navigation Assistant motes (GINA). The successful transmission count was calculated in this experiment, and it was found that it reduced 5.6%, when compared to the normal TSCH operation. Blacklisting, on the other hand, can also have a dire impact on the latency of the network, as many channels are blocked from transmitting data. Impact of blacklisting over latency and throughput and the quality trade-offs they have to endure, were not explored in this research work.

Under the DSME MAC protocol, a single Personal Area Network Coordinator (PAN C) can associate several slave devices with a negligible number of collisions, using Fast Association. Xuecheng Liu *et al.* [71] developed a technique, called Enhanced Fast association (EFastA), by which hundreds of slave devices could associate within 3 MDs (Multi-superframe Duration). The authors have analyzed the performance of EFastA in terms of network convergence time and mean number of retransmissions for association. However, the authors did not explore the possibilities of quality degradation when the network demands for higher scalability.

Research has been done by Gaetano *et al.* [72] to improve the scalability of the LLDN, by allowing a large number of network nodes while maintaining low cycle times. In their work, they have proposed a Multi-Channel-LLDN (MC-LLDN), under which nodes communicate on different channels in parallel. There are two levels in that model. There is a higher level network (HLN) and a lower level called sub-networks, which can support multiple networks. There is a network sub-coordinator that coordinates the sub-network and acts as an end node for the HLN. The authors believe this approach achieves lower cycle times, larger radio coverage and higher throughput than the legacy LLDN. OMNeT++ has been used to simulate this model. Assessments of cycle time, end-to-end latency and throughput were carried out in this research work. The attempt made by the authors to add multi-channel access to LLDN makes it look more like DSME. However, DSME supports techniques such as CAP reduction that further facilitate increased scalability. The impact of the fixed uplink and downlink timeslots, and the channel hopping and adaptation modes for the multi-channel access in LLDN, were not clearly discussed in their research work.

One of the limitations of the LLDN mode is that the uplink slots are reserved, whereas the downlink slots remain to be bidirectional. Hence, there is no explicit concept of reserved downlink. This results in the lack of determinism in the case of infrequent downlinks. Luca *et al.* [73]

proposed reserved downlink timeslots to overcome this limitation. An analytical comparison has been carried out in their research work to prove the efficiency of reserved timeslots over the traditional LLDN. By reserving timeslots at the association phase can be an impediment to the quality when dynamic networks are considered. A reserved downlink mode reserves a large amount of bandwidth, even without utilization. It may not be a right fit to support the emerging dynamic IoT applications.

A Mobility-Aware LLDN (MA-LLDN) scheme was proposed by Yaroob *et al.* [74] to overcome the node-mobility problem, using a multi-hop approach. The authors claim that their approach minimizes both latency and energy consumption, while maintaining total capability with the LLDN standard protocol. The impact of mobility on LLDN networks was presented in their paper using a Markov chain methodology. The usage of multi-hop techniques reduces the need for additional coordinators in the system. This will increase the overall power efficiency of the device. In our work, we introduce a "routing-aware" protocol that cumulatively also reduces the latency and the energy consumption of the network.

Mukesh Taneja *et al.* [75] in their paper provide a flexible resource management framework for DSME, to support delay-critical applications. When packets move from an originating device to a destination device (upward direction) several compensation parameters are computed. These parameters are conveyed as MAC packets. The receiving node uses these parameters to compute a compensation factor, and it conveys that to the originating device (to coordinator) (downward direction). Intermediate nodes use the compensation factor to do dynamic management of resources in DSME networks. The authors provide an extensive analytical model for this framework. The authors state that their framework can efficiently support real-time applications. In our work we take a similar approach by introducing an algorithm that helps to alter the network infrastructure based on the resources needed for allocation.

Variations in the superframes have been studied for LLDN by Mashood Anwar *et al.* [76]. In this work the authors provide an insight about the relationship of the superframe size, the base timeslot size and the data payload, with or without security implementation. In their work the authors configured the uplink slots for reconfiguration in order to improve the network performance. From research works like these, it evidently visible that the change in network infrastructures has a strong impact on the quality of the network itself. Most of these works consider a static scheduled network. In a dynamic real-life scenario, there is the possibility of multiple nodes to join and leave. A fixed static schedule would be an impediment for establishing large scalable dynamic networks. In our work, we propose a dynamically varying network infrastructure to alleviate the shortcomings a static network imposes.

Silvia Capone *et al.* [77] proposed enhancements for low-power instrumentation DSME applications. In their work, the MAC protocol was designed specifically for cluster-tree based networks. It was proved through simulations that this proposed MAC protocol reduces the power consumption drastically at the end nodes. MAC protocols like these can be efficient candidates for destination-oriented, ultra-low-power battery powered wireless sensor networks. With a routing based energy conserving approach, DSME also has the capability to be a strong com-

petitor to technologies like 6TiScH [78].

Prasan Kumar Sahoo *et al.* [79] provide a new channel access scheme and a beacon broadcast scheme for DSME, to avoid the collision in a mobile-dense wireless network. They provide analytical models to measure the data transmission reliability, throughput, energy consumption and success rate probability, and compare it with the IEEE 802.15.4e. They have devised a new retransmission scheme that helps the network to achieve better performance results.

One topic that was still open to research was proving DSME to be an apt candidate for IoT implementations. Such a claim can only be addressed by providing a suitable mechanism to support dynamically evolving mesh networks. From our survey on this topic, we are able to understand that the researchers have altered the infrastructure and introduced new mechanisms to support the quality demands of the network. In this thesis, we introduce a technique by which coordinators of the networks can adapt the network infrastructure based on the size of the network. We introduce a dynamic technique called DynaMO in Chapter 6 by which the MD duration can be altered on the go. This technique can adapt the superframe format and can provide support for dynamic and large-scale network infrastructures.

3.3   Need for highly reliable network architectures

Several technologies such as 6LoWPAN [80] have combined protocols such as IEEE 802.15.4 and IPV6 to make them suitable suitable for a variety of applications. The popularity of TSCH and Zigbee networks can be easily traced to the implementation of a standard network and a routing layer over them. The legacy IEEE 802.15.4 did not provide any network layer, which was also continued on to the IEEE 802.15.4e. An efficient network layer helps in improving the Quality of Service in terms of reliability. The RPL (Routing Protocol for Lossy networks) was introduced as one of the prominent network layers that supported the IEEE 802.15.4. This led to the evolution of the commercially adapted Zigbee protocol which was based on the architecture of RPL over the IEEE 802.15.4. TSCH followed this footsteps and efforts towards creating a RPL-based TSCH network architecture called 6TiSCH networks. The main problem with 6TisCH is that it does not have the capability to support both contention based and contention less communications. This can be a problem, as more guaranteed wasted bandwidth can result adversely in terms of quality.

Integration of the RPL with TSCH was carried out by Ren-Hung Hwang *et al.* [81]. Their network infrastructure called DIS-TSCH, allows every node in the network to independently calculate a timeslot offset and a channel offset within the window of a slotframe for conflict-free transmission. They also provide traffic differentiation by providing a scheduling priority to the nodes. This technique helps in minimizing the packet transfer delay of the IEEE 802.15.4e, while yielding a small network duty cycle. It must be noted that prioritizing timeslots for specific transmissions can have an impact on the overall latency of the network. Hence, there is a need for architectures that can efficiently mitigate the effects of these trade-offs.

Several recent studies (eg: [82]) indicate TSCH to be a very competitive industrial IoT MAC protocol. Thomas Watteyne *et al.* presented a model [83] to estimate the latency, power consumption and throughput of a TSCH network. They applied this model to SmartMesh IP, a commercial TSCH product, which claims that TSCH has a broad flexibility for implementation in industries. This TSCH model had an implementation of the RPL network layer on top of it. The RPL layer, in their work, was used to provide efficient routes based on a specific objective function.

Maria Rita *et al.,* in their paper [84], discussed an amalgamation of a power-efficient IEEE 802.15.4-2006 PHY layer, a power saving and reliable IEEE 802.15.4e MAC sublayer, a IETF 6LoWPAN adaptation layer enabling universal Internet connectivity, an IETF ROLL routing protocol, and an IETF CoAP. CoAP is a Constrained Application Protocol (CoAP) which is a specialized Internet Application Protocol for resource constrained devices. The authors of this work analyzed the TSCH mode of IEEE 802.15.4e by providing the working model of the MAC behavior and comparing its efficiency to the legacy IEEE 802.15.4. The take from these works is that having a routing layer which provides efficient routes based on a certain objective function will help in improving the quality without any dire trade-offs.

In our thesis, specifically in Chapter 5, we provide a network architecture in which the routing for the DSME MAC behavior is carried out by a RPL layer. We also provide an Enhanced Beacon model to support this cross-layer communication. A traffic differentiation mode to support prioritisation of transmissions is also introduced for the DSME MAC behavior in our network architecture. The motivation of this approach is to maintain both the routing quality aspects of the RPL and the deterministic features of the DSME within the same network. We analyzed the performance in terms of delay and energy efficiency with routing employed over DSME MAC behavior.

3.4   Need for algorithms that improve QoS with less trade offs

Scheduling techniques and algorithms have always been a key factor in determining the QoS of a network. Attaining a certain QoS can result in a trade-off; for instance, scheduling techniques that support high reliability can result in increased overall delay of the network. Therefore, there is a need for developing algorithms that take trade-offs into consideration and help the network to operate with optimal performance.

As the DSME MAC behavior has the capability to support complex topologies like mesh, there is a need for scheduling algorithms to support the multi-channel slot allocation. Researchers in [85] proposed a distributed algorithm for DSME by which they were able to show an improvement in throughput by 15%, when compared to other MACs like TSCH and CSMA (and even the original DSME). The algorithm was experimented over the QualNET platform [86] and it was confirmed that their algorithm was also able to reduce the power consumption by improving the availability of communication slots. In our work, in Chapter 5, we provide a

multi-channel scheduling algorithm called Symphony that is more focused towards reducing the overall latency between the nodes in the network.

H.Kwang-il *et al.*[87] have developed a new DSME beacon scheduling model. Based on experimental results they analyze the problems and limitations of DSME by trying a variety of experimental topologies to calculate the successful allocation ratio. By including new features like limited permission notification, they were able to improve the overall successful allocation rate. The scheduling algorithm Symphony (Chapter 5) that we address in this thesis is more in line with supporting complex topologies like mesh over multiple channels and improving the overall QoS.

Nicola *et al.* [88] have provided an implementation of a decentralized traffic aware scheduling algorithm for the OpenWSN stack [89]. Having the IEEE802.15.4e TSCH as the MAC sublayer, OpenWSN implements IoT-related standards such as 6LoWPAN, RPL, and CoAP. In their work, the researchers provide a scheduling technique that improves the performance in terms of duty cycle, delay, link and packet-loss ratio. One of the main drawbacks of this work is that, the authors did not provide any insight on the trade-offs incurred due to the distributed scheduling technique. In our work (in Chapter 5) we replace TSCH with DSME at the MAC sublayer and used RPL to provide routes to the datalink layer. In terms of scheduling, we propose a distributed algorithm that is more fit to reduce the latency of the network.

Chao Fang *et al.* proposed an algorithm [90] that generates a frequency hopping sequence in presence of interference without regeneration overhead, while maintaining the optimal throughput. Chao fang contradicts the algorithm of Peng Du [69] by stating that although blacklisting decreases power consumption, it changes the channel number, and the original Frequency Hopping Scheme (FHS) cannot be used. They provide a thorough analytical model which is supported with simulations using the Cooja network simulator [91] of the Contiki OS.

Domenico de Guglielmo *et al.* in their paper [92], provide an algorithm to improve the performance when shared links are utilized in a TSCH CSMA/CA network. Delivery probability, packet latency, and energy consumption of nodes are some of the metrics the model helps to evaluate. The model was simulated using the Cooja simulator in the Contiki OS. They claim that the performance of the CSMA/CA depends greatly on the capture effect [93] in the WSN. Despite providing some guarantees in terms of latency, CSMA/CA cannot assure accurate bounds like that of the guaranteed timeslots of DSME or the fixed timeslots of TSCH. In our thesis, in chapter 7, we showcase techniques that support time-critical applications and we keenly focus on contention-free features of DSME.

Many scheduling techniques in the literature have tried to improve the overall reliability of the network. In our thesis we design a scheduling algorithm (Symphony) that can utilize the reliability-preserving aspects of the RPL network layer and at the same time provide no compromise over the overall latency of the network. We also introduce a traffic differentiation technique in Symphony by which we can prioritize guaranteed transmissions in a network.

3.5   Need for validation of the protocol in real-life scenarios

The main goal of an industrial research is to bridge the gap between academics and real life industrial implementation. This can be achieved by identifying the application, which the research and innovation will be able to support. In our case, we explore the automobile industry, in specific, the use-case of intra-car communication.

There have been multiple research works that have looked into the possibility of using Wireless Sensor and Actuator networks (WSANs) in intra-car communications. One of the main common motivations for the implementation of WSAN in an intra-car communication is to reduce the weight of the car and increase the engine performance, better fuel economy, and reliability in sensor communication. Researchers have investigated [94] the design aspects of WSANs in intra-car systems and whether these could become a viable solution to partially replace or enhance the wired measurement and control subsystems.

In [95], the authors used IEEE 802.15.4 Compliant and ZigBee-based RF Transceivers to create a Blind Spot Information System (BLIS). Existing BLIS systems implemented by many car manufacturers (e.g., General Motors, Ford, and Volvo) are based on costly hardware components, such as cameras and radars. Their intra-car system was non-instrusive and at the same time cost-efficient. This work provides an ideal location for sensors in an intra-car system to obtain maximum perception. We took inspiration from this design and we have adopted similar location for our sensors in our scenarios taken in Chapter 7. Several case studies like [96], [97] have proven that multi-hop has the potential for significant reliability, robustness, and energy usage improvements over existing single-hop approaches. In their study the authors state that aggregating data in one or several processing centres is critical to the monitoring capabilities of the sensors, which are typically constrained in both energy and computational power. Multi-hop systems, despite its greater overhead, can enhance system reliability, provide better robust performance, and reduce the communication energy. In our work we look into the multi-hop and multi-channel features of DSME, which has the capability to enhance the overall performance of the network.

There have also been several simulation studies [98], [99], [100] on implementing low power and low rate wireless sensor networks for intra-vehicle communications. These works looked through the possibility of implementing a mesh network for intra-car communication. The authors of these works stated that ZigBee could be a good candidate because of its mesh networking capabilities and low power consumption. Zigbee solves multi-path fading using Direct Sequence Spread Spectrum (DSSS) technologies and interference resilience using Carrier Sense Multiple Access (CSMA). The propagation channel inside a vehicle is closed, and is affected by the mechanical vibrations caused by the movement of the vehicle, so authors propose a propagation channel inside a vehicle along with an adaptive equalizer at the receiver.

A major obstacle to be resolved in intra-car systems is to provide a deterministic communication infrastructure that can communicate with absolute bounds in terms of latency to assure a secure application. In this thesis, we test the DSME MAC behavior for intra-car communica-

Table 3.1: Comparison of the various MAC behaviors and its variants

MAC behaviors and variants	Scalability	Reliability	Power Efficiency	Multichannel Access	Group Acknowledgement	Available tools Simulations : S Analytical : A Hardware : H	Applications Intended
RFID	-	medium	high	no	no	S, A, H	Tracking and identification
AMCA	yes	medium	medium	yes	no	NA	non time sensitive applications with high scalability requirements
DSME	yes	high	Depends on the number of nodes	yes	yes	S, A	Time critical applications with high scalability
TSCH	yes	Very high	Higher than DSME	yes	no	S, A	Time critical applications with energy efficiency requirements
A-TSCH	yes	Higher than TSCH	Almost equal to TSCH	Uses blacklisting	no	S, H (GINA)	similar to TSCH -with high reliability requirements
LLDN	no	Very high	higher than DSME	no - Uses TDMA	yes	S	Industrial Automation - fixed number of nodes
MC-LLDN	yes	Very high	Lesser to LLDN	yes	yes	S	Industrial Automation - variable number of nodes
ELIPDA	yes	high	Lesser than DSME	yes	yes	S	Similar to DSME
MA- LLDN	yes	Very high	Higher than LLDN	Uses multi hop	yes	A-Markov chain	similar to LLDN

tion systems and venture into the possibilities of wide adoption of such low rate and low power protocols in the automobile domain.

In the Table 3.1, we compare all the MAC behaviors of IEEE 802.15.4e and the variants discussed above in this chapter. We also provide a brief information on the models (simulations/ analytical/ hardware) available for these protocols and the application they will be fit for.

3.6 Summary

In this chapter we discussed relevant related work addressing the enhancements and the challenges in IEEE 802.15.4e networks. From our analysis, we are able to infer that the protocol is still immature and is open to several possibilities of enhancements to improve the overall QoS especially in non functional parameters, such as energy consumption, latency, reliability and throughput in IEEE 802.15.4e networks.

Chapter 4

Worst Case Dimensioning of IEEE 802.15.4e

4.1 Technical background

Network calculus [68] is a theoretical formulation based on deterministic queuing systems. This formulation helps in understanding and determining structural properties like schedulability, delay bounds, and buffering requirements for specific bounds of traffic in an integrated network. Network calculus is well adapted to controlled traffic sources and it provides upper bounds on delays for traffic flows. This makes network calculus a tailored fit for analyzing and determining the worst-case bounds of the IEEE 802.15.4e MAC behaviors.

Before leaping into the scenarios where network calculus has been utilized to determine the bounds of the IEEE standard 802.15.4e, let us walk-through some of the basic concepts and definitions of network calculus.

Arrival curves: To define the bounds of any network, we first have to determine the guarantees to the specific data flows. These guarantees can be defined by the so called *arrival curves*. Given a wide-sense increasing function α defined for a $t \geq 0$, we say that a flow-through s with input R is constrained by α for all $s \leq t$:

Considering that the flow R has an arrival curve of α, then the bound is be given by:

$$R(t) - R(s) \leq \alpha(t - s) \tag{4.1}$$

The accumulation of all the arrivals for the respective flow of $R(t)$ can be given by $\alpha(t)$, as shown in Figure 4.1 - a. It is also equally important to learn the service provided by the system for the respective traffic flow. The *service curves* help in determining this bound.

Service curves: Depending on resource reservations in the network, the *Service Curve* can be defined as a means of expressing resource availability. Service curves can be categorized into strict, weakly strict, or even variable capacity. To guarantee a strict QoS, it is necessary to specify some minimal service bound for the server.

For a system S, and a flow-through s with input R and an output function $R*$, S offers to the flow, a service curve β, if and only if β is wide-sense increasing, $\beta(0) = 0$ and $R* \geq R \otimes \beta$.

As shown in Figure 4.1 -b, the envelope of the entire service provided $\beta(t)$ will determine the overall service curve of the system.
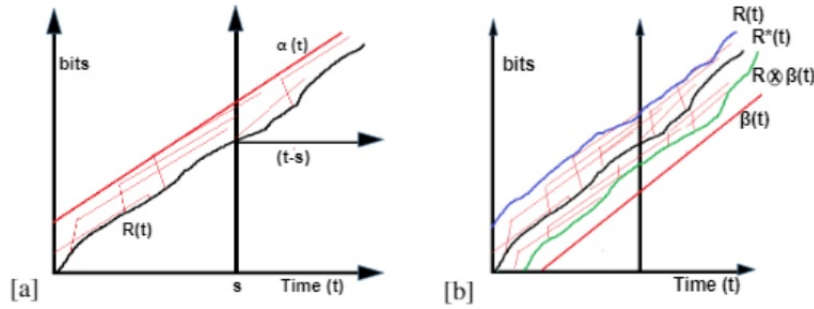
Figure 4.1: a - arrival curve function; b- service curve function

Delay bound using network calculus

For a cumulative arrival function $R(t)$, there exists an arrival curve $\alpha(t) = b + r.t$, Where b, r, t are the burst rate, the data rate, and the time interval, respectively. A minimum service curve β is guaranteed to the arrival function of $R(t)$. The maximum delay of the network is given by the horizontal distance between the arrival and service curves. The delay is computed by the summation of the maximum latency of the service T and the burst rate per data rate, as shown in Equation 4.2:

$$D_{max} = \frac{b}{r} + T \tag{4.2}$$

The variance between the (b, r) curve and the realistic model is adequate for a periodic traffic, which is often the case in Wireless Sensor Networks. Figure 4.2 depicts the basic (b, r) model with the arrival and service curves, and the delay bound (expressed as maximum delay).
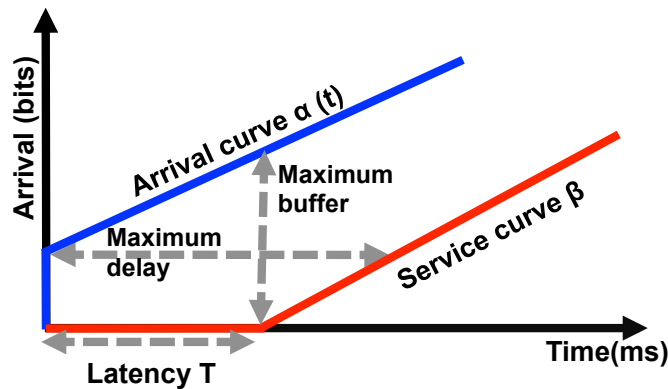
Figure 4.2: Arrival curve, service curve, delay bound

In our work, the leaky bucket (b, r) model has been used to derive the MAC models of IEEE 802.15.4e. It is simple, and it can represent the higher bound of any traffic. An example for the leaky bucket model [101] is shown in Figure 4.3. A bucket of size z is initially empty and is expected to leak at a rate of r units of fluid per second when it is not empty. Now when a fluid amount x has to be utilized, we have to add x amount of fluid into the bucket. When large quantities of x are added into the bucket, it would result in an overflow. In such a case, the system is declared as non-conformant and is then moved on to the next time frame or dropped. Till the case of an overflow, the fluid in the leaky bucket case will be constantly added. Considering communication networks as an analogy, the packets that are sent in a network will be continuously transmitted as long as a service is guaranteed to it.
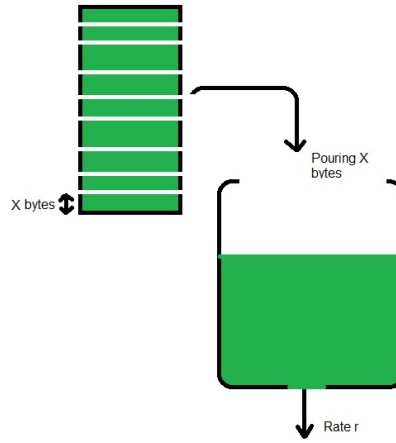
Figure 4.3: Illustration of the leaky bucket model

Keeping the leaky bucket model as a reference, we take the network model of the MAC behaviors of IEEE 802.15.4e and determine their respective arrival, service and delay bounds. Calculating worst-case bounds will help a network engineer to determine the qualities of the MAC behavior and help in devising accurate application scenarios.

In the following sections, we will identify the worst-case bounds on delay and throughput and concurrently learn the parameters of the standard that can have an impact on the overall QoS of an IEEE 802.15.4e network.

4.2 Network analysis for DSME

4.2.1 Worst-case bound analysis of DSME

For the analysis of DSME, let us consider a single PAN Coordinator and a set of coordinator and slave nodes forming a DSME network. This analysis is not topology dependent. The communication in the DSME network supported by the network infrastructure called the multi-superframe. The PAN Coordinator sends an Enhanced Beacon for every multi-superframe

interval and a beacon is sent for every superframe interval by the coordinators. The Beacon Interval (BI) and the Superframe Duration (SD) can be represented as shown in the Figure 4.4.
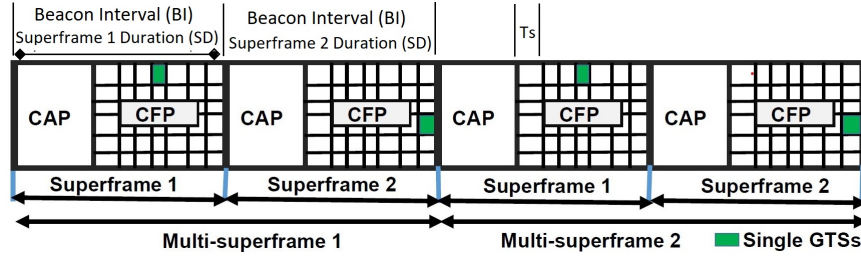
Figure 4.4: Structure of the DSME superframe

The BI and SD are computed as follows:

$$BI = aBaseSuperframeDuration \times 2^{BO} symbols \tag{4.3}$$

$$for\ 0 \le BO \le 14$$

$$SD = aBaseSuperframeDuration \times 2^{SO} symbols \tag{4.4}$$

$$for\ 0 \le SO \le BO \le 14$$

In these equations, $aBaseSuperframeDuration$ represents the minimum length of the superframe (i.e. SO = 0). The IEEE 802.15.4e standard has fixed this value to 960 symbols. Each symbol corresponds to 4 bits, resulting in a duration of 15.36 ms, considering an ideal data rate of 250 Kbps. It is mandatory that the data transmission, Inter Frame Spacing (IFS) and Acknowledgment/Group Acknowledgment (if requested) are accommodated within the end of a DSME GTS slot for the successful transmission of a message. For the sake of simplicity, we consider one data frame transmission in one DSME GTS per superframe. The size of a timeslot in a superframe, T_s, is given by,

$$T_s = \frac{SD}{16} = aBaseSuperframeDuration \times 2^{SO-4} \tag{4.5}$$

Every timeslot T_s in a superframe comprises T_{data} and T_{idle}. T_{data} is the maximum duration used for data transmission in a guaranteed timeslot. T_{idle} is the time period that accommodates the acknowledgments and IFS in the network. As shown in Equation 4.6, the latency T is the time for which a burst waits to get served. This is the difference between the bursts arrival at the Beacon Interval and the time at which data is served.

$$T = BI - T_s \tag{4.6}$$

The overall service provided by the network can be given by the product of the data rate and the time at which the system receives the service. As the overall service for DSME has to

be calculated in accordance to a data rate, we multiply the service offered by a constant data rate "C". The service given for the GTS, i.e., the number of bits that has to be sent during a GTS within a time t, is as given by Equation 4.7,

$$\beta_1 = \begin{cases} C\left((t-(BI-T_s))\right)^+, \forall\, 0 \le t \le BI - T_{idle} \\ 0, \quad otherwise \end{cases} \tag{4.7}$$

This value of the service curve can now be derived for N number of superframes, similar to the equation derived for the service curve for n superframes of the IEEE 802.15.4 in [102]. When considering N superframes, we must take the entire service by summing the overall service provided for the data frames. This service of the N_{th} superframe can be represented as:

$$\beta_N = \begin{cases} (N-1)CT_{data} + C\left(t-(N(BI-T_N))\right)^+ \\ \forall 0 \le t \le (N-1)BI - T_{idle} \\ 0, \quad otherwise \end{cases} \tag{4.8}$$

The resulting DSME GTS service curves is represented as a staircase in Figure 4.5. The elevation in the stairs (red line) occurs whenever a service is provided for a GTS. On the other hand, when the network is idle, the stair remains stagnant (green line) without any elevation. An active transmission is represented by a green timeslot in Figure 4.5.
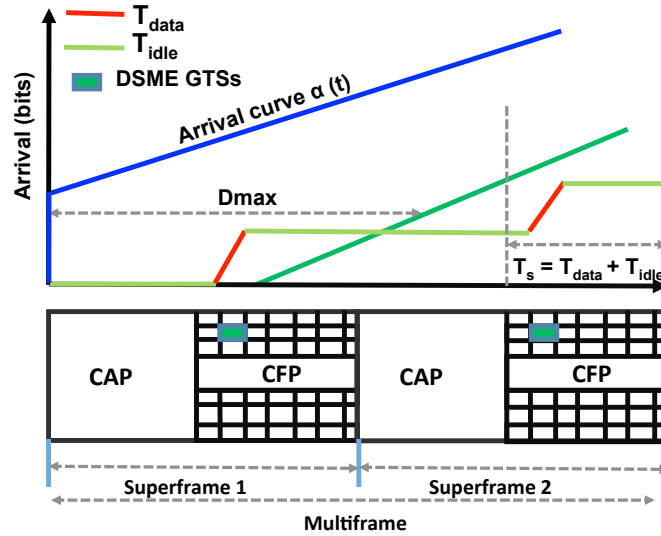
Figure 4.5: Service curve of DSME MAC

Delay bound and throughput analysis of DSME

The summation of every individual delay bounds in a superframe will give the overall delay bound of a multi-superframe. For a burst size b greater than CT_{data}, the maximum delay bound of the first superframe D_{max1} with m channels will be the horizontal angular distance between the arrival curve and the first stair shown in Figure 4.5. In accordance to Figure 4.1 (b),

a minimum service of $\beta(t)$ will be provided for the cumulative data flow $R(t)$, and the delay for this service will be given by:

$$D_{max1} = \frac{b}{C} + (BI - T_s) \ \ if \ \ b \leq CT_{data} \tag{4.9}$$

As the data in DSME has the possibility to be carried over different channels, the overall delay must also be calculated by taking the multi-channel feature into consideration. The data transmitted over different channels can significantly reduce the delay. In our case we consider m number of channels with good link quality. When $N(CT_{data}) < b \leq (N+1)CT_{data})$, the delay of the system with N number of super-frames and m channels is given by:

$$D_{maxN} = \frac{b}{m \times C} + ((N+1)BI - T_s) - Nt_{data} \tag{4.10}$$
$$if N(CT_{data}) < b \leq (N+1)CT_{data}$$

We use the method as used in [103] for the throughput calculation of all the time critical MAC behaviors. From a network perspective, considering the entire CFP, the channel capacity will have an increasing impact on the overall network throughput. The following Equation 4.11 is derived based on the throughput derivation devised in [103]. This equation represents the overall network throughput, which is defined as the maximum amount of traffic that can be transmitted simultaneously over the network. The throughput formulated for m channels and n superframes is given by:

$$Th_{max} = n \times \min \begin{cases} (b + rT_S)/BI, \\ \\ \max \begin{cases} ((T_s - (N_{LIFS} - 1) \cdot LIFS \\ -\triangle(IFS) \ C \cdot m/BI, \\ \\ \\ Ts - N_{SIFS} \cdot SIFS)) \ C \cdot m/BI \end{cases} \end{cases} \tag{4.11}$$

4.2.2 Quality of Service analysis for DSME

DSMEs´ unique multi-superframe can stack several superframes one after the other within a specific multi-superframe interval. It has the same functionalities of a superframe, but with additional features like multi-channel access and group acknowledgment. If compared, a single transmission in a GTS slot of a DSME and a IEEE 802.15.4 superframe with conditions like same superframe duration, same traffic and same burst size, the throughput will be the same.

However a comparison can be made for the overall network throughput, which is the maximum traffic transmitted simultaneously over the network, for a superframe of IEEE 802.15.4 and a multi-superframe of IEEE 802.15.4e. For this analysis, we consider a multi-superframe of IEEE 802.15.4e that accommodates three superframes in it, and we compare it with a IEEE 802.15.4 superframe of equal duration. The scenario taken for the experiment is depicted in Figure 4.6.

In this case, the GTS allocation throughput must be higher in DSME as three GTS bandwidths are utilized in a multi-superframe, whereas only a single GTS is been utilized under the same duration in IEEE 802.15.4. It must also be noted that this analysis is applicable only for transmissions of lengths equal to that of IEEE 802.5.4 guaranteed timeslots.
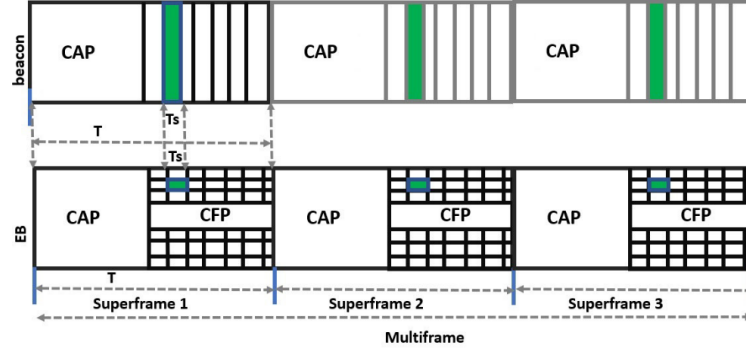
Figure 4.6: Scenario -DSME throughput analysis

Scenario 1: Throughput analysis - IEEE 802.15.4 vs DSME

The experiment was carried out for various Superframe Orders ranging in the interval [0,10], for arrival rates varying such as (5,50,100) Kbps at a constant burst size of 2 Kbits. These values are compared with the throughput of legacy IEEE 802.15.4 calculated in [103]. Based on the Equations (4.11, 4.10) provided in the previous section we calculate the throughput and the delay values for a DSME-enabled network.
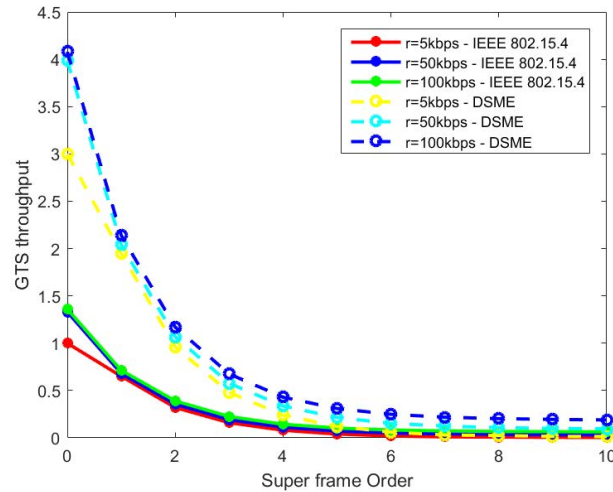
Figure 4.7: DSME throughput - function of arrival rates

It can be noticed that both IEEE 802.15.4 and IEEE 802.15.4e follow the same behavior; that is, the throughput starts to reduce when there is an increase in the value of the Superframe Order. As the Superframe Order increases, the length of the timeslot and beacon interval increase concurrently, evidently affecting the overall throughput. From Figure 4.7, it can be noticed that

the value of throughput is around 25% higher than the legacy IEEE 802.15.4. The increased throughput in DSME is a result of the modified superframe, which has the ability to support multi-channel access. This property helps us to understand why DSME is more suitable for applications with high scalability and throughput requirements.

Scenario 2: Impact of superframe structure and CAP reduction

Having established the advantage of DSME over IEEE 802.15.4 in terms of throughput, we tried to compare the unique features of DSME such as multi-superframe structure and CAP reduction. The CAP reduction technique is explained in Chapter 2 (see Figure 2.6). For this experiment we consider the same parameters as those taken for the throughput test, but now we use a single channel to clearly depict the differences between the cases with and without CAP reduction. The scenarios taken for comparison are as depicted in the Figure 4.8, in which we take two multi-superframes which accommodates three superframes of an equal duration. We consider the following scenarios: (i) without CAP reduction which has three CAP and CFP regions, within which three GTS are allocated; (ii) a scenario with CAP reduction employed in which five GTS are allocated in an extended Contention Free Period region.
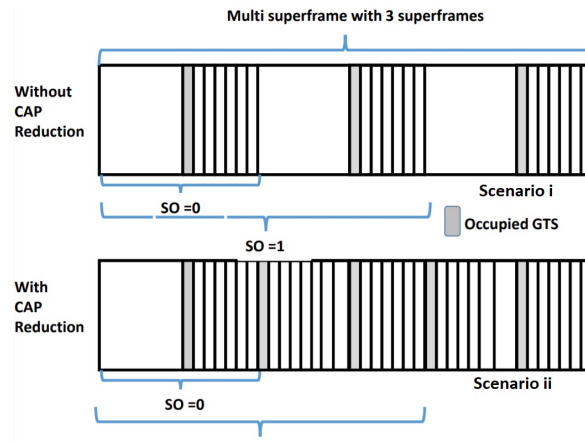
Figure 4.8: CAP reduction - scenarios

This comparative analysis is depicted in Figure 4.9 from which it can be inferred that a 7% increased network throughput is obtained under CAP reduction. CAP reduction radically increases the number of Guaranteed Timeslots (GTS), thus providing additional CFP bandwidth. It is also notable that the CAP reduction increases the overall scalability of the system by the inclusion of additional DSME GTS.. However, there can be some reduction in throughput if the bandwidth is not utilized properly, but unlike the traditional IEEE 802.15.4 the nodes need not wait for any contention based period to get accommodated in their respective timeslots.

Scenario 3: Impact on delay with multi-channel DSME

DSME also provides the capability to accommodate a higher number of data transmissions (flows) within a single timeslot because of its multi-channel capabilities. We compare the delay performance of a multi-channel DSME against the legacy IEEE 802.15.4. In the case of DSME, we use multiple (2, 3, 4, 6) channels providing an equal bandwidth of 20 Kbps for transmission.
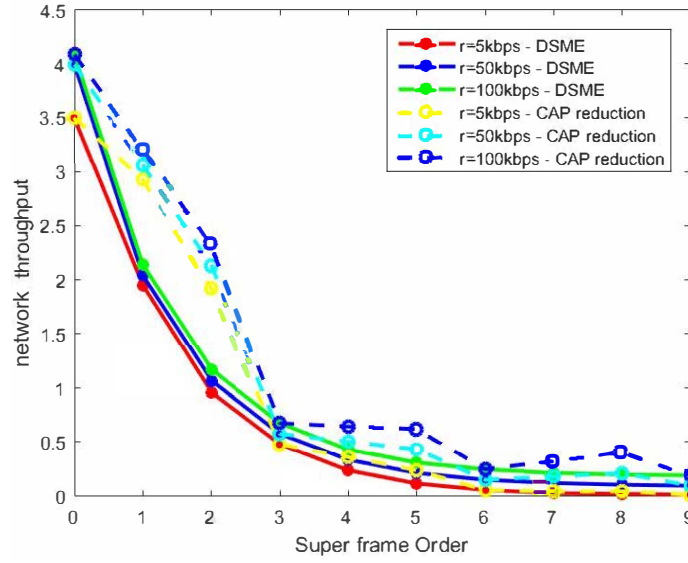
Figure 4.9: DSME throughput analysis - CAP reduction

Figure 4.10 shows the delay calculation of the DSME with respect to the legacy IEEE 802.15.4. It is clearly evident that DSME outperforms IEEE 802.15.4 because of the multi-channel capability. In this comparison analysis we infer that DSME is able to provide a reduction in the delay from 20% to 40%, based on the number of channels taken into consideration.
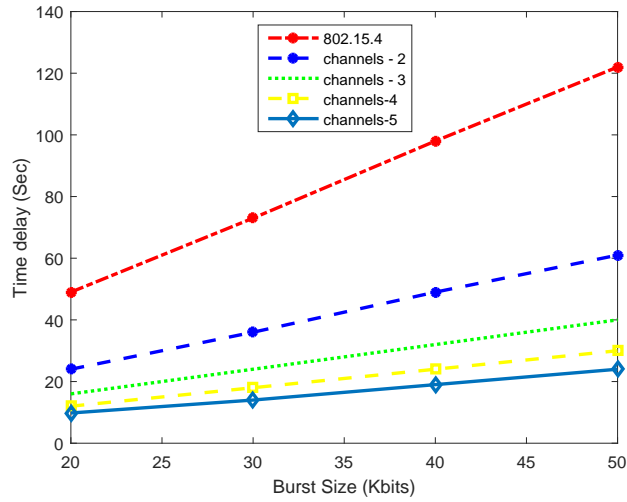
Figure 4.10: DSME delay analysis - function of burst size

4.3 Network analysis for TSCH

4.3.1 Worst-case bound analysis of TSCH

To obtain the worst-case bounds of the TSCH MAC behavior, first we derive an expression for the delay bound of an arrival rate $R(t)$ bounded by a (b,r) curve for a single timeslot in a non

contention-based slotframe. The default duration of every timeslot T_s is fixed at 10 ms [35], within which it has to accommodate the acknowledgment delays (on the receiver and transmitter end), and the receiver and transmitting frames during a transmission in non-shared dedicated timeslot. Unlike DSME, TSCH has a different network supporting infrastructure that can be fully set as a contention-free period.

Every timeslot T_s (time duration of a single timeslot) is of equal durations and is composed of T_{data} and T_{idle}. T_{data} is the time duration of a data transmission in the timeslot. T_{idle} comprises of the acknowledgment delays, MAC transmission and reception offsets. Let us consider T_{cycle} to be the duration for which the slotframes repeat periodically. The latency T for data transmission in one timeslot in a slotframe is as given by Equation 4.12, and the service obtained by the first slotframe for a time t is given by Equation 4.13.

$$T = T_{cycle} - T_s \tag{4.12}$$

$$\beta = \begin{cases} C\left(t - (T_{cycle} - T_s)\right)^+ & \forall\, 0 \le t \le T_{cycle} - T_{idle} \\ 0, & otherwise \end{cases} \tag{4.13}$$

Considering a TSCH-enabled network with N number of slotframes, the overall service of the system for the Nth timeslot can be computed as follows:

$$\beta_N = \begin{cases} (N-1)CT_{data} + C\left(t - (NT_{cycle} - T_N)\right)^+ \\ \forall 0 \le t \le (N-1)\ \ (T_{cycle} - T_{idle}) \\ 0, & otherwise \end{cases} \tag{4.14}$$

The service curve of the TSCH MAC behavior results in a staircase (see Figure 4.11). Akin to the DSME service curve, TSCH service curves also show a similar elevation when the service is provided; in the example, timelsot 2 carries data, and a service is provided to it with respect to the arrival curve. The service curve remains stagnant when there is no data to transmit.

Delay bound and throughput analysis of TSCH

The delay can be calculated through the horizontal distance between the arrival of the packets and the moment a service is provided to them. In this regard, for the first slotframe, let us assume that $b \le CT_{data}$, as we have a steady data transmission in every TSCH time intervals. The maximum delay bound D_{max1}, is the horizontal distance between the arrival curve and the first stair. For this analysis, we assume that a minimum service of $\beta(t)$ will be provided for the cumulative data flow $R(t)$. The delay for achieving service in this slotframe will be given by:

$$D_{max1} = \frac{b}{C} + \left(\,T_{cycle} - T_s\right) \tag{4.15}$$

Figure 4.11: Example of a service curve of TSCH MAC
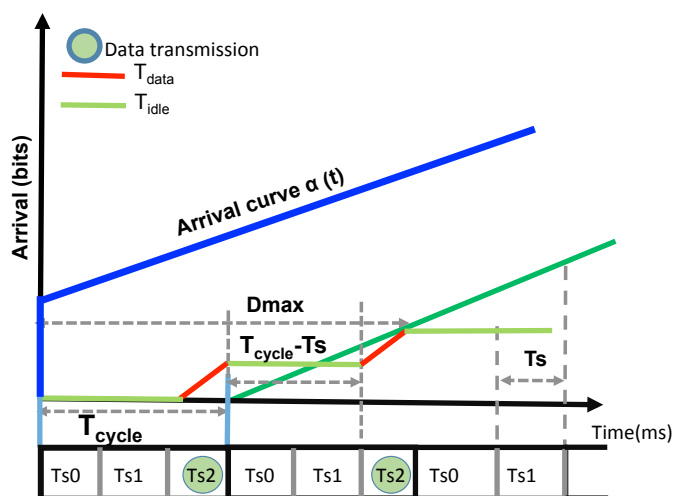
When $N(CT_{data}) < b \leq (N+1)CT_{data})$, (generalizing for N slotframes in the TSCH network), the overall network delay can be given by:

$$D_{maxN} = \frac{b}{C} + ((N+1)T_{cycle} - T_s) - Nt_{data} \tag{4.16}$$

As the nodes in the network transmit in a repetitive fashion, the overall throughput of TSCH networks will be a function of the duty cycle. For this analysis we consider a steady data rate of C in the TSCH network. $T_{dutyCycle}$ is the ratio of the T_{Cycle}, which is the active period of the network out of the total number of $T_{Cycles(N)}$ present in the network. The throughput with respect to a single timeslot can be then given by:

$$Throughput = (T_{data}/T_{dutyCycle})C \tag{4.17}$$

Considering the multi-channel features of TSCH and assuming m as the number of channels,then, the value of the network throughput Th_{Max} (which is the maximum traffic transmitted over the multi-channel TSCH network) is given by:

$$Th_{max} = ((b + r \cdot T_S)/T_{dutycycle})m \cdot C \tag{4.18}$$

4.3.2 Qualtiy of Service analysis for TSCH

TSCH is different from DSME and LLDN as it lacks a superframe structure; but it is made of slotframes and it works based on Time Division Multiplexing Access (TDMA). The throughput of the system can be calculated in accordance to the time the device remains active in the network. For this purpose, we propose a duty cycle which is "the ratio of the time that the device remains active in the system". This duty cycle will be a direct proportionate value of $Tcycle$. For this experiment of throughput analysis, we increase the duty cycles for different arrival rates

and calculate the corresponding throughput. In Figure 4.12, it is clear that the overall network throughput for the maximum traffic decreases with the increase of the duty cycle. In addition, higher arrival rates allow achieving higher throughputs. The duty cycles have a similar relation to the Superframe Orders of DSME and LLDN, and they follow the same behavior.
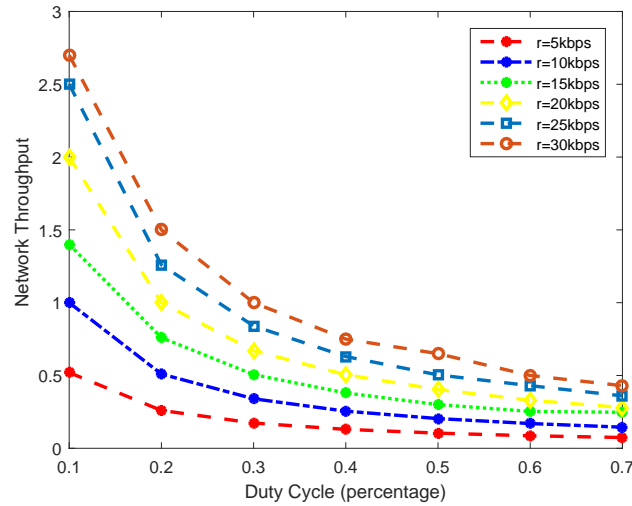
Figure 4.12: TSCH throughput - function of duty cycle

Concerning delay, TSCH has a very interesting property. The timeslot of TSCH has a fixed duration of 10ms. So when the $Tcycle$s value is within 10ms it makes sure the delay is almost 0 if it can accommodate the nodes within that respective time interval. When above 10ms it will accommodate another timeslot, thereby increasing the delay proportionately. The delay will be more dependent on the number of nodes that occupy the network and their respective $Tcycle$s. In Figure 4.13, after having a nil delay till the duration of 10ms, the delay starts to slowly increase proportionate to the number of nodes accommodated in the network.
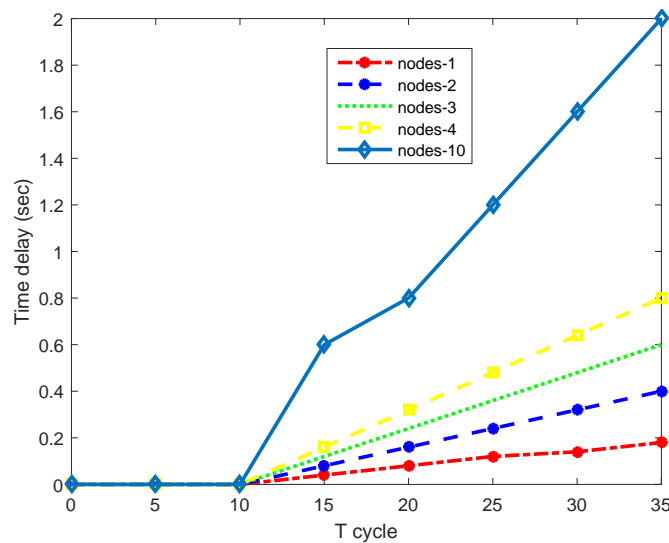
Figure 4.13: TSCH delay analysis - delay over $Tcycle$

4.4   Network analysis for LLDN

4.4.1   Worst-case bound analysis of LLDN

LLDN follows three different states for setting up a network namely: Discovery; Configuration and Online. In the *Discovery state*, the device that wants to join the network scans the available channels for a LLDN PAN Coordinator that is broadcasting beacons indicating discovery state. The scanning device sends its current configuration to the PAN Coordinator during this state, which is in-turn acknowledged. In the *Configuration state* the PAN Coordinator sends the new configuration for the receiving device such as the length of the management slots and the directions of the bidirectional frames. In the *Online state*, data transmission occurs. Each device will receive a number of shared/dedicated timeslots unique to its respective IDs. In case of collisions, the LLDN facilitates retransmission using uplink timeslots.

For the service curve analysis of LLDN, we consider the Online state. We design the network calculus model for the data transmission from a dedicated node to a PAN Coordinator (uplink timeslot) and the transmission from PAN Coordinator to the node using a bidirectional timeslot (configured to downlink). It is mandatory that the data transmission, Inter-Frame Spacing and acknowledgments/Group Acknowledgments (if requested) complete within the end of the allocated timeslot for a successful data transmission.

Let us consider a dedicated slot allocated for single node as T_{Uplink}. The timeslot allocated for the transmission of data from the PAN Coordinator will be $T_{downlink}$. Both are composed of T_{data} and T_{idle}. The maximum duration used for data transmission inside the dedicated timeslot T_{data} and T_{idle} comprises the time occupied by Inter-Frame Spacing (IFS) and Group Acknowledgments. The latency for a LLDN enabled network is the difference between the bursts arrival (start of the beacon interval) and the time the data is served either in the *uplink* or the *downlink*. The maximum latency, T, either in the *uplink* or in the *downlink* is the time a burst may wait for a service. It is given by Equation 4.19:

$$T = BI - [T_{uplink/downlink}] \tag{4.19}$$

The total service provided by the network is given as the product of the data rate and the time at which the system receives the service. The service curve measured over time t, is the service provided for the data that is transmitted during an uplink of a dedicated node. (Figure 4.14)

$$\beta_{uplink} = \begin{cases} C\left(t - (BI - T_{uplink})\right)^{+} \\ \forall 0 \le t \le (N-1)BI - T_{idle} \\ 0, \quad otherwise \end{cases} \tag{4.20}$$

Similarly the service curves for the downlink slots can be derived as,

$$\beta_{downlink} = \begin{cases} C\,(t - (BI - T_{downlink}))^+ \\ \forall 0 \leq t \leq (N-1)BI - T_{idle} \\ 0, \quad otherwise \end{cases} \tag{4.21}$$

LLDN works using a Time Division Multiplexing Access (TDMA) protocol. The superframes repeat in cyclic intervals. If N is the total number of cycles for which the superframe repeats, the service of the total system is then given by:

$$\beta_N = N \times \beta_{[uplink/downlink]} \tag{4.22}$$
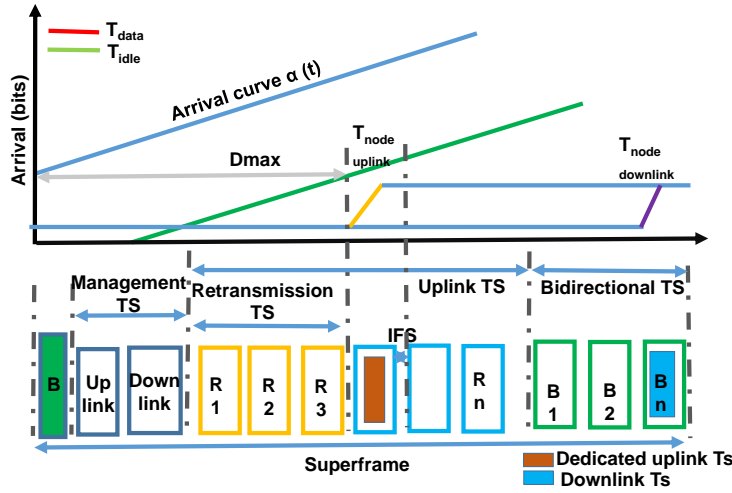
Figure 4.14: Service curve of LLDN MAC

Delay bound and throughput analysis of LLDN

To calculate the delay bound, we consider the transmission of data in one timeslot, T_s, in a single LLDN frame. T_s can either be an uplink timeslot or a downlink timeslot. The maximum delay bound will be the horizontal linear distance between the arrival curve and the first stair. The value of the delay can be given as follows:

$$D_{max} = \frac{b}{C} + (\,BI - T_s) \tag{4.23}$$

The maximum delay of a network having N superframes can be given by:

$$D_{max\,network} = \sum_1^N D_{maxn} \tag{4.24}$$

The throughput of a single LLDN node depends on the T_{data} which is composed of either T_{uplink} or $T_{downlink}$ or both, depending upon the configuration of the network. The throughput equation is then given by:

$$Th_{max} = (T_{uplink/downlink}/BI)C \tag{4.25}$$

The overall network throughput of a LLDN is similar to the one of the IEEE 802.15.4 derived in [103]. Considering that T_S can either be an *uplink* or a *downlink*, the maximum traffic transmitted simultaneously over the network, i.e the network throughput, is as given by Equation 4.26:

$$Th_{max} = \min \begin{cases} (b + rT_S)/BI, \\ \max \begin{cases} ((T_s - (N_{LIFS} - 1) \cdot LIFS \\ -\triangle(IFS) \; C/BI, \\ \\ Ts - N_{SIFS} \cdot SIFS)) \; C/BI \end{cases} \end{cases} \tag{4.26}$$

4.4.2 Quality of Service analysis for LLDN

Similar to the DSME, LLDN also follows a superframe structure. However, LLDN lacks multi-channel access and multi-superframe capabilities, and hence the throughput of a LLDN-enabled network will be very dependent on the bandwidth allotted for the uplink timeslots, downlink timeslots and the retransmission timeslots. We use the network calculus model of LLDN to analytically learn the impact of the Superframe Order on the overall network throughput. The respective results are illustrated in the Figure 4.15. The experiment was conducted for a constant burst rate of 5 Kbits and the arrival rate was varied from 60 -250 Kbps. This experiment, which was conducted for the maximum traffic the network can withhold, provided us similar results as for the case of DSME. There is steady decrease in throughput with the increase Superframe Order because of wasted bandwidth. The average throughput for LLDN is lesser DSME by almost 20% because of its lack of multi-superframe extension. LLDN assures a very stable delay as the values do not sway over impact of multi-channel, but rather it is more fixed depending on the size of the corresponding uplink and downlink timeslots. Being well known for providing retransmission timeslots as a part of the uplink timeslots, the time for retransmission contributes to the delay but helps in providing more reliability to the network.

This experiment on delay was carried out for a variable number (5-15) of timeslots (uplink and downlink) as shown in Figure 4.16. When there is higher number of timeslots to accommodate the nodes, there is a stern decrease in delay. It can be noticed that delay increases by almost 80% when the number of nodes are increased from 1 to 10. Unlike the TSCH, the size of timeslots in the network is quite flexible for a LLDN-oriented system, allowing it to perform better in terms of delay.
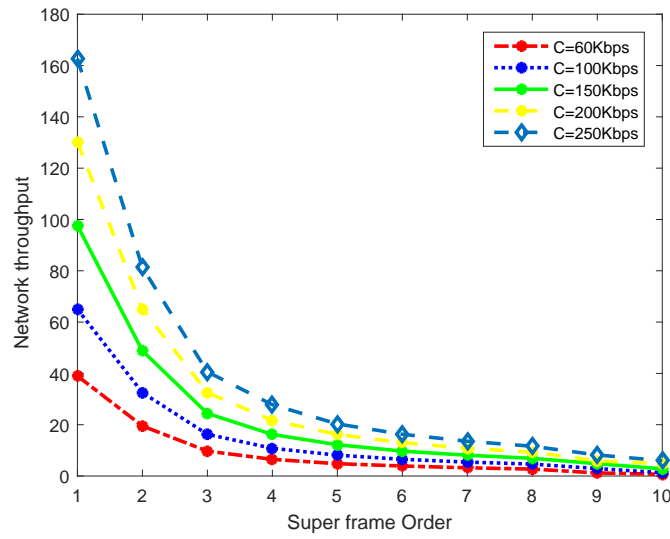
Figure 4.15: LLDN throughput - function of arrival rates
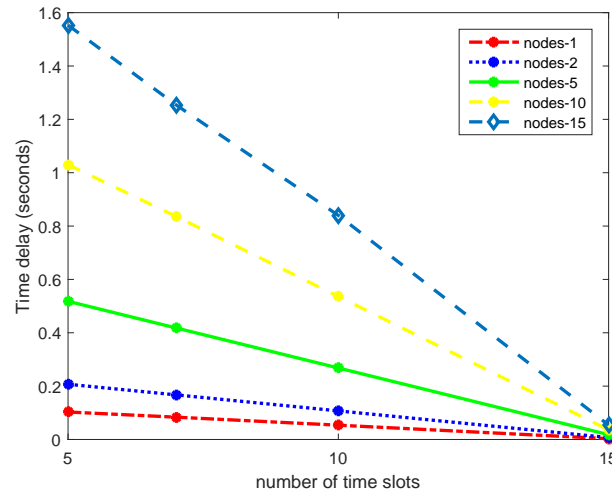
Figure 4.16: LLDN delay analysis

4.5   Open-ZB models for time-critical MAC behaviors

The Open-ZB [104] is a web site developed by CISTER Labs which provides open source tools for IEEE 802.15.4 and ZigBee. It provides several experimental tools (TinyOS implementation) for the time synchronized (beacon-enabled) mode of the IEEE 802.15.4 protocol, ZigBee Network Layer supporting the cluster-tree topology and security sublayer of the IEEE 802.15.4. It also provides simulation tools for IEEE 802.15.4 and a matlab tool for worst-case dimensioning of IEEE 802.15.4/ZigBee cluster-tree wireless sensor networks. We have developed a matlab tool that compliments to the Open-ZB open source platform. This tool supports the worst case dimensioning of the DSME, LLDN and TSCH MAC behaviors of the IEEE 802.15.4e. It helps in providing the throughput and latency for the network parameters given as an input. It also provides a graphical representation of the QoS such as throughput or latency.

Figure 4.17 shows the GUI of the MATLAB tool developed for LLDN.
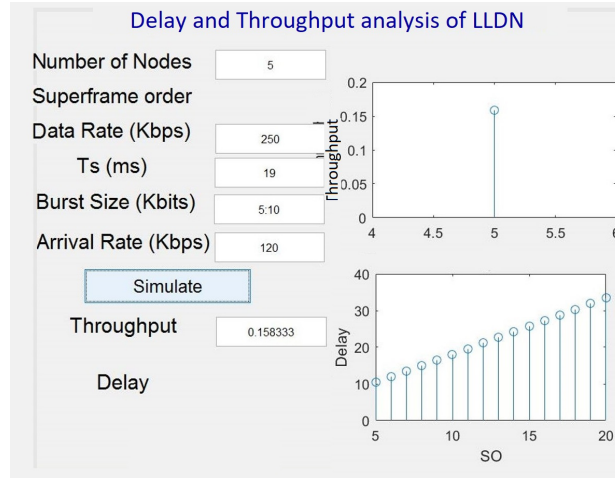
Figure 4.17: GUI of the MATLAB tool for LLDN

Using this tool, we can input the individual values of SO, number of nodes to visualize the impact of a specific parameter on QoS. In Figure 4.17 we can see the delay represented for a set of superframes, for an increasing order SO in a LLDN-enabled network. The value of throughput is also represented for a single burst size.

4.6 Summary

In this chapter, we derived the worst case bounds of the DSME, TSCH and LLDN MAC behaviors to guarantee latency bounds and reliability for a IEEE 802.15.4e network. We also provided a QoS analysis in terms of throughput and delay to understand the impact of several features the IEEE 802.15.4e. Because of multi-channel access, a DSME network is able to outperform the basic IEEE 802.15.4 in terms of both end-to-end delay and throughput. We further explored the different capabilities of DSME, such as CAP reduction, to analyze its features and advantages. We also analyzed the QoS of LLDN and TSCH to understand the boundaries of these MAC behaviors and how they can compare against the deterministic capabilities of the DSME. From our analyses, we can infer that these MAC behaviors have been designed to suit specific application scenarios. For example, the DSME is a suitable MAC behavior to implement large scale applications such as structural health monitoring where high number nodes have to be connected to a network. On the other hand, LLDN is more suitable for low-latency and dense applications, in which the network has to be robust and at the same time provide low latencies. TSCH would be efficient for applications that demand low end to end delays with short data transmissions. We believe that this work will enable the research and network engineering community to design more efficient ways of scheduling transmissions over these protocols and carrying out efficient network planning, by computing, in advance, the worst case service and needed resources.

Chapter 5

Symphony - A Cross Layer Approach on DSME Networks

IEEE 802.15.4e provides a rigid MAC layer and several infrastructures to support complex network topologies. Despite providing several infrastructures, the standard does not provide any strict guidelines on placing the transmissions of the nodes in the appropriate timeslots. Scheduling of the transmissions in a network will have a dire impact on the Quality of Service (QoS) of a network. For instance, providing additional timeslots to the nodes to retransmit can improve the overall reliability of the system. However, such a setup will reduce the useful bandwidth and affect the overall latency. The challenge here lies in determining a scheduling algorithm that could improve the QoS of the network without having more substantial trade-offs.

The word "Symphony" means an extended musical composition in western classical music, which is written for orchestras and musicals. They are usually scored for several instruments that are played together to produce a unified sound. It is derived from the Greek word "$\sigma\upsilon\mu\phi\psi\vartheta\grave{\iota}\alpha$ - Symphonia," which means agreement or concordance with sound. Isidore wrote the first-ever known symphony of Seville for a two-headed drum during c. 1155 to 1377. During the late 17th century, musicians like Mozart revolutionized the music theory of symphony with the addition of several movements like Sonata, Allegro, Minuet, and Rondo, but the general concept behind symphony remained the same. The concept was playing different musical notes within the same or a limited timeframe to produce a cordial tone.

Increased number of notes played within a limited time-frame produces a large amplitude of the sound. Musical movements like allegro hold a steady speed (tempo) to provide a stronger and louder musical tone than an adagio that has fewer music notes within a time-frame. In the Figure5.1 below, we compare an "adagio" a version of the music in which fewer music notes are present within a time-frame and an "allegro" in which more music notes are present within the same time-limit.

Considering that the airwaves in which music gets transmitted are channels, this basic ideology in music theory gave us a thought of making an algorithm that could help several different transmissions to be transmitted within the same timeframe. One such algorithm will have
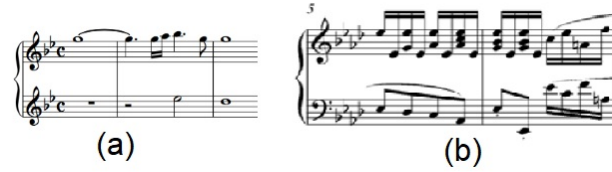
Figure 5.1: (a) adagio within 2 timeframes, (b) allegro within 2 timeframes

a chance to be a candidate for several IoT applications as it will be able to accommodate many devices within a limited or a defined time-frame. Having several transmissions within a limited time-frame can help achieve more scalability and timeliness. We ventured into this idea to form an algorithm named thereafter - "Symphony," which helps to achieve this.

During the initial stages of developing such a scheduling algorithm for IEEE 802.15.4e networks, both TSCH and DSME were equal contenders on which Symphony was intended to be implemented. Both DSME and TSCH have their own pros and cons in network implementation. For instance, TSCH, with its constant 10ms slot period has a very desirable feature for latency hungry applications. However, from our simulations and our analytical work, we were able to understand that DSME is a much desirable candidate when we consider QoS such as scalability and guaranteed latency. When the number of nodes increases above 30, the throughput of the DSME enabled networks can be much better because of supported features such as the CAP reduction. Our research work in [9] and other research studies like [67] prove the efficiency of DSME over TSCH when the network size is large. Other than scalability, communications using the GTS timeslots of DSME can be time-bounded and are very reliable and more suitable for low-rate IoT networking.

5.1   Integer Linear Programming

Schedule formation is one of the very common application areas in which Integer Linear Programming (ILP) can be used. ILP helps in the creation of communication schedules by determining optimal links to improve a certain QoS during communication. It is a handy tool for determining schedules for complex mesh networks with a large number of nodes. For formulating an ILP based solution, the user must set some constraints that can have an impact on QoS properties such as latency, throughput, and energy efficiency.

For obtaining optimal schedules in a wireless network, we should follow the methodology of a Constraint Satisfaction Problem (CSP) which comprises of the following factors:

- a set of variables $a_1, a_2, ..., a_n$

- for each variable a_i, a finite set D_i of possible values (which belongs to it or be its subset)

- a set of constraints that will restrict the variable in every subset

As constraints can affect the domain/subset of variables, the CSP need not be linear like that of the traditional ILP model. An optimal solution of a CSP is the assignment of every variable within the domain by meeting the required constraints.

We first set preferences for every individual link in the network and also can form a relation between the sets of transmissions (subsets) and determine the need for necessary preemption in allocation. CSP is usually set to obtain one of the feasible solutions, but in our case, we adapt it to make it more optimal. An initial feasible solution is first obtained. Then, with the introduction of constraints, a better solution is obtained. The constraints are checked periodically, and the solution (schedule) is accordingly updated. It is thus resulting in a better or optimal allocation of the GTS during the overall network time.

ILP problem statement:

Creating an optimal schedule for the guaranteed timeslots in a DSME mesh network such that minimal timeslots and maximum bandwidth are stringently utilized

Assumptions:

For the Integer linear programming (ILP) model, we take an example of a mesh network with 5 DSME-enabled nodes that are interconnected, as shown in Figure 5.2. For this network, we consider that all the nodes are FFDs that are capable of transmitting and receiving information. This model also can be extended to a network with several slave nodes with minimal functionality connected to a FFD. For this model, we only consider the guaranteed timeslots in the CFP region of the DSME superframe. For an optimal schedule, the transmissions are expected to be packed in the GTS in such a way that lesser timeslots are used, utilizing the advantage of multi-channel access. Using multi-channel access to its full extent raises a constraint that the same nodes must not transmit or receive within the same timeslot. This constraint helps in assuring that no interference or hidden node problem exists in the network.

Step 1: dependency graph formation: For formulating our ILP, we first define all the transmissions as flows for easy readability. The flows for the topology shown in the Figure 5.2 can be defined as:

$$c \rightarrow d(a1), c \rightarrow a(a2), a \rightarrow b(a3), b \rightarrow e(a4), b \rightarrow d(a5), d \rightarrow f(a6), f \rightarrow a(a7), e \rightarrow f(a8)$$
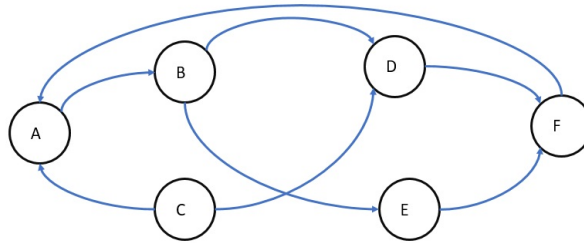
Figure 5.2: example of a mesh network

As two transmissions with identical nodes cannot go along the same timeslot, it can be defined that flow $a1$ cannot go along with $a2$, $a5$ and $a6$ in the same timeslot, because (c or d) is repeated in $a2$, $a5$ and $a6$, either in the transmission or in the reception. Hence, if we

consider $a1$ to be in the first timeslot, either of $a2$, $a5$, or $a6$ shall not be placed within that same timeslot. From this formulation, if we place $a1$ in the initial timeslot, the start time of $a1$ will be precedent to those which are not in dependency with it. Now we make a dependency graph based on the aforementioned factors. The dependency graph clearly visualizes flows that can go concomitantly with other flows.
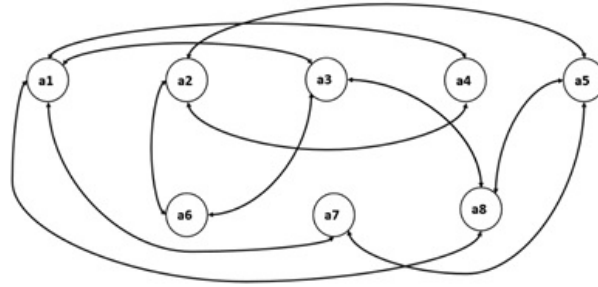
Figure 5.3: Dependency graph of the network

Figure 5.3 shows several loop cycles with the routes formed between nodes that have the capability of associating with each other. For instance, $a1$ forms a connection with $a3$ as different transmissions with nodes c, d, a, b can co-exist along the same timeslot. So if $a1$ is placed in the first timeslot, flows such as $a3$ and $a4$ can also be placed in the initial timeslot. Using this step, we primarily determine the set of variables and its respective subsets.

Step 2 - Cyclic scheduling formulation:

Having obtained the dependencies and formed subsets of transmissions that can go along with each other, we now try to obtain the optimal schedule using cyclic scheduling [105] which is an approach that is much suited for networks that get scheduled periodically. In our case, the DSME multi-superframe repeats for every multi-superframe duration, and the change in the schedule and the respective preemptions are updated periodically. Hence, we take the cyclic scheduling approach for solving our problem.

In a dependency graph formed with uniform constraints, the input taken is a T set of tasks with processing times $p1, p2, p3....$ For every single branch (Figure 5.4) in the dependency graph governed by constraints, the cyclic scheduling formulation can be given for every set of transmissions. For every branch $L(a)$ - length of the arc (delay/latency) and $H(a)$ - height of the arc (dependency length) must be defined. For any time instant within a fixed time of K, the time taken for the overall schedule (T_σ) in a periodically repeating ω_i can be given as:

$$T_\sigma(a1, a2) = T_\sigma(a1, 1) + (K - 1)\omega_i \tag{5.1}$$

also the iteration delay D_σ can defined as:

$$D_\sigma = \max_{i \epsilon T} T_\sigma(i, K) - \min_{i \epsilon T} T_\sigma(i, K) \rightarrow_{k \to \infty} \tag{5.2}$$
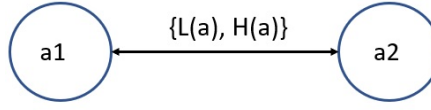
Figure 5.4: Branch of a dependency graph

For our problem let us consider α to be the start time of a transmission. Let $\widetilde{\alpha_1}$ be the start time within the initial period for the transmission from $a1$. In our problem we assume to have fixed guaranteed time that is supposed to accommodate the transmission and an eventual acknowledgment. We consider this time index value to be $T_{(i,j)}$. Considering the iteration of the schedule change that occurs every beacon period (BI), this start time can be defined as follows:

$$\alpha = \widetilde{\alpha_1} + T_{(i,j)} BI \forall \widetilde{\alpha_1} \in (0, BI - 1), T_{(i,j)} > 0 \quad (5.3)$$

Following $\widetilde{\alpha_1}$, let us consider another transmission, say $a2$, and define its formulation. It must be noted that $a1$ and $a2$ do not fall under the same time-line. If we take a time-line with $a1$ at $\widetilde{\alpha_1}$, $a2$ should definitely be placed at the end of the time-index of α_1. The value of X is a binary decision that be either (1,0) depending on the start value accepted for schedule. The ILP is governed by the binary decision of X that can be set to 0, so that $\widetilde{\alpha_1}$ starts before the $\widetilde{\alpha_2}$ and the value of BI will be set for a maximum value before the algorithm is processed. At this point, we determine the precedence of the transmissions to each other. The precedence constraints for the relative deadlines can be defined as follows:

$$\widetilde{\alpha}_1 + T_{(c,d)} - BI[X] \le \widetilde{\alpha}_2 \forall (0, BI), T_{(c,d)} > 0 \quad (5.4)$$

The above given notion is defined in [106] as a way to determine relative deadlines in a cyclic scheduling algorithm. $\widetilde{\alpha_2}$ on the other hand can be represented as follows:

$$\widetilde{\alpha}_2 + T_{(c,d)} - BI[1 - X] \le \widetilde{\alpha_1} \forall (0, BI - 1), T_{(c,d)} > 0 \quad (5.5)$$

Let us consider the time index for a transmission to be 1 for the sake a simplicity in the problem. It also should be noted that the transmissions within the dependency graph will have the same start time, for example $a1$ and $a8$ will have the same precedence constraints as both the transmissions fall under the same timeslot. This is in accordance with the constraint and the dependencies slated out at the beginning of this problem.

Step 3 - Schedule placement:

We solved this constraint Satisfaction Problem using the *linprog* functionality of Matlab. The *linprog* function, based on the set constraints, defines an output for the set of values defined as an input. Based on our defined constraints, we got the transmissions to be placed on slots

in such a way that any conflicts were avoided. For the alignment of the transmissions, we place them across several channels and various examples were taken to check its feasibility. Figure 5.5 shows the result for the example that we considered in our model. The constraint satisfying transmissions were placed in accordance to their specified start times.
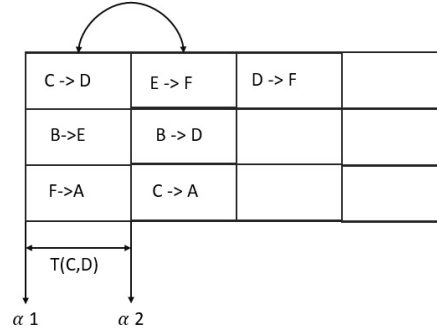
Figure 5.5: Scheduling results of ILP

Step 4 - Optimal check: On verifying the optimality of an ILP problem, it is debatable as optimality can be defined in several ways based on the users requirements. For instance, optimality can be determined as a term that satisfies higher reliability. In such a case, our solution may not be optimal and there will be a definite trade off between several quality guarantees. In our use case, as we consider minimal timeslots to be used, the major QoS satisfied is latency between transmissions. Hence the optimal check must be in such a way that a maximum number of transmissions is placed within minimal timeslots. The optimality of this condition is checked by the following calculation:

$$NT = \lceil (n/R) \rceil \tag{5.6}$$

In the above equation, NT is the number of timeslots occupied, n represents the total number of transmissions and R is the number of channels used. Using this equation, we obtain a value for the optimal number of timeslots that can be used specifically for the problem. However, this formulation has some anomalies which are later addressed under the heuristics.

Heuristics - Symphony

Having obtained the optimal schedule for the problem we started working on developing heuristics to verify the optimality of the solution. By re-iterating, our problem to find a schedule placement becomes a strategic one, like an *N queens problem* [107] [108]. "N queens" is a classic combinatorial problem in which positioning a "N" number of queens without clashing in a "$N \times N$" chessboard was the solution. Figure 5.6 illustrates a solution for a N queens problem, in which 8 queens are placed in a 64 slot chessboard without any possibility of clashing.

In our case, we have to follow a similar approach by which we have to place n number of transmissions in a "$R \times T$" GTS bitmap, where R represents rows (i.e channels ranging from 1

to 16) and T represents columns (i.e timeslots ranging from 1 to N). The number of timeslots may vary based on special functionalities of the DSME, such as CAP reduction.
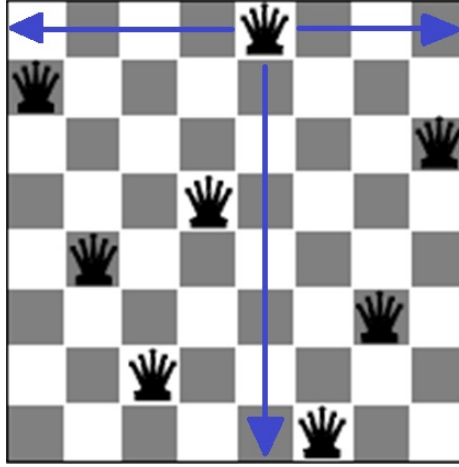
Figure 5.6: N queens problem solution

We looked into the state-of-the art hueristic solutions that could help in providing optimal solution which is similar or equal to that of our ILP formulation. Some of the heuristics we used to get the optimal solution is listed below:

Exhaustive search: Exhaustive search [109] is a brute force technique, which keeps on searching and iterating all the possible results till an optimal/sub-optimal result is obtained. In that technique, an allocation will be initialized and all possible solutions w.r.t the allocation will be checked exhaustively. Though this algorithm could bring us to an optimal solution, it very time consuming in terms of processing to build the overall schedule. With time consumption there will also be a huge processing power consumption involved in iterating every possible solution to receive the optimal schedule. This would be a significant impediment in supporting a dynamically changing WSN.

Just to give an example, we used exhaustive search technique on the same mesh network shown in Figure 5.2. Using a standard exhaustive search, it took 58 processing times of trial and error in order to reach an optimal solution where all the 7 transmissions were able to be placed within 3 timeslots.

Simulated Annealing: Simulated annealing [110] is a an allocation algorithm which was built based on the annealing which is a metallurgical process in which the material is heated and cooled in such a way the size of the crystals are altered to reduce the defect. Under simulated annealing the transmissions are chosen intuitively based on the neighboring transmission. At every iteration, the annealing process checks if the constraint is obtained and then moves on to the next level. Unlike the exhaustive search method, simulated annealing takes a subset of transmissions and checks its possibilities of optimal assignment. We provide the pseudo-code for simulated annealing under Algorithm 1.

We used simulated annealing over our example to obtain an optimal solution. We started the process at t_0 to place any transmissions (i, j) along the initial timeslot of the first channel

and its dependent values were then placed along the same timeslot till the subset became null. The process was continuously repeated till the optimal solution was reached.

Objective
get an optimized schedule
time initialization:
$t = t_0$
$X(t)$ = slot allocation for a transmission
$$Z(t) = \begin{cases} 0, & \text{for no scheduling at t} \\ 1, & \text{for scheduling at t} \end{cases}$$
Procedure
$X^*_{(i,j)} \leftarrow X_{(i,j)}$
$t \leftarrow 0$
Assign scheduling toward channel 1 timeslot 1
$t \leftarrow t + 1$
generate random (i,j) within the dependency tree
Assign scheduling toward channel 2 timeslot 1
$Z(t) = \left\{ X_{(i,j)} \epsilon [0,1] \right.$
iteration
$X_{(i,j)} \epsilon 0$

Algorithm 1: Simulated annealing for DSME scheduling

Though simulation annealing is not a brute force technique, it considers all the possibilities based on the dependency tree, and randomly selects the transmission for allocating towards a timeslot. There is also a chance we may not reach optimal or time-stringent solutions through this method. The randomness in the selection procedure also results in a higher processing time.

Maximum Dedicated Timeslot Algorithm Having checked some of the generic algorithms to get the optimal solution, we tried some algorithms that are designed specifically for the DSME MAC behavior. One such algorithms is the MDT scheduling algorithm [111], which aims at maximizing the number of dedicated timeslots to establish substitute paths. This algorithm was implemented for the tree topology, in which dedicated timeslots were used to accommodate the root paths of the tree.

The main aim of the algorithm is to place the transmissions within the superframe period in such a way that substitute paths were provided and were made available to re-accommodate in case of any failure. Very similar to the simulating annealing procedure, this algorithm also selects subsets and places them in the timeslots. The result of the algorithm is optimal if we focus only on the sheer reliability of the network.

The reliability attained through this schedule however compromises the quality aspects in terms of latency. Considering the transmissions in the dependency tree to be routes, we remapped it for a mesh network using this algorithm. For our algorithm, we obtained the schedule in such a away that 5 substitute paths were assigned for 7 transmissions. There is also a heavy wastage in bandwidth using this methodology.
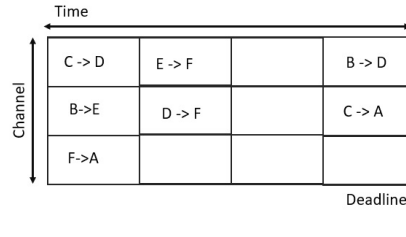
Figure 5.7: Scheduling results of MDT

5.2   The Symphony algorithm

The main objective of our algorithm was to improve scalability, increase the throughput and reduce the latency in latency-demanding DSME-based applications. To achieve this, the best possible way we design an ideal scheduling approach is through a Constraint Scheduled Programming. Our schedule is very similar to that of the simulated annealing by forming separate subsets of nodes based on its dependency tree. Then we formulate a set of rules by which we define the schedule. This makes Symphony more energy-efficient and faster in terms of computing time.

Problem Statement:
Placing the schedule in the available guaranteed timeslots in an optimal way, such that minimum number of timeslots are used and maximum bandwidth is utilized without any channel interference

The problem is bounded by the following two constraints, which will be a determining factor in establishing an optimal solution. Using these constraints, we aim at providing a schedule input for the DSME frame format. We summarize the two main constraints as follows:

Constraint 1: *No same nodes either involving in transmission or reception must fall under the same timeslot.* This constraint helps in avoiding all the interference in the network. We give a possibility for *different* nodes to communicate in a *same* timelot simultaneously in *different* channels, whereas, the same nodes can communicate in *different* timeslots within the *same* or *different* channels.

Constraint 2: *The maximum number of channels and the minimum number of timeslots should be used.* This constraint is more of a quality constraint that helps in establishing the optimality of the algorithm. This constraint helps in achieving the fact that the bandwidth will not be wasted and at the same time the minimal timeslots are used, so that the overall network throughput and scalability of the network is significantly increased, concomitantly achieving minimal latency.

Our algorithm is a two step process. First we get transmission ranks for the nodes based on the dependency graph. For example, in Figure 5.2, nodes B and C have a transmission rank of 2, as both the nodes have two links formed from them. We denote this Transmission Based Ranking as *TBR* in our algorithm. As an output of TBR, we group several sets of transmissions based on their respective ranks. There can be some transmissions which need to be scheduled

prior to others, in such cases, to include traffic differentiation, we also provide a priority indicator which can be issued as an Information Element (IE) through the Enhanced Beacon. The transmitting nodes that request for the priority indicator shall be provided a complete timeslot. Thereby, these transmitting nodes will have the liberty to choose among the channels with the best quality to transmit their data robustly.

Initialize
input: Updated schedule from RPL
step 1
Procedure: make TBR for all the nodes in the network
if TBR succesful **then**
  | return value : go to step 2 ;
else if *case of identical ranks* **then**
  | Place the elements in the adjacent subset;
else if *case priority indicator* **then**
  | assign the priority transmission in the full initial timeslot;
else
  | The transmissions are invalid
end
step 2
Procedure: place the subset with the highest rank adjacent to each other
Assign adjacent row slots till $subset1 \rightarrow null$
Assign subset 2 in the next row of the first column
if constraint *not satisfied* **then**
  | place the transmissions in the first row ;
else if constraint *satisfied* **then**
  | continue placing the transmissions till $all the subsets \rightarrow null$;
else
  | The transmissions are invalid
end

Algorithm 2: Symphony

In the example we provide in Figure 5.2, the rank of nodes C, B will be 2, and the ranks of nodes A, D, E, F will be 1. We start placing transmissions from C in adjacent timeslots followed by transmissions from B in the rows of the same timeslots. Then we follow backtracking to assign all the transmissions in accordance to the constraints.

Following the ranking procedure, the subset of the highest ranking will be placed in the row (i.e, different timeslots). Then the second subset will be placed in an adjacent timeslot with a different channel. This process will help in avoiding any channel conflict and optimal placement of the nodes. Compared with simulated annealing and exhaustive searching, the number of processing times are low in the Symphony algorithm. This is due to the ranking process and direct placement of a set of nodes in the timeslots using routing without any exhaustive constraint checking iterations.

Using this algorithm, we receive an optimal solution as shown in Figure 5.8

CAP	C→D	C→A	A→B				
	B→E	B→D	D→F				
	F→A	E→F					

CFP

Figure 5.8: Symphony schedule solution for the example scenario

The node carrying priority information can request for the initial timeslot, in which case the PAN Coordinator gives a priority flag in the IE carried in the payload of its Enhanced Beacon. In our network example, we gave the transmission from D with a high priority flag. This information must be conveyed to the network through the Enhanced Beacon to the owner of that specific superframe (the PAN Coordinator).

As a result of this priority indicator, we get a schedule, as shown in Figure 5.9. According to our algorithm, the transmission from D was given an entire timeslot. All the channels in the respective timeslot were used to occupy its transmission. Based on the channel adaptation technique of DSME, this transmission can sit in any channel over the initial timeslot. This gives the node the freedom to adapt its channel in case of any link quality deterioration, thereby improving robustness. Following this, the rest of all the nodes will follow the same procedure as mentioned above.

Though the priority timeslot feature, which we have introduced, may deteriorate the latency, it is a very valid idea to improve the robustness of the network. This is also an efficient traffic differentiation technique by which we can prioritize and preempt transmissions to achieve robust results.

However, the optimality of the algorithm is checked by Equation 5.6, this can be true only in certain cases. The anomalies are listed as follows:

anomaly 1: use of a priority indicator: When used a priority indicator, a priority-based transmission is given all the eligibility to occupy the full initial timeslot to any number of timeslots until it is allocated. Let us consider this time period as $t_{Priority-timeslot}$. In this case, the optimal solution is changed to the following formulation:

$$NT = \lceil (n/R) \rceil + t_{priority-timeslot} \tag{5.7}$$

anomaly 2: Rank $\geq NT$: This is a possibility that occurs when a higher number of nodes is associated only to a single node. In such a case, the usage of multiple channels will be useless. There is even a possibility that a single node will occupy the entire length of the timeslots, with different transmissions in them. In such a case, the optimal conditions cannot be met. In order to reach optimality, a larger superframe duration or a modified superframe with a large number of GTS must be provided.
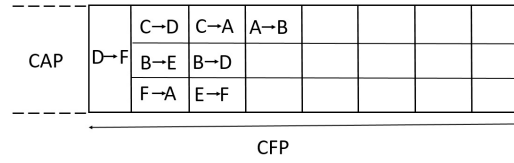
CAP	D→F	C→D	C→A	A→B				
		B→E	B→D					
		F→A	E→F					

CFP

Figure 5.9: example of priority based scheduling

5.3 RPL - Routing Protocol for Lossy networks

One of the main disadvantages of our algorithm is the reliability trade-off it has to endure against providing lower latency. We propose, however, a novel methodology to tackle this issue at the network layer. In the past decade, several technologies such as 6LoWPAN have combined protocols such as IEEE 802.15.4 and IPv6 to make it suitable for a variety of applications. The IEEE 802.15.4e itself is a haven of several MAC behaviors suitable for unique applications, but combining the technologies of IEEE 802.15.4e with routing protocols like RPL can make the standard even more effective.

Background on RPL

In order to ensure an effective routing in the 6LoWPAN network, the IETF ROLL working group proposed a routing protocol named RPL [112]. The Routing Protocol for Low Power and Lossy Networks (RPL) is a routing protocol that is designed for the Low power and Lossy Networks (LLN) supported by the 6LoWPAN protocol [113]. It integrates technologies such as the IEEE 802.15.4 and IPv6 protocols [114]. RPL supports both mesh as well as hierarchical topologies. RPL is specifically designed to support networks that are prone to high packet losses and limited resources in terms of computation and energy. RPL is a distance–vector (DV) and a source routing protocol that operates on the IEEE 802.15.4 PHY and MAC layers. It supports point-to-point and point-to-multipoint traffic. As DSME supports mesh networking, RPL would be a well-suited candidate for network integration.

RPL is based on hierarchical *Directed Acyclic Graphs* (DAGs). In contrast to a classical tree, in a DAG, a node can associate itself with many parent nodes. The destination nodes of a RPL are called a *sink*, and the nodes through which a route is provided to the Internet are called *gateways*. RPL organizes these nodes as *Destination-Oriented DAGs (DODAGs)*. Several DODAGs can be present in a network. Every node in a DODAG has a rank, which is the individual position of a node with respect to its neighbors in the system. A basic example of a RPL network is shown in Figure 5.10

The rank increases outwards from the DODAG root, as shown in Figure 5.10. In order to construct a network topology, every router in the system identifies and associates with a parent in a specific DODAG root. This is done based on an *objective function*. The *Objective function* helps in computing the rank of the node(s) and providing them an optimal routing path using metrics such as latency and power efficiency.
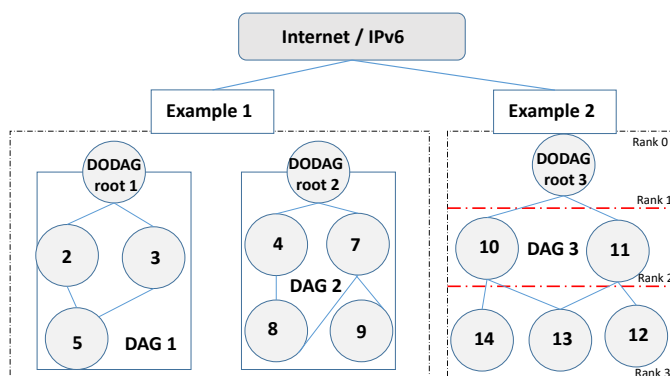
Figure 5.10: RPL - an Example

The Open WSN [88] research group already has designed a working model of TSCH in which RPL was implemented over the MAC layer. It was found to provide better results in terms of energy consumption and lower latency, when compared to the legacy TSCH. As RPL can be applied to low power lossy networks operating on mesh topologies, DSME will be an ideal candidate for its implementation. Our solution to this challenge is discussed next.

Integration of RPL with DSME

The fundamental research question which can arise now is "why RPL is needed and how it can be integrated alongside DSME." RPLs' objective function can be based on several QoS metrics like latency, power-efficiency, and even link-quality. RPL can provide links in accordance to high reliability and Symphony through its latency efficient scheduling can provide us a solution with a minimal trade-off.

As we already described, a DSME-enabled network consists of a central PAN Coordinator and several coordinators with routing capabilities; these nodes are called Full Function Devices (FFD). Reduced Function Devices (RFD) are the slave/end nodes with no routing capabilities. A RFD will always be connected to an associated coordinator within its range.

Symphony provides variable schedules to fit onto the multi-channel timeslots based on optimal routing decisions made by RPL. RPL supports either broadcast or unicast for disseminating the performance metrics using the DODAG Information Object (DIO), request DODAG information using the DODAG Information Solicitation (DIS) and disseminate the routing path using a Destination Advertisement Object (DAO). RPL will help in forming a complete mesh network based on specific quality metrics defined as an Objective Function.

The coordinators maintain schedules locally, and have their superframes to accommodate the nodes associated with them. They also maintain a routing table towards its routing children with a lesser rank in their respective DODAG. Every superframe comprises various kinds of traffic to support Symphony, such as the periodic beacons for synchronization, RPL signaling traffic, and application data traffic.

A concrete example of our architecture (Figure 5.11) is as follows:

- A dedicated beacon broadcast for synchronization between every superframe for every *X* slots, where *X* is the superframe duration of every individual superframe.

- A dedicated beacon broadcast for synchronization every multi-superframe for every *Y* slots, where *Y* is higher than *X* and is the multi-superframe duration coordinating every superframe with the duration of *X*.

- An Enhanced Beacon common for all coordinators in the network carrying the broadcast and unicast packets for RPL signaling (DIO, DIS, DAO), repeating every *Y* number of slots. In accordance with the standard, the Enhanced Beacon payload can be a variable, and it carries the RPL information. To form a DODAG, it can take a maximum of 128 bits for 1000 nodes.

- A dedicated unicast signal from the slave to the parent node, followed by *N* unicast signals from the coordinator to the slave nodes.
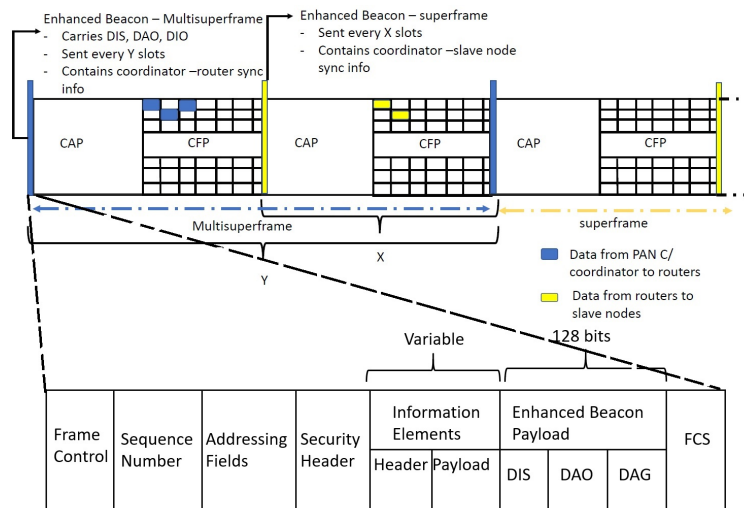
Figure 5.11: System architecture

Symphony maintains schedules for all the transmissions that occur in parallel without an overlap. It chooses the channels and the GTS timeslots for the schedule based on the nodes' unique MAC identifier, providing contention-free scheduling based on ranking and priority-based scheduling rules.

In contrast to the traditional IEEE 802.15.4 networks, which are predominantly organized in a star or a hierarchical topology, DSME supports a multi-hop network organized in a mesh structure, where the routes of the mesh are built using the RPL protocol.

Node Association

At the initiation of the network, the PAN Coordinator sends a broadcast of Enhanced Beacons to the nodes in its vicinity. The coordinators with routing capabilities can also start to send

Enhanced Beacons within their range to get associated with nodes, thus eventually leading to a formation of the network topology. The coordinators with routing capabilities and slave nodes at the vicinity of the range of the PAN-C can associate with the sink. The new node that needs to join the network requests the PAN-C in response to the PAN-Cs broadcast. This will be followed up with an acknowledgment. Following the node association, the router adds the newly joined node as a child.

At the network level, the association process can be dubbed as the RPL routing node joining process. The PAN Coordinator will act as a DODAG root and sends DODAG messages using its minimum transmission power. All the routers in the RPL overlay network keep sending their DIO messages to announce the DODAG.

A node will listen to the DIO message only if it joins the PAN by receiving a beacon frame from the PAN Coordinator. If the node does not receive a beacon frame, it will ignore all DIO messages. When a node wants to join the DODAG, it receives a DIO message from a neighbor router, it initially adds the DIO sender address to its parent list. Then it Computes its rank according to the Objective Function specified in the OCP (Objective Code Point) field, which is an identifier that specifies what Objective Function (OF) the DODAG uses. (The OF can be reliability determining element like LQI (Link Quality Indicator)) or RSSI (Received Signal Strength Indicator). Finally the DIO message is forwarded with the updated rank information. The client node chooses the most preferred parent among the list of its parents as the default node through which incoming traffic is forwarded.

We used the Contiki [115] to form a mesh network based on an objective function. The objective functions can be several variable QoS (Quality of Service) metrics such as power consumption and latency. In Figure 5.12, we show a mesh network with a single PAN Coordinator (green) and 7 coordinators (yellow) based on objective functions. The green circle shows the overall radio range, and the red circles represent the communicating nodes.
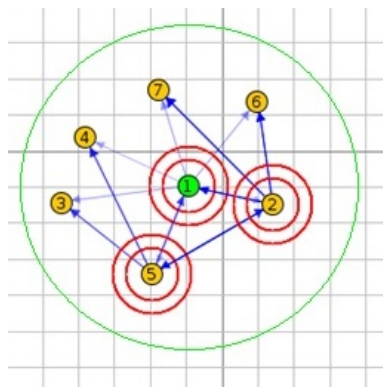

Figure 5.12: Mesh network using Contiki

Time-Slot Request/Response   For allocating a GTS an allocation-request is sent to the PAN Coordinator through the RPL network, either by single-hop or by multi-hop through intermediate routers (if the PAN-C is not in the vicinity). Once the request reaches the PAN Coordinator,

the Symphony algorithm helps to find the most efficient allocation in the time-frequency domain. This efficiency can be measured in terms of a specific Quality of Service like latency, reliability, or throughput. We present a sequence (Figure 5.13) diagram to illustrate this timeslot request process. In the figure, $t1$ represents the request, $t2$ represents the response, and $t3$ provides the change in the scheduling algorithm based on the evolution of the network as more changes in the topology is induced by the RPL. Therefore, we obtain a schedule that is QoS efficient without any considerable trade-off.
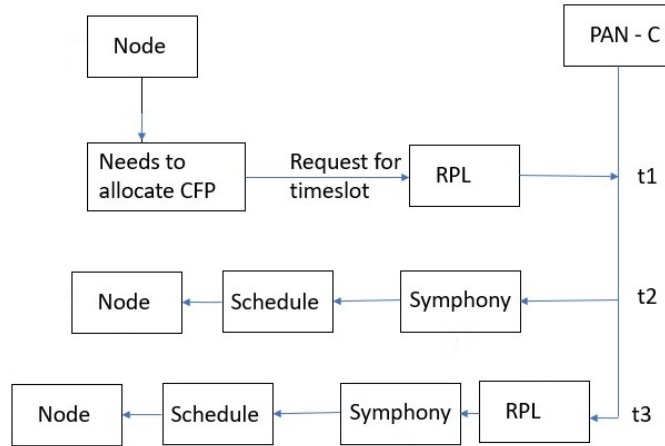
Figure 5.13: Timeslot request - response process

Once the allocation is done, the PAN Coordinator will send a response back to the requesting node using an Enhanced Beacon broadcast. The time synchronization that happens between the PAN Coordinator and all nodes through this beacon transmission can either be through multi-hop or single-hop.

Usecase of link- asymmetry

In the case of bi-directional communication, all the nodes will be able to reach the PAN Coordinator, and can also be reached by the PAN Coordinator. Scalability is naturally limited as all the nodes are constrained to be in each others communication range with the PAN Coordinator. Also, if all nodes use their maximum transmission power to extend their coverage, it will result in quickly draining their energy resources and thus dramatically shorten the network lifetime.

We looked into the concept of link asymmetry in an IEEE 802.15.4e/DSME network by adopting different transmission ranges in the PAN Coordinators and other nodes, which considers their role in the network, and their energy resources. We assume that the PAN Coordinator can transmit at the maximum transmission power, whereas other nodes can transmit at lower transmission powers, which will create a link asymmetry between the PAN Coordinator and the remaining sensor nodes in the network.

First, the link asymmetry resulting from different transmission power levels is nowadays feasible with current COTS technologies. The commercially available XBee IEEE 802.15.4-compliant modules [116] operate at different maximum transmission power levels allowing them to have

different communication ranges. Table 5.1 presents an overview of the different XBee modules and their characteristics. It is observed that the XBee module can allow a communication range for up to 1500 meters with a transmission power of 60 mW (18 dBm) and a communication range up to 100 meters with a transmission power of 1 mW (0dBm).

XBee Standards	Frequency	Data rate	protocol	Max transmit power	Channels
XBee ZigBee (S2C)	2.4 GHz	250kb/1Mb/s	ZigBee	6.3 mW (8dBm)	16
XBee PRO 900HP	900 MHz	10kb/200k/s	Proprietary	250 mW (24dBm)	FHSS
XBee 802.15.4	2.4 GHz	250kb/s	802.15.4	1 mW (0dBM)	16
XBee DigiMesh 2.4	2.4 GHz	250kb/s	Proprietary	1 mW (0dBm)	16
XBee 868LP	868 MHz	10/80kbit/s	Proprietary	25 mW (14dBm)	30
XBee PRO XSC	900 MHz	10/20kbit/s	Proprietary	250 mW (24dBm)	FHSS
XTend 900 MHz	900 MHz	10/125kbit/s	Proprietary	1000 mW (30dBm)	50
XBee Wi-Fi	2.4 GHz	72Mb/s	802.11b/g/n	16 mW	13

Table 5.1: XBee standards and characteristics

DSME also supports link asymmetry, a property very appropriate for varying energy resources. In IoT applications like intra-car communication [117], the PAN Coordinator can be general mains-powered as it can serve as the connection point between the low-power sensor networks that are spread across specific inaccessible locations of a car. Thus, the transmission at the highest power, and consequently a larger communication range, will not affect the energy of the PAN Coordinator. However, the remaining sensor nodes are typically battery powered and do not have the luxury to transmit at the highest transmission powers to avoid depleting their batteries. The PAN Coordinator and the coordinators can have varying ranges and can effectively utilize RPL to route and communicate. Techniques like link asymmetry can reduce energy consumption greatly.

5.4 Performance evaluation

The performance evaluation of Symphony is two-fold. First, we provide a detailed analysis of the advantages we get by implementing a RPL layer over DSME. We use simulations from Contiki [65] and OpenDSME [118] to showcase the efficiency of the architecture in terms of reliability and power consumption. In our simulations, RPL works based on the Objective Function (OF) of link quality and thus providing better reliability. Then we showcase the advantage of Symphony in terms of delay. For our delay analysis, we use probabilistic methods, and we compare our algorithm with the existing state-of-the-art algorithms for DSME.

5.4.1 Reliability and energy consumption

Every node in the network derives an ETX (Expected Transmission Count). This is a parameter that is helpful in estimating the frame loss ratio at the link. The ETX is dependent on the forward (P_f) and the backward frame losses (P_b) of the nodes in a network. Forward losses occur

because of any failure or link deterioration during transmission, and the backward losses occur due to problems during reception. The ETX value can be given by the following expression:

$$ETX = 1/(1 - P_f)(1 - P_b) \tag{5.8}$$

ETX determines the reliability of the links, and this parameter can be also represented as the inverse of successful packet delivery(P_S):

$$ETX = 1/(P_{Sf} + P_{Sb}) = 1/Reliability \tag{5.9}$$

In a RPL-enabled network, the nodes will alter the routes to the sink when there is a deterioration of the link quality and eventually improving the overall ETX. However, the delay using RPL can increase, if additional routes are deployed to reach the sink. In such cases, not only delay, but also other parameters like energy consumption can have a severe toll.

To calculate the energy consumption of the entire network, we must determine the average energy consumption of a device to transmit one slot amount of data. Then the values are generalized for the whole network. For this analysis, we consider the GTS slots and the energy that is wasted along the CAP region and idle frames. Let us consider the probability of successful transmission to be P_S. For the sake of simplicity, we consider this value to be 1, and we define an ideal network for our numerical analysis.

Let us consider P_{rx} and P_{tx} to be the energy consumed to receive and transmit a data packet. The energy consumption can be a turnaround process for a time of T_{ta}, as the sensor nodes move from a sleep to an ON state during the data transmission. The energy consumption during this turnaround process can be computed as follows:

$$E_{ta} = \left(\frac{p_{rx} + p_{tx}}{2}\right) \tag{5.10}$$

In case of a successful transmission, the energy consumption will be as follows:

$$E_s = \left(2T_{ta} \times \frac{p_{rx} + p_{tx}}{2}\right) + T_L P_{tx} + T_A P_{rx} \tag{5.11}$$

In case of an unsuccessful transmission, the energy consumed depends on the time until an acknowledgment is received. This phenomenon is completely avoided in a RPL-enabled network as it can immediately switch on to alternate routes based on the link quality. Let us assume the time for retransmission as α_n, and then the energy dissipated by a failed transmission is given as follows:

$$E_c = \left(2T_{ta} \times \frac{p_{rx} + p_{tx}}{2}\right) + T_L P_{tx} + \alpha_n P_{rx} \tag{5.12}$$

The overall energy consumption is then be given by:

$$E_{total} = \left(\frac{P_{rx} + (1 - P_s)E_c + P_s E_s}{P_s(T_{data}) + (T_{CAP})}\right) \tag{5.13}$$

T_{data} includes the time duration of the data consumption; it includes successfully transmitted timeslots and the retransmission timeslots in case of any collision. T_{CAP} will include the time duration of the data that is transmitted under CSMA/CA. In case considering just GTS transmissions, this parameter can be neglected. The idle GTS will not affect power consumption; in fact, saving energy can be proposed by introducing some inactive periods in the network, but they will affect the overall latency of the network.

An experiment was conducted over 30 nodes that transmit data in a multi-hop mode, and the energy consumption was calculated with and without RPL. In a lossy network, the average ETX starts to degrade with time, eventually leading to a failed link and contributing to E_c. We used the Cooja simulator of Contiki [65] to perform the reliability analysis. From the results shown in the Figure 5.14, it can be noted that the link degrades by time, and the power consumption increases as compared to that of a RPL-based network. RPL also shows a linear increase in power consumption due to energy used up in altering the routes.
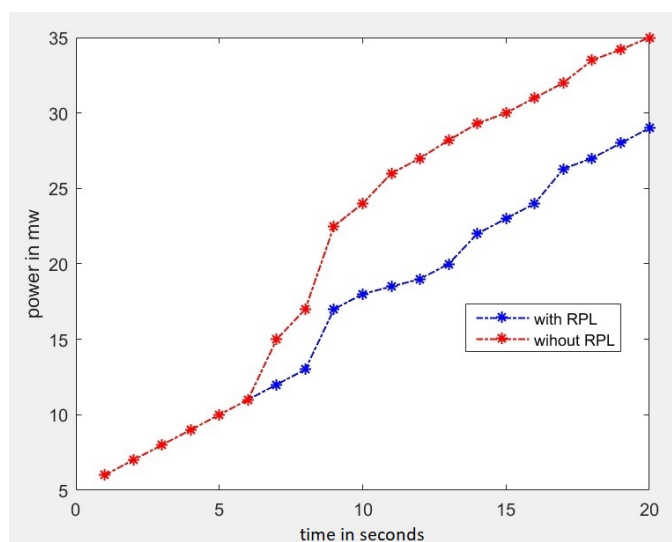
Figure 5.14: Energy consumption with and without RPL

We also conducted experiments on the OpenDSME platform [118], which is a C++ based simulation environment that provides the adaptation layer of DSME and CSMA structures of IEEE 802.15.4e. OpenDSME also provides the possibility of the implementation of a viable network layer on top of it. The DSME sublayer of OpenDSME employs a typical slot-based reservation system for a schedule that is provided by the application layer.

In our experiment, we calculated the reliability over a network of 25 nodes with a static concentric mobility type. The reliability of the network was calculated based on the number of successful packet delivery as shown in Equation 5.9. In the radio medium, we introduce a constant interference range to emulate a real-time wireless network. 16 channels are utilized in the DSME network in accordance with the standard parameters. Without having routes established for the network layer, it was noted that the reliability of the network depletes steadily with the increase in the number of nodes. We repeated the same experiment with the same

network configuration but with generic routing employed at the network layer. We were able to observe almost 40% increased reliability with generic routing. It can also be noticed that with a additional CAP reduction employed when the number of nodes increases, there is a chance of achieving higher reliability. This is because of the ready availability of guaranteed bandwidth.
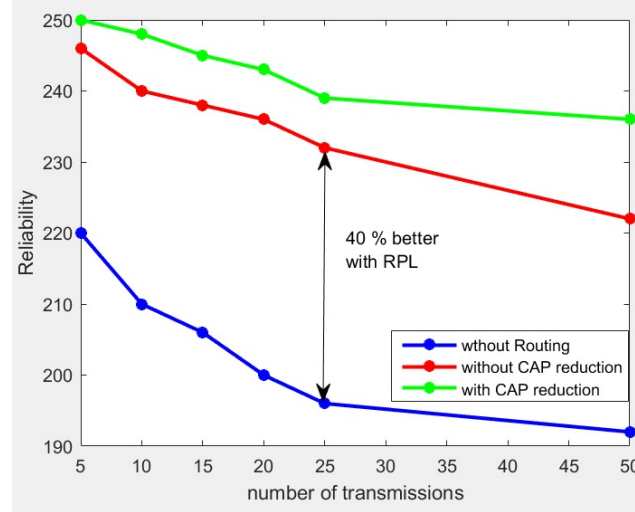
Figure 5.15: Reliability with and without RPL

5.4.2 Delay analysis

The average transmission delay for successfully transmitted GTS frames in the multi-superframe can be calculated as follows:

$$\delta = \sum_{i=0}^{\infty} P_{(i,m)}^f (i(MI)) \tag{5.14}$$

Considering that the schedule for routing is carried out for every multi-superframe, $P_{(i,m)}^f$ is the probability that the GTS is transmitted in the ith superframe of the multi-superframe m. MI is the summation of all the individual BIs (Beacon Interval) within the multi-superframe. To calculate this probability, let us take two parameters: X^s, the total number of GTS that is successfully transmitted; and $X_{(i,m)}^S$, the number of GTS that has to wait for i superframes within a multi-superframe for its successful transmission. The probability will be 1 if considered all the superframes in case of delayed transmission. Using these parameters the probability $P_{(i,m)}^f$ can be formulated as follows:

$$P_{(i,m)}^f = X_{(i,m)}^S / X^S \tag{5.15}$$

The first set of GTS frames based on the Symphony schedule that gets successfully placed in the initial attempt need not wait the next; let us consider this as $X_{(o,m)}^S$. These include all the

transmissions in all the available channels (m) of the initial superframe.

$$X_{(o,m)}^{S} = K(1 - P_e),$$
$$where \; m = |0 - 16| \; and \; K = (0, 1)$$

(5.16)

The value of $X_{(i)}$ will be incrementing as with the failures to accommodate a successful transmission. Now the GTS superframes that wait till the first adjacent superframe to get transmitted successfully can be denoted as $X_{(1,m)}^{S}$, this value can be formulated as:

$$X_{(1,m)}^{S} = HK(1 - P_e)$$

(5.17)

Where H is the probability of failure to get accommodated within the initial transmission. The value of H can be given as $P_e e^{-BI \cdot m \cdot i\lambda}$, this probability is with an assumption that all the transmissions shall be carried out within the multi-superframe with i superframes and m channels with a GTS arrival rate of λ. Generalizing for all the i superframes, the successful transmissions can be denoted as:

$$X_{(i,m)}^{S} = H^{(i)} K(1 - P_e)$$

(5.18)

The value of the successfully transmitted GTS can be formulated as:

$$X^{S} = \sum_{i=0}^{\infty} H^{(i)} K(1 - P_e)$$

(5.19)

By using the aforementioned equations, the probability to be transmitted in the ith superframe can be calculated as:

$$P_{(i,m)}^{f} = (1 - H) \cdot H^i$$

(5.20)

and the overall average delay of the network can be given as:

$$\delta = \sum_{i=0}^{\infty} (1 - H) \cdot H^i (\epsilon + i(MI))$$

(5.21)

Let us consider an example scenario where a multi-superframe with 2 superframes over 3 channels. We take three arrival rates for the numerical analysis. From the Fig 5.16, it can be understood that lower the values of λ the delay is significantly decreased, this is due to the decrease of the delay in the inter-arrival rates, this will also increase the overall throughput of the network. The possibility of multi-channel in DSME also contributes to smaller delay.

We use the same formulation to calculate the delay of the schedule placement within a superframe. For this very special case, we take the value of H and replace it with H_{tslot}, which
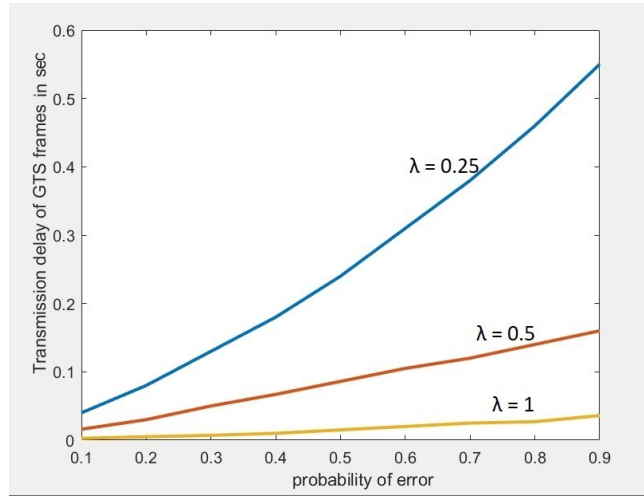
Figure 5.16: Probability error vs GTS delay

is the probability of failure to accommodate within the initial timeslot. This aforementioned value can be expressed as follows:

$$H_{tslot} = P_e e^{-T_s \cdot m \cdot i \lambda} \tag{5.22}$$

For our analysis, let us consider that all the timeslots have an equal size for all the i superframes in the multi-superframe. We now derive a formulation for the delay for a single GTS that fails to occupy the first timeslot and moves to the next. Utilizing the aforementioned details in Equation 5.22, we can derive the delay for a timeslot to be as follows:

$$\delta_{timeslot} = P_e e^{-T_s \cdot m \cdot i \lambda} (T_l) / (1 - P_e e^{-T_s \cdot m \cdot i \lambda}) \tag{5.23}$$

For this analysis, we compared Symphony with MDT and brute-force FIFO algorithms. Brute-force FIFO is based on the PID scheduling used in [119]. This method is also used for the GTS allocation in the OpenDSME framework [118] for the DSME implementation.

As the nodes start to get accommodating in the channels and move from one timeslot to another, the value of T_l (length of the timeslots) starts to increase with one. The analysis shown in Figure 5.18 provides the Transmission delay of the GTS frames for a set of transmissions for different arrival rates. With the change in the topology of the network, RPL updates a new set of transmissions to be scheduled in the following multi-superframe. The new nodes have to wait for the length of the current multi-superframe to be accommodated next.

Symphony provides better results when compared to the MDT and the random FIFO methods. MDT under-performs because it spares timeslots aiming for better reliability of the network. When the number of transmissions increases, more timeslots are wasted, thus contributing to delay. Symphony, on the other hand, aims at filling all the timeslots stringently based on channels available, thus eventually leading to lesser transmission delays and also increased robustness.
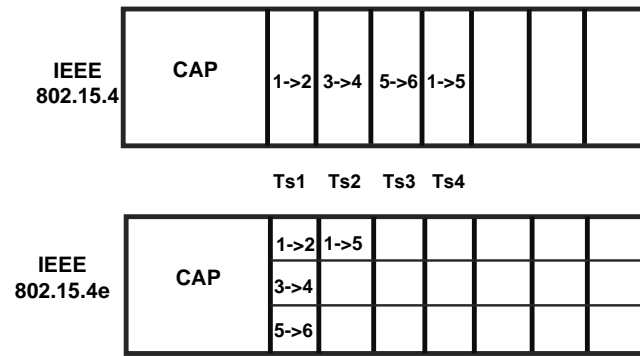
Figure 5.17: Scenario taken for Symphony comparison against IEEE 802.15.4

We also carried out simulations for Symphony using the OpenDSME platform. We conducted experiments for the delay over several GTS transmissions. Our simulations complement our numerical analysis and show a linear increase in delay with the increase in the number of transmissions. In our simulations, we pitted Symphony against MDT, standard DSME, and CSMA/CA. The experiment was carried out for a multi-superframe time interval that repeats periodically with varying schedules. To be in-line with real scenarios, we introduced some interference in all of our simulations.
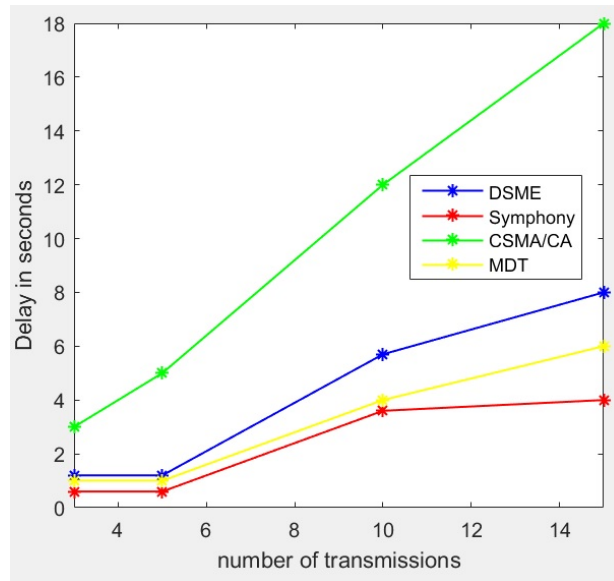
Figure 5.18: number of transmissions vs GTS delay

The simulation was carried over 3 channels and 7 GTSs as shown in Figure 5.17. From the results shown in Figure 5.18 the CSMA/CA, due to its contention-based nature, results in a more significant delay when compared to the CFP based scenarios. Traditional DSME works based on best effort, the transmission in the GTS is carried out when a timeslot is available for transmission. Under MDT, the schedule is fixed, but as the number of transmissions increases, additional free slots are assigned to reduce link failure, which in turn increases the overall-delay.

Under Symphony, we set a schedule that is tailored to reduce the delay by efficient packing of the timeslots, thus obtaining the most minimal delay in all these tested cases.

From our performance analysis and simulations, we learn that RPL helps in improving the reliability of the network. Coupled with an efficient algorithm like Symphony, we are able to achieve lesser latencies without any dire trade-offs.

5.5   Final remarks

In this Chapter, we introduce a novel solution for improving the Quality of Service in a periodically evolving real-time DSME network. We present aan approach of integrating RPL over DSME and an efficient scheduling algorithm called Symphony which supports it. We design a methodology to provide optimal schedules for GTS allocation based upon the RPL routing information. Symphony helps in scheduling transmissions in a stringent way by avoiding interference and helping to obtain lesser network latencies.

In this work, we focused on key QoS parameters, such as energy efficiency and latency which are some major demands in a industrial application. From extensive simulation and numerical analysis, we find that RPL provides a robust support to the mesh networking of the DSME protocol. Including RPL helps the network improve the energy efficient by 10-15 %. With a generic routing mechanism over dynamically growing DSME network, the reliability of the network increases by 40 % compared against a network without routing.

We compared Symphony to some of the state-of-the-art algorithms implemented for DSME using probabilistic analysis. We were able to learn that Symphony, with its stringent packing strategy, performs better in terms of latency. Using Symphony, we were able to witness a decrease in latency by 3-10 %. There is also a possibility to extend the Symphony algorithm can also be extended onto the other MAC behaviors like TSCH that support multi-channel mesh networks.

Chapter 6

DynaMO - Dynamically Tuning DSME Networks

6.1 Context and motivation

Recent advancements in the IoT domain have been pushing for stronger demands of Quality-of-Service (QoS) and in particular for improved determinism in time-critical wireless communications with power constraints. The new MAC behaviors of the IEEE 802.15.4e provide support for time-critical and reliable communications. Specifically the Deterministic Synchronous Multi-channel Extension (DSME) combines features such as contention-based and contention-free communication, guaranteeing bounded delays and improved reliability and scalability by leveraging functionalities such as multi-channel access and CAP reduction.

However, the DSME has a multi-superframe structure that is statically defined at the beginning of the network. Therefore, when the network evolves dynamically in accordance with its traffic characteristics, these static settings can affect the overall throughput and increase the network delay because of improper allocation of bandwidth. In this chapter, we address this problem, and we present a novel dynamic multi-superframe tuning technique that dynamically adapts the multi-superframe structure based on the size of the network.

In DSME, many superframes can be stacked within a multi-superframe period, which is defined by the *Multi-superframe Order* (MO), and as observed in the previous chapters, this parameter has a significant impact on determining the QoS of these networks. Traditionally, DSME networks require careful planning of the several MAC parameters, such as MO, SO, BI and CAP reduction by an experienced network engineer to achieve adequate QoS levels. However, if this is already an impediment for easy and straight-forward network deployment, in highly dynamic and unpredictable environments finding the right balance is borderline impossible. In scenarios where the traffic or the number of nodes can change, which is increasingly becoming a commonplace, static settings inevitably lead to some compromise in terms of delay or throughput that can only be addressed by devising mechanisms that can adapt during run-time to the new conditions.

DSME has the ability to satisfy the QoS requirements of several applications using several presets defined in the standard. However, for a dynamically evolving network, these static configurations can dramatically affect the overall Quality of Service, and the network will also suffer from dire trade-offs. There have been several research works like [67] and [120] in which the performance of DSME was analyzed. However, in the literature, features like the CAP reduction and superframe structure were always kept static.

The following section further details the problem we want to mitigate; we introduce a new tuning mechanism in this chapter by which the overall QoS provided by the network can increase significantly.

6.2   System architecture

For our system architecture, we consider a DSME enabled 802.15.4e mesh network with nodes dynamically joining and leaving the network, as illustrated in Figure 6.1. The network consists of a central PAN Coordinator (node a_1 in Figure 6.1), which can receive and transmit beacons and messages. Then we have the coordinators, which can transmit routing information and also send Enhanced Beacons for association and timing synchronization. Unlike the coordinators, the Reduced Functional Devices (RFD) in the network have the capability of only receiving information.
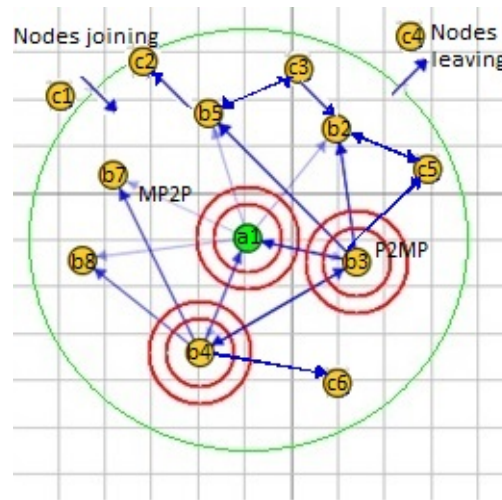
Figure 6.1: System model

We consider a dynamic architecture with nodes entering and leaving the network, as most of the emerging IoT applications bank on the dynamic nature of Wireless Sensor Networks. Our network is formulated with the help of RPL by which a point to many points (P2MP) or the vice-versa topology can be achieved. Our algorithm will help to provide a multi-superframe structure for the PAN Coordinator of this network, which will improve the scalability and also other QoS properties of the network.

Problem statement

In a dynamic mesh network, when new Fully Functional Devices (FFD) are added or removed, a statically defined MO and CAP reduction primitive at the inception of the network can cause adverse results.

Problem 1: There can be a *"need formore guaranteed bandwidth than what is available"*. This could be avoided if a bigger MO was defined at the beginning of the multi-superframe in accordance with the required resources.

Problem 2: There can be *"excess of guaranteed timeslots"*, affecting the throughput of the overall network due to a wastage of bandwidth also leading to increased latency.

This imbalance constitutes the fundamental cause of the decreased performance against TSCH [67]. TSCH overcomes the limitations as it does not have a fixed superframe like that of a DSME network. However, these issues can be avoided by a multi-superframe tuning technique that can: (i) employ/deploy CAP reduction based on the number of transmissions that need to be accommodated in the superframe; and (ii) provide a new Multi-superframe Order (MO) better suited to the number of pairwise transmissions scheduled for the GTS service. Such information can be made available by implementing a routing layer over DSME that helps to provide the scheduling information to the link layers similar to the Orchestra schedules [121] used in 6TiScH [122].

Node association and routing

The coordinators (FFDs) advertise their superframe periodically by sending Enhanced Beacons. New nodes can join the network by associating either to a coordinator or directly to the PAN Coordinator, via an Association request, eventually leading to a topology formation.

At the network level, the association process follows the RPL routing node joining process [123]. The PAN Coordinator will act as a DODAG (Destination Oriented Direct Acyclic Graph) root and will send DODAG messages. All the routers in the RPL overlay network keep sending their DIO (DAG Information Object) messages to announce the DODAG. A node will listen to DIO messages only if it joined the WPAN via association process. When a node wants to join the DODAG, it receives a DIO message from a neighbor router, and then it:
(*i.*) adds the DIO sender address to its parent list,
(*ii.*) computes its rank according to the Objective Function (OF) specified in the OCP (Objective Code Point) field, which is an identifier that specifies the Objective Function that the DODAG uses. The OF can be LQI (Link Quality Indicator), Packet Delivery Ratio (PDR) or even Power Consumption [124],
(*iii.*) forwards the DIO message with the updated rank information. The client node chooses the most preferred parent among the list of its parents as the default node through which inbound traffic is forwarded.

Notations	Description
N	total number of nodes
a_i	node a_n where $i\epsilon(1, N)$
$N_{Channels}$	number of channels = 16
T_i	index of the timeslot in the multi-superframe
N_{CFP}	total number of GTSs in the CFP of a multi-superframe
$NCAP$	number of GTS added when CAP reduction is activated

Table 6.1: Notations for DynaMO

DynaMO algorithm

In this section, we introduce an efficient multi-superframe tuning algorithm called DynaMO. The general idea of this algorithm is *adaptively increasing and decreasing the multi-superframe structure based on the evolution of GTS allocation requirements over time.*

Algorithm 6.1 presents the DynaMO adaptive network algorithm, and Table 6.1 shows the notations used for the description of the algorithm.

As the network grows/diminishes dynamically, the routing layer will update the topology and forward the respective, which contain the list of pair-wise GTSs transmissions. This is provided as an **input** (Algorithm Line 1). Let us consider pairs of neighbor nodes (a_i, a_j) to transmit between them. This transmission list will be provided as a bitmap to the link layers using the RPL backbone for every beacon interval. An example of a transmission bitmap for the network shown in Figure 6.2 is presented in Figure 6.3. Zero means that there is no transmission in a GTS between the two nodes and one means there is a transmission in a GTS between the two nodes.

The PAN Coordinator has access to all information needed to establish a multi-channel GTS allocation, including the number of channels ($N_{Channels}$), the number of the GTS (N_{TS}) and the total available GTS resources ($N_{CFP} = N_{Channels} * N_{TS}$). The number of timeslots can sometimes vary if the CAP reduction primitive is activated. In such a case, the number of timeslots will be $7 + N_{CAP}$, where N_{CAP} is the number of timeslots added via CAP reduction. The PAN-C initially randomly determines the values of BO, MO, and SO and the CAP reduction primitive.

In our algorithm, we first determine the number of resources that need to be allocated in the network. This number of required resources is obtained through a near-optimal scheduling algorithm like simulated annealing [110] or Symphony [125].

An optimal schedule must ensure minimal latency in the data transmission. The nodes must also be placed in such a way that there are no overlapping transmissions amongst them. By using Symphony, a (near) optimal solution for the mesh network in Figure 6.2 is given as:

$$\begin{bmatrix} c \rightarrow d & c \rightarrow a & a \rightarrow b \\ b \rightarrow e & b \rightarrow d & d \rightarrow f \\ f \rightarrow a & e \rightarrow f & -- \end{bmatrix}$$

Algorithm 6.1 DynaMO

1: **Input** BO, SO, MO, CAP reduction Primitive
2:     Pairwise transmissions from RPL: $((a_1,a_2),(a_1,a_3).....(a_2,a_1).....(a_i,a_N))$
3: $N_{Channels}$ and $N_{TS} \in (1,7+(NCAP))$
4:
5: Initialization
6: **repeat**
7: **Schedule** R= Required number of resources to accommodate the network
8: **Resource test:** check $N_{CFP} = R$ in a multi superframe
9:
10: **Case 1: less resources**
11:     **while** $N_{CFP} < R$ **do**
12: CAP Reduction = ON;
13:       **if** resource test = true **then**
14: **Print:** DynaMO is successful.
15:       **else** MO = MO + 1;
16:       **end if**
17:     **end while**
18:
19: **Case 2: abundant resources**
20:     **while** $N_{CFP} > R$ **do**
21: CAP Reduction = OFF;
22:       **if** Resource test = true **then**
23: **Print:** DynaMO is successful.
24:       **else** MO = MO - 1;
25:       **end if**
26:     **end while**
      Loop Repeat: Every multi superframe duration
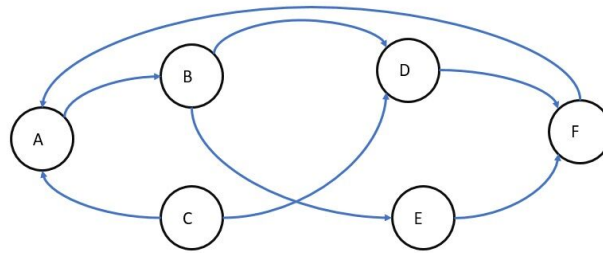
Figure 6.2: Example of a mesh network

An optimal schedule gives us an idea of the total number of resources that need to be accommodated in the network. For the aforementioned example, we need 9 GTSs spanning across 3 channels and 3 timeslots to accommodate the network. Let us call this amount of GTSs as R. Now we follow this by a resource check function (Resource test, Line 8, Algorithm 6.1). This function checks if the number of resources in the network (N_{CFP}) are enough to

	a	b	c	d	e	f
a	0	1	0	0	0	0
b	0	0	0	1	1	0
c	1	0	0	1	0	0
d	0	0	0	0	0	1
e	0	0	0	0	0	1
f	1	0	0	0	0	0

Figure 6.3: Transmissions bitmap for the network example of Figure 6.2

accommodate the total number of GTS transmissions (R).

Based on this check, the PAN-C determines the value of the multi-superframe Order (MO) and the CAP Reduction primitive. When the resource requirement is not satisfied (case 1 - algorithm line 10), DynaMO is initiated. When more resources are needed, the PAN-C initializes the CAP reduction and checks whether the schedule can be placed within the multi-superframe. Even after switching ON the CAP reduction primitive, if the number of available resources is not enough, the PAN-C will increase the MO, eventually adding another superframe. This process continues until the schedule is placed adequately.

On the other hand (i.e., Case 2 - algorithm line 19), when abundant resources are available, the PAN-C decreases the MO dynamically and can also switch off CAP reduction. The change to the MO and CAP reduction primitive is sent through the DSME PAN descriptor IE (Figure 6.4) in the Enhanced Beacon at the start of every new multi-superframe.
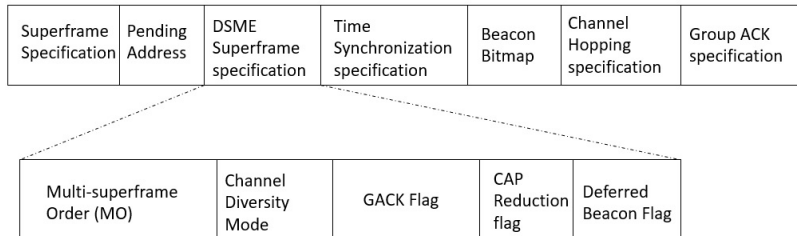
Figure 6.4: DSME PAN Descriptor Structure

In the next section, we provide an analytical model for the GTS allocation and then in Section 6.4, we carry out throughput and delay analysis for a DynaMO-enabled DSME network.

6.3 Mathematical modeling

For our analysis, we consider a DSME mesh network with N nodes under the network coverage of the PAN Coordinator. Being a time critical network, we assign guaranteed transmissions for all the nodes that are associated in the network. The RPL routing protocol [126] forms a network topology based on an Objective Function that defines a routing strategy [127, 123]. This schedule for an updated topology will be sent at every multi-superframe duration. Every

device is allocated one or more GTSs based on the topology issued by the RPL. For our analysis, as we only consider deterministic bounds, we assume that all the transmissions are successful.

The IEEE 802.15.4e does not provide any bound on the number GTSs that a device can allocate, in contrast to the legacy IEEE 802.15.4, which only allows a maximum of seven timeslots. This gives us the freedom to consider that a device can occupy as many GTSs as needed. Let us consider the maximum number of GTS in a superframe to be N_{CFP1} and when considering 2 superframes with CAP reduction the maximum number of GTS in a superframe will be N_{CFP3}, as it encompasses 3 CFPs in a 2 superframe period. Hence, in general, the maximum number of GTS available in a multi-superframe can be given by $N_{CFP(n)}$, where n is the number of CFPs encompassed in the specific multi-superframe.

In what follows we present the constraints on the number of timeslots that can be allocated based on the specifics of the DSME GTSs allocation mechanism of the IEEE 802.15.4e.

Number of GTSs with CAP reduction

In this subsection, we derive the value of $N_{CFP(n)}$, which is dependent on the values of the MO, BO and SO. D_{Max} represents the maximum delay a transmission has to undergo a successful GTS allocation in a multi-superframe.

In accordance to the standard, there will be an Inter Frame Spacing (IFS) period between every successful transmission. Depending on their size if smaller than $aMaxSIFSFrameSize$, it is called Short Inter Frame Spacing (SIFS), else it is called Long Inter Frame Spacing (LIFS). Under LIFS, the size extends for a minimum period of $minLIFSPeriod$ symbols. This IFS contributes to the delay along with other parameters such as L_{frame} (the frame length), R_s (the symbol rate) and R_b (the bit rate). In accordance to some previous research work [128], the maximum delay between superframe intervals can be given as:

$$D_{max} = \begin{cases} D_{SIFS} = \dfrac{(L_{frame} \times R_s)}{R_b} + minSIFSPeriod, \\ D_{LIFS} = \dfrac{(L_{frame} \times R_s)}{R_b} + minLIFSPeriod \end{cases} \tag{6.1}$$

The duration of the multi-superframe slot will depend on the Multi-superframe Order (MO) issued by the PAN Coordinator. Let T_{MS} be the duration of the multi-superframe slot, N_{MD} be the total number of symbols forming the multi-superframe, and N_{MD_i} be the total number of symbols constituting the multi-superframe since the value of $SO = 0$,

$$T_{MS} = \frac{N_{MD}}{T_{CAP} + T_{CFP}} = N_{MD_i} \times 2^{MO-4} \tag{6.2}$$

Equation 6.2 stands true for a scenario with CAP reduction for a single multi-superframe period encompassing all the GTSs in the CFP time period. It also considers a CAP region of duration T_{CAP}. A single GTS can span across several superframe slots, and so we should provide

a constraint on it. GTS must be greater than the total forward delay D_{max}. Let us consider N_{min} to be the minimum number of superframe slots a single GTS can extend over:

$$N_{min} = \left\lceil \frac{D_{max}}{T_{MS}} \right\rceil \tag{6.3}$$

As we consider a critical data-oriented network, we neglect the delay that occurs in the CAP region of the traditional IEEE 802.15.4. Under CAP reduction, the absolute number of GTSs is not specified, however, it can be expressed as $m \times N_{CFP}$, where m is the number of channels and N_{CFP} is the number of timeslots in CFP. From these, the maximum number of GTSs that can be allocated to devices can be given by:

$$N_{CFP(n)} = min\left(\left\lceil \frac{(T_{CAP} + T_{CFP})(1 - \frac{T_{CAP}}{T_{MS}})}{N_{min}} \right\rceil, m \times N_{CFP} \right) \tag{6.4}$$

Throughput analysis under CAP reduction

Though in DSME there is no inactive slots, the time-frames spent for IFS and acknowledgment contribute to the delay as the overhead timing. Let us call this time as T_{idle} and it can be given by:

$$T_{idle} = T_{overhead} + T_{wasted} \tag{6.5}$$

T_{wasted} will include the time-frames that were lost due to failures and delay of transmission due to queuing in the CFP. T_{idle} has to be calculated separately for every superframe in a multi-superframe, because in a dynamic CAP reduction scenario, we can have multi-superframes with and without CAP reduction primitives utilized. In the throughput equation 6.6, the available resources of GTS $N_{CFP(n)}$ should be considered as they contribute to the bandwidth. The throughput can be defined as the ratio of the data transmitted (T_{data}) to the total amount of bandwidth available for transmission. The maximum throughput for a multi-superframe repeating every MD with n superframes and a data rate of C can be given by:

$$TH_{max} = \left(\frac{T_{data}}{MD \times N_{CFP(n)}} \right) \times m * C, where T_{data} = n(T_{MS}) - T_{idle} \tag{6.6}$$

,

6.4   Numerical analysis

For the numerical analysis, we take the timeline of events as shown in Figure 6.5. The entire timeline is divided from T1 to T7, which comprises 7 multi-superframes that in turn are composed of 12 individual superframes. It should be noted that for this numerical analysis we consider all the transmissions in the guaranteed timeslots to be successfully accommodated.

The size of every timeslot taken for this analysis is 1ms. For the sake of simplicity, we have provided 2 timeslots over 2 channels in every CFP, overall providing 4 GTSs in an individual superframe. An Enhanced Beacon (EB) will be sent at the start of every multi-superframe. This EB will contain the primitive to activate the CAP reduction and the schedule with channel and slot information.
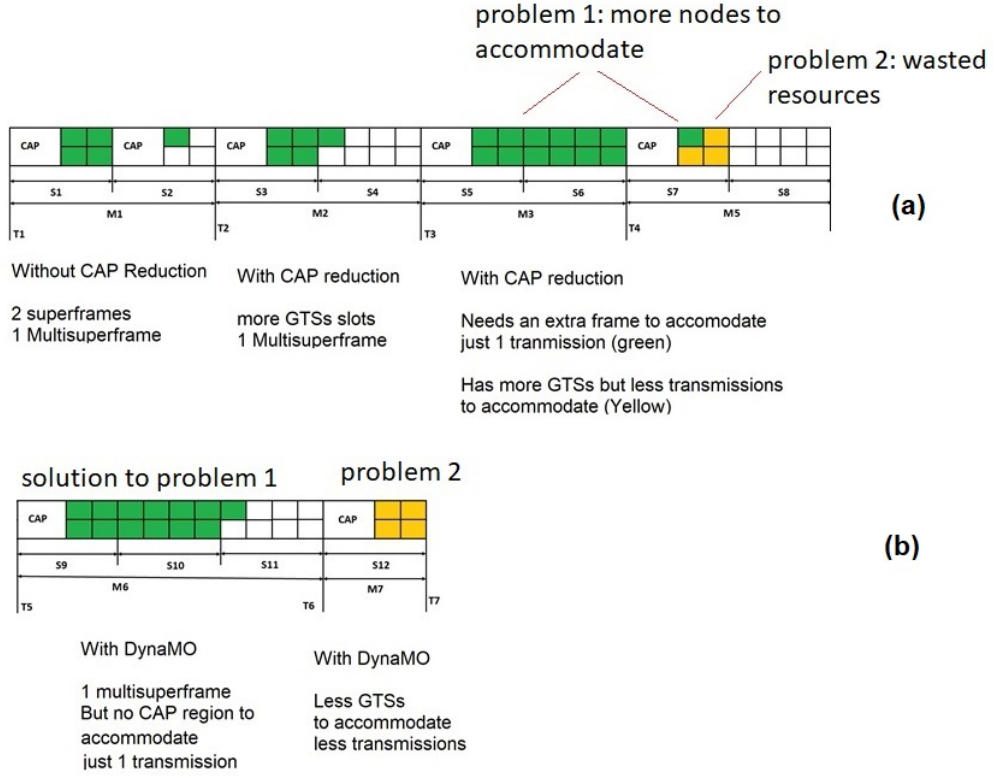
Figure 6.5: Scenarios taken for numerical analysis

Further details in the scenarios used for the numerical analysis:

(i) From T1 to T2: This is a multi-superframe in which normal DSME without CAP reduction is employed. The multi-superframe in this scenario is expected to support 5 GTS transmissions. It should be noted that without CAP reduction, the superframe has to wait for a "duration of CAP" before it is able to transmit.

(ii) From T2 to T3: This is a multi-superframe with CAP reduction employed in it. Unlike the previously discussed case, the final transmission need not wait for a CAP.

(iii) From T3 to T4: This is a multi-superframe with CAP reduction employed, and the number of transmissions it has to accommodate is 13. But the MO in this scenario is static; the final transmission also has to wait for an entire CAP period before its transmission.

(iv) From T4 to T5: This is a multi-superframe with CAP reduction employed with a static MO, but it should be noted that it just needs to accommodate 3 GTSs. As a result of this, 8 GTS remain unoccupied contributing to the wasted bandwidth eventually affecting the overall throughput of the network.

(v) From T5 to T6: This holds the same condition as in scenario *iii* , but with DynaMO, PAN-C counts the number of transmissions to be accommodated by the CFP. As the value is above the number of timeslots available, it increases the MO by 1 adding a superframe to the multi-superframe. In this case, the MO is 2, thus joining 3 superframes within a multi-superframe, eventually reducing the overall delay.

(vi) From T6 to T7: In this case the number of GTSs to be accommodated is 4. The PAN-C deploys CAP reduction in this scenario eventually providing a single superframe to accommodate the 4 transmissions. This method will reduce the wastage of bandwidth thus increasing the throughput.

We calculated the delay of the network for all the cases. We considered a network that dynamically grows and thus demanding more GTSs resources. For CAP reduction scenarios, we take the value of MO to be 1. For this numerical analysis, we consider the idle time to be 0 and a constant data rate of 1 Kbps.
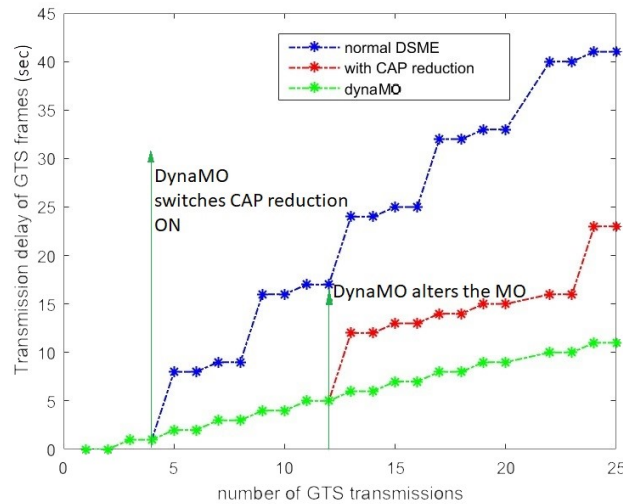
Figure 6.6: Comparison in terms of delay for the example scenario

From Figure 6.6, it can be noted that for a traditional DSME, the transmission delay of the GTS frames starts to increase at a point where the multi-superframe cannot allocate more GTSs. This results in a wait until the next multi-superframe to accommodate the transmission. However, if CAP reduction is triggered, the delay is much smaller when compared to the normal DSME, as more GTSs resources are available. However, as the MO is static, the delay inevitably starts to increase when enough resources are not available, imposing transmission deference to the next superframe. With DynaMO, the MO is increased when more resources are needed. Hence, it provides better results than networks with solely CAP reduction enabled (by 15%) and DSME networks with constant, non-dynamic settings (by 35%).

We also analyzed the throughput of the DSME network with several scenarios using Equation 6.6. For this analysis, we take a DSME network equipped with 2 channels and with a constant data rate of 250 kbps. We compared the normal DSME with the DSME under CAP reduction that employs a fixed MO of 2 and a DynaMO-enabled DSME network. We analyse the

network throughput w.r.t the number of GTSs accommodated in the timeslots. An average 10% improvement in throughput was observed in a DynaMO enabled network.

The results are illustrated in Figure 6.7. In case of DSME with fixed CAP reduction, it can be noticed that for a reduced number of nodes, there is an excess of resources. This affects the overall network throughput because of wasted bandwidth. As the number of GTS transmissions increases, the throughput under normal DSME steadily decreases because traffic (as shown in scenario T3 -T4) has to wait en-queued for an entire superframe duration to be granted service. In the case of DynaMO, dynamic CAP reduction and the efficient tune of the MO results in a better throughput. If the number of resources is abundant, the MO is reduced or the CAP reduction primitive is switched off in such a way that less bandwidth is wasted, thus resulting in better throughput.
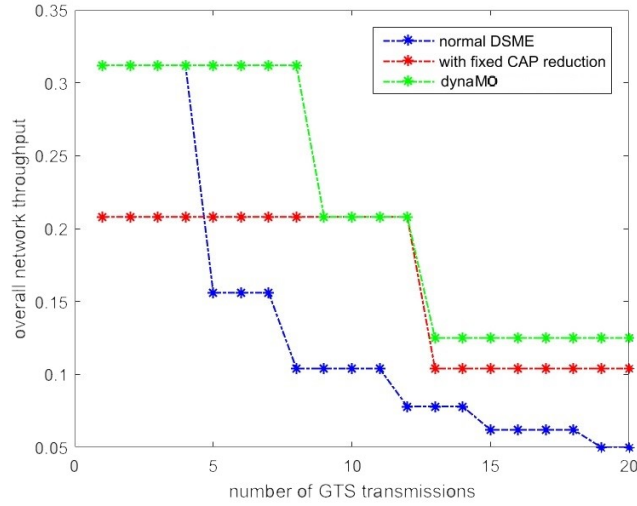
Figure 6.7: Comparison in terms of overall network throughput for the example scenario

6.5 Simulation results

For evaluating DynaMO, we use the OpenDSME simulation platform [118]. OpenDSME is an OMNeT++/C++ simulation-based environment that is dedicated to the simulation of the IEEE 802.15.4e DSME protocol. OpenDSME also provides the possibility of implementing a viable network layer on top of it. The DSME sublayer of OpenDSME employs a typical slot-based reservation system for a schedule that is provided by the top layer.

In our model, we provide BO, MO, SO and the CAP reduction primitives as a direct input. Other network simulation parameters such as traffic rate, the burst size, the interference, and the mobility models are also given directly. Furthermore, there is also a possibility to input the schedule based on a static schedule. We have also incorporated delay and throughput parameters [129] in the network definition files to obtain the appropriate output for the network simulated.

For our simulation set up, we consider several nodes that are arranged in a static concentric mobility pattern around the PAN Coordinator [130]. The static concentric mobility pattern is one of the several mobility patterns in OpenDSME, and it places several nodes in a set of concentric circles around the PAN Coordinator. A static concentric pattern can be used to represent several DSME based use cases like intra-car communication and smart area monitoring.

We conducted our experiments using contention-based and non-contention communication over IEEE 802.15.4e. A traffic of 100 packets of 75B length was generated from the node to the sink. In our first three scenarios (scenarios 1-3), we understand the impact of the change in throughput and delay with respect to the change in MO and CAP reduction primitive in a DSME network without DynaMO, and then in the next following three scenarios (scenarios 4-6), we demonstrate the impact of DynaMO on a DSME network.

We demonstrate the performance of the DSME in terms of throughput with and without CAP reduction in Scenario 1. In Scenario 2, we vary the MO and analyse the throughput to infer a general understanding of its behavior without DynaMO. In Scenario 3 we compare the throughput and the delay obtained through several presets (refer to Table III) in accordance to the standard. In the Scenario 4, we study the impact of DynaMO on throughput and bandwidth in a DSME network. In Scenarios 5 and 6, we study the performance of DymaMO against high throughput and delay-sensitive settings in terms of delay for different traffic configurations. In Table 6.2, we provide the parameters that we have used for all the scenarios we put under extensive simulations.

Parameters	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6
Packet Length	75B	75B	75B	75B	75, 100B	75, 100B
Packet Traffic Interval	50ms	50ms	50ms	50, 30, 15ms	50, 30, 15ms	50, 30, 15ms
Destination	sink	sink	sink	sink	sink	sink
MAC Queue Length	30	30	30	30	30	30
MAC Frame Retries	7	7	7	7	7	7
BO	6	6	6, 10	6	10	6, 10
SO	3	3	3, 5	3	5	3, 5
MO	4	4,5,6	4, 6	DynaMO	6, DynaMO	4, 6, DynaMO
Number of Nodes	5 to 50	5 to 50	5 to 50	5 to 50	5 to 50	5 to 50
Traffic Rate	50 Kbps	50 Kbps	50 Kbps	15, 25, 75k Kps	15, 25, 50, 75 Kbps	25, 50, 75, 100 Kbps
CAP Reduction	ON/OFF	ON	ON	DynaMO	OFF	ON/OFF/DynaMO

Table 6.2: Simulation Parameters

Scenario 1: Impact of CAP Reduction on throughput

The objective of Scenario 1 is to illustrate the base throughput of IEEE 802.15.4e DSME with and without CAP reduction. We calculate the throughput of IEEE 802.15.4e under the parameters of BO=6, SO=3, MO=4. These parameters result in 2 superframes per every multi-superframe and 4 multi-superframes for a beacon interval. The throughput is calculated for a varying number of nodes ranging from 5 to 50. We also present the results of CSMA/CA throughput under the same conditions as a baseline for comparison.
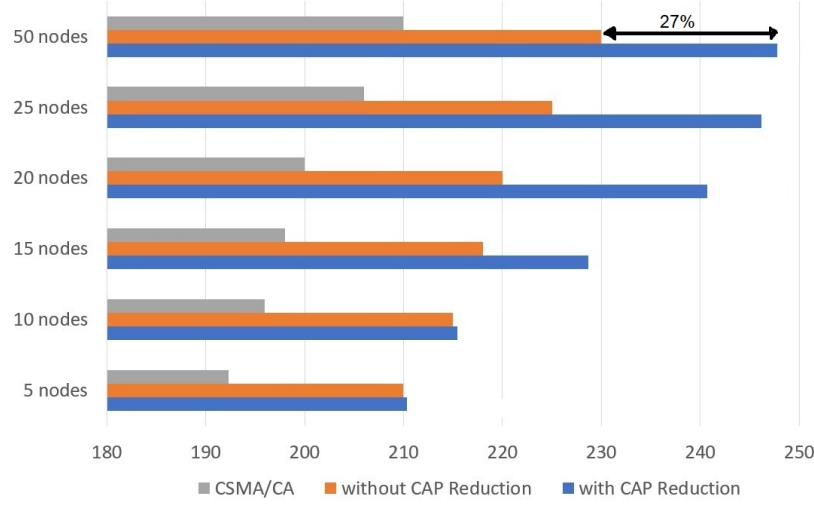
Figure 6.8: Throughput under different configurations

As expected, the throughput under guaranteed bandwidth is constantly higher than that of CSMA/CA ($\geq$ 50%). This is because CSMA/CA is contention-based, which will, in turn, affect the bandwidth of the network and eventually affect the throughput. Up to 10 nodes, there is no relevant difference in the throughput for scenarios with and without CAP Reduction. However, as the number of nodes increase, the number of transmissions to be scheduled also increases. In such a case, traffic must wait till the next superframe to be granted service, thus reducing the throughput. In contrast, by using CAP reduction, the number of resources available increases. It is thus resulting in better service and increased throughput, reaching around 20-30 % for the provided scenario.

Scenario 2: Impact of MO variation on throughput

The objective of this experiment is to investigate the impact in terms of throughput with respect to the Multi-superframe Order (MO) setting without using DynaMO.

When increasing the MO, the number of superframes carried inside a multi-superframe increases. This helps in increasing the number of available resources to accommodate the transmissions. We calculate the throughput for MO ranging from 4-6. For these values of MO, the number of superframes inside a multi-superframes are 2, 4 and 8, respectively. The experiment is conducted for a varying number of nodes from 5-50, and the CAP reduction is set permanently "ON" for this scenario. The Beacon Order (BO) and the Superframe Order (SO) are kept constant at 6 and 3, respectively.

The experimental results are shown in Figure 6.9. It can be understood that when the number of nodes is small, the throughput of the network remains approximately the same, independently of the MO setting. However, as the number of nodes increases, more resources are needed to accommodate the transmissions. At higher MO settings, more superframes are packed within the multi-superframe duration, resulting in improved throughput. DynaMO
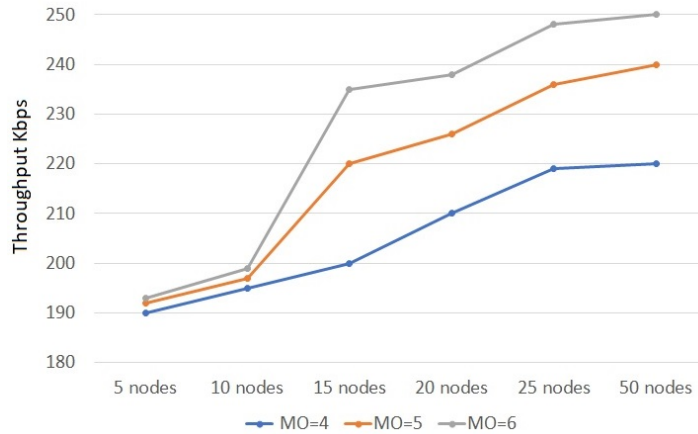
Figure 6.9: Throughput for MO=4,5,6

triggers the appropriate change in MO to maintain a high throughput. Also, as shown previously in Scenario 1, the CAP reduction also plays an integral part in determining the throughput. However, one cannot blindly increase MO or trigger CAP reduction as this has an impact on delay. Also, there is a trade-off between employing CAP reduction and changing the MO. For a small number of nodes, it is preferable to use CAP reduction if sufficient. As we will see next, DynaMO adapts these parameters to obtain a better throughput and delay for the overall network, independently of the scenarios and MO settings initially setup.

Scenario 3: Impact of standard presets on throughput

The IEEE 802.15.4e standard provides several preset settings to support various application scenarios, as given in Table 6.3. The objective of this experiment is to learn how the throughput and delay are affected with respect to these settings.

In our experiment, we take the high throughput specification provided in the standard (BO = 10, SO = 5, MO = 6). Under these parameters it can be calculated that the number of superframes in a multi-superframe interval is 2, and the number of multi-superframes within a beacon duration is 16, providing 1792 GTSs. The delay sensitive parameters will only provide 2 superframes within a multi-superframe interval, and 32 multi-superframes in a beacon duration. The throughput will not be very high in this scenario, but traffic will be serviced with minimal latency. As the QoS requirements of the network can change at run-time, an algorithm like DynaMO can trigger the appropriate changes in the MO and CAP-reduction settings, so that the throughput, latency and even the reliability of the network can be maintained without any dire compromise or trade-off.

Figure 6.10, presents the delay and throughput analysis carried out for high throughput setting of DSME with a capability of accommodating 50 nodes in the network. These are compared against a normal DSME with CAP reduction that can accommodate only 15 nodes within the beacon interval. We can infer that the delay under high throughput settings is higher for a reduced set of nodes than in the case of CAP reduction. Under high throughput settings, nodes

Application type	BO	SO	MO	CAP reduction
Delay sensitive	6	0	1	Enabled
Reliability sensitive	8	3		Disabled
Energy Critical	14	1	14	Enabled
High throughput	10	5	6	Disabled
Large scale	10	1	8	Enabled

Table 6.3: Application scenarios for BO, MO, SO variation

have to wait the entire beacon period for the next transmission, due to the static nature of its lengthy schedule. This results in wastage of bandwidth, thus affecting the overall throughput. Whereas, if we just rely on CAP reduction, only a small amount of bandwidth is wasted in the allocation process, and the delay is thus minimal. Now as the number of nodes is increased, there is a need for additional resources, the throughput steadily drops in the case of CAP reduction, as the delay starts to increase.

The opposite occurs in a high throughput setting as adequate resources are available to service the transmissions. Maximum throughput is achieved because of sending data within small intervals. From this experiment, we conclude that relying on static high throughput settings, corresponding to the allocation of larger MOs in a DSME network, to achieve higher throughput, is not always the best option in terms of performance, due to the increased delay. Better behaviour can be guaranteed if one relies upon a dynamic tuning mechanism, capable of dynamically varying MO to provide optimal throughput and delay. This is the objective of DynaMO.

Scenario 4: Impact of DynaMO on throughput

The objective of this experiment is to investigate the impact of DynaMO with respect to overall network throughput and spare bandwidth. DynaMO dynamically varies the MO and CAP reduction primitives to provide better throughput. Figure 6.11 shows the results of a throughput analysis of DynaMO with respect to different traffic rates. In addition to the throughput, we also analysed the spare bandwidth for each case. A comparison was conducted for variable number of nodes and different traffic rates (15, 25, 75 kbps), under static settings (i.e., DSME with CAP reduction, and high throughput setting) against a DSME with DynaMO. When we compare to the DSME settings with CAP reduction, a high throughput setting is able to achieve almost 20 - 30% higher throughput for higher traffic rates (75Kb - 25, 50 nodes), since throughput under static CAP reduction setting deteriorates when no more GTS resources are available (it cannot scale up). This is visible in the steep decrease in available bandwidth (BW) in the first case as the number of nodes increases (red line - CAP Reduction setting).

We initialize the DynaMO scenario with a static CAP reduction DSME setting (5 nodes) . As seen, when the number of nodes increases, the static CAP reduction, and high throughput settings lose the ability to guarantee the necessary throughput. Contrary, in the DynaMO case, as the network evolves with the addition of more nodes, DynaMO turns on CAP reduction and also
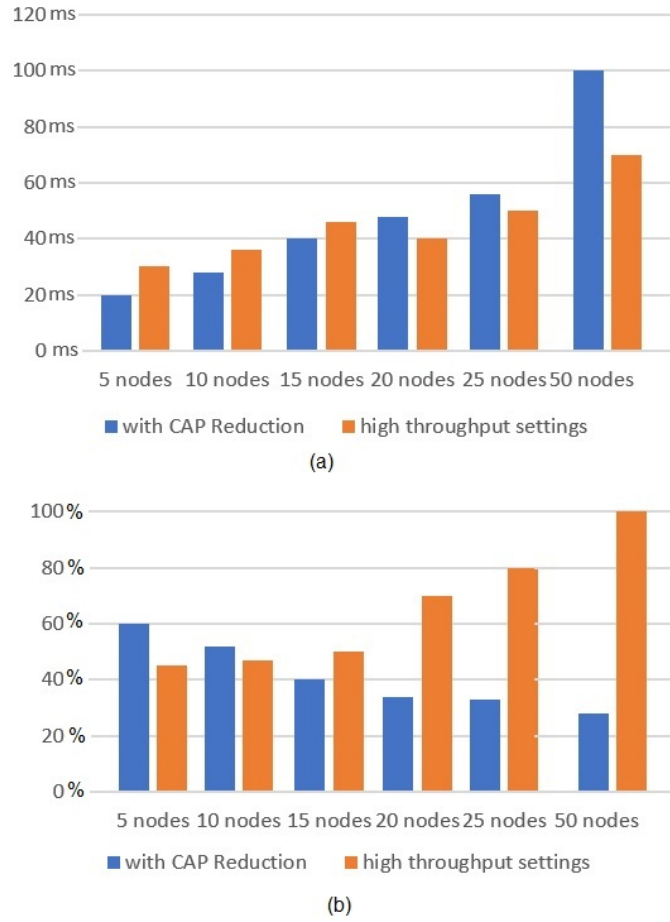
Figure 6.10: delay (a) and throughput (b) analysis with high throughput settings

increases the MO as follows: when the number of nodes increases past 5, DynaMO switches on CAP reduction. As the number of nodes rises above 10, DynaMO increases the MO, providing more superframes to accommodate data, thus increasing the throughput effectively. This results in almost 15-20 % increase in throughput under DynaMO, against a static CAP reduction enabled network.

We also notice that unlike the static CAP reduction setting, the spare bandwidth does not deteriorate steeply but gives us more bandwidth (green line in Figure 6.11) for utilization as the number of nodes increase. Though high throughput settings can provide on-par throughput results, they have a decline in terms of spare bandwidth. This also has an effect on the delay which we will later investigate in Scenario 6.

Scenario 5: Impact of DynaMO on Delay against CAP reduction settings

The objective of this experiment is to investigate the impact of DynaMO with respect to network delay. In this experiment, we compare a static CAP reduction settings against DynaMO for several data rates.
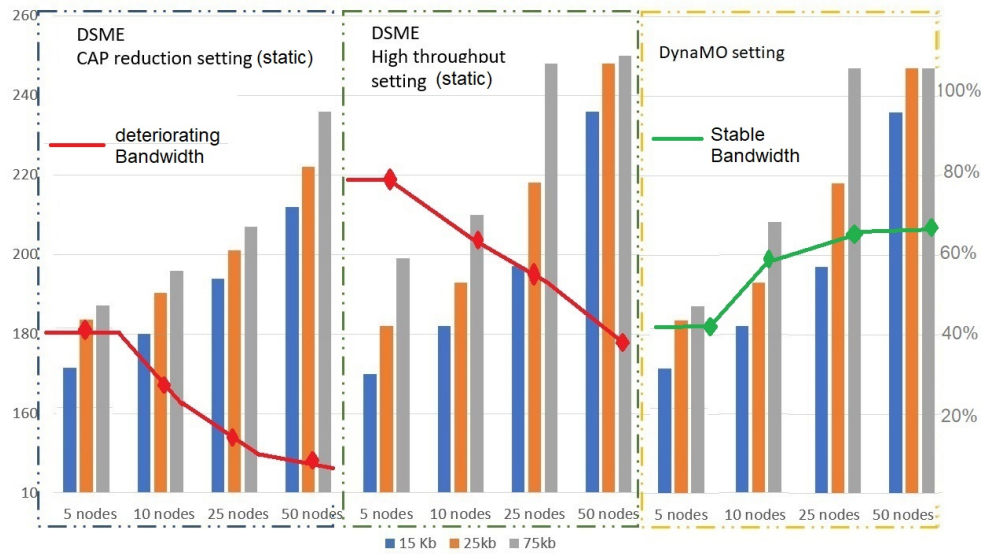
Figure 6.11: Throughput and bandwidth analysis for various data rates

For this experiment, we calculate the values of the overall delay of the network with respect to the number of GTS transmissions, over 50 nodes under different data rates ranging from 5-75 Kbps for CAP reduction and without CAP reduction scenarios in Fig 6.12. This result complements our theoretical analysis shown in Figure 6.10, clearly showing DynaMO in action.
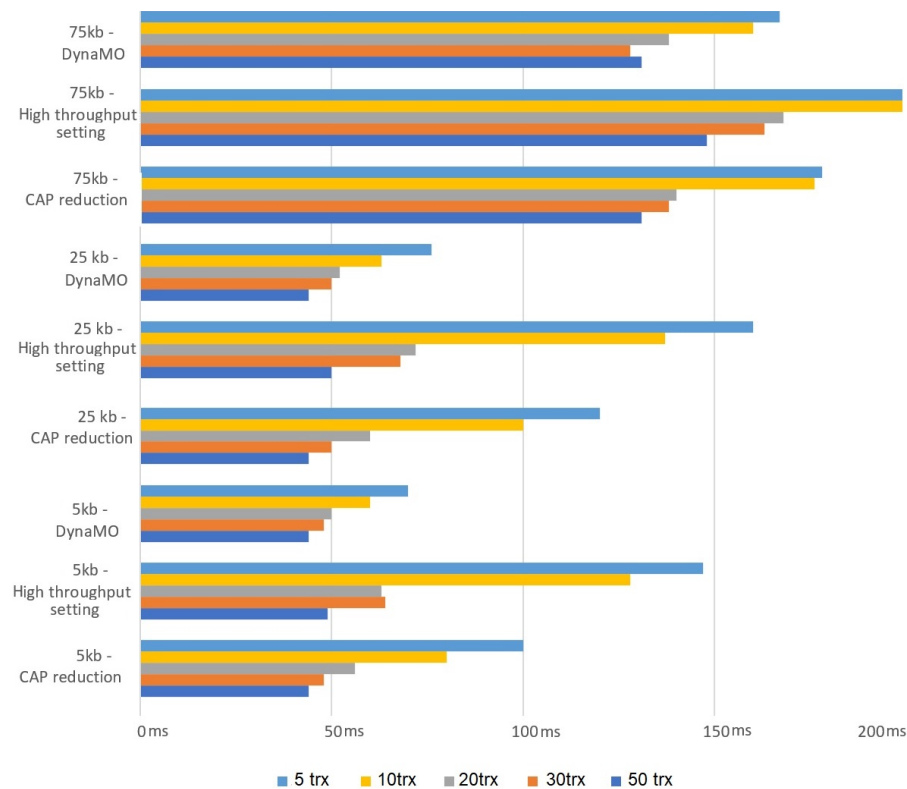
Figure 6.12: Delay analysis for DynaMO against static settings

We use the high throughput parameter settings for this experiment (mentioned in Table 6.3) against a static CAP reduction setting. With a limited number of GTSs transmissions, the delay performance does not have a significant decrease with the scenarios without CAP reduction (5,10,15 transmissions). Delay performance is in-fact sometimes better without CAP reduction when the number of nodes are smaller than 10, due to less wasted bandwidth. However, as the number of transmissions increases, with CAP reduction, delay is minimized. This is due to the fact that nodes need not wait till another superframe duration to accommodate the transmissions that did not occur during the first superframe interval. DynaMO switches the CAP reduction parameters according to the resource requirements and hence doesn't compromise on the delay for those scenarios in which CAP reduction is still not needed, offering a clear advantage over static settings.

For a clear understanding, the example of DynaMO is demonstrated along with the 75 Kbps case in Figure 6.13. At T0, the CAP reduction is OFF providing minimal delay (similar to the scenario without CAP reduction), whereas at T1, due to the scarcity of the resources, the CAP reduction is turned ON dynamically and we can witness a reduction in delay by almost 30%. Above 30 scheduled transmissions, an increase in MO under DynaMO further maintains a lower delay in comparison to static settings including the CAP reduction enabled setting, again in the order of 30%.
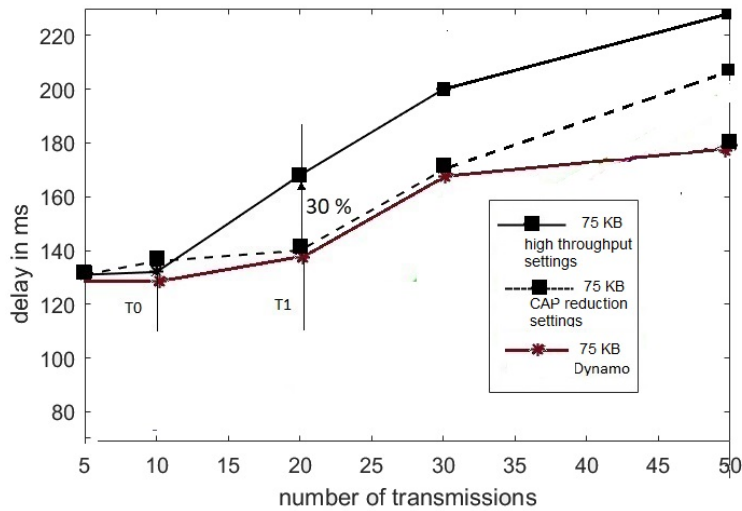
Figure 6.13: Delay analysis for for 75 Kbps data rate

Scenario 6: Impact of DynaMO on delay against delay sensitive and high throughput presets

The objective of this final experiment is to investigate the impact of DynaMO with respect to network delay against the delay sensitive settings provided in Table 6.3. In this experiment, we compare the static high throughput settings and the static delay sensitive settings (dotted lines) with DynaMO. In Figure 6.14, we demonstrate this comparison over 100 Kbps. The other data

rates also have similar behavior. OpenDSME does not allow the value of SO to be set to '0' by default. So we took another delay sensitive setting of BO, SO and MO to be 6, 3, 4 such that the number of superframes within a multi-superframe will be 2 and every beacon interval will have 4 multi-superframes.
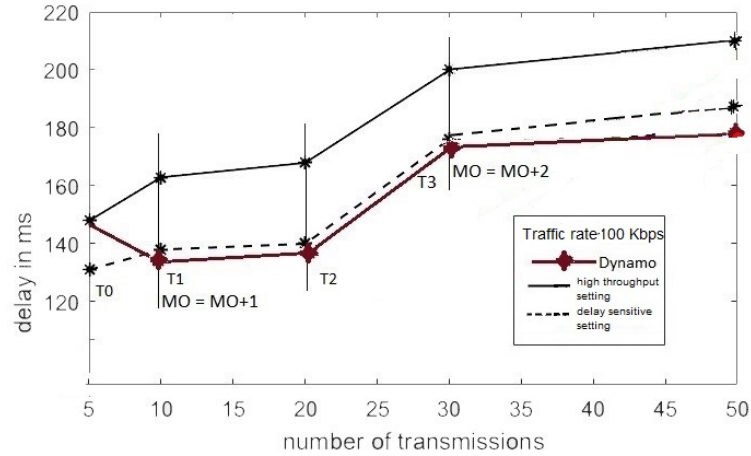
Figure 6.14: Delay Analysis against delay-sensitive settings

The delay is always higher in the high throughput setting, and this gap increases with data rate. The higher MO in the high throughput settings causes a wastage of bandwidth, which results in additional delay, contrary to the time-sensitive settings in which the superframes are tightly packed. We observe a 20-25% reduction of delay under delay sensitive settings, when the number of transmissions is maximized. However, as previously shown in Scenario 4, relying on static settings, which provide shorter a MO, is often not an adequate solution, as it can compromise throughput if the network needs to accommodate an increase in traffic.

In Figure 6.14, at T0, we start DynaMO with a high throughput setting, consisting of one superframe in a multi-superframe. However, as the time-frame moves on to T1 and the number of transmissions increases, DynaMO automatically adapts its MO based on the number of resources. In this case, by increasing MO, DynaMO packs more superframes within the beacon interval, providing more GTS bandwidth and eventually obtaining smaller delays. We can observe a significant reduction in delay, even below the delay-sensitive setting scenario. Notice, that the delay-sensitive setting does not outperform DynaMO in terms of delay when the number of transmissions are small. Although this could somewhat appear counter-intuitive, as the number of transmissions increases, the small MO is not able to accommodate the transmissions causing deference of transmissions to the subsequent superframes. These increase delay, and its effect is particularly visible above 35 scheduled transmissions.

6.6   Final remarks

Traditionally, IEEE 802.15.4 enabled networks require a careful planning of its several MAC parameters, such as MO, SO, BI and CAP Reduction, to achieve adequate QoS levels. However, if this is already an impediment for easy and straight-forward network deployment, in highly dynamic or unpredictable environments finding the right balance is an arduous task. In a complete dynamic evolving network (nodes joining and leaving), static settings can inevitably lead to some compromise in terms of delay or throughput. These compromises can only be addressed by devising mechanisms that can adapt dynamically to conditions that change during run-time.

In this chapter, we proposed an efficient multi-superframe tuning mechanism for DSME networks called DynaMO that dynamically toggles the CAP reduction functionality and adapts the Multi-superframe Order to obtain an improved Quality of Service (QoS) in terms of throughput and delay. We provided a detailed mathematical model of the network and complemented it with a performance analysis. We also used OpenDSME, a simulation platform for DSME, to evaluate the advantages of DynaMO over several DSME network configurations, focusing on throughput and delay. DynaMO dynamically adapts the network parameters at run-time and helps to obtain a better QoS, coping with on-demand changes to traffic and scheduled transmissions. With DynaMO, we were able to increase the average throughput by 15-30% and achieve a 15-35% reduction in delay against a DSME network with static settings.

We believe that the IEEE 802.15.4 and in particular the DSME MAC behaviour is a prominent candidate to become a de-facto standard for IoT implementations, although some mechanisms such as DynaMO can and should be devised to improve its efficiency. Although we believe this analysis is quite conclusive in regards to the impact of this mechanism, we intend to develop an open-source implementation of this protocol for Commercially Off The Shelf (COTS) WSN platforms, to validate the results over a real WSN hardware.

Chapter 7

A deterministic approach towards intra-car communication

7.1 Introduction

In the past decade, Wireless Sensor Networks (WSN) have been supporting several innovative applications in a multitude of domains, such as in health, security, and agriculture [131], [132]. The increasing miniaturization of modern embedded systems, together with the advancements in the area of WSNs and energy harvesting, have opened up new possibilities to fit wireless communications into a previously unexpected series of applications. However, the automotive industry has understandably, been reluctant to adopt WSN, mostly pointing out its non-deterministic communication behaviour and unreliability due to electro-magnetic interference and security issues. Therefore, wireless networking has been so far confined to some limited functionalities of infotainment systems in the automotive domain. Its adoption in critical systems has been virtually non-existent for vehicles, although it has been already enabling a series of critical scenarios in several other industrial domains [133].

The day-to-day automobile has gradually evolved from fully electric and mechanical design to a fully electronically equipped modern car, enabling several additional safety and comfort functionalities. The current production vehicle is fitted with over 150 sensors, and actuators, coupled with hundreds of Electronic Control Units (ECU) [134], interconnected through thick wired harnesses and they communicate based on real-time communication protocols. The wiring harness is the heaviest, most complex, bulky and expensive electrical component in a vehicle and it can contribute up to 70 kg to the vehicle mass. This impacts the vehicle's performance in terms of fuel consumption and increases its environmental footprint. Additionally, maintenance of the vehicle can become cumbersome if the harness experiences deterioration over time [135]. The current trend, is to continue to increase the number of application modules and complexity in the vehicle, by fitting newer models with improved Advanced Driving Assistance Systems (ADAS) to increase their safety and autonomy. However, this effort is not being applied to the millions of older vehicles that will continue to share the roads in the next

15 years, partially due to the tremendous complexity involved in retrofitting such vehicles. In all these fronts, wireless communications can potentially become the enabling technology to address these problems, considering its flexibility and ease of deployment, by exploring the innovative plug-and-play possibilities. However, ADAS pose stringent requirements to a system's control and communications, in terms of timeliness and reliability, and therefore these properties must be ensured by the underlying communications technology. Additionally, energy-efficiency must also be considered, if some of these sensors are to be powered by energy scavenging methods or to avoid cabling for power.

The features of the low-power, low-rate IEEE 802.15.4e amendment enable interesting features such as guaranteed bandwidth, deterministic delay and improved reliability by the introduction of multi-channel techniques. These characteristics make this communication protocol as a prominent candidate to support wireless ADAS as well as other non-critical applications. The DSME MAC behavior supports both contention and non-contention-based traffic, offering a good flexibility to support such systems, by enabling multi-channel and mesh capabilities. From our previous works [25], [24], we provided the groundwork to understand the performance limits of the protocol, such that it can be assessed to enable such application scenarios.

Importantly, such cyber-physical systems impose an unprecedented integration between communication, sensing and actuation, alongside the particular characteristics of the dynamics of the vehicle and the physical environment. This significantly increases the complexity of these systems, which due to their critical safety requirements must undergo additional and prolonged test and validation in different situations. However, the expensive equipment and safety risks involved in testing, demands for comprehensive simulation tools that can as accurate as possible mimic the real-life scenarios, from the autonomous driving or control perspective, as well as from the communications perspective, focusing on their interplay.

The Robotic Operating System (ROS) framework [136] is widely used to design robotics applications, and aims at easing the development process by providing multiple libraries, tools and a publish/subscribe transport mechanism. On top of it, several simulation tools are capable to simulate the the sensor/actuator and control components of these vehicles. On the other hand, several network simulators are available and capable of carrying out network simulation of various networking protocols. Nonetheless, these tools remain mostly separated from the autonomous driving reality, offering none or very limited capabilities in terms of evaluating ADAS in such scenarios.

In this chapter we introduce a co-simulation framework that enables the simulation of an ADAS application system from these two perspectives: a physics perspective that helps in building different vehicle dynamics and a network perspective that helps in designing the system to communicate safely by exploring the impact and the performance limits of different network configurations. We carried out the integration of a ROS-based robotics simulator (Gazebo) [137] with a network simulator (OMNeT++), enabling a novel framework to test and validate wireless ADAS. In the one hand, we leverage upon Gazebo's most prominent features,

such as its support for multiple physics engines and vehicles in integration with ROS, which enables us to build realistic vehicle control scenarios. On the other hand, OMNet++ supports the underlying network simulation via the OpenDSME communications' stack [118]. This integration helps us to accurately analyse the communications impact upon wireless ADAS applications.

7.2   Specification of the use-case

Automobile industry giants like Tesla are now venturing into the idea of intra-car communication. Some of the models include functionalities like automatic driving from highway on-ramp to off-ramp including interchanges and overtaking cars. They can also automatically change lanes while driving on the highway and auto-park within both parallel and perpendicular spaces. There are also functionalities such as summoning a parked car to maneuver and reach the owner anywhere in a parking lot.

For our use case, we aim to implement a classic intra-car system that is used to detect the surroundings for any threats and alert the user and actuate accordingly. Sensors in the intra-car model have to be strategically placed towards the corners of the car and must be tilted to a specific angle to achieve maximum field of view, without any blind spots. In Figure 7.1, we provide a pictorial representation of a production model Tesla model X car [138] that moves safely by detecting the other cars in the environment. We intend to use the same application of a high end Tesla vehicles intra car functionality of alerting a driver and helping in Automatic Driving Assist System (ADAS) using wireless technology (specifically DSME).
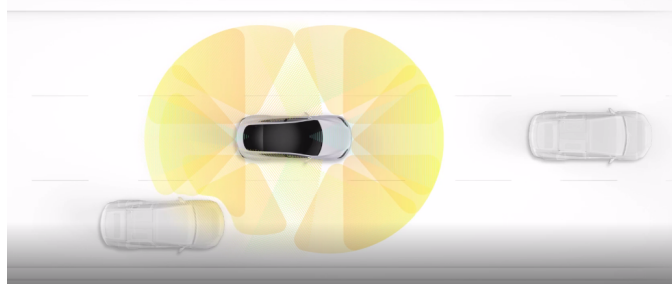
Figure 7.1: Tesla ADAS system - model X

In this work, we simulate two commonly occurring parking lot scenarios to learn the impact of the communication protocol:

Scenario A: This scenario involves a case where the vehicle has to maneuver towards a parking spot; the sensors around the vehicle have to pick the readings of the obstacles around (either stationary or moving in variable speeds) and maneuver safely. In this scenario as the vehicle is maneuvering towards a spot, it must have a keen sensing capability to maintain a steady awareness of its surroundings, and should maintain a deterministic and low delay in the communication between the sensors and the actuation unit. (Figure 7.2)
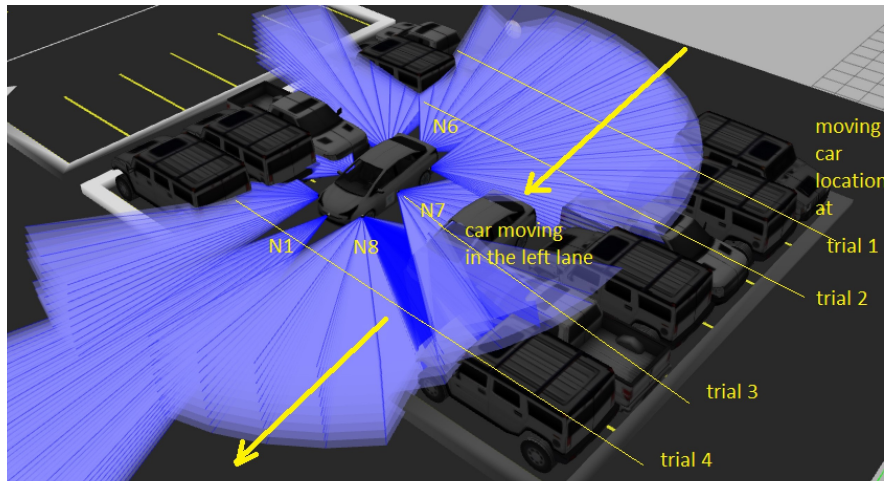
Figure 7.2: Gazebo Simulation - Scenario A

Scenario B: This scenario involves a case where a vehicle enters a parking compound with an intention of finding a spot to park, during this time, the driver should be able to properly detect and actuate accordingly to the other vehicles moving around it. Depending on the vehicles in its surroundings, the intra-car system must take the decision to apply brakes when a crash scenario might happen from a driver's flaw, like loss of attention. It should be noted that inside the parking lot the vehicle must be able to maneuver till maximum limit of at least 30 Km/h. Taking in account the obstacles and the other vehicles within the line of sight, we consider a worst case scenario where the vehicle must detect under 360 milliseconds in order to avoid a potential crash (Figure 7.3). This value was gathered in simulation analyzing a scenario where a vehicle was running at 30 Km/h and its front radar detected the obstacle within a 9 meters distance, within these conditions the vehicle and its control systems were able to properly stop even with the distance information arriving with a 360 milliseconds delay. Values above this threshold would make the vehicle control system not being able to stop "in time" resulting in a crash.

Hardware specification:

Two types of sensors are integrated within the vehicle's parking assist system: firstly, a radar for sensing and detection of objects/obstacles directly in front of the vehicle. We use a radar with a range of 30 meters but, obviously, should be upgraded to detect at higher ranges of sight for different long sight ADAS scenario implementations. 7 sonars that detect obstacles up to 9 meters of distance is also incorporated for this use case. The sonars taken for this use case consume significantly less power (15mA) and can detect up to 16 returning echoes, which can be very helpful if several objects are at different distances within the field of view.

These sonars are placed all around the vehicle and in conjunction with the radar they can detect obstacles in a 360° field of view in the vehicle's surroundings. They were set to a Horizontal angle of ±0.30 rad and a Vertical angle of ±0.14 rad towards the corners of the vehicle, so that
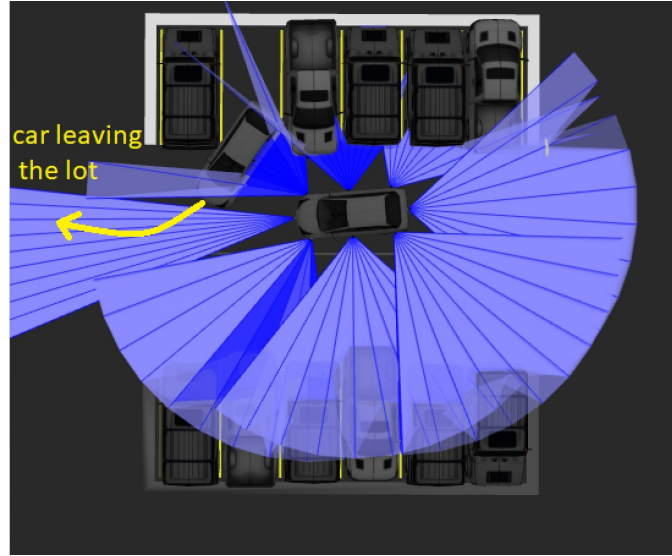
Figure 7.3: Gazebo Simulation - Scenario B

they achieve a larger line of sight. Moreover, these sensors would be a right fit for retrofitting purposes as they are not expensive, or have high energy consumption, while still able to provide enough range for the functionality of the proposed scenarios.

Network specification

For our intra-car system, we intend to use the beacon-enabled mode of DSME MAC behavior of IEEE 802.15.4e because of its deterministic capabilities. In this work, We vary the network architecture defining parameters like the Multi-superframe Order (MO) and the CAP reduction technique to learn the impact in the overall performance. Based on the network settings, we learn the co-relation in terms of speed and latency and the safe functioning of the intra-car system. Based on the network presets discussed in Chapter 6, we use the standard based delay sensitive settings for our scenarios and also propose appropriate settings of MO and SO for the efficient functioning of the intra-car communications system.

7.3 Implementation tools

To evaluate our model in real-time, we synchronize several tools to recreate an intra-car simulation model. For evaluating such complex models it is vital to rely on robotic simulators to emulate the environment and accurate network simulators that can work synchronously with the inputs provided through the robotic emulators. In what follows, we briefly describe the list of tools used to establish this model.

OMNeT++ and Gazebo

OMNeT++ is a C++ based discrete event simulator for modeling communication networks, multiprocessors and other distributed or parallel systems. Some of the major advantages of this simulation tool are its capabilities to enable large-scale wireless simulation and its support for hierarchical simulation models. In our model we use OpenDSME [139], an OMNeT++-based network simulator exclusively designed for DSME networks. In OpenDSME, there are several compound modules such as the DSME data link layer which handles the GTSs allocations. The structure of a network model is defined in OMNeT++'s topology description language called NED(Network Definition file). Typical components of a NED file are the module declarations, compound module definitions and network definitions. Figure 7.4 provides a representation of a NED file of a DSME enabled node. It can be noticed that the DSME nodes operate on a complete OSI (Open Systems Interconnection model) structure comprising the application, network, datalink and the physical layers. Under dynamic network control, every DSME enabled node in our network model uses a generic routing mechanism to establish efficient routes to the DSME sink. We use a traffic generator to generate regular traffic from the sensor nodes.
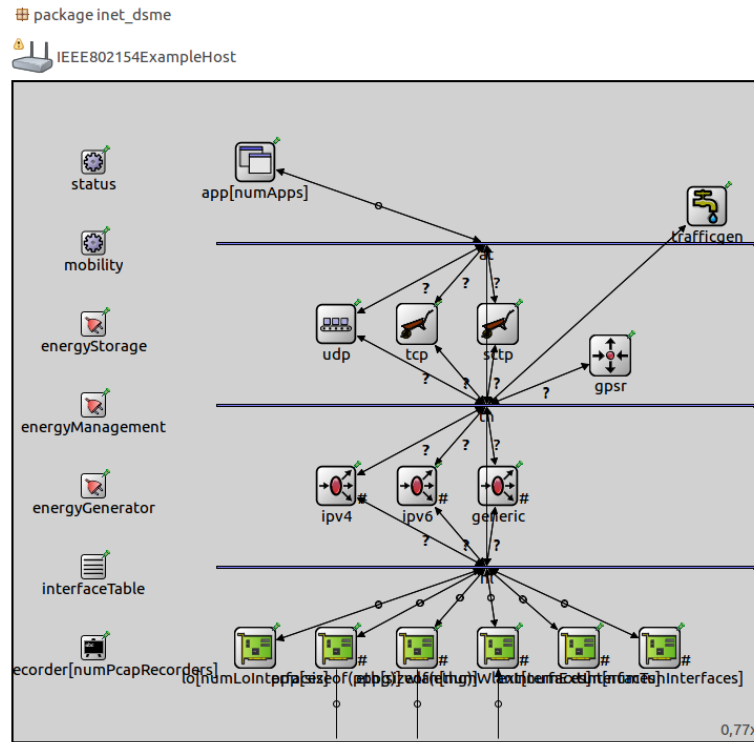
Figure 7.4: NED file for a DSME node

The Figure 7.5 illustrates an example of wireless nodes in an intra-car simulation network synchronized with Gazebo. Gazebo is a robotic simulator that helps in creating dynamic simulations by providing access to multiple high-performance physics engines. It can also generate sensor data, optionally with noise, from laser range finders, 2D/3D cameras, or even contact sensors. It also gives access to several plugins for robots/environment control, developed with

the Gazebo's API. Figures 7.3 and 7.2 were rendered in Gazebo by taking a real-life Toyota Prius model and assembling it with the sensors required for the functionality of this application. In Figure 7.5, we can see the placement of the nodes that cover a 360° line of sight which are translated to the NED file of the OMNeT++. Using our novel synchronization approach the network directly corresponds to the robotic model of the intra-car communication system designed in Gazebo with ROS nodes.

ROS nodes   OMNeT++ nodes
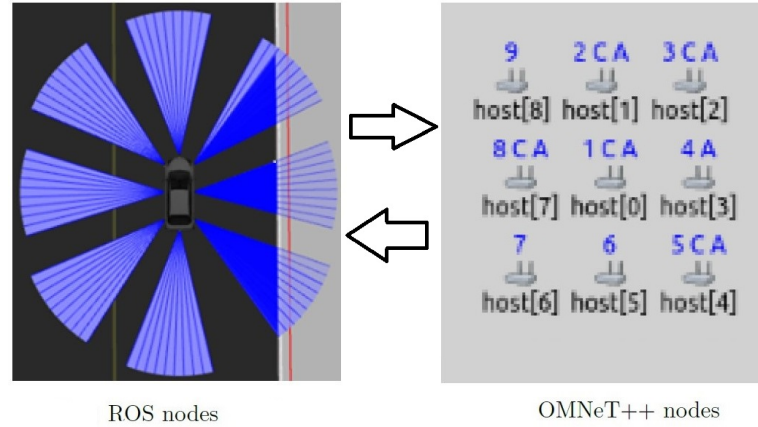
Figure 7.5: integration of OMNeT and Gazebo

Robotic-Network co-simulator

Simulation of an integrated application can be done in diverse ways, either by integrating two different simulators or by expanding a network simulator with physical models [140] or by expanding the physical simulator with network model [141]. However, integrating two simulators is the best approach to replicate a real-life scenario. Kudelski *et al.*, in their work [142], propose an integrated framework to support multi-robot and network simulation. The authors propose an integration of three simulators namely ARGoS [143], NS-2 [144] and NS-3 [145] that can be used in co-simulation scenarios. ARGoS is a multi-physics robot simulator that can simulate large-scale swarms of robots of multiple variants. ArGoS also provides a possibility to add new plug-ins to perform additional functions. This is very similar to the Gazebo that we use in our work which presents the possibility to add modules to the existing framework to represent newer complex scenarios. In this work the authors propose a synchronization approach between the simulators in which the number of nodes, characteristics of the equipment and the simulation area are synchronized. At every simulation step the ARGoS sends the updated robot position to the network simulator and the communication is transferred back to the robotic simulator. In our work, we take a similar approach by integrating Gazebo with OMNeT++. In our case, we use the ROS sync application to handle the synchronization in our simulations.

BARAKA [146] is another co-simulator tool introduced by Thomas Halva Labella *et al.* In this work the authors provide a tool that is able to perform integrated simulation from a communication standpoint using OMNeT++ and a robotic standpoint using the Open Dynamics

Engine (ODE) [147] for rigid-body physics simulation. The integration in this simulation is done in two steps, they first integrate the collision/detection step loop in the OMNeT++. Then they create modules that simulate the robots and motes.These modules are accessed by an agent program to control the communication in the system. The ODE loop of the OMNeT++ in this case has no connection to any other module in the simulation. Contrast to this approach, in our work, for every simulation step, the simulators are synchronized to provide a better realistic simulation environment.

The integration in our work is done over the Robotics Operating System (ROS), based on previous works of the CISTER lab [148], [149], which focused on inter-vehicle communications (i.e. using ETSI ITS-G5) to enable a cooperative platooning function. A general architecture for our framework is presented in Figure 7.6. The integration of the network model is supported by the OpenDSME open-source framework [139] to implement the DSME protocol on top of the IEEE 802.15.4 physical layer. Two kinds of nodes are implemented in OMNeT++/INET simulation: the sensor nodes corresponding to 8 end-devices and a sink corresponding to the PAN Coordinator, respectively. As shown in Figure 7.5, the displayed outward 8 nodes of the the OMNeT++ correspond to the wireless radar/sonar modules implemented in the Gazebo vehicle model. At the center of the layout, the "sink" node (host(0)) is also displayed and corresponds to the Application Unit (AU). The AU is responsible for the ADAS system control implementation; it processes the sensor inputs and reacts accordingly, by interfacing the vehicle's steering and braking systems.

Synchronization Approach

OMNeT++ is an event-driven simulator and Gazebo a time-driven simulator. In order to synchronize these simulators, a synchronization module was implemented in OMNeT++ to carry out this task, relying upon the ROS "/Clock" topic as clock reference. The OMNeT++ synchronization module subscribes to ROS' "/Clock" topic, published at every Gazebo simulation step and proceeds to schedule a custom made OMNeT++ message for this purpose ("syncMsg") to an exact ROS time, which allows the OMNeT++ simulator engine to generate an event upon reaching that timestamp and executes any other simulation process that must run.

Data Workflow

The ROS publish/subscribe middleware is used to support data flow between the Gazebo and OMNeT++ simulators. For each node in the OmNet++/INET simulation, there is a corresponding sensor in the Gazebo vehicle model which publishes its relevant data into a rostopic. In the OmNet++/INET side, each node subscribes to the corresponding ROS module and prepares a message that is en-queued into the OpenDSME MAC layer to be transmitted to the sink node (PAN Coordinator). OpenDSME handles the transmission and, if successful, the sink node publishes a rostopic with the sensor data that is subscribed by the AU. The AU then uses this input to feed its control loop.

As for the Gazebo model, a Toyota Prius car model (visible at fig.7.6) is used as the baseline deployment for this WSN layout. With this general layout architecture, different ADAS scenarios can be implemented, by changing sensors or their characteristics, the vehicle model, the track and the surrounding environment, enabling the possibility to extensively test and validate a ADAS behaviour and explore its performance limits pre-deployment.
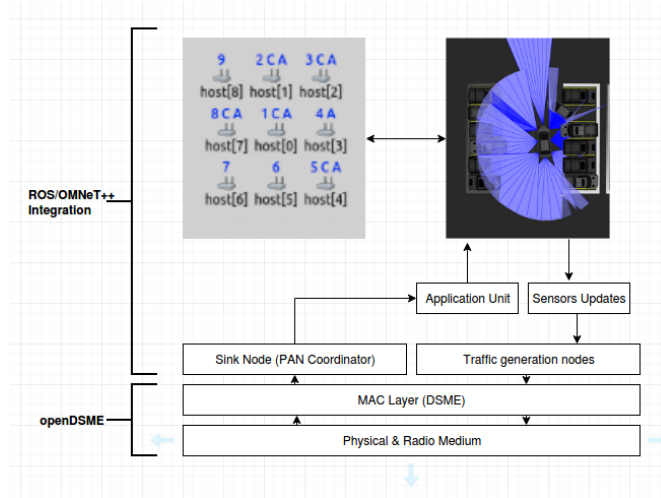
Figure 7.6: Integration Architecture

The modern ADAS in vehicles are increasingly being equipped with a wide variety of sensors, in order to get a good awareness of their surroundings. Some of these sensors include the pressure, fuel and even air to detect performance and prevent any failure. In this framework, all these sensors, can be implemented as modules and be integrated with the network model as a new node that feeds data into the AU, thus enabling the possibility of devising a multitude of scenarios.

Path loss model specification

We tested several path loss models for our intra-car modelling. We used the log normal shadowing path-loss model [150] for our application scenarios. Unlike the Friis free space model [151] that is used as a path loss model for clear path between the transmitter and the receiver, we must choose a model that encompasses random losses due to signal blockage. In an intra-car scenario the sensors will usually be blocked by the mechanical chassis of the vehicle and an ideal path loss model for this scenario must include the path loss attributed by the signal blockage. The propagation path loss in free space (denoted as PL) is the loss incurred by the transmitted signal during propagation at a wavelength of λ for an arbitrary distance of d meters. This value is given by:

$$PL(dB) = -10log10[\lambda^2/(4\pi d)^2] = +20log_{10}[4\pi d/\lambda] \tag{7.1}$$

For a distance beyond df, if $PL(n)$ is the path loss at a distance n meters from the transmitter, then the path loss at an arbitrary distance $d > n$ is given by:

$$[PL(d)]dB = [PL(n)]dB + 10n log_{10}(d/n) + X, df \geq n \geq d \tag{7.2}$$

where X is a zero-mean Gaussian distributed random variable with standard deviation σ expressed in dB, used only when there is a shadowing effect.

In the literature (e.g [152]), researchers have measured the Ultra Wide Band (UWB) channel between a receiver installed in the rear view mirror and transmitters near the tyres for an intra-car communication. In our case we fit a path-loss exponent of 2.2 as we fit the sensors near the tires and small-scale fading following the log normal distribution with a standard deviation of 0.8 dB.

7.4   Simulation results

7.4.1   Impact of MO on the maximum acceptable speed:

The aim of this experiment is to determine the maximum acceptable speed for the safe functioning of a DSME-enabled intra-car communication. The scenario taken for this experiment is depicted in Figure 7.3. Despite the capability of the sensor to timely communicate accurately to the PAN Coordinator, a crash could happen because of the delay in actuation and high velocity. For this experiment, we consider several delay sensitive values to define a bound for the maximum speed each setting could support.

We conduct this experiment for varying speeds in the range of 15-35 Km/hr for various MO and SO settings. In this scenario, the sensing vehicle has to detect both the car in-front of it and the car trying to overtake it and stop without crashing. Even some commercial experiments done by companies like Benz that have utilized radars in the past have proved an assured success rate only when the car travels within 50 km/h. However, these systems were wired-based and cannot be retro-fitted for any other older models.

SO	MO	SD	MD	Maximum Acceptable speed
4	6	245.7 ms	982.8 ms	27.4 Km/hr
4	5	245.7 ms	491.4 ms	29.4 Km/hr
4	4	245.7 ms	245.7 ms	31 Km/hr
3	3	122.8 ms	122.8 ms	33.3 Km/hr

Table 7.1: Maximum acceptable speeds for various settings

As depicted in Table 7.1, the maximum acceptable speed increases for smaller values of MO. In the case of the delay sensitive setting BO=6, MO=6, SO=4, within 27.4 Km/h, all the sensor data were received by the PAN Coordinator (in less than a second in the worst case), and there was enough time to actuate and stop the vehicle by detecting the obstacle in the front and the side. The crash rate went on increasing exponentially with the increase in speed. In the event of

27.92 Km/h for the same setting (BO=6, MO=6, SO=4), the system witnessed 1 crash in 10 trials. In this specific trial, the sensor was able to communicate with the coordinator in 1.2 seconds, which in turn caused a delay in actuation, making the crash imminent.

However, when we reduce the value of MO further to 5, 4 and 3, we reduce the size of the superframes within the multi-superframe interval and also the size of the multi-superframe. In the case of MO = 3, we even encountered a very minimal delay of just 0.013 seconds. Despite successful reception of the packet in this scenario, we encountered a crash due to higher velocity of the vehicle. This could have been avoided if we use sensors that provide a longer line of sight, however by keeping the minimal speeds the vehicle operates in a parking-lot the choice of the sonars taken was a right fit. From a network perspective, this delay is a result of the waiting of respective data packets until another superframe to get accommodated. This can be avoided if the length of the superframe was smaller and repeated for shorter intervals.

7.4.2 Impact of CAP reduction

The aim of this experiment is to understand the impact of CAP reduction setting in a parking-lot scenario. The setting requires 1 radar and 7 sonars for its functionality (see Figure 7.3). For the first round of experiments, we take a standard defined delay sensitive settings where BO, SO, MO = 6,4,6 and we use the CAP reduction primitive. In this case, the entire multi-superframe duration will be 982.8 ms. In this setting, when we switch on the CAP reduction, 4 superframes will be engulfed in a multi-superframe duration and we will have 49 GTS to accommodate the transmissions.
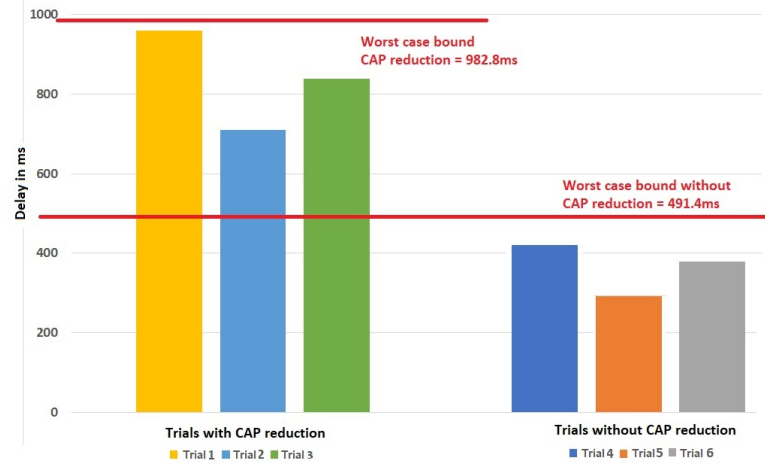
Figure 7.7: Comparison of setting BO, MO, SO = 6, 4, 6 with and without cap reduction

As shown in Figure 7.7 for the first three trials we keep the CAP reduction primitive ON for the aforementioned delay sensitive settings. Every trial in this result is an average of 10 simulations with the same settings. For the functioning of this scenario, only 8 sensors are needed for the GTSs allocation. Having more unnecessary bandwidth allocated would increase the latency as the allocation of the GTSs occurs every multi-superframe duration. As for this

scenario, with CAP reduction 49 GTSs are available and most of the GTSs do not get allocated. Further, the next set of transmissions have to wait an entire multi-superframe to get allocated adjacently resulting in an increase in the overall delay.

Without CAP reduction, there will be only 28 GTSs and there will be a possibility of allocation of the timeslots within the third superframe interval, drastically reducing the overall delay. The next set of trials (4-6) have the CAP reduction primitive switched OFF based on the number of resources that need to be allocated. When the CAP reduction primitive is switched OFF it is clearly visible that the worst case delay bound drops drastically, and we will be able to accommodate all the transmissions within a shorter time period, thus reducing the latency. It must be noted that in both of these cases, as the length of the Multi-superframe was larger, they were not able to communicate within the prerequisite time bound (360 ms). This can be alleviated by the reduction of the Multi superframe order.

When the vehicle needs to operate on information from various sensors, like the fuel sensors and pressure sensors, we need to have CAP reduction and sometimes a larger MO to accommodate multiple transmissions. However adding such additional functionality will have a dire increase on the delay of the system and the system will have to operate on reduced speeds. Having a dynamic technique like DynaMO, as discussed in the previous chapter, would help in the allocation of resources in dynamically varying ADAS systems by varying the CAP reduction primitive and the Multi superframe Order.

7.4.3   Delay Analysis with dynamic scheduling:

Despite the lower speeds required for the operation of the parking lot scenario, the delay must be bounded to avoid a potential crash. We take Scenario A, as depicted in Figure 7.2, for this experiment. The aim of this experiment is to learn the delay of the reception in every sensor that impacts in the maneuvering of the vehicle in a parking lot with dynamic allocation of the GTSs. In this scenario, we take a stationary vehicle placed in front of our sensing vehicle and a moving vehicle passing by the left side. In such a case, the sonars (N6, N7, N8, N1) have to pick data from both the moving vehicle at Trials 1, 2, 3 and 4, respectively. For instance at Trial 1, the vehicle has to sense the moving vehicle at the line of sight of N6 and N1, and initiate a braking sequence.

We conducted our experiments with a delay sensitive setting (BO=6, SO=4, MO=4) for all trials and with the sonars operating at a steady frequency of 5 Hz. The recorded communication delays are presented in Figure 7.8. Interestingly, although these delay values are within the scenario safety requirement, they are not bounded. The variations of delay for each sensor are a result of the dynamic allocation of the sensor data in the guaranteed timeslots of the DSME superframe. It can be seen the delay can be less as 250 ms, but it can also grow up to 880 ms as a result of the slot being allocated for transmission several superframes after the arrival of sensor data. This is an implementation issue that poses a safety hazard in these scenarios, and this would be fatal if these scenarios are even more demanding in terms of communications delay.
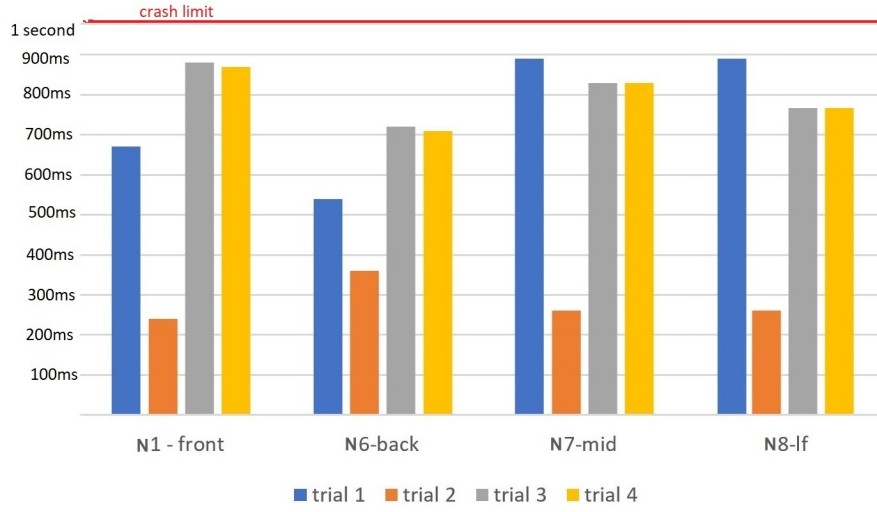
Figure 7.8: Delay analysis for the sonars

7.4.4 Delay analysis with Static Scheduling

From our work in Chapter 5, we can understand that determining an appropriate static scheduling can help in reducing the latency of the network. In a worst-case scenario the maximum time a superframe can take to accommodate a transmission will be the size of the superframe. Hence, by varying the size of the superframe, we will be able to control the latency of the network and determine clear bounds.

In Figure 7.9 we illustrate a superframe with the parameters BO=6, MO=4 and SO=4. Considering the duration of a symbol in a 2.4 GHz band is 16 μs, the duration of every timeslot (T_s) in the superframe is given by:

$$T_s = 16\mu s \times aBaseslotDuration \times 2^{SO} \tag{7.3}$$

Considering a superframe comprising of 8 contention-based timeslots and 7 GTSs (over 16 channels) the size of the entire superframe (SD) can be given by:

$$SD = (7+8) \times T_s \tag{7.4}$$

In this case (BO=6, MO=4, SO=4), the size of the superframe duration is 0.23 ms, which means when a schedule is statically allocated, the worst-case bound delay should be 0.23 ms. This bound can be further minimized by reducing the values of the MO and SO and providing them a static schedule. For the case of MO=3 and SO =3, we can obtain a smaller worst-case bound of 0.11 ms.

For this experiment we took the scenario as depicted in Figure 7.3, in which a maneuvering vehicle has to detect its surroundings and move towards an exit. With the theoretic formulation of the worst-case bounds in mind, we emulated the scenario with static scheduling using our robotic-network co-simulator. In Figure 7.10 we show the results of the experiment. It is clearly
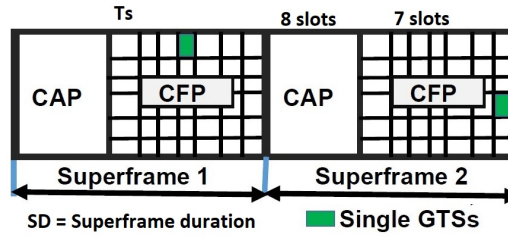
Figure 7.9: Superframe with MO=4 and SO=4

visible that the bounds are maintained properly for the respective network parameters. The values with the worst-case delays usually arrive at the end of the superframe and are scheduled at the beginning of the subsequent superframe resulting in the worst case delay. The delay results in Figure 7.10 are for the settings with MO=4 and SO=4. As previously mentioned in the theoretic analysis, the results of the experimental evaluation also strictly adhere within the limit of the worst-case delay. In this case, we experience a maximum delay bound of 0.23 s.
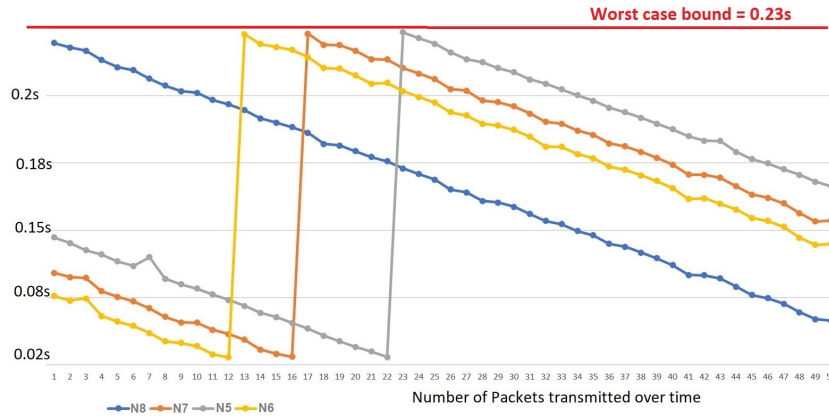
Figure 7.10: Delay analysis with static scheduling for MO=4 and SO=4

From our previous experiments, we understood that the reduction in MO value to an adequate value to support the length of the packet will be very beneficial when the overall latency needs to be reduced. In our case, we were able to reduce the MO till a value of 3. The results of this experiment are depicted in Figure 7.11. This experiment also confirmed our theoretic calculations. We obtained a steady worst case delay of 0.11 s throughout our experiment. The fluctuations of delay values in these static settings can be attributed to the arrival time of the packet. As discussed previously in Chapter 3, the arrival of the packets have direct impact on the overall delay of the network. In our network analysis, we calculated the value for the delay (D_{max}) based on the arrival of the packets and the time they are served, as shown in Figure 7.12.

The packets that get served immediately with respect to its arrival result in a much lower delay. The worst-case delay might be when the arrival starts at the end of the first superframe and gets scheduled at its adjacent frame. One big advantage of static scheduling is that the user has the possibility to vary the network settings and fix a steady worst case bound for the entire

Figure 7.11: Delay analysis with static scheduling for MO=3 and SO=3

network. With using dynamic network settings it will not be possible to fix bounds on the delay of the network and be deterministic.
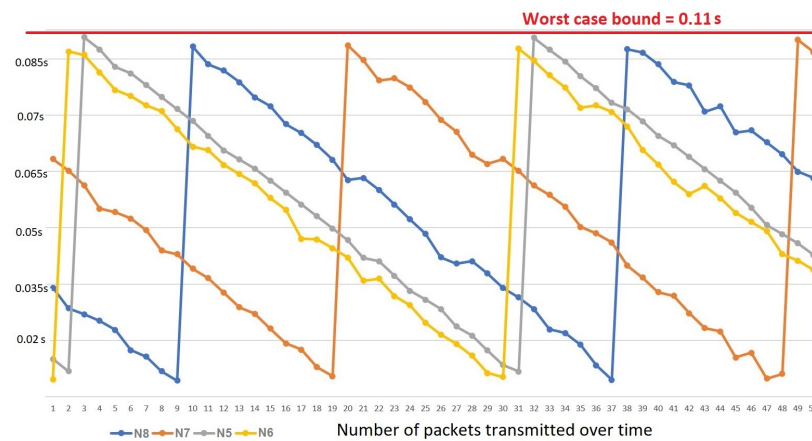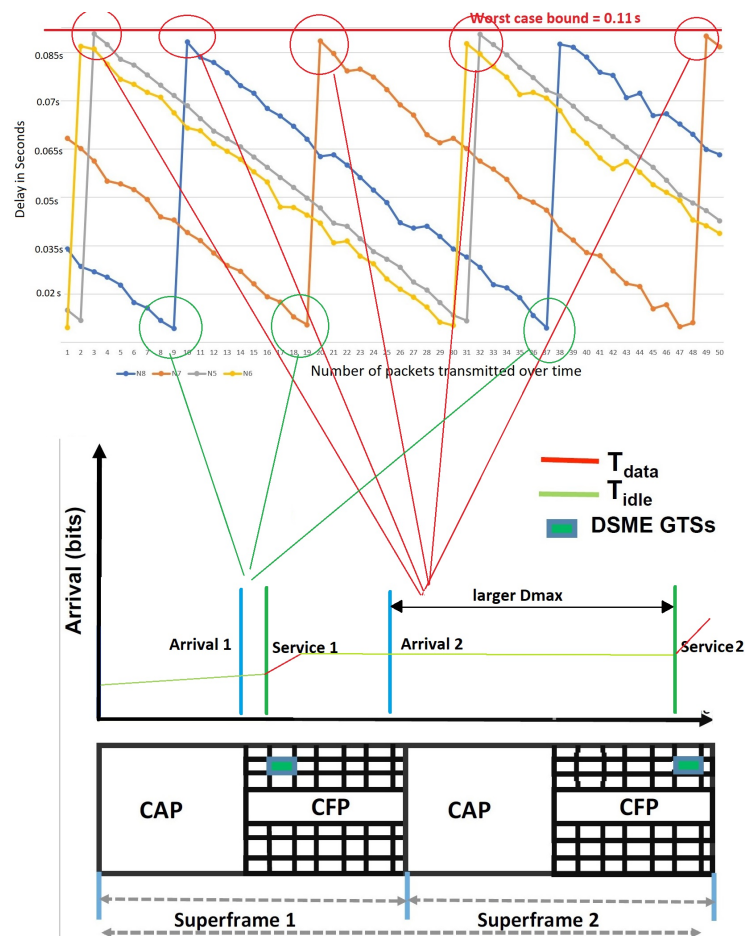

Figure 7.12: Comparison with the analytical modeling

7.4.5   Impact of braking force loss on the maximum delay:

Braking force is one of the common parameters in any car that deteriorates over time. This is a result of the loss of friction in the the clamping mechanism while actuating a brake. In a 100 % operational brake, the clamping load is assumed to act on all friction surfaces equally. The loss in this force is only generated when the wheel does not lock because the friction of a sliding wheel is much lower than a rotating one. The aim of this experiment is to learn the impact of the braking force over the maximum acceptable speed of the vehicle for various speeds.

In what follows, we do an analysis of several braking forces and the delay it takes to actuate and result in a scenario without a crash. In the best case scenario with a 100% braking force, the scenario allows a maximum bound of 360 ms delay to actuate for a 30 Km/h constant speed. Having set the 30 Km/h as the application prerequisite, we used this speed and delay sensitive network setting to assess the bound of the braking force required for this application to operate.

In this experiment we study the limits of this system by averaging the results for 10 trials for different braking forces and calculating the maximum acceptable delay for the vehicle to operate without a crash. The results are illustrated in Figure 7.13. As expected, the braking force and vehicle speed impose different constraints into the network delay. Decreased braking capacity, or higher speeds demand smaller communication delays to avoid the crash. At 30 Km/h, with a 50% braking capacity, the vehicle is unable to avoid hitting the car leaving the parking space, independently of the delay. This is the point where we reach the performance limit of the control system and vehicle dynamics.
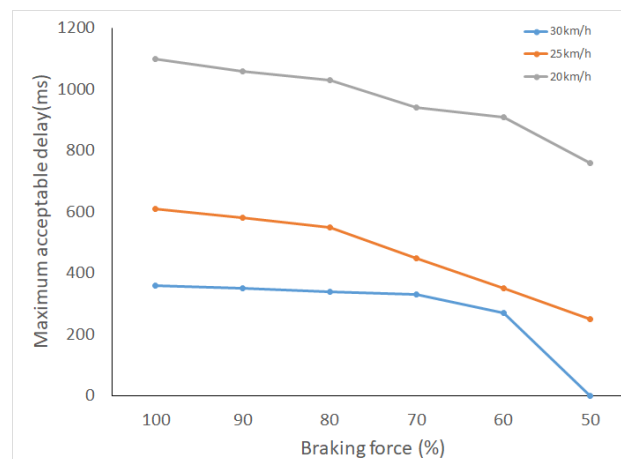
Figure 7.13: Maximum acceptable delay for crash prevention to the braking force applied

7.4.6   Impact of static scheduling and braking force on the crash rate

We carried out several trials for these application settings, and network MO/SO settings, to explore the performance limits of the Wireless ADAS scenario. In this experiment we fit lower latency obtaining static schedule for every DSME superframe and compare various delay sensitive setting and their limits. Figure 7.14 presents the communication's delay tolerances, for

different speeds (25 and 30 km/h) and braking capacities (100% to 50%), to prevent a crash, superimposed by the overall bounded delay at different network settings.
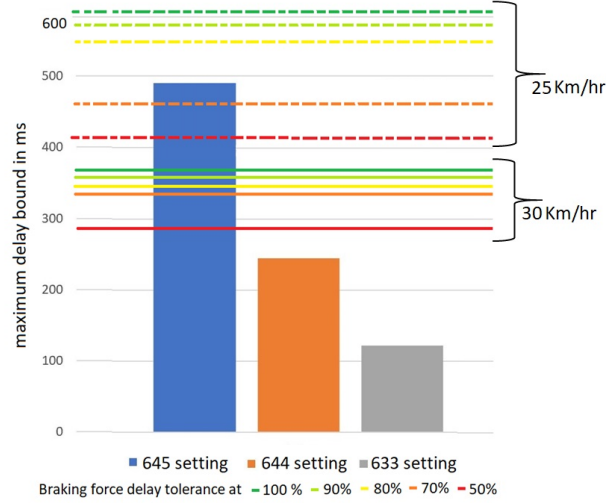
Figure 7.14: Impact of static scheduling and braking force on the crash rate

As observed in Figure 7.14, if the vehicle travels at 25 km/h in the parking lot, and has its braking capacity at 80%, it can still allow approximately 550 ms of delay in the ADAS communications, and therefore a (MO,SO) = (5,4) setting suffices. This is important considering the usage of a higher MO can support the allocation of additional superframes and support additional nodes, particularly, if CAP reduction is activated, increasing the scalability of the system. Thus finding this trade-of between delay and scalability, in parallel with speed and braking capacity can lead to increased efficiency and safety. When the braking capacity reduces to 70 or 60 % the maximum acceptable delay decreases steeply and can only be met by lower MO/SO network settings. This is also the case for a speed of 30Km/h, that even at 100% braking capacity, only (MO,SO) = (4,4) settings or lower, can meet the imposed delay requirement of approximately 360 ms. These results show us that for those settings, at the targeted speed for our scenario of 30 km/h, our system can still guarantee the safety of the vehicle even with its braking capacity impaired by 50%.

7.5   Summary

The contributions of this chapter are two-fold; first, we provide a novel architecture and implementation of a robotic-network co-simulator to render and simulate our intra-car based parking assist system. Secondly, we simulate the DSME stack and confirm it to be a possible contender for the use in intra-car communication scenarios. With retro-fitting systems in mind, we believe this architecture can complement the existing wired communication systems. We provide the worst-case limits from the network and the application perspectives, as well as, show that with a proper network design and configuration, the DSME stack can be used as a

valid MAC protocol that can facilitate the requirements of time-critical and safety-critical scenarios as the ones presented.

In this work, as these are time-critical scenarios, end to end delay is a very important metric to analyze and to try and reduce it as much as possible. With this in mind, we propose custom MO settings that can significantly reduce the delay in the network. From our findings, using a custom MO setting, we were able to obtain a significant lesser delay than the standard delay-sensitive DSME settings. Additionally, we also found that a dynamic slot allocation settings, due to its unpredictability is not a right fit to serve the safety-critical nature of this application scenario. Using static scheduling with our custom MO settings we were able to reduce the overall delay, as well as have a deterministic approach to assure a crash-free application.

As a future scope, we intend to extend this work to different urban traffic scenarios, as well as, scenarios where the vehicle has to operate at higher speeds and experiment them with different sensors that should better fit the requirements provided by these.

Chapter 8

Conclusions and Future Scope

8.1 Summary of the results

The domain of Wireless Sensor Networks has had a tremendous leap in world-wide adoption since the advent of communication standards that support low power and low rate communication. The new innovations in this field constantly expanding its horizons to several application domains that demand various levels of Quality of Service like latency, reliability, sustainability and energy efficiency. These demands are pushing forth this domain to newer standards and the next generation of Internet of Things (IoT) that support a variety of QoS.

The current state-of-the-art solutions aim at providing these QoS but are still immature and there is still open research to be done in terms of protocols, network architectures and newer techniques to support the QoS the upcoming IoT architectures may impose. There is an ardent need to re-design and enhance the existing architectures. In this dissertation we focused on four key pointers: (i) understanding the technology; (ii) learning the drawbacks and identifying the challenges; (iii) providing new solutions and methodologies and (iv) finally bridging the gap between academic solutions and the industrial need.

This thesis relies on the use of standard protocols, particularly the DSME MAC behavior of the IEEE 802.15.4e communication protocol. This dissertation also pushes forth in the direction of understanding the impact of this standard on real-time applications with a variant of Quality of Service demand. We provided a detailed state-of-the-art that enables us to understand the drawbacks of the protocol, open issues and some possible application scenarios. Some of the QoS aspects for the IEEE 802.15.4e that were addressed as a part of this dissertation are latency, reliability, energy consumption and scalability. Recalling the contributions (C1-C5) we stated in Chapter 1, we present them as follows:

Latency
This thesis addresses latency in the MAC sub-layer of IEEE 802.15.4e DSME network. The delay in the DSME network can be attributed to several aspects like scheduling, network infrastructure and routing. We carried out mathematical modeling (C1) to understand the bounds of the DSME network with respect to its time sensitive counter-parts like LLDN and TSCH. Comple-

menting DSME, we introduced a scheduling algorithm called "Symphony" (C2,C3) to schedule the transmissions stringently and reduce the overall delay of the network. We also introduced a traffic-differentiation mechanism under the Symphony architecture to facilitate priority based transmission without hindering the overall delay of the network. We also introduced a multi superframe tuning architecture that helps to alter the network infrastructure to support the incoming data traffic. We finally instantiated our models in a real-life intra-vehicle communication scenario (C5) where the theoretic findings based on latency were proved experimentally.

Scalability

Scalability is addressed in this thesis by the mechanism called "DynaMO" (C4) that can change the structure of the mutisuperframe and add additional timeslots based upon the demand of the network. DynaMO is a right suite for scalable dynamic networks, where the the resource demand can fluctuate constantly. The use of DynaMO was clearly demonstrated using the intra-car network system we designed by keeping DSME in the MAC sublayer.

Energy efficiency

In this dissertation, this specific QoS is addressed indirectly by the previous proposals. In line to the Symphony algorithm (C2, C3), we provide a detailed network architecture by which RPL (Routing protocol for lossy networks) can be integrated over the DSME MAC sub-layer. From our simulations, we confirm that this architecture can provide better results in terms both energy efficiency and reliability.

Robustness

Robustness is an aspect of the network on how well it can cope up to different circumstances like variant traffic flows, timeliness bounds and throughput. It is a daunting task for a network engineer to find the right setup for an application that may impose strict QoS demands.

In this dissertation, we propose a cross-layer architecture that can greatly improve the probability of successful transmissions through a link reliant routing layer on top of the deterministic MAC structure of DSME. This architecture concurrently works with symphony saving the network from any dire trade-offs. (C2)

Validation of thesis statement

In summary, we confirmed the initial hypothesis of this thesis, i.e., ***The augmentation of the IEEE 802.15.4e protocol with a set of QoS mechanisms that can effectively support the demands that futuristic IoT deployments may impose.***

In this dissertation, we proposed a set of mechanisms that effectively support the Quality of Service that Dynamic IoT networks demand. Furthermore, we incorporated our proposals over the intra-car communication use-case presenting ´valuable contributions to both the communication and application domains.

8.2   Future Research Directions

In this dissertation we analyzed the features of IEEE 802.15.4 and in particular the DSME MAC behaviour. We find that DSME is a prominent candidate to become a de-facto standard for IoT implementation. All throughout this thesis we provide several extensions and mechanisms that can further improve the standards efficiency and the overall Quality of Service. However, there is still room for development of an open-source implementation of this protocol for Commercially Off The Shelf WSN platforms (COTS), to validate the results over real WSN hardware. This dissertation focused on some of the prominent QoS issues in a wireless sensor network. However, there is enough room to address some of the aspects to bring these systems close to real-life implementation.

Though this dissertation indirectly provides a way to alleviate fault tolerance and increase the reliability of the network, there is much scope to improve the reliability by efficient techniques like blacklisting. From the network layer perspective, there can be several efficient routing techniques introduced to improve the reliability of a lossy mesh wireless network. One such future scope would be the a channel quality aware mesh network that can eliminate the channels with lesser link quality and then apply latency efficient algorithms like Symphony over it. This can reduce several trade-offs in terms of energy efficiency, reliability and robustness.

In our initial network analysis phase, we provided a tool that can facilitate a network engineer to understand the QoS aspects and the bound the MAC behavior can imply. However more ideal tools with other non-functional properties like reliability and energy efficiency must be integrated to enable an easier network setup and deployment even for a non expert.

The application scenario set in this dissertation was designed in such a way to push the paradigm of wireless sensor network towards the automobile industry domain. The existing networks in the automobile industry are completely wired. Despite being a drawback in terms of performance, maintenance and the possibility of single point failure the automobile industry has strongly adopted the wired network over the wireless. This decision can be attributed to several non functional properties like higher throughput, more reliability and possibilities to carry large amount of data which is not possible in a low rate wireless sensor network.

Upcoming technologies like the pre-processing cameras can detect the surroundings and deliver information with lesser payloads. The rise of techniques like machine learning also provide chances for low rate and low power wireless sensor network to be an alternate candidate for futuristic intra-car ADAS systems. There is also the possibility better sensor layouts for these systems to fully utilize the mesh capabilities of a DSME network.

This dissertation ventures into the aspects of safety when experimenting on the application domain. We set several experiments to obtain optimal network settings to ensure a crash proof ADAS system. Apart from safety, there is also room to look into the security aspects of the network. Efficient key generation techniques is one possible way to ensure better safety for these low power wireless networks.

One step closer...

A decade ago, IoT was just a buzzword but now it is shaping into reality. This gives the research community several challenges in building an efficient infrastructure to support it. In the paradigm of IoT every object is expected to be interconnected which will eventually change the way how we as human beings perceive the world and be an integrated part of it. Several communication protocols have the capability to become the de-facto standard for certain application domain. There is eventually going to be no winner as new standards pop up every single day. The winner for this race can only be determined by the QoS every standard can offer for the respective application domain. Almost ten years back ZigBee was in the research phase and only recently it is being adopted in some industries. DSME also has such potential to become a de-facto standard for futuristic low power IoT deployments.

All throughout this dissertation, we venture into challenges and contribute methods to alleviate them. This thesis can be a small step towards the improvement of Quality of Service and adoption of DSME networks for more real-time and time-critical application domains. We believe that the work carried out in this thesis can push the current state-of-the-art technology and bring one step closer towards the reality of Internet of Things.

References

[1] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, "Monitoring volcanic eruptions with a wireless sensor network," in *Wireless Sensor Networks, 2005. Proceeedings of the Second European Workshop on.* IEEE, 2005, pp. 108–120.

[2] K. Martinez, J. K. Hart, and R. Ong, "Environmental sensor networks," *Computer*, vol. 37, no. 8, pp. 50–56, 2004.

[3] H. Bogena, J. Huisman, C. Oberdörster, and H. Vereecken, "Evaluation of a low-cost soil water content sensor for wireless network applications," *Journal of Hydrology*, vol. 344, no. 1, pp. 32–42, 2007.

[4] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Wireless sensor networks for structural health monitoring," in *Proceedings of the 4th international conference on Embedded networked sensor systems.* ACM, 2006, pp. 427–428.

[5] T. Harms, S. Sedigh, and F. Bastianini, "Structural health monitoring of bridges using wireless sensor networks," *Instrumentation & Measurement Magazine, IEEE*, vol. 13, no. 6, pp. 14–18, 2010.

[6] M. Li and Y. Liu, "Underground coal mine monitoring with wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 5, no. 2, p. 10, 2009.

[7] M. Rodriguez-Sanchez, S. Borromeo, and J. Hernández-Tamames, "Wireless sensor networks for conservation and monitoring cultural assets," *Sensors Journal, IEEE*, vol. 11, no. 6, pp. 1382–1389, 2011.

[8] A. Milenković, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation," *Computer communications*, vol. 29, no. 13, pp. 2521–2533, 2006.

[9] J. H. Kurunathan, "Study and overview on wban under ieee 802.15. 6," *U. Porto Journal of Engineering*, vol. 1, no. 1, pp. 11–21, 2015.

[10] E. Jovanov, A. Milenkovic, C. Otto, and P. C. De Groen, "A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation," *Journal of NeuroEngineering and rehabilitation*, vol. 2, no. 1, p. 6, 2005.

[11] B. P. Lo, S. Thiemjarus, R. King, and G.-Z. Yang, *Body sensor network–a wireless sensor platform for pervasive healthcare monitoring.* na, 2005.

[12] S. N. Ramli, R. Ahmad, M. F. Abdollah, and E. Dutkiewicz, "A biometric-based security for data authentication in wireless body area network (wban)," in *2013 15th International Conference on Advanced Communications Technology (ICACT).* IEEE, 2013, pp. 998–1001.

[13] M. Hefeeda and M. Bagheri, "Wireless sensor networks for early detection of forest fires," in *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE International Conference on*. IEEE, 2007, pp. 1–6.

[14] M. Suzuki, S. Saruwatari, N. Kurata, and H. Morikawa, "A high-density earthquake monitoring system using wireless sensor networks," in *Proceedings of the 5th international conference on Embedded networked sensor systems*.   ACM, 2007, pp. 373–374.

[15] A. Ko and H. Y. Lau, "Robot assisted emergency search and rescue system with a wireless sensor network," *International Journal of Advanced Science and Technology*, vol. 3, pp. 69–78, 2009.

[16] E. T. Alotaibi, S. S. Alqefari, and A. Koubaa, "Lsar: Multi-uav collaboration for search and rescue missions," *IEEE Access*, vol. 7, pp. 55 817–55 832, 2019.

[17] K. S. Low, W. N. N. Win, and M. J. Er, "Wireless sensor networks for industrial environments," in *Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on*, vol. 2.   IEEE, 2005, pp. 271–276.

[18] M. R. Akhondi, A. Talevski, S. Carlsen, and S. Petersen, "Applications of wireless sensor networks in the oil, gas and resources industries," in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*.   IEEE, 2010, pp. 941–948.

[19] I. Johnstone, J. Nicholson, B. Shehzad, and J. Slipp, "Experiences from a wireless sensor network deployment in a petroleum environment," in *Proceedings of the 2007 international conference on Wireless communications and mobile computing*.   ACM, 2007, pp. 382–387.

[20] K. S. Kwak, S. Ullah, and N. Ullah, "An overview of ieee 802.15. 6 standard," in *2010 3rd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL 2010)*.   IEEE, 2010, pp. 1–6.

[21] "Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (wpans)," *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, pp. 1–320, Sept 2006.

[22] S. Farahani, *ZigBee wireless networks and transceivers*.   Newnes, 2011.

[23] J. Song, S. Han, A. Mok, D. Chen, M. Lucas, M. Nixon, and W. Pratt, "Wirelesshart: Applying wireless technology in real-time industrial process control," in *2008 IEEE Real-Time and Embedded Technology and Applications Symposium*.   IEEE, 2008, pp. 377–386.

[24] H. Kurunathan, R. Severino, A. Koubaa, and E. Tovar, "Towards worst-case bounds analysis of the ieee 802.15. 4e," in *2016 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*.   IEEE, 2016, p. 51.

[25] H. Kurunathan, "Ieee 802.15. 4e in a nutshell: Survey and performance evaluation," *IEEE Communications Surveys & Tutorials*, 2018.

[26] H. Kurunathan, R. Severino, A. Koubâa, and E. Tovar, "Symphony: routing aware scheduling for dsme networks," *ACM Sigbed Review*, vol. 16, no. 4, pp. 26–31, 2020.

[27] H. Kurunathan, R. Severino, A. Koubaa, and E. Tovar, "Dynamo: dynamically tuning dsme networks," *ACM Sigbed Review*, vol. 16, no. 4, pp. 8–13, 2020.

[28] H. Kurunathan, R. Severino, A. Koubâa, and E. Tovar, "An efficient approach to multisuperframe tuning for dsme networks: Poster abstract," in *Proceedings of the 17th ACM/IEEE International Conference on Information Processing in Sensor Networks*, ser. IPSN '18. IEEE Press, 2018, p. 162–163. [Online]. Available: https://doi.org/10.1109/IPSN.2018.00044

[29] H. Kurunathan, R. Severino, and E. Tovar, "Wicar - simulating towards the wireless car," in *2020 International Workshop on Dependable Smart Embedded Cyber-Physical Systems and Systems-of-Systems (DECSoS)*. Springer, 2020.

[30] C. Lu, B. M. Blum, T. F. Abdelzaher, J. A. Stankovic, and T. He, "Rap: A real-time communication architecture for large-scale wireless sensor networks," in *Real-Time and Embedded Technology and Applications Symposium, 2002. Proceedings. Eighth IEEE*. IEEE, 2002, pp. 55–66.

[31] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3. IEEE, 2002, pp. 1567–1576.

[32] T. Van Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*. ACM, 2003, pp. 171–180.

[33] Z. Alliance *et al.*, "Zigbee specification," 2006.

[34] T. Winter, "Rpl: Ipv6 routing protocol for low-power and lossy networks," 2012.

[35] "Ieee standard for local and metropolitan area networks–part 15.4: Low-rate wireless personal area networks (lr-wpans) amendment 1: Mac sublayer," *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–225, April 2012.

[36] "Ieee standard for smart transducer interface for sensors and actuators–transducers to radio frequency identification (rfid) systems communication protocols and transducer electronic data sheet formats," *IEEE Std 1451.7-2010*, pp. 1–99, June 2010.

[37] D. Williams, "The strategic implications of wal-mart's rfid mandate," *Directions Magazine*, vol. 28, 2004.

[38] Y. Xiao, X. Shen, B. Sun, and L. Cai, "Security and privacy in rfid and applications in telemedicine," *Communications Magazine, IEEE*, vol. 44, no. 4, pp. 64–72, 2006.

[39] Y. Li, Y.-Q. Song, R. Schott, Z. Wang, and Y. Sun, "Impact of link unreliability and asymmetry on the quality of connectivity in large-scale sensor networks," *Sensors*, vol. 8, no. 10, pp. 6674–6691, 2008.

[40] L. Hou and N. W. Bergmann, "System requirements for industrial wireless sensor networks," in *Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on*. IEEE, 2010, pp. 1–8.

[41] E. Karulf, "Body area networks (ban)," 2008.

[42] P. Dutta, J. Hui, J. Jeong, S. Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse, and D. Culler, "Trio: enabling sustainable and scalable outdoor wireless sensor network deployments," in *Proceedings of the 5th international conference on Information processing in sensor networks.* ACM, 2006, pp. 407–415.

[43] J. Suhonen, M. Kohvakka, V. Kaseva, T. D. Hämäläinen, and M. Hännikäinen, *Low-power wireless sensor networks: protocols, services and applications.* Springer Science & Business Media, 2012.

[44] CISCO, "20 myths of wi-fi interference: Dispel myths to gain high- performing and reliable wireless," http://www.cisco.com/c/en/us/products/collateral/wireless/spectrum-expert-wi-fi.

[45] J. W. Smith, "Radio communication system using synchronous frequency hopping transmissions," Jul. 18 1989, uS Patent 4,850,036.

[46] S. Williams, L. T. Parker, and A. M. Howard, "Terrain reconstruction of glacial surfaces: Robotic surveying techniques," *Robotics & Automation Magazine, IEEE*, vol. 19, no. 4, pp. 59–71, 2012.

[47] M. M. Hanna, A. Buck, and R. Smith, "Fuzzy petri nets with neural networks to model products quality from a cnc-milling machining centre," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 26, no. 5, pp. 638–645, 1996.

[48] S. Petersen, P. Doyle, S. Vatland, C. S. Aasland, T. M. Andersen, and D. Sjong, "Requirements, drivers and analysis of wireless sensor network solutions for the oil & gas industry," in *Emerging Technologies and Factory Automation, 2007. ETFA. IEEE Conference on.* IEEE, 2007, pp. 219–226.

[49] N. Pereira, S. Tennina, and E. Tovar, "Building a microscope for the data center," in *Wireless Algorithms, Systems, and Applications.* Springer, 2012, pp. 619–630.

[50] D. Chen, M. Nixon, S. Han, A. K. Mok, and X. Zhu, "Wirelesshart and ieee 802.15. 4e," in *2014 IEEE International conference on industrial technology (ICIT).* IEEE, 2014, pp. 760–765.

[51] C. De Dominicis, P. Ferrari, A. Flammini, E. Sisinni, M. Bertocco, G. Giorgi, C. Narduzzi, and F. Tramarin, "Investigating wirelesshart coexistence issues through a specifically designed simulator," in *2009 IEEE Instrumentation and Measurement Technology Conference.* IEEE, 2009, pp. 1085–1090.

[52] M. S. Costa and J. Amaral, "Analysis of wireless industrial automation standards: Isa-100.11 a and wirelesshart," *InTech Magazine*, 2012.

[53] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of lorawan," *IEEE Communications magazine*, vol. 55, no. 9, pp. 34–40, 2017.

[54] A. Lavric and A. I. Petrariu, "Lorawan communication protocol: The new era of iot," in *2018 International Conference on Development and Application Systems (DAS).* IEEE, 2018, pp. 74–77.

[55] P. Kinney *et al.*, "Zigbee technology: Wireless control that simply works," in *Communications design conference*, vol. 2, 2003, pp. 1–7.

[56] J. Leal, A. Cunha, M. Alves, and A. Koubaa, "On a ieee 802.15. 4/zigbee to ieee 802.11 gateway for the art-wise architecture," in *2007 IEEE Conference on Emerging Technologies and Factory Automation (EFTA 2007)*.   IEEE, 2007, pp. 1388–1391.

[57] W.-C. Jeong and J. Lee, "Performance evaluation of ieee 802.15. 4e dsme mac protocol for wireless sensor networks," in *Enabling Technologies for Smartphone and Internet of Things (ETSIoT), 2012 First IEEE Workshop on*.   IEEE, 2012, pp. 7–12.

[58] I. Juc, O. Alphand, R. Guizzetti, M. Favre, and A. Duda, "Energy consumption and performance of ieee 802.15.4e tsch and dsme," in *2016 IEEE Wireless Communications and Networking Conference*, April 2016, pp. 1–7.

[59] J. Lee and W.-C. Jeong, "Performance analysis of ieee 802.15. 4e dsme mac protocol under wlan interference," in *ICT Convergence (ICTC), 2012 International Conference on*.   IEEE, 2012, pp. 741–746.

[60] C. Ouanteur, D. Aïssani, L. Bouallouche-Medjkoune, M. Yazid, and H. Castel-Taleb, "Modeling and performance evaluation of the ieee 802.15. 4e lldn mechanism designed for industrial applications in wsns," *Wireless Networks*, vol. 23, no. 5, pp. 1343–1358, 2017.

[61] A. Berger, A. Entinger, A. Potsch, and A. Springer, "Improving ieee 802.15. 4e lldn performance by relaying and extension of combinatorial testing," in *Emerging Technology and Factory Automation (ETFA), 2014 IEEE*.   IEEE, 2014, pp. 1–4.

[62] T. Paso, J. Haapola, and J. Iinatti, "Feasibility study of ieee 802.15. 4e dsme utilizing ir-uwb and s-aloha," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*.   IEEE, 2013, pp. 1863–1867.

[63] D. De Guglielmo, A. Seghetti, G. Anastasi, and M. Conti, "A performance analysis of the network formation process in ieee 802.15. 4e tsch wireless sensor/actuator networks," in *Computers and Communication (ISCC), 2014 IEEE Symposium on*.   IEEE, 2014, pp. 1–6.

[64] J. Zhou, A. E. Xhafa, R. Vedantham, R. Nuzzaci, A. Kandhalu, and X. Lu, "Comparison of ieee 802.15. 4e mac features," in *2014 IEEE World Forum on Internet of Things (WF-IoT)*. IEEE, 2014, pp. 203–207.

[65] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level sensor network simulation with cooja," in *Proceedings. 2006 31st IEEE Conference on Local Computer Networks*.   IEEE, 2006, pp. 641–648.

[66] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki-a lightweight and flexible operating system for tiny networked sensors," in *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*.   IEEE, 2004, pp. 455–462.

[67] G. Alderisi, G. Patti, O. Mirabella, and L. L. Bello, "Simulative assessments of the ieee 802.15. 4e dsme and tsch in realistic process automation scenarios," in *Industrial Informatics (INDIN), 2015 IEEE 13th International Conference on*.   IEEE, 2015, pp. 948–955.

[68] J.-Y. Le Boudec and P. Thiran, *Network calculus: a theory of deterministic queuing systems for the internet*.   Springer Science & Business Media, 2001, vol. 2050.

[69] P. Du and G. Roussos, "Adaptive time slotted channel hopping for wireless sensor networks," in *Computer Science and Electronic Engineering Conference (CEEC), 2012 4th*. IEEE, 2012, pp. 29–34.

[70] T. Watteyne, X. Vilajosana, B. Kerkez, F. Chraim, K. Weekly, Q. Wang, S. Glaser, and K. Pister, "Openwsn: a standards-based low-power wireless development environment," *Transactions on Emerging Telecommunications Technologies*, vol. 23, no. 5, pp. 480–493, 2012.

[71] X. Liu, X. Li, S. Su, Z. Fan, and G. Wang, "Enhanced fast association for 802.15. 4e-2012 dsme mac protocol," in *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering*.   Atlantis Press, 2013.

[72] G. Patti, G. Alderisi, and L. L. Bello, "Introducing multi-level communication in the ieee 802.15. 4e protocol: the multichannel-lldn," in *Emerging Technology and Factory Automation (ETFA), 2014 IEEE*.   IEEE, 2014, pp. 1–8.

[73] L. Dariz, M. Ruggeri, and G. Malaguti, "A proposal for enhancement towards bidirectional quasi-deterministic communications using ieee 802.15. 4," in *2013 21st Telecommunications Forum Telfor (TELFOR)*, 2013.

[74] Y. Al-Nidawi, H. Yahya, and A. Kemp, "Tackling mobility in low latency deterministic multihop ieee 802.15. 4e sensor network," 2015.

[75] M. Taneja, "A framework to support real-time applications over ieee802. 15.4 dsme," in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on*.   IEEE, 2015, pp. 1–6.

[76] M. Anwar and X. Yuanqing, "Ieee 802.15. 4e lldn: Superframe configuration for networked control systems," in *Control Conference (CCC), 2014 33rd Chinese*.   IEEE, 2014, pp. 5568–5573.

[77] S. Capone, R. Brama, F. Ricciato, G. Boggia, and A. Malvasi, "Modeling and simulation of energy efficient enhancements for ieee 802.15. 4e dsme," in *Wireless Telecommunications Symposium (WTS), 2014*.   IEEE, 2014, pp. 1–6.

[78] M. R. Palattella, P. Thubert, X. Vilajosana, T. Watteyne, Q. Wang, and T. Engel, "6tisch wireless industrial networks: Determinism meets ipv6," in *Internet of Things*.   Springer, 2014, pp. 111–141.

[79] P. K. Sahoo, S. R. Pattanaik, and S.-L. Wu, "A reliable data transmission model for ieee 802.15. 4e enabled wireless sensor network under wifi interference," *Sensors*, vol. 17, no. 6, p. 1320, 2017.

[80] G. Mulligan, "The 6lowpan architecture," in *Proceedings of the 4th workshop on Embedded networked sensors*.   ACM, 2007, pp. 78–82.

[81] R.-H. Hwang, C.-C. Wang, and W.-B. Wang, "A distributed scheduling algorithm for ieee 802.15. 4e wireless sensor networks," *Computer Standards & Interfaces*, vol. 52, pp. 63–70, 2017.

[82] K. T. Kim, H. Kim, H. Park, and S.-T. Kim, "An industrial iot mac protocol based on ieee 802.15. 4e tsch for a large-scale network," in *Advanced Communication Technology (ICACT), 2017 19th International Conference on*.   IEEE, 2017, pp. 721–724.

[83] T. Watteyne, J. Weiss, L. Doherty, and J. Simon, "Industrial ieee802. 15.4 e networks: Performance and trade-offs," in *Communications (ICC), 2015 IEEE International Conference on*.   IEEE, 2015, pp. 604–609.

[84] M. R. Palattella, N. Accettura, M. Dohler, L. A. Grieco, and G. Boggia, "Traffic aware scheduling algorithm for reliable low-power multi-hop ieee 802.15. 4e networks," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on*. IEEE, 2012, pp. 327–332.

[85] Y.-S. Lee and S.-H. Chung, "An efficient distributed scheduling algorithm for mobility support in ieee 802.15. 4e dsme-based industrial wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 12, no. 2, p. 9837625, 2016.

[86] M. Subramanya Bhat, D. Shwetha, and J. Devaraju, "A performance study of proactive, reactive and hybrid routing protocols using qualnet simulator," *International Journal of Computer Applications*, vol. 28, no. 5, pp. 10–17, 2011.

[87] H. Kwang-il and N. SungWook, "Analysis and enhancement of ieee 802.15.4e dsme beacon scheduling model," in *Journal of Applied Mathematics, vol. 2014, Article ID 934610*, April 2014, pp. 1–15.

[88] N. Accettura, E. Vogli, M. R. Palattella, L. A. Grieco, G. Boggia, and M. Dohler, "Decentralized traffic aware scheduling in 6tisch networks: Design and experimental evaluation," *Internet of Things Journal, IEEE*, vol. 2, no. 6, pp. 455–470, 2015.

[89] O. WSN, "Orchestra protocol stack," 2016, https://github.com/EIT-ICT-RICH.

[90] C.-F. Shih, A. E. Xhafa, and J. Zhou, "Practical frequency hopping sequence design for interference avoidance in 802.15. 4e tsch networks," in *Communications (ICC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 6494–6499.

[91] J. Eriksson, F. Österlind, N. Finne, N. Tsiftes, A. Dunkels, T. Voigt, R. Sauter, and P. J. Marrón, "Cooja/mspsim: interoperability testing for wireless sensor networks," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009, p. 27.

[92] D. De Guglielmo, B. Al Nahas, S. Duquennoy, T. Voigt, and G. Anastasi, "Analysis and experimental evaluation of ieee 802.15. 4e tsch csma-ca algorithm," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1573–1588, 2017.

[93] M. Mohammad, M. Doddavenkatappa, and M. C. Chan, "Improving performance of synchronous transmission-based protocols using capture effect over multichannels," *ACM Transactions on Sensor Networks (TOSN)*, vol. 13, no. 2, p. 10, 2017.

[94] G. Stamatescu, D. Popescu, and I. Stamatescu, "Modeling for deployment techniques for intra-car wireless sensor networks," in *2014 18th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, 2014, pp. 501–505.

[95] J.-R. Lin, T. Talty, and O. K. Tonguz, "Feasibility of safety applications based on intra-car wireless sensor networks: A case study," in *2011 IEEE Vehicular Technology Conference (VTC Fall)*. IEEE, 2011, pp. 1–5.

[96] M. Hashemi, W. Si, M. Laifenfeld, D. Starobinski, and A. Trachtenberg, "Intra-car multi-hop wireless sensor networking: a case study," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 183–191, 2014.

[97]   M. Hashemi, "Intra-car multihop wireless sensor networking: a case study," *2013 IEEE 77th Vehicular Technology Conference*, pp. 1–5, 2013.

[98]   A. D. G. Reddy and B. Ramkumar, "Simulation studies on zigbee network for in-vehicle wireless communications," in *2014 International Conference on Computer Communication and Informatics*.   IEEE, 2014, pp. 1–6.

[99]   H.-M. Tsai, O. K. Tonguz, C. Saraydar, T. Talty, M. Ames, and A. Macdonald, "Zigbee-based intra-car wireless sensor networks: a case study," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 67–77, 2007.

[100] H.-M. Tsai, C. Saraydar, T. Talty, M. Ames, A. Macdonald, and O. K. Tonguz, "Zigbee-based intra-car wireless sensor network," in *2007 IEEE International Conference on Communications*.   IEEE, 2007, pp. 3965–3971.

[101] J.-Y. Le Boudec, "Application of network calculus to guaranteed service networks," *IEEE Transactions on Information theory*, vol. 44, no. 3, pp. 1087–1096, 1998.

[102] A. Koubâa, M. Alves, and E. Tovar, "Energy and delay trade-off of the gts allocation mechanism in ieee 802.15. 4 for wireless sensor networks," *International Journal of Communication Systems*, vol. 20, no. 7, pp. 791–808, 2007.

[103] A. Koubaa, M. Alves, and E. Tovar, "Gts allocation analysis in ieee 802.15.4 for real-time wireless sensor networks," in *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, April 2006, pp. 8 pp.–.

[104] A. Cunha, A. Koubaa, R. Severino, and M. Alves, "Open-zb: an open-source implementation of the ieee 802.15. 4/zigbee protocol stack on tinyos," in *2007 IEEE International Conference on Mobile Adhoc and Sensor Systems*.   IEEE, 2007, pp. 1–12.

[105] A. Ahmad and Z. Hanzálek, "Distributed real time tdma scheduling algorithm for tree topology wsns," in *IFAC-PapersOnLine*, vol. 50, no. 1.   Elsevier, 2017, pp. 5926–5933.

[106] P. Jurčík and Z. Hanzálek, "Simulation study of energy efficient scheduling for ieee 802.15. 4/zigbee cluster-tree wireless sensor networks with time-bounded data flows," in *Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on*.   IEEE, 2010, pp. 1–8.

[107] Q. Weisheng, "N queens problem," *Journal of Mathematics*, vol. 2, p. 002, 1986.

[108] I. Rivin, I. Vardi, and P. Zimmermann, "The n-queens problem," *The American Mathematical Monthly*, vol. 101, no. 7, pp. 629–639, 1994.

[109] K. Ueda, "Making exhaustive search programs deterministic," in *International Conference on Logic Programming*.   Springer, 1986, pp. 270–282.

[110] P. J. Van Laarhoven and E. H. Aarts, "Simulated annealing," pp. 7–15, 1987.

[111] J. Lee, W.-C. Jeong, and B.-C. Choi, "A multi-channel timeslot scheduling algorithm for link recovery in wireless multi-hop sensor networks," in *Information and Communication Technology Convergence (ICTC), 2016 International Conference on*.   IEEE, 2016, pp. 871–876.

[112] IETF, "Routing over low power and lossy networks (roll)„" 2004, https://datatracker.ietf.org/wg/roll/charter/.

[113] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of ipv6 packets over ieee 802.15. 4 networks," Tech. Rep., 2007.

[114] J. W. Hui and D. E. Culler, "Ip is dead, long live ip for wireless sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems.* ACM, 2008, pp. 15–28.

[115] A. Dunkels, O. Schmidt, N. Finne, J. Eriksson, F. Österlind, and N. T. M. Durvy, "The contiki os: The operating system for the internet of things," *Online], at http://www. contikios. org*, 2011.

[116] V. Mayalarp, N. Limpaswadpaisarn, T. Poombansao, and S. Kittipiyakul, "Wireless mesh networking with xbee," in *2nd ECTI-Conference on Application Research and Development (ECTI-CARD 2010), Pattaya, Chonburi, Thailand*, 2010, pp. 10–12.

[117] G. Stamatescu, C. Chiţu, C. Vasile, I. Stamatescu, D. Popescu, and V. Sgârciu, "Analytical and experimental sensor node energy modeling in ambient monitoring," in *Industrial Electronics and Applications (ICIEA), 2014 IEEE 9th Conference on.* IEEE, 2014, pp. 1615–1620.

[118] F. Kauer, M. Köstler, T. Lübkert, and V. Turau, "Opendsme-a portable framework for reliable wireless sensor and actuator networks," in *2017 International Conference on Networked Systems (NetSys).* IEEE, 2017, pp. 1–2.

[119] M. Domingo-Prieto, T. Chang, X. Vilajosana, and T. Watteyne, "Distributed pid-based scheduling for 6tisch networks," *IEEE Communications Letters*, vol. 20, no. 5, pp. 1006–1009, 2016.

[120] S. Capone, R. Brama, F. Ricciato, G. Boggia, and A. Malvasi, "Modeling and simulation of energy efficient enhancements for ieee 802.15.4e dsme," in *2014 Wireless Telecommunications Symposium*, April 2014, pp. 1–6.

[121] S. Duquennoy, B. Al Nahas, O. Landsiedel, and T. Watteyne, "Orchestra: Robust mesh networks through autonomously scheduled tsch," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems.* ACM, 2015, pp. 337–350.

[122] D. Dujovne, T. Watteyne, X. Vilajosana, and P. Thubert, "6tisch: deterministic ip-enabled industrial internet (of things)," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 36–41, 2014.

[123] O. Gaddour and A. Koubâa, "Rpl in a nutshell: A survey," *Computer Networks*, vol. 56, no. 14, pp. 3163 – 3178, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128612002423

[124] M. Qasem, H. Altawssi, M. B. Yassien, and A. Al-Dubai, "Performance evaluation of rpl objective functions," in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, Oct 2015, pp. 1606–1613.

[125] H. Kurunathan, R. Severino, A. Koubâa, and E. Tovar, "Rpl over dsme: A technical report," CISTER-Research Centre in Realtime and Embedded Computing Systems, Tech. Rep., 2018.

[126] P. Thubert, "Objective function zero for the routing protocol for low-power and lossy networks (rpl)," 2012.

[127] M. Ghazi Amor, A. Koubâa, E. Tovar, and M. Khalgui, "Cyber-of: An adaptive cyber-physical objective function for smart cities applications," in *28th Euromicro Conference on Real-Time Systems*, 2016.

[128] P. Park, C. Fischione, and K. H. Johansson, "Performance analysis of gts allocation in beacon enabled ieee 802.15. 4," in *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*.   IEEE, 2009, pp. 1–9.

[129] H. Kurunathan, "Opendsme support file," https://github.com/harrisonkurunathan/throughputnedfile, 2019.

[130] M. I. Khan, W. N. Gansterer, and G. Haring, "Static vs. mobile sink: The influence of basic parameters on energy efficiency in wireless sensor networks," *Computer communications*, vol. 36, no. 9, pp. 965–978, 2013.

[131] D. Puccinelli and M. Haenggi, "Wireless sensor networks: applications and challenges of ubiquitous sensing," *IEEE Circuits and systems magazine*, vol. 5, no. 3, pp. 19–31, 2005.

[132] V. Ç. Güngör and G. P. Hancke, *Industrial wireless sensor networks: Applications, protocols, and standards*.   Crc Press, 2013.

[133] D. Yang, J. Ma, Y. Xu, and M. Gidlund, "Safe-wirelesshart: A novel framework enabling safety-critical applications over industrial wsns," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3513–3523, 2018.

[134] T. ElBatt, C. Saraydar, M. Ames, and T. Talty, "Potential for intra-vehicle wireless automotive sensor networks," in *2006 IEEE Sarnoff Symposium*, March 2006, pp. 1–4.

[135] O. K. Tonguz, H.-M. Tsai, C. Saraydar, T. Talty, and A. Macdonald, "Intra-car wireless sensor networks using rfid: Opportunities and challenges," in *2007 Mobile Networking for Vehicular Environments*.   IEEE, 2007, pp. 43–48.

[136] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2.   Kobe, Japan, 2009, p. 5.

[137] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 3.   IEEE, 2004, pp. 2149–2154.

[138] S. Ingle and M. Phute, "Tesla autopilot: semi autonomous driving, an uptick for future autonomy," *International Research Journal of Engineering and Technology*, vol. 3, no. 9, 2016.

[139] F. Kauer, M. Köstler, T. Lübkert, and V. Turau, "Opendsme - a portable framework for reliable wireless sensor and actuator networks (demonstration)," in *Proceedings of the 3rd International Conference on Networked Systems (NetSys 2017)*, Mar. 2017.

[140] D. Ramos, L. Oliveira, L. Almeida, and U. Moreno, "Network interference on cooperative mobile robots consensus," in *Robot 2015: Second Iberian Robotics Conference*. Springer, 2016, pp. 651–663.

[141] W. Li, X. Zhang, and H. Li, "Co-simulation platforms for co-design of networked control systems: An overview," *Control Engineering Practice*, vol. 23, pp. 44–56, 2014.

[142] M. Kudelski, L. M. Gambardella, and G. A. Di Caro, "Robonetsim: An integrated framework for multi-robot and network simulation," *Robotics and Autonomous Systems*, vol. 61, no. 5, pp. 483–496, 2013.

[143] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle *et al.*, "Argos: a modular, parallel, multi-engine simulator for multi-robot systems," *Swarm intelligence*, vol. 6, no. 4, pp. 271–295, 2012.

[144] T. Issariyakul and E. Hossain, "Introduction to network simulator 2 (ns2)," in *Introduction to network simulator NS2*. Springer, 2009, pp. 1–18.

[145] G. Carneiro, "Ns-3: Network simulator 3," in *UTM Lab Meeting April*, vol. 20, 2010, pp. 4–5.

[146] T. H. Labella, I. Dietrich, and F. Dressler, "Baraka: A hybrid simulator of sanets," in *2007 2nd International Conference on Communication Systems Software and Middleware*. IEEE, 2007, pp. 1–8.

[147] R. Smith *et al.*, "Open dynamics engine," 2005.

[148] B. Vieira, R. Severino, A. Koubaa, and E. Tovar, "Towards a realistic simulation framework for vehicular platooning applications." in *22nd IEEE International Symposium on Real-Time Computing (ISORC 2019)*. Institute of Electrical and Electronics Engineers, 2019.

[149] R. Severino, E. Vasconcelos Filho, B. Vieira, A. Koubaa, and E. Tovar, "COPADRIVe - a realistic simulation framework for cooperative autonomous driving applications," in *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE) (IEEE ICCVE 2019)*, Graz, Austria, Nov. 2019.

[150] B. Blaszczyszyn and M. K. Karray, "Quality of service in wireless cellular networks subject to log-normal shadowing," *IEEE transactions on communications*, vol. 61, no. 2, pp. 781–791, 2012.

[151] F. Lassabe, P. Canalda, P. Chatonnay, F. Spies, and O. Baala, "A friis-based calibrated model for wifi terminals positioning," in *Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks*. IEEE, 2005, pp. 382–387.

[152] C. U. Bas and S. C. Ergen, "Ultra-wideband channel model for intra-vehicular wireless sensor networks beneath the chassis: From statistical model to simulations," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pp. 14–25, 2013.