Load Balancing and Context Aware Enhancements for RPL Routed Internet of Things

Mamoun Qasem

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS OF EDINBURGH NAPIER UNIVERSITY, FOR THE AWARD

DOCTOR OF PHILOSOPHY

Aug 2018

Abstract

Internet of Things (IoT) has been paving the way for a plethora of potential applications, which becomes more spatial and demanding. The goal of this work is to optimise the performance within the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) in the network layer.

RPL still suffers from unbalanced load traffic among the candidate parents. Consequently, the overloaded parent node drains its energy much faster than other candidate parent nodes. This may lead to an early disconnection of a part of the network topology and affect the overall network reliability. To solve this problem, a new objective function (OF) has been proposed to usher better load balancing among the bottleneck candidate parents, and keep the overloaded nodes lifetime thriving to longer survival.

Moreover, several IoT applications have antagonistic requirements but pertinent, which results in a greater risk of affecting the network reliability, especially within the emergency scenarios. With the presence of this challenging issue, the current standardised RPL OFs cannot sufficiently fulfil the antagonistic needs of Low-power and Lossy Networks (LLNs) applications. In response to the above issues, a context adaptive OF has been proposed to facilitate exchanging the synergy information between the application and network layers. Thus, the impact of the antagonistic requirements based on context parameters will be mitigated via rationalizing the selection decision of the routing path towards the root node.

We implemented the proposed protocol and verified all our findings through excessive measurements via simulations and a realistic deployment using a real testbed of a multi-hop LLNs motes. The results proved the superiority of our solution over the existing ones with respect to end-to-end delay, packet delivery ratio and network lifetime. Our contribution has been accepted initially to be adopted within the standard body Internet Engineering Task Force (IETF).

Declaration

I hereby declare that the work presented in this thesis was solely carried out by myself, except where due acknowledgement or specific reference is made, and that it has not been submitted for any other degree professional qualification.

Mamoun Qasem

Aug 2018

To my parents,

My lovely wife and my children

Acknowledgment

First and foremost, I thank Allah Almighty for giving me the strength and the means to complete my doctoral degree.

This wonderful and challenging journey would not have been possible without the guidance and support of several individuals. I owe my deepest gratitude to my supervisor Professor Ahmed Al-Dubai for his continuous support and guidance, which steered me in to the right direction and made my doctoral journey a worthwhile and pleasant experience. Special thanks goes to my second supervisor Dr Imed Romdhani for his support and valuable comments.

I am also thankful to my friends and colleagues who supported me throughout the up and down moments. I would also like to acknowledge the funding support of the School of Computing, Edinburgh Napier University.

I would like to express my deepest gratitude to my beloved parents for their support, love, prayers, patience, and sacrifices. It is to them that I owe all the success in my life. I would like also to extend my sincere thanks to the members of my family for their support and encouragements. Big ups for my lovely daughters and son. Finally, I accomplish this acknowledgment by thanking my wife Sara for giving me more than enough love, understanding, and strength necessary to close this part of the life journey.

Contents

Declaration **II**

Acknowledgment **IV**

Contents **I**

List of Figures **V**

List of Tables **VII**

List of Abbreviations **XI**

1 Introduction **1**

1.1 Shallow Foundation....1

1.2 Motivation....2

1.3 Problem Definition....3

1.4 Aim and Objectives....4

1.5 Thesis Contributions....5

1.5.1 Key Outcomes....5

1.5.2 List of Publications during this research....6

1.6 Thesis Structure....7

2 Background **9**

2.1 Introduction....9

2.2 Historical view of WSNs....10

2.3 Low-power and Lossy Networks....10

2.4 Internet of Things....12

2.5 IoT Applications....15

2.6 Overview standardisation efforts on LLNs 17
2.7 IEEE 802.15.4 Protocol 18
2.8 IEEE 802.15.4e (TSCH) 20
2.9 6TOP 21
2.10 6LoWPAN 21
2.11 IPv6 Routing Protocol for Low-Power and Lossy Networks 23
2.11.1 RPL Control Packets 24
2.11.2 RANK and DODAG Construction 26
2.11.3 Communication Patterns 28
2.11.4 Objective Functions 31
2.12 Constrained Application Protocol 33
2.12.1 CoAP Message Model 35
2.12.2 Message Format 36
2.13 Summery 38

Literature Review **39**

2.14 Introduction 39
2.15 Load balancing solutions in LLNs 39
2.16 Load balancing in RPL 40
2.16.1 Energy-Aware 40
2.16.2 Queue and MAC aware 43
2.16.3 Congestion Avoidance 44
2.16.4 Multi-path approach 46
2.17 Context-awareness Definition 47
2.18 Context awareness in RPL 48
2.18.1 Single Metric 49
2.18.2 Composite Metric 50
2.18.3 Further discussion on RPL OFs: 55

2.19 Current Limitations in RPL from the OF perspective 56
2.20 Load Balancing Solutions Summery 58
2.21 Summary 61

3 Tools for IoT 62

3.1 Introduction 62
3.2 LLNs Operating Systems (Overview) 63
3.2.1 Contiki 63
3.2.2 TinyOS 65
3.2.3 RIOT 67
3.2.4 Common IoT OS Comparison 68
3.3 Methods and Tools 69
3.3.1 Simulation Environment: Cooja 69
3.3.2 IoT Testbed Environment 71
3.4 Summary 74

4 An Efficient Load Balancing Objective Function for RPL 75

4.1 Introduction 75
4.2 Background and Motivation 76
4.3 The Proposed Load Balanced OF 81
4.4 Experiments and Evaluation 85
4.4.1 Simulation Setup 86
4.4.2 Results Analysis 86
4.5 Conclusion 91

5 Context-adaptive RPL for Antagonistic Requirements of LLNs Applications 93

5.1 Background and Motivation 93
5.2 The problem statement 95
5.2.1 The impact of antagonistic requirements on an emergency scenario 96

5.2.2 Survival time scenario....97

5.3 The Proposed Solution 97

5.3.1 Minimize the power consumption within the Lifetime Scenario....98

5.3.2 The Emergency Context Scenario....101

5.4 IPv6 Traffic Class (Tagging)....105

5.5 Experiments and Evaluation....107

5.5.1 Testbed-Lossy Network....110

5.5.2 Fixed Lossy network by the simulator....115

5.5.3 Tagging scenario....119

5.6 Conclusion....121

6 Conclusion and Future Work 123

6.1 Introduction....123

6.2 Thesis Summary....124

6.3 Summary of Contributions....125

6.4 Border impact....127

6.5 Future Directions....128

References 130

List of Figures

FIGURE 1.1 PROTOCOL STACK FOR IOT 2
FIGURE 2.1 ARCHITECTURE OF CONSTRAINED DEVICE 11
FIGURE 2.2 WIRELESS DEVICES POWER CONSUMPTIONS COMPARISON WITH RESPECT TO THEIR DATA RATES (REDRAWN FROM [21]) 12
FIGURE 2.3 SENSOR MTM-CM5000-MSP [27] 14
FIGURE 2.4 FORECAST CONNECTIONS BY SECTOR FROM 2016 TO 2024 16
FIGURE 2.5 THE EVOLUTIONARY OF IOT STACK 17
FIGURE 2.6 IEEE 802.15.4 TOPOLOGIES 19
FIGURE 2.7 6LOWPAN HAS THREE ARCHITECTURES TYPES 22
FIGURE 2.8 RPL ICMP MESSAGES [58] 24
FIGURE 2.9 DIO FORMAT 26
FIGURE 2.10 THE SEQUENCE OF DIOS IN DODAG CONSTRUCTION IN RPL 28
FIGURE 2.11 UPWARDS VS. DOWNWARDS ROUTS 29
FIGURE 2.12 STORING MODE: EVERY NODE HAS THE MAP OF SUBTREE 31
FIGURE 2.13 NON-STORING MODE: ONLY THE DODAG ROOT HAS THE MAP OF THE TREE 31
FIGURE 2.14 COAP OPERATES INDEPENDENTLY OF LOWER LAYER TECHNOLOGIES 34
FIGURE 2.15 LAYERING OF COAP 34
FIGURE 2.16 UNRELIABLE MESSAGE TRANSMISSION 36
FIGURE 2.17 COAP MESSAGE FORMAT 37
FIGURE 3.1 LOAD BALANCING SOLUTIONS FOR RPL 40
FIGURE. 3.2 CONGESTED ROUTE[88], [89] 45
FIGURE 3.3 APPLICATION, ROUTING AND MAC INTERACTION 49
FIGURE 3.4 CURRENT OF METRICS APPROACHES 50
FIGURE 3.5 FUZZY MODEL 54
FIGURE 4.1 PARTITIONING INTO CORE AND LOADED PROGRAMS 64
FIGURE 4.2 CONTIKI ARCHITECTURE 65
FIGURE 4.3 TINYOS ARCHITECTURE 66
FIGURE 4.4 RIOT ARCHITECTURE 67
FIGURE 4.5 CONTIKI UIPV6 STACK 70
FIGURE 4.6 TESTBED NODES TOPOLOGY IN MERCHISTON CAMPUS PREMISES 71
FIGURE 4.7 TMOTE SKY SENSOR BOARDS FRONT SIDE 72
FIGURE 4.8 TMOTE SKY SENSOR BOARDS BACK SIDE 72
FIGURE 5.1 RANDOM TOPOLOGY 77

FIGURE 5.2 THE SEQUENCE OF DIO IN DODAG CONSTRUCTION IN RPL 78
FIGURE 5.3 THE UNBALANCED PARENT SELECTION APPROACH IN RPL 80
FIGURE 5.4 UTILISING THE IGNORED DIO 83
FIGURE 5.5 THE BOTTLENECK NODES 85
FIGURE 5.6 POWER CONSUMPTION BALANCING 18 NODES 87
FIGURE 5.7 POWER CONSUMPTION BALANCING 50 NODES 88
FIGURE 5.8 NUMBER OF CHILDREN FOR N2 AND N3 (CHURN) 89
FIGURE 5.9 FIRST NODE TO DIE 90
FIGURE 5.10 NETWORK PDR 91
FIGURE 6.1 OF APPROACHES FOR COMPOSITE METRICS 96
FIGURE 6.2 (B) THE CALCULATION OF ACCUMULATIVE NUMBER OF CHILD NODES FLOWCHART 100
FIGURE 6.3 REGULAR (LIFE-TIME) SCENARIO 102
FIGURE 6.4 EMERGENCY (MINIMUM DELAY) SCENARIO 103
FIGURE 6.5 DODAG METRIC CONTAINER FORMAT 105
FIGURE 6.6 IPV6 HEADER FORMAT 106
FIGURE 6.7 RPL TARGET DESCRIPTOR OPTION FORMAT 106
FIGURE 6.8 TAGGING PACKET IN A CONGESTED PATH 107
FIGURE 6.9 THE TESTBED NODES 108
FIGURE 6.10 NETWORK DEPLOYED IN MERCHISTON CAMPUS PREMISES, EDINBURGH NAPIER UNIVERSITY 109
FIGURE 6.11 THE TEMPERATURE'S IMPACT ON DELAY 111
FIGURE 6.12 DELAY OVER THE TIME 111
FIGURE 6.13 AVERAGE DELAY FOR EACH METRIC 112
FIGURE 6.14. NUMBER OF DIOS 112
FIGURE 6.15 THE CHURN IN NODE7 113
FIGURE 6.16 POWER CONSUMPTION FOR NODE 8 114
FIGURE 6.17 POWER CONSUMPTION FOR NODE 3 114
FIGURE 6.18 POWER CONSUMPTION FOR THE THREE NODES IN EACH OF 115
FIGURE 6.19 DELAY WITH FIXED LOSSY RATIO 117
FIGURE 6.20 AVERAGE DIO 117
FIGURE 6.21 POWER CONSUMPTION FOR NODE 3, 7, AND 8 IN EACH OF 118
FIGURE 6.22 POWER CONSUMPTION FOR NODE 3 118
FIGURE 6.23 POWER CONSUMPTION FOR NODE 8 119
FIGURE 6.24 PDR FOR TAGGED AND NON-TAGGED PACKETS 120
FIGURE 6.25 END-TO-END DELAY FOR TAGGED AND NON-TAGGED PACKETS 120
FIGURE 6.26 LOST PACKET FOR TAGGED AND NON-TAGGED 121

List of Tables

TABLE 2.1 LR-WPAN FEATURES 18

TABLE 2.2 DIO MODE OF OPERATION (MOP) ENCODING 30

TABLE 2.3 COAP MESSAGES TYPE 35

TABLE 2.4 COAP MESSAGE HEADER DESCRIPTIONS 37

TABLE 4.1 SPECIFICATIONS OF SKYMOTE [155] 73

TABLE 4.2 SPECIFICATIONS OF TYPICAL CONSTRAINED DEVICES[21] 73

TABLE 5.1 SIMULATION CONFIGURATIONS FOR EXPERIMENTS 86

TABLE 6.1 SIMULATION PARAMETERS 116

List of Abbreviations

6LoWPAN	IPv6 Over Low Power WPAN
AMCA	Asynchronous Multi-Channel Adaptation
AODV	Ad Hoc On-Demand Distance Vector
AMI	Advanced Metering Infrastructure
ARPANET	Advanced Research Projects Agency Network
CNNs	Constrained Node Networks
CoAP	Constrained Application Protocol
CSMA-CA	Carrier Sense Multiple Access with Collision Avoidance
DAG	Directed Acyclic Graph
DAO	Destination Advertisement Object
DAO-ACK	Destination Advertisement Object Acknowledgement
DARPA	Defence Advanced Research Projects Agency
DIO	DODAG Information Object
DIS	DODAG Information Solicitation
DMR	DAG-Based Multipath Routing Protocol
DODAG	Destination Oriented Directed Acyclic Graph
DSME	Deterministic and Synchronous Multi-Channel Extension
ED	Energy Detection
ELT	Expected Lifetime
ETX	Expected Transmission Count
EWMA	Exponentially Weighted Moving Average
FFD	Full-Function Device
GTSs	Optional Allocation of Guaranteed Time Slots
ICMPv6	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoT	Internet of Things
JNI	Java Native Interface
LCI	Lifecycle Index
LLNs	Low-Power And Lossy Networks
LOAD	On-Demand Distance Vector
LQI	Link Quality Indication

LR-WPANs	Low-Rate Wireless Personal Area Networks
M2M	Machine-to-Machine
MAC	Media Access Control
MANET	Mobile Ad Hoc Network
MCU	Controller Unit
MOP	Mode of Operation
MRHOF	Minimum Rank with Hysteresis Objective Function
MP2P	Multi-Point-to-Point
MSNs	Mobile Sensor Networks
NIC	National Intelligence Council
OF	Objective Function
OF0	Objective Function Zero
OS	Operating Systems
OSI	Open Systems Interconnection
P2P	Point-to-Point
PAN	Personal Area Network
P2MP	Point-to-Multipoint
QoS	Quality of Service
RFD	Reduced-Function Device
RFID	Radio Frequency Identification
ROLL	Routing Over Low-Power and Lossy Networks
RPL	IPv6 Routing Protocol for Low-Power and Lossy Networks
RSSI	Signal Strength Indicator
SN	Sensor Node
SICS	Swedish Institute of Computer Science
SOSUS	Sound Surveillance System
TSCH	Time Synchronized Channel Hopping
TSMP	Time Synchronized Mesh Protocol
UDGM	Unit Disk Graph Medium
UDP	User Datagram Protocol
WSNs	Wireless Sensor Networks

1

Introduction

1.1 Shallow Foundation

With unprecedented potentialities of Low-power and Lossy Networks (LLNs) and Machine-to-Machine (M2M) [1], the connectivity of the constrained devices (*things*) have become a reality, which opened the door to its fullest extent for the Internet of Things (IoT). Indeed, the prosperity of IoT is owed essentially to the standardisation efforts along with academia for new standards protocols, which enabled the LLNs to afford a durable infrastructure for a plenty of IoT applications with a low intrusive. Therefore, by the end of this decade, it is widely expected to observe the massive increase of the connected things that roughly touch every aspect of our daily life. This chapter gives an overview about the research conducted in this thesis.

1.2 Motivation

With IoT era, the cost of miniaturisation, computing, networking has been reduced, thus, this bunch of tiny and cheap devices paves the way for more proliferation of new IoT applications, starting from smart homes to smart cities. This will connect heterogeneous devices that will be counted in billions within the next decade, and the vast majority of those connected *things* will be low-power, with varying degrees of complexity, according to industry anticipation [2]. In this vision, the research community strives with many precious efforts to fit the needs and the challenges of this newcomer (i.e., IoT) since the last decade [3] [4] [5] [6] in terms of the restricted processing and memory, the scarcity of energy, and their high susceptibility to intervention. Their efforts concluded to the deficiency of the current protocols and there is a persistent need for new protocols in line with the emergence of the IoT applications. The standards bodies, the Internet Engineering Task Force (IETF) and the Institute of Electrical and Electronics Engineers (IEEE), have anticipated this necessity. Thus, up to now, their efforts have been fruiting different protocols across all layers, such as the IEEE 802.15.4 protocol in the physical and MAC layer, Pv6 Routing Protocol for Low-Power and Lossy (RPL) in the network layer, and the Constrained Application Protocol (CoAP) in the

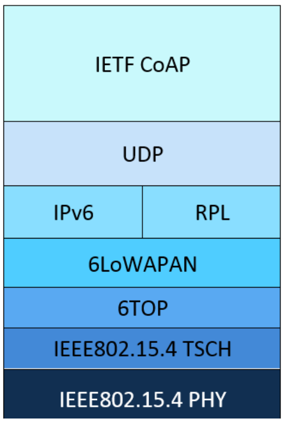


Figure 1.1 Protocol Stack for IoT

application layer, and ended up with a new IoT stack as depicted in the Figure 1.1.

Given the fact that the routing protocol is one of the main pillars of networking architecture, and confidently foreseeing its necessity for LLNs, RPL has swiftly become the de-facto routing protocol for IoT. Moreover, as RPL is the only standardised routing protocol for LLNs so far, then the number of the published work about RPL has been distinctly increasing every year according to the statistical distribution curve shown in [7]. In fact, it is expected to keep this proliferation increasing monotonically as RPL is meant to be one of the main building blocks to underpin LLNs within the IoT trend. However, this protocol is still in need for more enhancements, in terms of load balancing, power consumption, and context-awareness. Henceforth, here from this point, our motivation took place and the research within this thesis came to focus on.

1.3 Problem Definition

Despite the existing research efforts such as in [8] [9] [10], which have been focused to address the limitations of the constrained devices within the design of the standards protocols, yet there is a room for enhancing the standard protocols and dealing with the gaps that need to be filled. Hence, these gaps can be identified as follows:

- ***First,*** RPL is dealing with non-uniform distribution in large-scale LLNs in addition to the uniform ones, which leads to unfairness distribution for the number of children and to inequitable data traffic. Hence, the load imbalance is considered as a significant weakness in this protocol in terms of fairness in the number of children among parents. Thus, the energy of the overloaded nodes will be drained much faster than other nodes. Moreover, this problem has more harmful impacts if the overloaded node is a bottleneck node (i.e. those are within the first hop to the root), which may result in early disconnection in the part of the network that is covered by that overloaded parent, and consequently affect the network reliability negatively. Therefore, the traffic balancing is a key issue that urges to be considered within the design of RPL protocol.

- ***Second***, with IoT era, a plethora of applications has been emerged. Nevertheless, these quotidian applications may have antagonistic requirements (e.g. end-to-end delay and lifetime) which are susceptible to increase the risk of performance degradation of RPL protocol. In fact, that is due to changing the priority of these antagonistic requirements as a result of emergency scenarios or in response to a new context. Consequently, this challenging issue is associated with building the best topology in RPL, which cannot sufficiently fulfil the antagonistic requirements of the LLNs applications.

This thesis strives to address the following questions:

1) What constitutes a better traffic distribution among the overloaded nodes in RPL compared to the existing solutions?
2) Can the RPL Destination Oriented Directed Acyclic Graph (DODAG) construction mitigate the risk of the bottleneck nodes in LLNs?
3) How can RPL be a more reliable protocol in emergency scenarios considering the coexistence of the antagonistic requirements within IoT applications?
4) Will the benefit of the adaptive routing technique for the antagonistic requirements lead to improvements in the performance and quality of networks?

1.4 Aim and Objectives

The main aim of this thesis is to enhance the performance of the current standard routing protocols of IoT, in particular of RPL protocol. To focus on that goal and get past that standstill, we proposed several algorithms to ensure better load balancing, confirming energy-balanced, and constructing paths based on the context in RPL to mitigate the impact of the antagonistic requirements.

To achieve the overall aim, several objectives have been set as follows:

- **Objective 1:** The first objective is to gain an in-depth knowledge and master the state-of-the-art about IoT concepts, the potential applications, the new standards stack of IoT with more focus on routing protocol, and investigate the major issues and concerns related to RPL protocol.
- **Objective 2:** The second objective is to scrutinise and analyse the features and the drawbacks of the current standard RPL objective functions (OFs).
- **Objective 3:** Designing new algorithms with the aim of endorsing the reliability in RPL in terms of load balancing, context adaptive, and mitigating power consumption and the antagonistic impact on LLNs applications.
- **Objective 4:** Evaluating the proposed algorithms performance metrics, such as lifetime, PDR, and end-to-end delay, and their techniques via configuring a real testbed nodes environment alongside appropriate software simulator tool.

1.5 Thesis Contributions

1.5.1 Key Outcomes

The impact and applicability of this work goes beyond academia and extends to be included within the standards body, which has been initially accepted to be adopted within the standard.

The key outcomes of this study in the form algorithms, and protocol improvements are listed as follows:

- Firstly, a new OF has been proposed using a new metric and novel technique, to achieve a better distribution in the number of direct children among the candidate parents, in an attempt to balance the load traffic and keep the battery lifetime thriving for a longer time.
- Secondly, new technique of distribution in the number of children has been widen to accommodate the entire path towards the root rather than the number of direct children only.

- Thirdly, a context-adaptive OF has been proposed to facilitate exchanging the information between application and network layers. The key property is to optimise the selection decision of the routing path towards the root node adaptively, by juxtaposing the antagonistic requirements within the application, according to the new context triggers.

1.5.2 List of Publications during this research

- **M. Qasem**, A. Y. Al-Dubai, I. Romdhani and B. Ghaleb, "*Load Balancing Objective Function in RPL*" Internet-Draft-roll-rpl-load-balancing-02. Internet Engineering Task Force (IETF) 2017.
- **M. Qasem**, A. Y. Al-Dubai, I. Romdhani and B. Ghaleb, "*A New Efficient Objective Function for Routing in Internet of Things Paradigm*" in IEEE Conference on Standards for Communications & Networking (IEEE CSCN'16), Berlin, Germany, 2016.

- **M. Qasem**, A. Y. Al-Dubai, I. Romdhani and M. B. Yassien, "*A Dynamic Power Tuning for the Constrained Application Protocol of Internet of Things*" Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on, Liverpool, 2015, pp. 1118-1122.
- **M. Qasem**, A. Y. Al-Dubai, I. Romdhani and M. B. Yassien," *ATP: Adaptive Tuning Protocol for Service Discovery in Wireless Sensor Networks*", ACM 2015. Istanbul, Turkey.

Under Review

- **M. Qasem**, A. Y. Al-Dubai, I. Romdhani, "Context Adaptive RPL for Antagonistic Requirements in Low Power and Lossy Networks" IEEE Internet of Things Journal 2018.

1.6 Thesis Structure

The remainder of this thesis is organised as follows. In Chapter 2 more details about IoT definition and their potential applications, with more focus on the applications with emergency scenarios. Chapter 3, the literature review, provides a further discussion of the state of the art research presents, across all layers, the different protocols of the standardisation efforts on Low-power and Lossy Networks (LLNs) of IoT stack with more attention has been given for the RPL protocol. It also gives the reader the essential foundation for understanding the rest of this thesis. The next chapter is Chapter 4, the tools for IoT are presented, including the existing LLNs operating systems along with their main features, and the testbed hardware as an experimental environment along with main features of the simulator used in this thesis. In Chapter 5, the load-balancing issue in RPL is identified and investigated. Accordingly, a new Load Balancing OF for RPL is introduced to improve the balance of traffic via improving based on the number of child nodes among the overburdened parent nodes in the OF, to ensure network lifetime maximization. The thesis also explored the synergy between the context parameters and RPL routing protocol, which further alleviates risks caused by the application antagonistic needs as given in Chapter 6. Furthermore, an adaptive OF based on the context is proposed to decrease the impact of the antagonistic requirements throughout the emergency scenarios. Additionally, the network lifetime is optimised during the non-emergency scenarios via the accumulated number of children towards the root node. The implemented OF has been tested using real testbed measurements of a multi-hop LLNs nodes in conjunction with simulator. The thesis ends by giving conclusions, in Chapter 7, shaped from the current research and outlining avenues for future directions.

2

Background

2.1 Introduction

In this Chapter, a generic historical background of Wireless Sensor Networks (WSNs) is given. Then, the IoT definition as a term along with its potential applications are also presented with more attention paid to the near future trends of such applications and their influences. Moreover, this chapter discusses the essential background to facilitate the understanding of the research problems within this thesis. Across all layers, we survey the different protocols of the standardisation efforts on Low-power and Lossy Networks (LLNs) of IoT stack, with more focus on the protocol RPL in the network layer.

2.2 Historical view of WSNs

The story of the sensors network began with the U.S. Sound Surveillance System (SOSUS) in 1952 by U.S. Navy. The point behind it was to sense the Soviet submarines in the sea via underwater microphones called hydrophones [11]. Currently, these sensors are utilised in notification systems for earthquake [12].

Also, in 1980 the U.S defence launched another program for WSN called the Distributed Sensor Networks (DSN) at the Defence Advanced Research Projects Agency (DARPA)[13]. Before that in a couple of years, the Advanced Research Projects Agency Network (ARPANET) was working. The aim of the project was ambitious: "*The network was assumed to have many spatially distributed low-cost sensing nodes that collaborate with each other but operate autonomously, with information being routed to whichever node can best use the information"*[13]. The point was to figure out whether it is possible to extend the ARPANET to sensors network.

2.3 Low-power and Lossy Networks

With the beginning of this decade, new terminologies have been proposed to describe the new appearances of networks that include the constrained devices [14]. A good example of these terminologies is the LLNs (*Low-power and Lossy Networks*) [15] and Constrained-Node Networks (CNNs) [6]. The definition of LLNs is "*typically composed of many embedded devices with limited power, memory, and processing resources interconnected by a variety of links, such as IEEE 802.15.4 or low-power Wi-Fi*" [15]. Furthermore, an LLN introduced in [16] as "a constrained-node network with certain network characteristics, which include constraints on the network as well". This attention of LLN is owed essentially from the wide range of applications that can be under the umbrella of LLNs, comprising, industrial monitoring, smart building and homes, lighting, access control, e-health , environmental monitoring, urban sensor networks, energy management and so forth [15]. To enable LLNS devices to the mentioned applications, the Standard bodies IETF and IEEE have standardized new protocols, as detailed in Section 2.6. Thus, utilising the constrained devices has become possible, despite the fact of the limitations of such devices in terms of memory and processing capacities, small data rates, and limited energy which typically powered by non-rechargeable batteries.

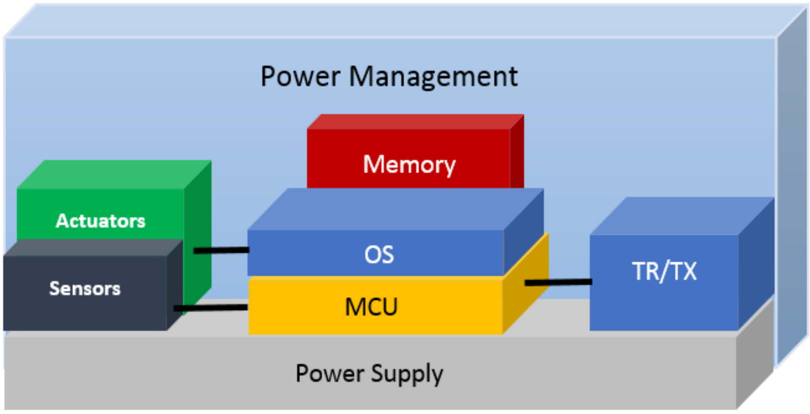


Figure 2.1 Architecture of Constrained Device

Commonly, the architecture of the constrained devices is almost similar. Figure 2.1 represents the main components of a constrained device, namely the sensing/actuating unit, a central processing (MCU) unit along with operating system (OS) and Memory units, communication (sending TX and receiving TR) unit and the power unit [17].

As the power consumption is a significant issue for the LLNs devices, it is important to get them work at a very low data rate and transmission power aiming at an extended lifetime. Figures 2.2 shows a comparison between different wireless devices, in terms of power consumptions of transceivers and the microprocessors with respect to their data rates. Among all illustrated devices, MSP430 LLNs devices show the minimum transmission power. However, the ZigBee and the Bluetooth devices are located within the medium average in terms of power consumption and with similar data rate compares to LLNs devices. However, when it comes to Wifi the data rate is much better but on the express of more power consumption [18] [19] [20], which does not suit the IoT devices.

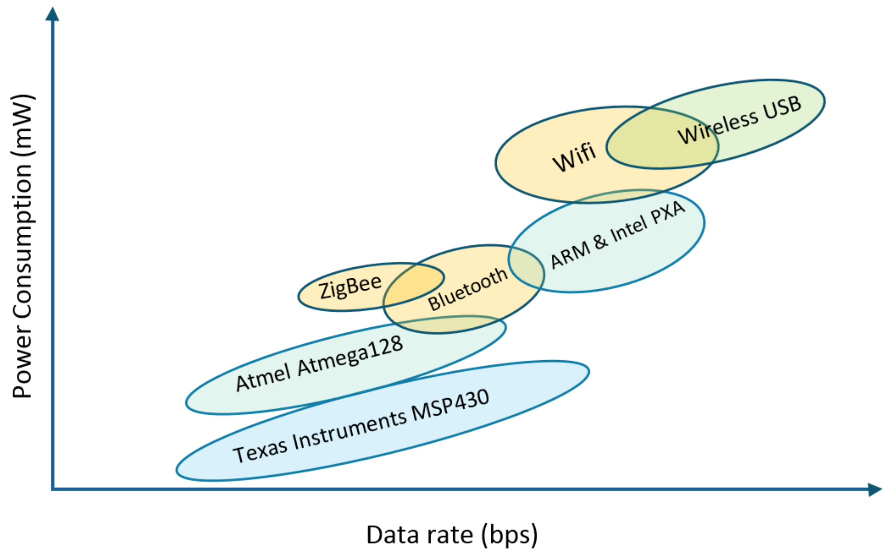


Figure 2.2 Wireless devices power consumptions comparison with respect to their data rates (Redrawn from [21])

2.4 Internet of Things

Capturing the gist of the Internet of Things (IoT) in one definition is quite hard, thus, there is a considerable variation in the definition and explanation of IoT. Historically, Kevin Ashton from MIT has pioneered the term "Internet of Things" at the beginning of the previous decade[22], and then he added:

> *"The Internet of Things has the potential to change the world, just as the Internet did. Maybe even more so."[23]*

In the last few years, the interest in IoT has been a growing due to the advent of new applications widely for WSN in different fields. Indeed, this proliferation can be recoganised due to the decreasing cost of the maintenance and configuration for IoT devices (things). For instance, the lower energy consumption in sensors is preferable, especially in the presence of energy shortage that we have on our planet. Furthermore, LLNs as infrastructure for IoT are able to work independently without the need to attend from human and that another cutting-cost factor can be considered. As the future trend for IoT, the U.S. National Intelligence Council (NIC) has anticipated the benefits of IoT over the conventional Internet in the civil technologies and reported that:

> *"By 2025 Internet nodes may reside in everyday things, food packages, furniture, paper documents, and more. Today's developments point to future opportunities and risks that will arise when people can remotely control, locate, and monitor even the most mundane devices and articles. Popular demand combined with technology advances could drive widespread diffusion of an Internet of Things (IoT) that could, like the present Internet, contribute invaluably to economic development and military capability"[24].*

Placing the virtual world (i.e., Internet) and the physical world (i.e., Things) together semantically means a world-wide network of interconnected objects uniquely addressable [25]. Internet is considered as an immense revolution for human, and then IoT shapes the next revolution of Internet, taking into account the new manner of collecting information and then providing it as services.

As mentioned earlier, the outstanding concept of IoT is to allow sharing the resources and provide them as services within a minimum level of power consumption. Thus, the option to replace the typical ways of using and restricting these resources is on the table. Moreover, it can be considered as a further step to reduce the gap between poor and rich people as these services will become more affordable.

In general, IoT architecture has three main components [26]:

1) **Hardware (Things):** includes all different devices such sensors, actuator, smartphones and Radio Frequency Identification (RFID). Generally, any physical object can deal with at least one of the following in digital way [27]:

    - Sensors (light, humidity, temperature, etc.)
    - Communication transceiver (either wired or wireless)
    - Actuators (sound, motors, etc.)
    - Computation (programmable)

    These things are varied in size, capabilities, and functionalities. Figures 2.3 shows a widely used IoT device called sky motes [28]. Further details about this device in Chapter 4.

Figure 2.3 Sensor MTM-CM5000-MSP [27]

2) **Data storage and analytics**: this part represents the storage technique in addition to analytics. As the core role of things is to gather information and then offer it as services

on demand. In addition, the huge number of these devices is indicating to the massive amount of the collected information that should be stowed. Due to this reason, the machine learning algorithms have been used at analysis process.

3) **Visualization**: this part permits the end-user to interact with the IoT application. Moreover, this constituent has been given more consideration to meet the end-user needs as it forms the business oriented.

2.5 IoT Applications

Over the recent years, there has been a prolific growth in the research of IoT. Nowadays, the potential of IoT applications can be recognised in a vast range of different domains, with varying degrees of complexity, due to the wealthy market opportunities in this field. To this end, the research teams strive to sort out the challenging problems of IoT that are standing as an obstacle to release the best IoT products, and so far their progress is due to some applications in different disciplines [3] [29], such as:

- Home (Health, Security, Utilities, and appliances)
- Transport (logistics, Traffic, Parking, emergency services)
- E-health (remote care, monitoring)
- Community (smart metering, factories, retail, Monitoring environment)
- Defence and Military
- Smart Cities

In addition to the disciplines mentioned above, IoT has a significant potential in Emergency Management and Real-time applications [30] [31]. IoT applications can be mostly classified into two categories based on the data assembly approach: time-driven and event-driven. In the first category, the date is collected periodically based on a predefined interval to monitor the environment, a place, or for instance a particular incident or a phenomenon in nature. The second one can be used to start reading the data once a specific event is detected where the emergency management can take a place for IoT applications. In general, IoT application will be part of every aspect of our life as Figure 2.4 explains.

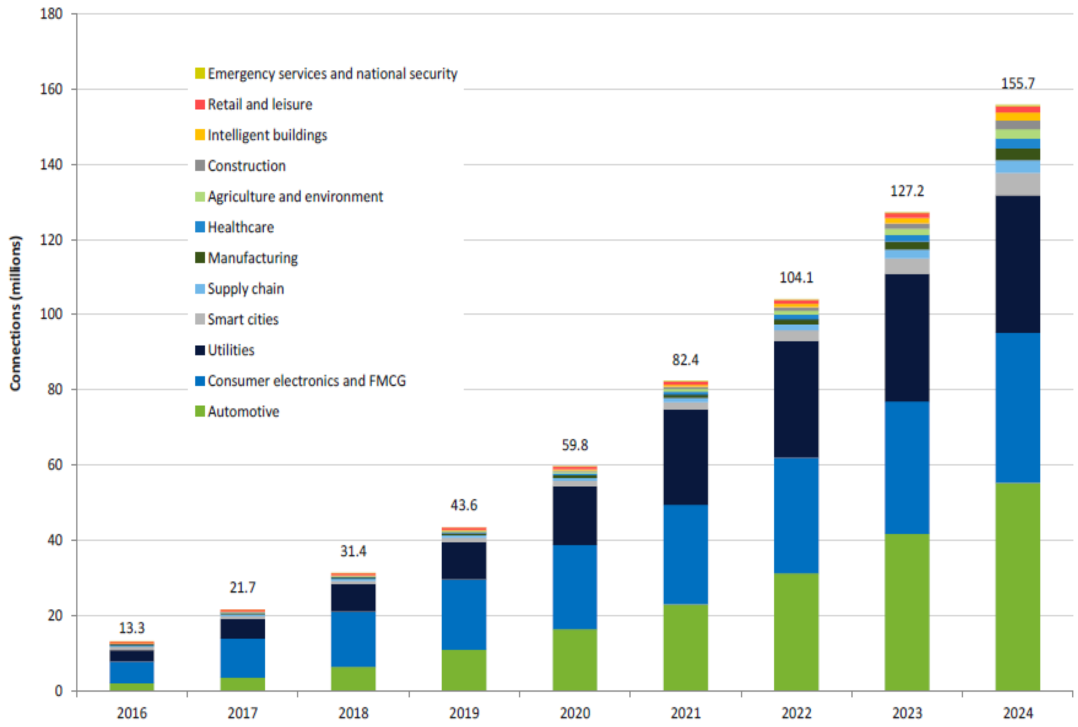


Figure 2.4 Forecast connections by sector from 2016 to 2024

Figure 2.4 [32] shows clearly the potential of IoT market in several disciplines. This forecasting study, which was conducted in 2017 by Ofcom, illustrates the number of connected IoT devices in the UK was roughly 13.3 million by the end of 2016. However, this number is predicted to rise up [32] at a Compound Average Growth Rate (CAGR) of roughly 36%, to 155.7 million connected devices by the end of 2024.

Moreover, the study concluded that there are several factors will influence the adoption of the IoT across all disciplines and domains.

2.6 Overview standardisation efforts on LLNs

The Open Systems Interconnection (OSI) stack conceptualises networking into seven layers as shown in Figure 2.5.a. However, this stack has been minified to five layers as shown in TCP/IP stack [33] Figure 2.5.b, where both of the presentation and session layers have been merged into the application layer. With IoT era, the current protocol in TCP/IP stack cannot accommodate the needs of the newcomer (IoT), thus, the standards bodies IETF and IEEE anticipated this necessity and introduced new stack to meet the needs of the constrained devices (*things*) as shown in Figure 2.5.c [3], [34]. In the coming sections, we will come across different layers and protocols of the standardisation efforts on LLNs of IoT stack.

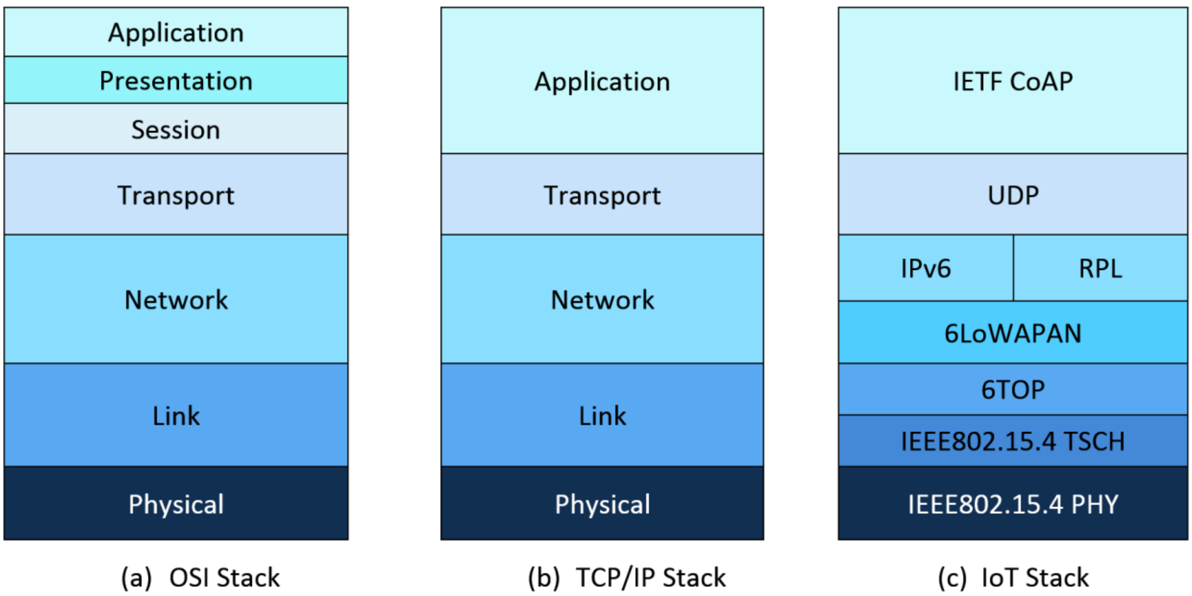


Figure 2.5 The Evolutionary of IoT stack

2.7 IEEE 802.15.4 Protocol

The IEEE 802.15.4 standard [35]designates the physical and the Media Access Control (MAC) for low-rate wireless personal area networks (LR-WPANs). Mainly the IEEE 802.15.4 protocol aims to easy installation, reliable data transfer, short-range operation, extremely low cost, and a sensible battery life while maintaining a simple and elastic. A summary of some features of a LR-WPAN is given in Table 2.1.

Table 2.1 LR-WPAN Features

Feature	Specification
Over-the-air data rates	250 kb/s, 100kb/s, 40 kb/s, and 20 kb/s
Topologies	Star or peer-to-peer
Allocated extended addresses	16-bit short or 64-bit
Time slots allocation	Optional allocation of guaranteed time slots (GTSs)
MAC	Carrier sense multiple access with collision avoidance (CSMA-CA) channel access
Reliability	Fully acknowledged protocol
Power consumption	Low power consumption
Energy constraint	Energy detection (ED)
Link quality	Link quality indication (LQI)
Number of channels vs Band	16 channels in the 2450 MHz band, 30 channels in the 915 MHz band, and 3 channels in the 868 MHz band

The participated devices within IEEE 802.15.4 could be one of two types: either a full-function device (FFD) or a reduced-function device (RFD). An FFD device can act as a Personal Area Network (PAN) coordinator or a coordinator, or a device. Thus, an FDD can communicate to RFDs or other FFDs. Whereas an RFD communication is restricted only to an FFD device. Given this limitation in RFD devices, then the amount of data that intended to transfer to FDD is low.

Another main difference between FDD and RFD based on the intended role for each type. Considering the un-slotted access approach, a device, normally the gateway (FFD), needs to remain always awake, however, the awake of RFD is on demand i.e. when it is required. This extremely mitigates the power consumption to the least level, on the other hand, consequently, increases the router the power consumption.

According to the application requirements, an IEEE 802.15.4 LR-WPAN may operate in one of two topologies: either the star topology or the peer-to-peer topology.

The topologies within IEEE 802.15.4 LR-WPAN fall into one of three types according to the requirements of the application: the star, the peer-to-peer or the cluster-tree topologies as illustrated in Figure 2.6.

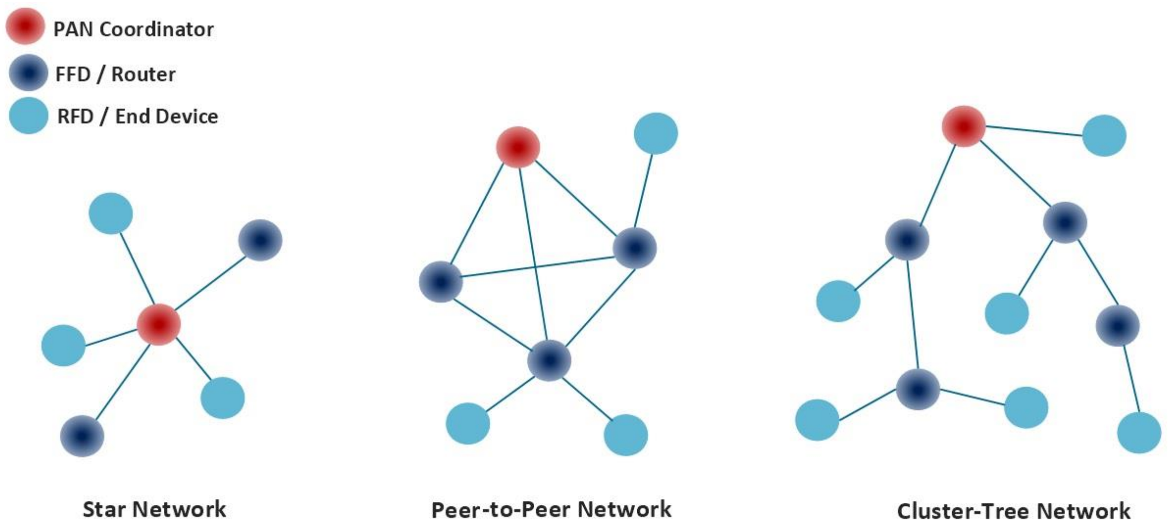


Figure 2.6 IEEE 802.15.4 Topologies

In a star topology, all the traffic between devices must pass through the primary controller, i.e. the PAN coordinator. The main responsibilities of the PAN are, initiating, routing, and terminating the communication in the network. Conversely, in a peer-to-peer network, any two nodes within the range of each other they can communicate, where one node is chosen as the PAN coordinator. Obviously, the elasticity in this type of topology is much better, enabling all types of mesh networks, but on the expenses of increasing the power consumption. For this reason, it is recommended to keep the PAN coordinator with mains powered, whereas powered the rest devices with battery. The cluster-tree topology comes to combine the previous two topologies together. In this case, some of the edge nodes could be out of the radio coverage, however, these nodes might still interconnect through the clusters.

2.8 IEEE 802.15.4e (TSCH)

IEEE 802.15.4e-amendment1 was released in 2012 [36] which has been combed later on to the next revision of IEEE 802.15.4 in 2015 focusing on the key feature named Time Synchronized Channel Hopping (TSCH). TSCH provides a robust channel hopping technique to address the emerging needs of industrial requirements. The first commercialization of TSCH technology is owing to Dust Networks, driven from Time Synchronized Mesh Protocol (TSMP) [37] which has been effectively adopted in Wireless HART networks [38] [39]. In short, the authors in [40] study the general functional enhancements of IEEE 802.15.4e (Low Energy, Information Elements, Enhanced Beacons, MAC Performance Metric, and Fast Association). Moreover, along with TSCH mode, IEEE 802.15.4e states another four MAC behaviour modes[40], namely, Deterministic and Synchronous Multi-channel Extension (DSME), Low Latency Deter Radio Frequency Identification Blink (BLINK) ministic Network (LLDN), Asynchronous multi-channel adaptation (AMCA).

TSCH integrates time-slotted access, which is originally defined in 802.15.4MAC protocol, with channel hopping capabilities. This smart combination provides the ability to anticipate energy efficiency, communication reliability, and high network throughput [41]. Furthermore, by eliminating collision among competing nodes can, in turn, achieve deterministic latency. Furthermore, the time synchronization in TSCH is vital to attain ultra-low-power operation in addition to channel hopping to secure a reliable network [42] [39].

As the topology TSCH is self-governing, it can be used to accommodate any network topology such as a tree, star, partial or full mesh. Typically, all the aforementioned features of TSCH are applied in wide domains include oil and gas industry, manufacturing different products such as (food/brew, chemical, pharmacological). water/waste treatments, green energy production, and climate control [36].

2.9 6TOP

However, TSCH schedule still lacks for management, thus, in Nov 2013 IETF charted a working group called 6TiSCH (IPv6 over the TSCH mode of IEEE 802.15.4e) to enable IPv6 on top of TSCH [44] [39]. The goal of 6TiSCH is to provide high throughput and minimum latency that are ideal for Industrial WSNs such as WirelessHART and ISA100 [45]. Given the heterogeneity of the standards and devices, 6TiSCH introduces a so-called minimal 6TiSCH configuration [46] to guarantee a reasonable level of compatibility. The key point in 6TiSCH networks is the distributed scheduling among the nodes, which can be achieved by the 6TiSCH Operation sublayer protocol named 6top Protocol (6P) [47], this protocol enables the interaction among the neighbour nodes to add/delete timeslots called cells. Moreover, 6top can run one or several scheduling functions which specifies when to add/delete the cell [46]. Finally, the work on 6TiSCH is still ongoing, however, it is anticipated to be the standard for LLNs industrial monitoring applications.

2.10 6LoWPAN

The adaption layer (header format) 6LoWPAN (IPv6 over Low power WPAN)[35] [48] standardised by IETF to adopt the connection between IPv6 protocol (network layer) and IEEE

802.15.4 protocol (Mac layer). Enabling the transmissions of IPv6 Packets over IEEE 802.15.4 Networks of the constrained devices is demanding for many LLNs applications. To open the door for such tiny, ubiquitous devices to participating natively over the Internet, several efforts have been taken place to arm the intermediate layer 6LoWPAN with a number of techniques [49] as follow:

A) Compression technique: IETF standardised RFC 6282 [50] due to the limited bandwidth (127 bytes) provided by the IEEE 802.15.4 protocol to fit only 60-80 bytes for a User Datagram Protocol (UDP) payload. In this technique, the extra bytes have been minimized and resized. In more specific words, compressing the IPv6 header without the 8 bits field of the hop limit.

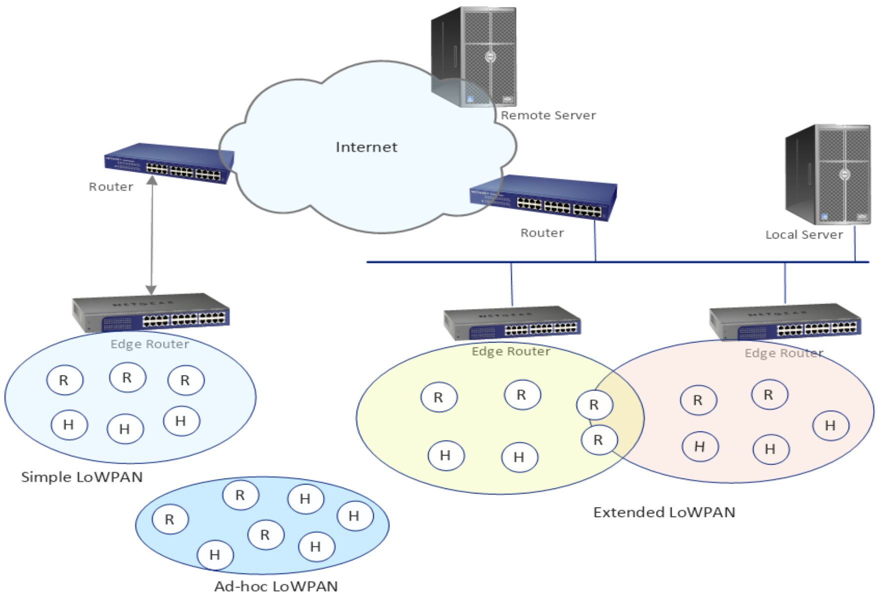


Figure 2.7 6LoWPAN has three architectures types

B) Fragmentation: Normally, the IPv6 payload does not fit the payload of the IEEE 802.15.4, so the fragmentation is the only option in this case. 6LoWPAN plays an essential role to guarantee the right assembling of the fragmented packets over multiple hops.

Figure 2.7 illustrates the 6LoWPAN three architectures types, namely, the simple LoWPAN architecture, the extended LoWPAN architecture and the ad-hoc LoWPAN architecture. Unlike the ad-hoc, the simple and the extended types need an infrastructure to operate which are mainly used in smart homes and buildings applications. Clearly, every node within the architecture can be either a router (R) and edge router, or a host (H).

2.11 IPv6 Routing Protocol for Low-Power and Lossy Networks

Given the remarkable pragmatic connection between LLNs and IoT, and the fact that LLNs is emerging and underpinning the infrastructure of IoT, IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) has swiftly become the de-facto routing protocol for IoT. Indeed, that achievement can obviously be seen from adopting RPL by the ZigBee Alliance companies within their technologies along with IEEE 802.15.4 MAC and PHY layers [51]. The standards body IETF established the Routing Over Low-power and Lossy networks (ROLL) working group to design a routing protocol for LLNs. So, ROLL considered the route-over approach [33] to propose RPL in 2012 [53].

Generally, existing routing protocols for WSNs and LLNs can be categorised either as proactive, reactive, or hybrid. In proactive protocol every node within the topology maintains a routing table that continuously evaluates the routes [54] [55]. RPL is considered as a proactive (*table driven)* distance vector routing protocol, as the routing tables are updated frequently in two different modes, namely, storing and non-storing modes, as detailed in Section 2.11.3.

While, in the reactive approach, the route is created towards the destination on demand, so-called on-Demand routing protocols. In this approach the routing information, driven form route discovery events, play an essential role to build the cache tables. These types of protocols are ideal only when the load traffic is low, but not an option in the real-time applications where the delay is a critical factor. The 6LoWPAN Ad Hoc On-Demand Distance Vector (LOAD) Routing protocol [56] relying on the flooding technique, if any node is willing to send then the network will be flooded with the route request. The following sections will present in brief the major components of RPL.

2.11.1 RPL Control Packets

To maintain the Destination Oriented Directed Acyclic Graph (DODAG) construction, new types of Internet Control Message Protocol (ICMPv6) [57] control messages have been proposed in RPL. The type of each message can be identified using the code field within the message format as shown in Figure 2.8.

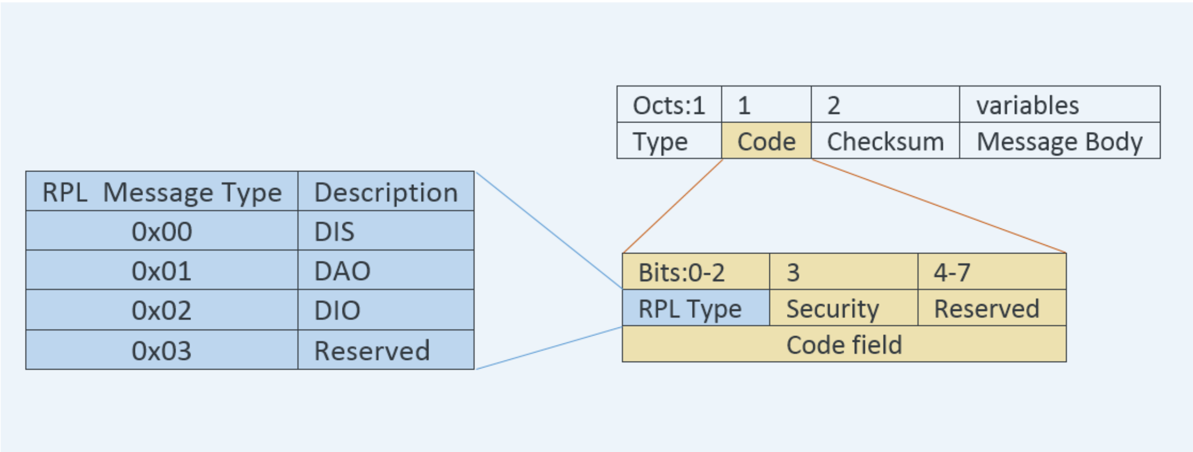


Figure 2.8 RPL ICMP messages [58]

- **DODAG Information Object (DIO):** This control message holds the configurable parameters RPL that can be learned (such as RPLInstanceID) to find an RPL Instance, select a DODAG parent set, and maintain the DODAG construction. The starting point begins when the DODAG root broadcasts the

DIO message to the downstream neighbour nodes and enable Point-to-Multipoint traffic in an upward direction. DIO messages hold the RPLInstanceID, DODAG root identity, routing metrics, rank, objective function and DODAGID. These messages are sent periodically with the cumulative sequence number in order to start the parent selection process.

- **Destination Advertisement Object (DAO):** The DAO message is dedicated to propagating destination information upwards along the DODAG. Sending DAO message, which is in response to a DIO message, is an optional feature in RPL for the applications with point-to-multipoint or point-to-point traffic. This feature can be enabled, by the identifier Mode of Operation (MOP) within the DIO message, either as a storing or non-storing mode. In storing mode, the child node unicasts the DAO message to the selected preferred parent, whilst unicasting DAO message directly to the DODAG root if the mode is non-storing. It is worth to point that, in both modes, the feature of receiving an acknowledgment (i.e. DAO-ACK) for the sent DAO is optional.

- **Destination Advertisement Object Acknowledgement (DAO-ACK):** This message is sent, in the response to receiving a DAO message, in a unicast manner to the DAO sender to acknowledge its willingness to act as a next hop node towards the DODAG root.

- **DODAG Information Solicitation (DIS):** Multicasting the DIS message by an RPL node, willing to join the DODAG, to solicit a DODAG Information Object from neighbourhood nodes.

2.11.2 RANK and DODAG Construction

To construct and maintain the topology (DODAG) an explicit routing information can be transported. According to the DIO base rules [53], RPL employs a number of different identifiers within DIO message, among other control information, as shown in Figure 2.9 [53], part of these identifiers are (1) RPLInstanceID to identify the number of DODAGs if it is more than one, (2) DODAGID represents the RPL Instance. (3) DODAGVersionNumber shows how many times the DODAG has been reconstructed as part a technique of free loop, (4) Rank which will be calculated based on the designated objective function.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RPLInstanceID |Version Number |             Rank              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|G|0| MOP | Prf |     DTSN      |     Flags     |   Reserved    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                            DODAGID                            +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Option(s)...
+-+-+-+-+-+-+-+-+
```

Figure 2.9 DIO Format

RPL relies on DIO control message to initiate the DODAG construction. The first move comes when the DODAG root node emits the DIO message to all neighbour nodes. As soon as the nearest node receives the DIO message, it adds the sender to its parent list, then calculates the rank using equations (2.1) and (2.2) [59] associated with the selected OF. The node decides to join this DODAG based on the rank value. If the sender has the same or less rank compared to the child node's rank then it would opt as a preferred parent, and thus permits all traffic from the child node to be received.

$$\text{Rank}(_{N}) = \text{Rank}(_{PN}) + \text{RankIncrease} \quad (2.1)$$

$$\text{RankIncrease} = \text{Step} \times \mathit{MinHopRankIncrease} \quad (2.2)$$

where Rank($_{N}$) is the rank of the sender node and Rank($_{PN}$) is the Rank of preferred parent, *Step* represents a scalar value and *MinHopRankIncrease* represents the minimum increase in rank between a node and any of its DAG parents. *RankIncrease* is represented in units expressed by the variable *MinHopRankIncrease* which can be fixed according to the configuration.

variable MinHopRankIncrease

According to the predefined criteria, the selected objective function defines the way of calculating the rank with a monotonical increase fashion in a downward direction, wherein the DODAG root has the least rank to guarantee loop-free topology. In this regards, the rank embodies the virtual location of the node itself compares to the other nodes within the DODAG. In fact, this scaler position might be improved by associating with the lower rank node, which is at the same time act as an anchor for the path towards the DODAG root.

Afterward, the node propagates its own DIO with all updated information (i.e. identifiers) to its neighbours as shown in Figure 10.b. In the same context, another scenario could emit the DIO, when the node intends to join DODAG it transmits DIS control message to its neighbours to solicit DIO message, then join by choosing the preferred parent according to the rank. An example of the DIO messages sequence into DODAG construction is depicted in Figure 2.10. Heretofore, two objective functions have been standardised by IETF, OF0 [60] and MRHOF [59] relying on the metrics hop count and ETX respectively. More metrics have been proposed in [61] such as node energy, throughput, latency, and others. Nevertheless, the RPL specification [53] does not obligate any certain OF neither routing metrics to be considered and left the door open for further improvements for any new metrics or OF.

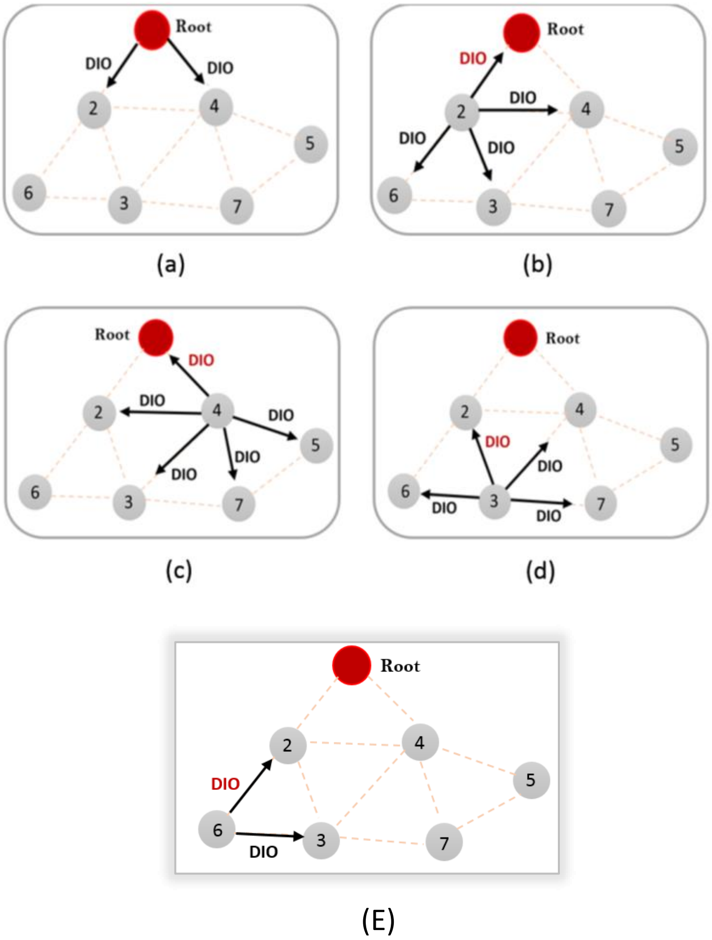


Figure 2.10 The sequence of DIOs in DoDAG construction in RPL

2.11.3 Communication Patterns

Generally speaking, RPL facilitates three patterns of communications: Multi-Point-to-Point (MP2P) which is considered as the superior traffic flow in most of LLN applications [53], and it performs the routes from multiple nodes up to the DODAG root. In contrast, the traffic in Point-to-Multipoint (P2MP) pattern transmitted from the DODAG root along to the multiple nodes according to the needs of some applications. The last pattern called point-to-point (P2P) which facilitates the traffic between any two nodes within the same DODAG as depicted in Figure 2.11.

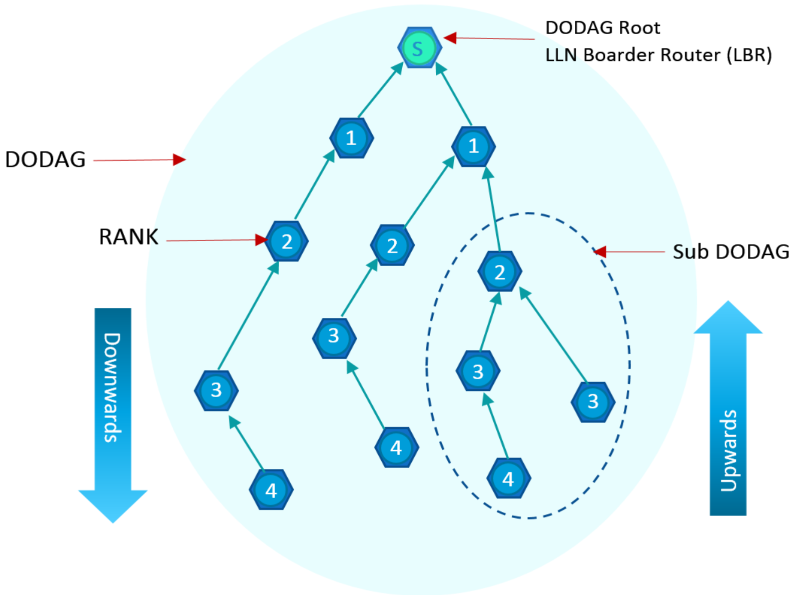


Figure 2.11 Upwards vs. Downwards routs

A) Upwards Routes

In this approach, the DIO messages are utilised to disclose and maintain the routs towards the DODAG root as shown in Figure 2.11. Upward routes support MP2P traffic flows from the leaves nodes towards the DODAG root. In such a traffic pattern, each node should select one node as a preferred parent from the set of one-hop neighbours according to the calculated rank that driven from the criteria's that have been injected within the OF to fit the needs of the application. Thus, all traffic in MP2P will be transmitted to the DODAG root via these constructed routes in upwards manner. The initiation step of Upwards routs begins with broadcasting DIOs by the DODAG root as detailed in Section 2.11.2.

B) Downwards Routes

The Downward routes are constructed to support the traffic in both P2MP and P2P patterns. The designated routes are maintained via the DAO messages which have been proposed as an optional feature [53] in P2P and P2MP traffics. In this context, every node associated to the DODAG sends DAO to its preferred parent in a unicast manner. Accordingly, the received DAO will be handled based on one of two modes, storing and non-storing modes. To identify the mode that should be considered in each node a variable called mode of operation (MOP) is identified along with other values in DIO as DIO Base Rules [53], and then signaling Down the DODAG values that should be broadcasted by each node has a route to that DODAG root. The possible values of MOP have been proposed in [53] as given in Table 2.2.

In the storing mode shown in Figure 2.12, the sub-DODAG Downward routing tables will be stored in addition to child address, based on these routing tables the current node can decide to which neighbour to send it, i.e. unicasting the DAO messages only to the DAO parent node.

However, unlike the storing mode, in the non-storing mode shown in Figure 2.13, the sub-DODAG Downward routing tables stored only in the DODAG root, i.e. unicasting the DAO messages only to the DODAG root node.

Table 2.2 DIO Mode of Operation (MOP) Encoding

MOP Value	Description
0	No Downward routes maintained by RPL
1	Non-Storing Mode of Operation
2	Storing Mode of Operation with no multicast support
3	Storing Mode of Operation with multicast support
	All other values are unassigned

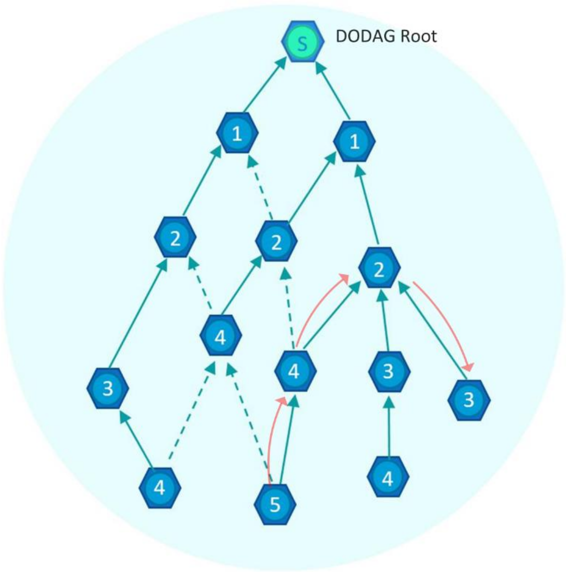


Figure 2.12 Storing Mode: Every node has the map of subtree

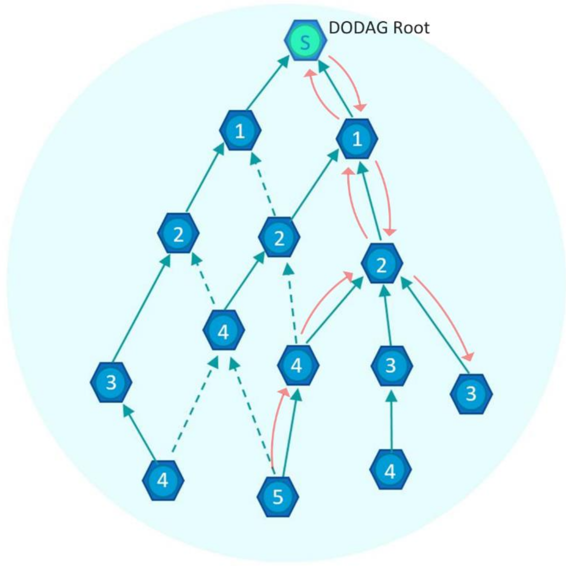


Figure 2.13 Non-Storing Mode: Only the DODAG root has the map of the tree

2.11.4 Objective Functions

The Objective Function (OF) specifies how RPL nodes choose routes within a RPL Instance via selecting the potential parents by translating one or more metrics into the rank value. The used objective function calculates the rank based on some routing metrics such as hop-count, delay, energy, and so forth.

Up to now, the IETF ROLL working group standardised only two different objective functions, namely, Objective Function zero (OF0) [60] and the Minimum Rank with Hysteresis Objective Function (MRHOF) [59]. The OF0 was first developed and known as a basic objective function that uses only the hop count as a routing metric. In other words, this objective function does not take into account any routing metric listed in [61] such as throughput, latency, link quality, and node energy. Moreover, the preferred parent in OF0 will be selected based on the minimum rank for the neighbour nodes. That means the rank increases strictly from the node towards the sink monotonically. Therefore, in OF0, all the forwarded traffic load by the preferred parents towards the sink, do not fulfil any load balancing requirements. On the other hand, unlike OF0, MRHOF is slightly more complicated by emphasizing the hysteresis feature along with the rank calculation in resulting minimizing the churn (i.e. the frequent change of the preferred parent) to mitigate the path cost towards the DODAG root results in better network stability. Typically, the hysteresis feature plays a crucial role via permitting the node to change its preferred parent only if the new candidate parent rank differs remarkably from the current parent rank. A threshold can be pre-defined to compromise between quality and stability. If the threshold is too small, then the churn will be increased for the sake of quality.

Thus, the node's rank that calculated based on the metrics that are injected into the metrics container, which encompasses both metrics and constraints, considered either as link or node metrics. The default metric for the standard MRHOF is the Expected Transmission Count (ETX) metric. The main goal of ETX is to select routes with high end-to-end throughput, which is defined as follows [62]:

$$ETX = \frac{1}{d_f \times d_r} \tag{2.3}$$

where d_f indicates to the forward delivery ratio which is the measured probability that a packet is received by a neighbour. The d_r indicates to the reverse delivery ratio which is the probability that an acknowledgment packet is successfully received.

The expected probability that a transmission is successfully received and acknowledged which is represented in $d_f \times d_r$. A sender will retransmit a packet that is not successfully acknowledged.

With eyes wide open, the Objective Function is still a controversial part of the specification. The argument of many that OF swells the traffic load so it is superfluous for RPL that is observable in [63], the authors proposed DAG-based multipath routing protocol (DMR) for routing in mobile sensor networks (MSNs). DMR provides multiple paths to allow mobile sensors to find alternative routes on local and global route failures. DMR constructs a DODAG by using two routing metrics, hop count and LQI without using the objective function. However, the authors compared their protocol to the existing mobile ad hoc network (MANET) routing protocols Ad hoc On-Demand Distance Vector (AODV) [64] and the multipath AOMDV[65], respectively not to the standard RPL that essentially designed for LLNs.

Moreover, the key idea of the OFs separation is to bolster RPL elasticity to fulfil the variety optimisation requirements, which are highly demanding in handling LLNs applications. This freely choose attracts the researchers to propose new metrics and algorithms to enhance RPL.

2.12 Constrained Application Protocol

Constrained Application Protocol (CoAP) [66] is a web transfer protocol, designed by CORE working group of IETF to fulfill the limited capabilities of the constrained nodes. Many applications can benefit from this protocol among the constrained networks and LLNs particularly machine-to-machine (M2M) applications and automated buildings [67] [68] [66] , where there is no place for human intervention. To make this possible, CoAP introduces a resource discovery technique by listening to port 5683.The aim of CoAP is to act as HTTP but in a compact way within the constrained networks. Thus, CoAP is quite similar to HTTP in terms of client/ server model, i.e. the client can make a request on a certain resource already identified by URI, then the server response by code. While, HTTP messages interchange between client and server simultaneously, whereas, CoAP interchanges are asynchronous over UDP, which affected positively on saving energy consumption. CoAP has four different types of messages with a specific role for each, namely, Confirmable, Non-confirmable, Acknowledgement, and Reset.

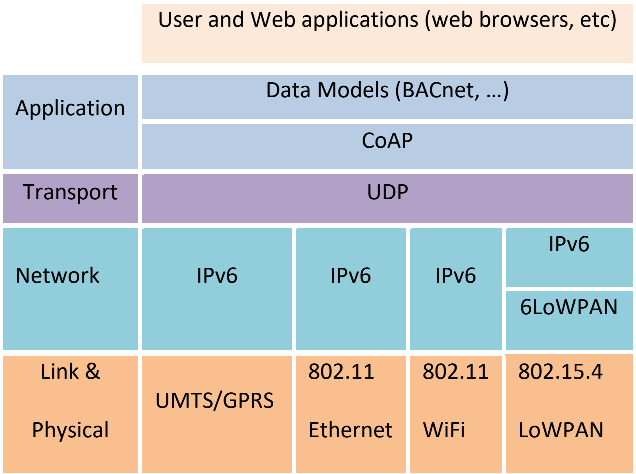


Figure 2.14 CoAP Operates Independently of Lower Layer Technologies

CoAP located within the depicted entire stack in Figure 2.14 [69] of Lower Layer Technologies. In more particular, CoAP can be logically formed in two layers [66], the messaging layer to deal with UDP and asynchronous interactions, and the request/response layer to handle the methods and codes as shown in Figure 2.15.

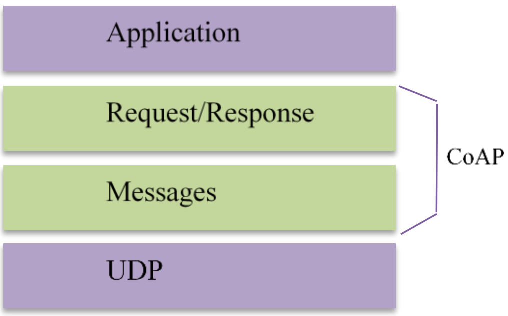


Figure 2.15 Layering of CoAP

2.12.1 CoAP Message Model

CoAP defines four messages types: Confirmable (CON), Non-confirmable (NON), Acknowledgement (ACK) and Reset (RST). To identify the message type, a 2 bit within the message header has been used as listed in Table 2.3.

Table 2.3 CoAP messages type

Type	Name
0	CONfirmable
1	NON-confirmable
2	ACKnowledgement
3	REST(RST)

Generally, each CoAP message has a unique ID to avoid any possibility of duplication, which is compacted to 16-bit size and can hold 250 messages/sec from one end-point to another. However, due to the asynchronous exchanging of the CoAP messages, they may arrive out of order leading to duplication or missing part of the messages.

To tackle this problem, a mechanism has been used to ensure the reliability in CoAP via marking the sent message as a CON message within the CoAP header, which always carries either a request or response. Having this approach means that the sender should retransmit the confirmable message until receives an acknowledgment message (ACK) with the same message ID.

In this context, the retransmission process can be determined in two parameters: a timeout and the retransmission counter. The default timeout is assigned randomly and the retransmission counter is set zero. Thus, if the timeout is over that triggers the retransmission process to proceed only if the retransmission counter is still less than MAX_RETRANSMIT. Meanwhile, the counter will be increased and the timeout will be doubled (which means the waiting time interval (backing-off) will be increased exponentially to ensure congestion control comfortably. Once the retransmission counter is equaled to MAX_RETRANSMIT or the requester receives a Reset message then the failure of transmission process is confirmed. In contrast, the transmission will be considered successful if the sender receives acknowledgment.

Nevertheless, this message could be rejected if the receiver could not process the confirmable message (i.e., the message is empty or has an error); in this case, the receiver consequently must either reply with a reset message (RST) or simply ignore it.

On the other hand, if the message is unreliable type then a NON message can be used instead. The NON message does not need any ACK message but the message must contain an ID for duplicate detection (see Figure 2.16, in this example, 12y7xe). Unlike CON messages, the NON-message is dedicated to carry either a request or a response and should not be empty.

A good example of such a message is the regular update messages collected from the peripherals sensors. Nevertheless, if the receiver struggled to process the message then the only option is to reject it and reply with reset message (RST) similarly to CON messages.

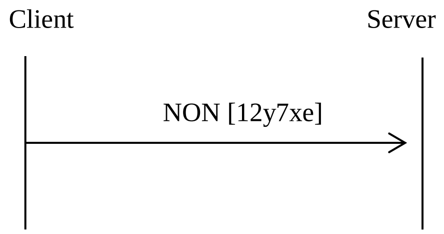


Figure 2.16 Unreliable Message Transmission

2.12.2 Message Format

The CoAP message components are simply: header, token, optional information, and payload. In this regard, it is worth to describe each component and what does it contain in a brief. The token value follows the header in the message format, and the given size for the token is up to 8-byte length.

The main role of the token value is to narrate all requests and responses together based on request/response matching rules which token is part of them.

Ver	T	TKL	Code	Message ID
Token (if any, TKL bytes) ...				
Options (if any) ...				
1 1 1 1 1 1 1 1			Payload (if any) ...	

Figure 2.17 CoAP message Format

. The last part of the message is the optional payload, which mainly carries the content of the message itself to the recipient as shown in Figure 2.17.

The description of each component within the CoAP message header are summarised in Table 2.4.

Table 2.4 CoAP message header descriptions

Char	**Stands for**	**Description**
Ver	Version	2-bit unsigned integer. Indicates the CoAP version number. Implementations of this specification MUST set this field to 1 (01 binary). Other values are reserved for future versions. Messages with unknown version numbers MUST be silently ignored.
T	Type	2-bit unsigned integer. Indicates if this message is of type Confirmable (0), Non-confirmable (1), Acknowledgement (2), or Reset (3)
TKL	Token Length	(TKL): 4-bit unsigned integer. Indicates the length of the variable-length Token field (0-8 bytes). Lengths 9-15 are reserved, MUST NOT be sent, and MUST be processed as a message format error.
Code	Code	8-bit unsigned integer, split into a 3-bit class which indicates a request (0), a success response (2), a client error response (4), or a server error response (5).
Message ID		16-bit unsigned integer in network byte order. Used to detect message duplication and to match messages of type Acknowledgement/Reset to messages of type Confirmable/ Non-confirmable.

2.13 Summery

In this Chapter, we provided a historical and essential background of WSNs. The IoT potential applications are also presented with more focus on the near future trend of such applications and their impacts. Moreover, this chapter discussed the essential background to facilitate the understanding of the research problems within this thesis. Thus, the evolutionary process of the IoT stack has been introduced and how accordingly this stack was generated from the OSI and WSNs stacks. We also made an overview of the protocols accommodated in the current IoT standardised stack. The protocol RPL in the network layer has been given more attention

In the next chapter, we survey a range of existing load balancing solutions in RPL as part of the literature review along with the current approaches that have been used for the antagonistic metrics.

3

Literature Review

2.14 Introduction

In this chapter, we survey a range of existing load balancing solutions in RPL and then categorise them as given in Section 3.3. Moreover, the context-awareness in RPL and the attributed metrics are presented with new classification given in Section 3.6. Then, the chapter concludes with a discussion on the gaps found in RPL protocol, which consequently initiated the aim of this study.

2.15 Load balancing solutions in LLNs

Given the remarkable pragmatic connection between load balancing and network lifetime, and the fact that minimizing the power consumption in WSNs [70] and LLNs plays a key role to extend the network lifetime, which is the focus of the most existing protocols. However, the traffic load balancing is also obligatory to improve the entire network lifetime, where each node consumes the comparable amount of energy. Generally, when it comes to the structure or to the participating style of the load balancing, the authors in [71] categorised the existing approaches into hierarchical and flat routing protocols. In hierarchical approach, the network is formed into clusters to prolong the network lifetime and allowing scalability. There are two main phases in clustering approach, bootstrapping and clustering. While in the flat approach, which is more preferable within the heterogeneous traffic pattern and non-uniform distribution [72], the design comes with multi-hop fashion which makes the load balancing in this approach more challenging due to the lack of global information.

2.16Load balancing in RPL

In the literature, different classifications can be found for WSNs [73] from different perspectives. However, the existing solutions for load balancing in RPL can be categorized as follows:

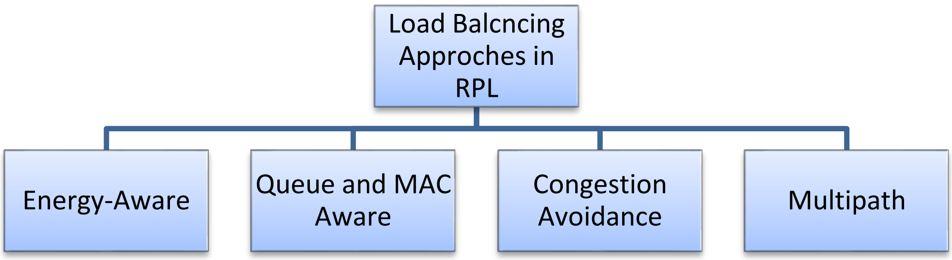


Figure 0.1 Load balancing Solutions for RPL

2.16.1Energy-Aware

Chen et al. were the first ones to perform energy-balancing routing in WSNs [73]. Periodically, the sink selects the path on which to send the data packets, based on the energy level of the different paths it maintains. The authors proposed to compute the optimal paths in a centralized way, by collecting the whole radio topology at the base station. Since the topology may change dynamically, and the overhead is important for such a network, this approach may perform poorly in practice. In [74], the energy-balancing routing approach has been proposed in WSNs for the first time . It is notable that the vast majority of the existing work in the literature focus on the energy issue [75], which normally affected by the imbalance load within the RPL. Therefore, most of the proposed solutions strived to introduce the lifetime issue in line with load balancing issue.

In [10] the authors presented an algorithm called A Load Balancing Model for RPL (ALABAMO) to tackle the unbalance load traffic problem, by considering the traffic profile. Within the proposed OF, ALABAMO employs a hysteresis technique to decrease the oscillation of changing the parent. The weighting approach has been used in this solution along with two auxiliary constants (MaxETXRatio, MaxWorkloadRatio) to optimise the Expected Transmission Count (ETX) with less churn, more elasticity, and better load traffic distribution. Thus, the preferred parent will be chosen with the least number of transmitted packets. Despite the proposed protocol normalized the network lifetime fairly, however, it could not provide better packet delivery ratio (with less 22% in LABAMO-80 and with less 44% in LABAMO-90) compared to the standards OF MHROF due to selecting the parent with lower quality links. This PDR ratio, that this study came with, might be not acceptable in many LLNs applications.

Another proposed protocol for load balancing in LLNs can be found in [86]. This protocol spreads the load through a set of braided paths to minimize the holistic transmission cost. However, the minimum cost load balanced multipath protocol omitted the QoS with the multi-class minimum cost load distribution scheme. The authors of [87] proposed a mechanism to anticipate the remaining energy within bottleneck nodes via suggested a new metric called expected lifetime metric. Accordingly, an algorithm has been proposed for multiple parents in RPL to boost the network lifetime. That can be achieved through mitigating the number of DODAG reconstructions

To prolong the survival time of the network, the authors in [76] proposed an algorithm combines the node metric along with the link metric to push the energy utilisation to the full extent limit as they stated. In this study, a lifecycle index (LCI) measurement has been proposed to predict the bottleneck of candidate path. In [77] the authors conducted a performance evaluation study for RPL energy-aware routing metrics in grid and random topologies. They concluded that using ETX metric in the standard MRHOF [59] might benefit indirectly the end-to-end delay towards the DODAG root, however, that affects the network lifetime negatively. In [78], the authors proposed a dynamic parent selection algorithm considering the recent traffic on the path towards the sink and the remaining energy besides the signal strength indicator (RSSI) as a composite metric. The collected information from DIO and beacon in MAC layer keeps those metrics updated. Another energy-aware algorithm presented in [79], the energy-aware Objective Function (EAOF) improves the network lifetime using the ETX metric along with residual energy metric in a lexical manner. The proposed OF has been implemented via Contiki/Cooja simulator, and investigated in terms of packet reception ratio and network lifetime.

In [80] the authors propose a remaining energy-aware objective function to equalize the residual energy among the nodes in an attempt to prolong the lifetime for the entire network. The proposed objective function considers the remaining energy metric along with ETX metric in two ways, either considering the two metrics for only the node with the next-hop, or calculate both metrics for the entire path towards the DODAG root. Their findings, which have been generated from two scenarios in dense symmetric and sparse asymmetric networks, present a good progression in terms of the lifetime. However, in terms of the throughput, the MRHOF behaves better than the proposed OFs. Moreover, the end-to-end delay was omitted and never statistically studied to show the superiority of the proposed OFs over the standard ones.

Lastly, the main criticism against relying on the remaining energy as a metric is relatively the long time that the OF has to wait/run before recognising the difference in the remaining energy among the candidate parents as result of unbalanced load traffic. Moreover, the attempt of equalising the power consumption among the nodes could induce extra power consumption, especially when selecting a long bypass route to avert the risk of low power nodes.

2.16.2 Queue and MAC aware

The authors in [44], [95], [82], [83] highlighted the congested (overloaded) parent node problem in RPL, which results in losing many numbers of children packets due to the parent's buffer limitation, in addition to unfair energy consumption. In [8], [81] the proposed protocol queue utilisation based RPL (QU-RPL) balances the load and enhances the end-to-end packet delivery performance. Fairly, it selects the parent node based on the queue utilisation, in addition to the number of hops towards the root by considering the both in the rank calculation, which is not completely clear how the hop count has been used in equation 5. The implemented protocol has been investigated extensively using testbed multi-hop nodes and the findings show a good improvement in terms of end-to-end packet delivery performance. However, the attendance of power consumption results, which have been omitted in this study, will add an essential clarity element to the entire picture of the proposed protocol performance. While in [82], the authors proposed an Adaptive Binary Exponential Backoff (ABEB) Algorithm, which defines several parameters to calculate the buffer overflow probability. The proposed algorithm presented a good superiority over the compared protocols in terms of packet drop count. However, it would be better if the study provided a sheer investigation in terms of the power consumption which obviously reflects the quality of the used links instead of the congested ones. The author in [84] [85] [86] presented dynamical interaction of MAC IEEE802.15.4 and RPL layers in LLNs in a cross-layer manner using the mathematical model approach. Two metrics considering the dynamic behaviour of the MAC and routing. Particularly, the proposed protocol relied on the effects of the level of contention to extend the ETX reliability and, thus, to improve the load balancing and prolong the lifetime. In this joint model, two new metrics have been introduced in MAC layer, the first is *R*-metric which reflects the extended ETX via including the number of losses packets, thus, reflecting the end-to-end reliability between two nodes. The second metric is *Q*-metric which measures the contention level without measuring the queues. In [87] the authors propose the M-CoLBA Multichannel Collaborative Load Balancing Algorithm with queue overflow avoidance protocol aiming at improving the throughput and upgrading the bandwidth capacity via reducing interference and collisions, and accordingly congestion awareness. Also, this protocol targeting the losing packets minimisation as result of queues overflow, and in return achieving better load balancing. Generally, their findings show good progression over the standard RPL, however, one of the key elements for any protocol in LLNs is the power consumption. It would be more beneficial if this key element has been considered in this analysis.

2.16.3 Congestion Avoidance

One of the main issues that can affect the LLNs performance negatively is the congestion. The congestion issue has been widely debated and challenged in the literature [88], [89] as the crucial impact which can cause as shown in Fig. 3.2. In this section, congestion avoidance routing techniques, aimed at averting congestion from incidence.

In [90] the authors propose a congestion avoidance (CA-RPL) routing protocol. Using a composite metric which comprises several metrics, the link reliability, load balance, and end-to-end delay alleviation have been targeted to be improved in the proposed protocol. Contiki/Cooja has been used to evaluate the proposed protocol. Another new approach has been used in [91] called content centric routing (CCR), where the content has been used to construct the routing paths in an attempt to reduce the delay and balance the traffic. While in [92] the authors use Buffer Occupancy (BO) alongside ETX as a composite metric to balance the traffic and in return mitigate the number of lost packets.

However, in [93] the authors present the residual energy and the queue utilisation of neighbouring nodes as one composite metric, considering the weighting approach to prioritise each metric with a higher constant. In this solution, the authors aim to improve the lifetime among Smart Grid Advanced Metering Infrastructure (AMI) Networks in Smart City via creating energy and congestion routing awareness.

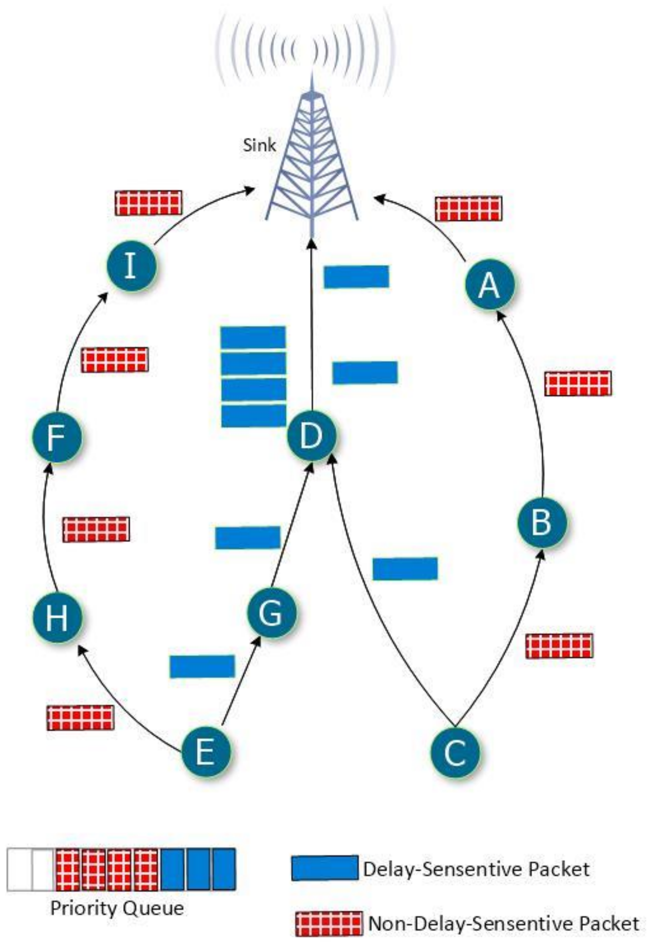


Figure. 0.2 Congested route[88], [89]

2.16.4Multi-path approach

The multi-path approach has been widely considered in the literature[76][94]in an attempt to achieve multifold benefits, such as fault-tolerance, better QaS, and of course better load balancing. Among the tree structure, the multi-path or multi-parents approach is not preferred to avoid loops. However, in RPL this problem has been tackled using the rank, as aforementioned in the previous chapters, the OF calculating the rank and monotonically decreases toward the DODAG root [53]. Thus, the multi-path approach has been considered in several solutions in the literature.

The authors in [9] [95] present a new routing metric, namely, expected lifetime (ELT) that combines traffic load and link quality [68]. Hence, by utilising ELT, a multi-parent routing for RPL has been proposed with aim at balancing the energy consumption over the path towards the DODAG root. The performance evaluation of the proposed algorithms has been conducted via WSNet simulations [96], which still insufficient to show the truly behaviour of the resource constrained nodes.

Also, A multi-path opportunistic algorithm has been proposed in [97]. The authors integrate RPL and IEEE 802.15.4 to enable QoS routing and achieve balance power consumption among the nodes. Using the WSNet/Worldsens event-driven simulator [96], the proposed algorithm attains a slight improvement in terms of end-to-end delay and PDR.

The authors in [98] attempt to enhance RPL with a multipath trait. The proposed algorithm considers the node with the same rank to be selected as a preferred parent rather than the node with lower rank as the case in the standard RPL. The load balancing has been considered as well by including the residual energy metric along with the hop count to calculate the rank. Their findings have been conducted using the OMNET++ simulator which show slightly a good improvement, however, considering the churn analysis of switching parents in such an algorithm could give better clarity and accuracy to recognise the superiority of the proposed algorithm over the standard OFs. Furthermore, it is not fully cleared from the provided results the behaviour of the proposed algorithm compares to the basic or standard RPL.

Another solution based on multi-path approach proposed in [99]. The authors present an energy equalization routing protocol to push the surviving time longer. In the same study, a multi-path forwarding route based on the cache utilisation has been proposed, where the mechanism which has been used for parent selection was according to a steady probability along with the preferred node to mitigate the buffer overload. Their findings show a good progress in terms of decreasing the end-to-end delay and the number of DAG reconfigurations, and extending the life time. However, the load balancing has not been achieved for the bottleneck nodes, i.e. the nodes with one hop towards the DAG root. Surely the role of this node is crucial and acting as a gateway, so it is important for any proposed solution for load balancing to engage these nodes with positive impact. Finally, the authors in [100] Introduces the Parent-Aware Objective Function (PAOF). The proposed solution uses both ETX along with the parent count metric, in a lexical manner, to select the best rout towards the DODAG root. The authors used Cooja simulator to conduct their results that show a good improvement, compared to MRHOF, in terms of Delay and Average Parent Load Density. However, it would be nice to show include the power consumption as it is one of the key metrics for such devices.

2.17 Context-awareness Definition

Among many descriptions of the context, the most comprehensive description has been stated by Dey: *"Context is any information that can be used to characterize the situation of an* *entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.”*[101].

However, the mentioned interaction in the previous definition is not necessary to be exclusive between the user and the application only, it could be between the objects (i.e. sensors) themselves [102] as the case in the proposed protocol. The context that can be acquired via the sensor nodes could be :(temperature, location, power level, and so forth), generally, any information affects the routing might be considered as a context. Thus, the context-aware routing, which has a dynamic network, employs the above information (context) to fulfil the application requirements. However, the obvious argument in the sense that the context-aware routing would be more beneficial in the reactive or proactive routing. It depends on how often the context is changing [102], for example, if the context becomes different infrequently then the context-aware proactive routing is ideally and vice versa.

As the RPL is a proactive routing protocol and the context changes in the emergency systems is scarce and uncommon, but could be a life-threatening case, then it is highly recommended to be a context-aware protocol. In addition, the reliability impact caused by the delay during the emergency scenarios is truly critical, which undoubtedly play a crucial role in rescue coordination and saving people’s lives.

2.18 Context awareness in RPL

Hence, the context contains the information that can be utilised, with utmost elasticity without rigid boundaries, to re-prioritise the best metrics that should be used in this case, i.e. according to the new context, in such type of applications, in particular within the real-time and emergency applications. For this reason, context information is highly in need to be considered in RPL.

In response to the above, we need to allow the application layer to feed the routing layer with the context information, which can effectively optimise the routing decision as shown the cross-layer interaction between the three layers in Figure 3.3 [85]. That could be only possible if more than one metric/constraint have been considered within the routing decision to fulfil the needs of the application. Hence, the current work of RPL OFs depends either on an individual routing metric or on a combination of multiple metrics (composite metric) to build the DODAG as given in Fig. 3.4.

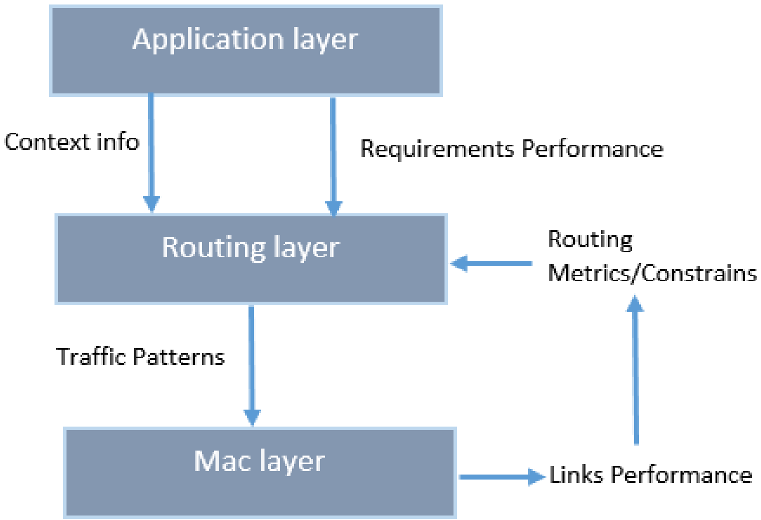


Figure 0.3 Application, Routing and Mac interaction

2.18.1 Single Metric

Undoubtedly using only one metric in the OF, such as the standard OFs i.e., OF0 and MRHOF, is a beneficial form to decrease the amount of complexity within the OF. However, from the other side, there are some drawbacks that affect the optimisation of DODAG construction negatively as result of omitting other metrics within the OF.

In the single metric approach, the performance of DODAG would be deflated due to the application needs that have not been fitted sufficiently. For instance, if the residual power has not been taken into account, then the OF with hop-count metric [60] who are targeting the shortest path, may suffer from a failure because of the battery power may run out. Another example can be noticed, when the OF depends only on ETX [59] as a routing metric, then the latency would be increased as a result of unbalanced load traffic [103] [8], [81], without qualms, this has a serious impact on the real time LLNs applications reliability or delay.

Thus, using the ETX metric in the OF alone is inadequate, as the needed requirements are not satisfied in some applications.

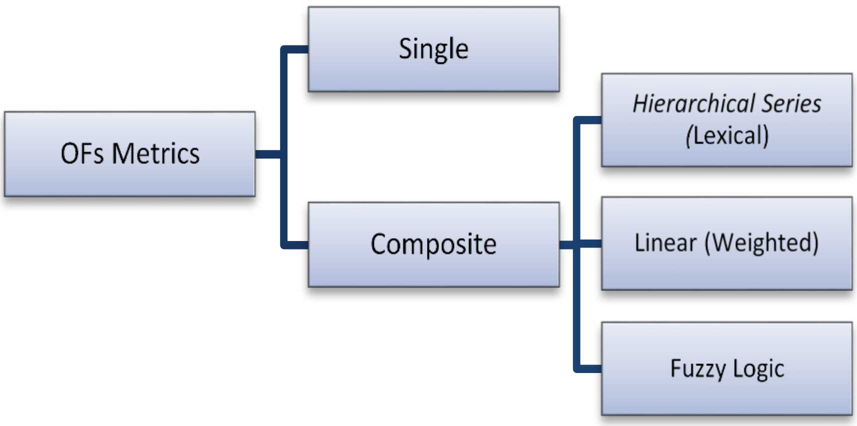


Figure 0.4 Current OF Metrics Approaches

2.18.2 Composite Metric

A number of research efforts have been strived to develop RPL by combining more than one metric in the OF. The existing approaches of combining multiple metrics (composite) fall into one of three categories [104]:

(a) Hierarchical concatenation (Lexical)

If two candidate parents provide the same value of the main metric (i.e. the driven rank), the secondary metric or metrics are triggered to outweigh one of the parents as preferred parent. However, because this similarity case barely happens, it has been considered as the main drawback of this approach and discouraged researchers to use it in RPL enhancement compares to the other approaches. To overcome this point, combining the metrics lexically could be beneficial if the matching of the value conducted based on a range of values rather than on a particular value only.

The key point of the lexical combination is that it is not mandatory to keep the order of the participating routing metrics. For instance, the authors in [105] proposed an algorithm that builds short paths in an attempt to mitigate misbehaving nodes that acting greedily or maliciously. They combined the Hop Count (HC) metric with Packet Forwarding Indication (PFI) metric lexically. In addition, the authors in [79] combined the ETX metric and the residual energy (RE) metric lexically in the proposed protocol Energy-Aware Objective Function (EAOF). The RE metric will be considered only if the calculated rank via ETX for the neighbouring nodes (i.e. potential preferred parents) are quite similar or within the same level. In this case, relaying on the maximum RE metric is the option to decide which one is the preferred parent. Their findings presented improvements in terms of balance energy consumption, and network lifetime. Another Energy-Aware Objective Function in a lexical manner can be seen in [80]. The authors used the remaining energy as a metric besides the ETX metric to optimise the preferred parent selection process.

In [106], a new solution has been proposed named, Cyber-OF: An Adaptive Cyber-Physical Objective Function for Smart Cities Applications. According to the design of the Cyber-OF, the authors inject the alarming information within the DIO packet. Afterward, they consider if the event has been detected then the alarmed node will send a data packet to notify the sink (DODAG root). Consequently, that packet triggers the sink to re-build the DODAG (topology) again according to a new pre-defined objective function that fits the needs of that event. However, being the DODAG re-construction is an event-based will cost more delay and more power consumption. Indeed, this technique will affect the quality of service and the reliability of another network especially within the big networks such as 200 nodes and more.

CAEsAR [107], a Context-Aware Addressing and Routing scheme for service discovery and to support data-centric communication. The authors utilise RPL trees and aggregates context information using Bloom-filters in a probabilistic way [108]. The proposed CAEsAR has been compared with two protocols, namely, an adaptive and context-aware service discovery protocol for 6Lowpans (TRENDY) protocol [109], and MQTT-SN protocol [110], which are both located in the application layer.

(b) Linear combination (Additive)

In this approach, all the considered metrics will be combined in a linear or additive manner to be calculated in one composite metric as shown in equation 3.1. Using the weighting technique, the variable (β) for example, to priories each input (metric).

$$C_{metric}=\beta_1 m_1+\beta_2 m_2+\cdots+\beta_n m_n \tag{3.1}$$

Wherein C_{metric} is the output resulting individual metric for parent election, mi is different input variables and β_i is the corresponding constants factor. The weight of coefficients for the input variables is varied based on the importance of the predefined output. This means that a larger coefficients factor to a certain variable will result in its higher impact on the findings. Thus, the most important parameter(s) can be prioritised according to the application needs. However, this approach cannot deal with the opposed requirements. Thus, unlike the

Hierarchical concatenation (Lexical) approach, it is obligatory to stick to the order of the participating routing metrics. Also, the authors in [78] proposed a composite metric including the residual energy, the hop count, the Received Signal Strength Indicator (RSSI), and the ETX using the additive approach along with lexical approach as aforementioned. They stated that combining metrics shows that more than one routing metric is applicable. The rank in this algorithm has been evaluated according to a composition function and then injected into the DIO message. Each metric has been given a weight, and the rank is driven from the sum of these weights based on the equation 3.1. Although combining two routing metrics or more could ameliorate the performance in the DAG but it may, in return, degenerate other performance parameters. Thus, considering many metrics within one OF should always reflect the needs of the application, otherwise, it would be pointless.

The authors in [111] attempt to optimise the power consumption via considering the battery level into the routing decision. They consider the capabilities of the nodes, either the ones that can be recognised prior, such as the nodes battery level, or the ones that can be known posterior, for instance, the node’s sleeping mechanism for saving power. They cogitate the approach in Context-aware Adaptive Routing (CAR) [112] and Sensor Context-aware Adaptive Routing (SCAR) [113], to build the routing decision in a multi-criteria prediction technique. In other words, the driven instantaneous suitability (Si) takes into account the node available resources (i.e., battery level and connectivity (duty-cycle)) in a composite metric weighting (linear) manner to calculate the node rank. Moreover, [114] introduces a solution for packet loss and power depletion issues in RPL under a hefty load traffic. Hence, the authors propose a Context-Aware Routing Metric (CARF) by considering the residual energy and the queue utilisation in an attempt to address the dynamicity of traffic and the so-called Equality Illusion state issue. However, the proposed OF does not consider the impact of antagonistic metrics which has a crucial consequence on the network reliability in emergency scenarios.

(c) Fuzzy logic

Fuzzy logic was first known in 1965 [115], unlike the typical logic concepts (i.e. true and false) that also known as Boolean logic or binary. It mainly deals with the partial truth rather than the full truth. Therefore, the decision that is provided by fuzzy logic combines confronted parameters with various weights to create algebra-reasoning rules (membership functions) via probabilistic logic to express the human reasoning (flexible and nonlinear). As result, generating an accurate mathematical representation to create a decision matrix. Generally, a fuzzy logic system has four units: Fuzzifier, Inference, Fuzzy Knowledge Base, and Defuzzifier, as illustrated in Figure 3.5.

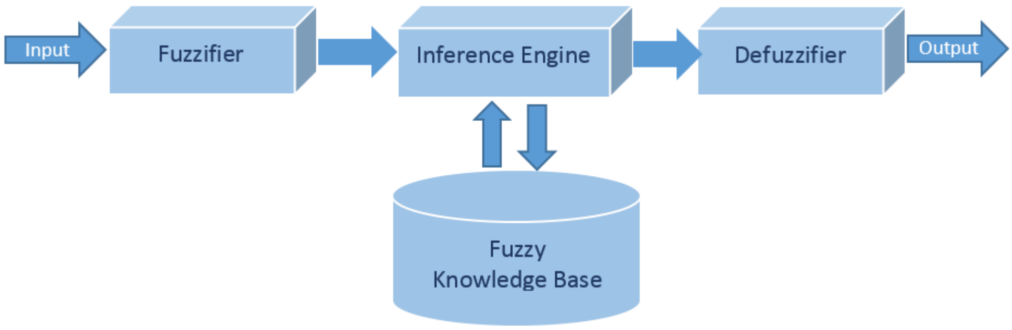


Figure 0.5 Fuzzy Model

Fuzzy logic has been used for WSN for routing such as in [116], for logistics applications [117], for MAC optimisation [118] [119]and for others approaches [120]. The authors in[121] proposed a QoS-aware fuzzy logic objective function. The holistic objective function combines four different metrics, namely, battery level, hop count, delay, and ETX to select the best route towards the DODAG root. Primarily, the authors defined and categorized three properties of the good route (1) real-time: which can be evaluated according to the end-to-end delay, (2) Reliability: this property can be judged based on the packet reception ratio (PRR) in addition to the RSS and ETX, (3) Energy Efficiency: based on the power consumption or the residual energy. The add-on from this study can be recognised by using the fuzzy logic to combine different and heterogeneous metrics in RPL.

Also in [117], the authors used the fuzzy logic to propose Context-aware Monitoring Model for Wireless Sensor Networks in Logistics to extend the node lifetime. Their proposed technique used the temperature as a context input to build the fuzzy rules within the fuzzy knowledge base (*rules base*). Hence, the set of the corresponding actions have been classified in three categories (Alarm, Monitoring, and Normal). Overall, their model utilised the context to adjust the rate of sending the data packets according to the mentioned categories 10sec, 1 minutes, and 2 minutes respectively. However, the routing protocol i.e., RPL has not been considered.

The authors in [122] propose four different OFs to use in one application. These OFs are stored in a DATABASE module to construct new DODAG by switching between them, using fuzzy logic rules, according to the requirements of the application. However, building new DODAG to fit the application requirements according to the context, results in more delay especially in the big network and that may not work effectively in the real-time and emergency applications.

2.18.3 Further discussion on RPL OFs:

In [123] and [124] the authors analysed the performance of the network formation process using a ContikiRPL simulator. Among other parameters, they verified how the two different OFs (i.e. OF0 and MRHOF) influence the average number of hops and the average node energy. The observed differences are insignificant due to the choice of the OFs and their specific parametrization, which results in analogous outcomes when computing the rank. Authors in [125] indicated initial simulation results on the performance of RPL and loading in centralized architecture and scenarios that used less than or equal 50 nodes. However, the traffic patterns, as well as the size of the network tested are still limited. Several researchers have also tried different methods to optimise routing metrics, and OFs for RPL to meet different requirements in specific application scenarios [126] [127]. Moreover, in [128] the authors use the clustering mechanism to improve the energy efficiency of RPL, By using the cluster head node, the clustering probability approach has been considered in competition mechanism

In [129] the authors investigated two OFs using the simulator in addition to testbed motes. The results of the simulation and experimental measurements revealed that a simple hop-count OF leads to a shorter path length at the cost of a higher power consumption. As a mean to increase the lifetime of the network besides the efficient packet delivery ratio, both the energy and the link quality metrics should be used in the OF to obtain an energy efficient network performance. However, if the energy routing metric has been used alone in the OF then it may result in a high packet loss ratio [130].

2.19 Current Limitations in RPL from the OF perspective

The design of RPL tends to lid the limitations of LLNs nodes. However, the load balancing is still a challenging issue in every tree-based topology, and RPL one of them. Typically, every single node has more than one decision to take regarding the parent, consequently, the parent node either has the entire traffic or nearly no traffic at all [131]. Thus, the standard RPL OFs are used to build a DODAG where the bottleneck nodes may suffer from unbalanced traffic load. As a result, the overloaded parent node would drain its energy much faster than the other candidate parent nodes, which might result in early disconnection the part of the network that is covered by that overloaded parent. It is expected that RPL comes with a load balancing-aware OF along with the standards, however, none of the standardised OFs support load balancing [7]. This serious issue has a crucial impact on the lifetime of these types of nodes and may affect the network reliability negatively. Therefore, a reliable and energy efficient topologies balancing the load traffic among the overloaded nodes to ensure node lifetime maximization is highly demanding especially with the scarcity of energy in LLNs.

Another key issue should be addressed in the current RPL in terms of the antagonistic requirements. Some of the quotidian applications consider or use the antagonistic metrics within their routing protocols due to their needs in different scenarios. For example, the contradiction between the metrics end-to-end delay and lifetime, which will result in a destructive impact on the performance of the parent selection as well as the quality of service (QoS) of the protocol. Consequently, the decision accuracy of prioritising the best metric to use to build the routing topology (DODAG in RPL case), which is driven by the OF according to the new context, may not pay off the expected result.

2.20 Load Balancing Solutions Summery

References	Load Balancing Approach: Energy-aware	Load Balancing Approach: Queue & MAC aware	Load Balancing Approach: Congestion Avoidance	Load Balancing Approach: Multipath	Number of Nodes	Environment	Metrics	Composite Metric Approach: Lexical	Composite Metric Approach: Weighted	Composite Metric Approach: Fuzzy	Comments
ALABAMO [10]	√				41	Testbed	Optimised ETX		√		Could not provide better PDR Compares to MRHOF
[81]	√	√			100	Cooja/ Contiki 3.0	Hop count, residual energy, Queue utilisation		√		It would be better if it is tested using real testbed as well. Also, the best weight for each metric is varied and depends on the application.
[78]	√	√		√	144	WSNet	RSSI, residual energy	√	√		WSNet still insufficient to show the truly behaviour of the resource constrained nodes
EAOF [79]	√				26	Cooja/ Contiki	ETX, residual energy	√			The number of nodes is quite small, can be increased easily, to optimise the result more, as the tool is a simulator.

[94]	√				35	Cooja/ Contiki 2.7	ETX, residual energy	√			The throughput in MRHOF behaves better than the proposed OF
QU-RPL [81] [8]		√				Testbed	hop count, queue utilisation		√		Power consumption analysis has been omitted
ABEB [82]		√			N/A	Cooja/ Contiki	Buffer overflow	√			End –to-end delay and power consumption analysis are needed
[84] [85] [86]		√			N/A	Mathe-matical model	(Extended ETX) R-metric and Q-metric		√		Synchronising the MAC layer with the routing layer to balance the traffic.
M-CoLBA [87]		√			80	Cooja/ Contiki 2.7	Node delay, queue utilisation		√		The power consumption has not been investigated in the purposed solution.
CA-RPL [90]			√	√		Contiki/ Cooja	link reliability, load balance, end to end delay		√		Optimising the proposed solution using Fuzzy logic technique might be worth idea to compromise all these metrics together.
CCR [91]			√		20	Cooja/ Contiki 2.7	data traffic status, remaining energy, content variation		√		Giving the limited memory of the LLNs devices, it cannot fit all the different types of contents.
[92]			√		35	Cooja/	Buffer Occupancy, ETX		√		The proposed protocol could generate more churn as result of changing the parents.

						Contiki 2.7					
[93]			√		100	Cooja/ Contiki 3.0	residual energy, queue utilisation, ETX	√	√		Mismatching between the proposed rank calculation equation 9 (based on weight) and the proposed scheme description (based on lexical approach).
[9] [95]				√	50	WSNet	traffic load, link quality		√		WSNet still insufficient to show the truly behaviour of the resource constrained nodes.
[97]				√	256	WSNet/ Worlds-ens	ETX, Delay, time budget	√			It would be better to be deployed in in testbed devices.
[98]				√		OMNET ++	residual energy, hop count	√			Churn analysis was not investigated, and the behaviour of the proposed algorithm is not fully clear compares to the basic or standard RPL
[99]				√		N/A	cache utilisation, less churn	√			The load balancing has not been achieved for the nodes with one hop towards the DAG root.
[122]	√				14	Cooja	Hop count, ETX, residual energy,			√	Building new DODAG to fit the applicat-ion needs according to the context, results in more delay especially in the big network and that may not work effectively in the real time and emergency applications.

2.21 Summary

In this chapter, load balancing techniques for RPL were surveyed and classified based on the energy-aware, queue and MAC aware, congestion avoidance, and multi-path. Then, the context-awareness has been discussed as a definition and its impact on the routing protocols, in particular on RPL. Further discussion has been conducted in terms of compromising more than one metric in RPL in three different approaches to comprise more than one metric within one OF, namely, Lexical, Weighted, and Fuzzy logic. The chapter ended up with categorizing the current approaches in the literature for all current solutions for load balancing in RPL. Afterward, we concluded the current limitations in RPL from the OF point of view, which shaped the aforementioned researched questions in Chapter 1.

The following chapter presents a new solution to balance the traffic considering the number of children within the OF.

4

Tools for IoT

3.1 Introduction

This Chapter surveys the existing LLNs operating systems along with their main features. Moreover, the setting up of the real testbed hardware as an experimental environment also will be presented. Then, an overview description of the tools that have been used as a simulator or a testbed is given, alongside the methodology that has been followed in this research. From the existing work, the chapter concludes with a discussion of the baseline specifications of the LLNs operating systems.

3.2 LLNs Operating Systems (Overview)

The importance of LLNs Operating Systems (OS) lies in the vital prop to simplify the design, coding, testing, and maintenance of IoT software. Typically, a decent combined solution needs to take into account both the hardware platform and the OS (which comprises a protocol stack). The most common OS among IoT teams is Contiki, however, other OSs are flattering the researchers as well, such as RIOT and Tiny OS. The next sections present several OSs that are the most active in the IoT domain.

3.2.1 Contiki

Contiki is a lightweight, highly portable open source operating system developed in C language and dedicated for devices with constrained resources such cases in WSNs and IoT. The Swedish Institute of Computer Science (SICS) released the first version of Contiki in 2003[132], which was supporting only IPv4, however, afterward the uIPv6 stack has been included and became fully supported in addition to the other IoT standards such as 6LoWPAN, RPL, and CoAP. Technically, Contiki has two main partitions the core and the loaded programs [132] as illustrated in Figure 4.1 [96]. The core part contains the kernel, libraries, the program loader, device drivers along with communication stack for the communication hardware. Due to the limited memory within the constrained devices, the Event-driven programming is ideal solution in such case to allocate the needed memory on demand of event. However, the tools to help the programmer control the coding of the explicit state machines, i.e., the blocking wait abstraction within the event-driven, are missing among the common programming languages for embedded systems, for example, C and nesC [133] . Therefore, Contiki uses a unique extremely lightweight, stackless threads called the *protothreads* [133], enabling highly efficient context switching. Unlike the conventional multi-threading, which needs too large memory overhead because of allocating a different stack for each thread, while a protothread only needs to allocate two bytes of memory for each protothread. This novel technique offers a conditional blocking wait statement, PT_WAIT_UNTIL(), to compromise between multi-threaded and event-driven kernel to mitigate the memory-constrained embedded systems consumption to the minimum level alongside with more compact code required. In this regard, two types of event are supported within Contiki kernel: asynchronous and synchronous events.

Furthermore, to build an efficient and portable storage abstractions, Contiki enables interfaces via a file system called *Coffee* file with a tiny and constant RAM footprint per file [134] with low overhead. Another important feature is the timer, hence, the real-time clock for synchronization reasons and for real-time applications can be provided as well. Contiki has mainly two types of timers, namely, etimer and rtimer. The etimer generates timed event, however, the rtimer usage for the callback functions.

The first version of Contiki was released in 2003. A team of developers from the industry and academia developed it. The Contiki project was led by Adam Dunkels of the Networked Embedded Systems Group at the SICS, and since then other developers were

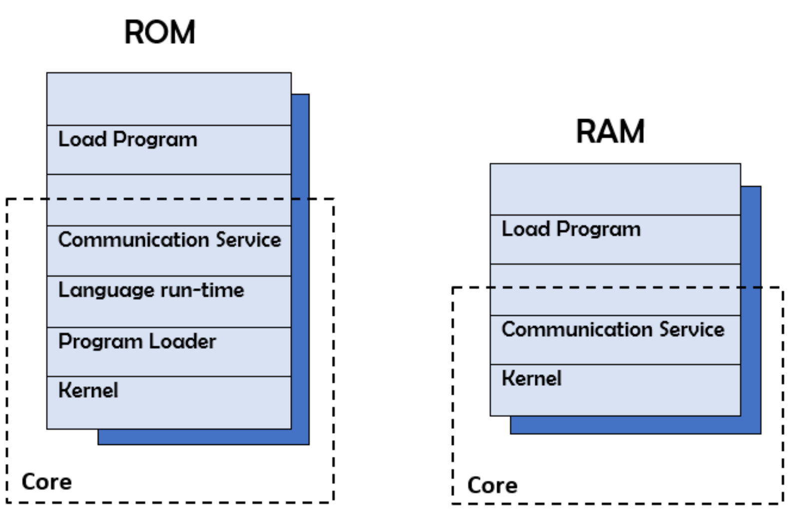


Figure 3.1 Partitioning into core and loaded programs

involved to provide it several new features, and the whole picture of Contiki architecture shown in Figure 4.2 [135].

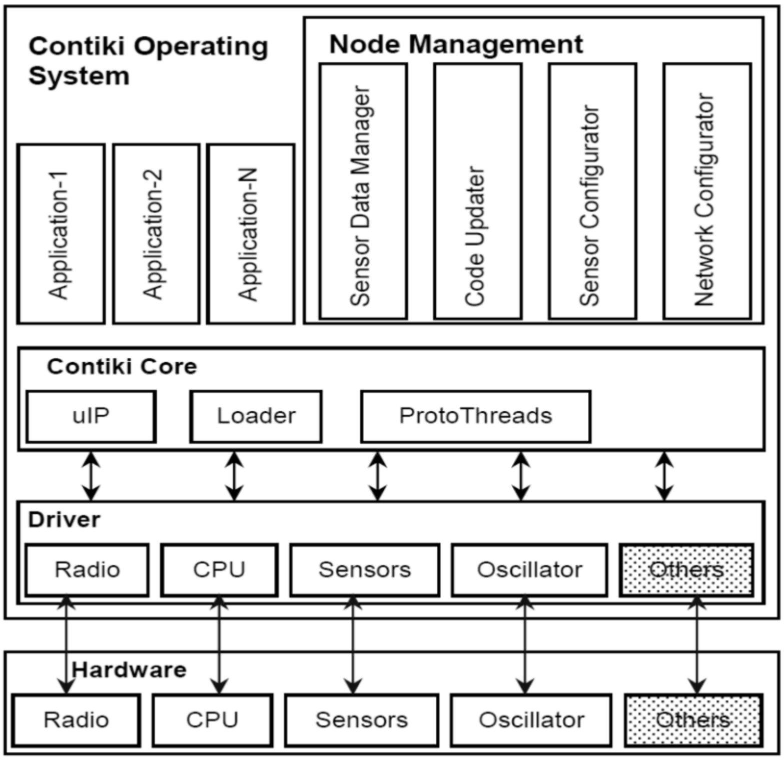


Figure 3.2 Contiki Architecture

A couple of versions have been released for Contiki, the last one was Contiki 3.0 [136]. However, recently a new Contiki cross-platform has been released on Nov 2017 named Contiki-NG 4.0, which licensed under the 3-clause BSD license. Contiki-NG concentrates on one hand, on the Next-Generation IoT devices such as ARM Cortex M3 and other 32-bit MCUs. On the other hand, on reliable and secure standard protocols like IPv6/6LoWPAN, 6TiSCH, RPL, and CoAP [137].

3.2.2 TinyOS

TinyOS is a free event-driven operating system written in nesC, a dialect of C, and developed at the University of California at Berkeley for WSNs nodes to meet their limited resources (e.g., 8K bytes of program memory) [138] [139]. The footprint of OS fits in 400 bytes [140]. The components of TinyOS programs are independent computational entities. These components have three computational elements: *commands, events, and tasks*. Both events and commends are techniques for inter-component communication. However, tasks are dedicated for intra-component concurrency [139], [141]. Overall, TinyOS can be considered as an event driven approach [142] similar to Contiki.

TinyOS has a sole execution stack and considered as a non-blocking approach. Consequently, all I/Os with a duration more than a few hundred microseconds are asynchronous and have a callback. To optimise the native compiler calling, TinyOS utilises nesC's features to associate these callbacks, named events. Considering TinyOS, as enables to achieve high synchronization with one stack in a non-blocking TinyOS, then the complex logic components should be written by linking many small events handlers together. A TinyOS component can execute a task that has been scheduled to be run later by OS. Hence, using GNU toolchain, the compiling of TinyOS code into a small binary became possible [19] [143]. The architecture of TinyOS illustrated in Figure. 4.3.

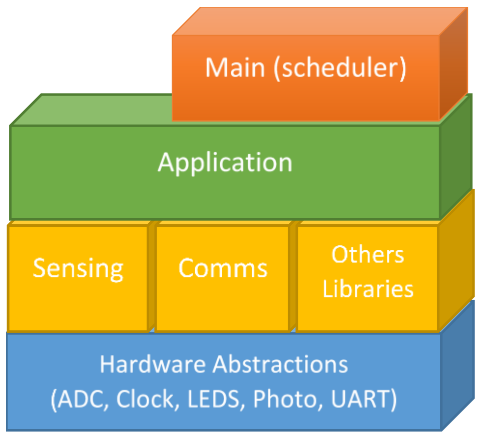


Figure 3.3 TinyOS Architecture

3.2.3 RIOT

RIOT is a free, open source microkernel operating system, developed by a grassroots open community to bridge the gap between OS for WSNs and traditional full-fledged OS on the Internet and to meet the requirements of IoT[144] [145]. The energy-efficiency, small memory footprint, modularity, real-time capabilities, and supporting the low-power constrained devices using C and C++ languages, all together shape the identity of RIOT OS. In addition to including cryptographic libraries, data structures (bloom filters, hash tables, priority queues) [146] and more.

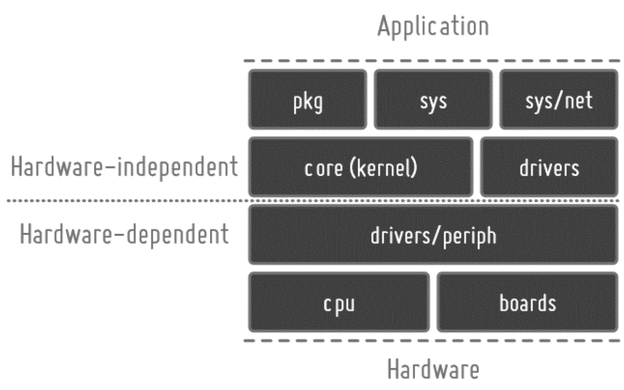


Figure 3.4 RIOT Architecture

Figure. 4.4 concludes the different components of the RIOT structure. The kernel contains the scheduler, inter-process-communication, thread synchronization, and supporting data-structures and type definitions. In terms of CPU and board, one CPU dedicated for each board, but the CPU can be part of many boards. The driver is required for any external device such as sensor or actuator, which is placed on a subdirectory named with a certain device. All the network stack implementations and the network stack agnostic code as header definitions or network types can be found within the sys/net component.

RIOT comes with support for some external libraries such as ccn-lite, microcoap. RIOT comprises that with a custom Makefile for each supported downloadable library with several patches to guarantee the compatibility with RIOT. The Makefiles along with patches are within the pkg directory.

3.2.4 Common IoT OS Comparison

To summarize, the below table provides several features of the afore-discussed OS [29] [147] [148], which can be used as a check-list to choose the proper OS.

	Contiki	Tiny OS	Riot OS
Language support	C	nesC	C/C++
Programming Model	event-driven, protothreads	event-driven	Multi-threading
Supported Protocols	RPL, COAP, uIPv4, uIPv6, , 6LowPAN, MAC IEEE802.15.4, TCP/ UDP	RPL, 6LowPAN, COAP, MAC IEEE802.15.4, TCP	uIPv6, 6LowPAN, MAC IEEE802.15.4, RPL
Real-Time	Partially	No	Yes
Scheduler	Cooperative	Cooperative	Pre-emptive, tickless
Supported Miro Controller Unit (MCU)	AVR®, MSP430™, ARM®, Cortex-M®, PIC32, Skymote/TelosB	AVR®, MSP430™, Skymote/TelosB	AVR®, MSP430™, ARM®, Cortex-M®, x89
Min RAM	<2KB	<1KB	~1.5KB
Min ROM	<30KB	<4KB	~5KB
Dynamic Memory	Yes	Yes	Yes

Table 4.1 Common IoT Operating System Features

3.3 Methods and Tools

In this study, two types of tools have been used to investigate the aforementioned research problems:

3.3.1 Simulation Environment: Cooja

The Cooja simulator[149] is widely used among the research community of IoT, and the percentage of the published work that has been conducted via Cooja in this area reaches 63% [7]. Cooja can be considered as a hybrid approach in terms of the cross-level emulation and simulation tool based on Instant Contiki 2.7 operating system.

Cooja [150] is an elastic Java-based simulator dedicated for WSNs based on Contiki Operating System [132]. The simulated node in Cooja can be of a different type in terms of on-board software, also the simulated hardware. Clearly, the flexibility feature of Cooja enables it to replace or extend many parts smoothly, such as plug-ins or radio medium.

Being Cooja is written in Java, which facilitates any willing for extension via Java Native Interface (JNI) calls from the Java environment, to comprise with Contiki OS, to link it with sensor node software that implemented in C. Additionally, the software of sensor node itself can be run in two ways: running on the simulator as compiled native code for the platform, or running on the emulator as an actual sensor node MSP430 at the hardware level [151]. The stack which is considered for Cooja is Contiki uIPv6 stack as shown in Figure. 4.5:

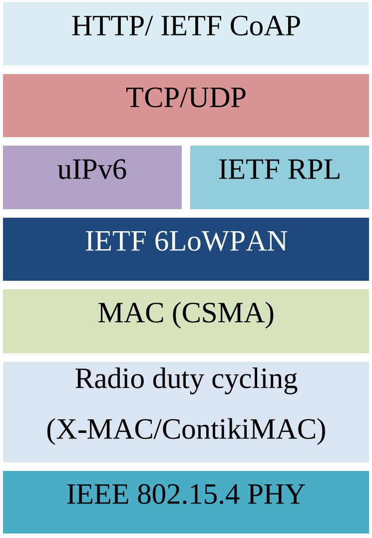


Figure 3.5 Contiki uIPv6 stack

In this regard, Cooja is a hybrid tool in terms of the cross-level emulation for real hardware and simulation virtual nodes. In other words, joining the low-level simulation of device hardware with the simulation of the high-level protocols [150], [152]. That leads to getting one of the significant features of Cooja, which is allowing the simultaneous simulations at three different levels: Network Level, Operating System Level, and Machine code instruction level. Hence, Cooja has been designed to work on three different levels - that it enables the so-called cross level simulations [28].

The key features can be concluded as follows:

- The simulation can be conducted for varied types of motes within a single simulation.
- It supports various radio mediums [153].
- Can be extended easily, for instance adding new plug-ins.
- It allows simulation at three different levels simultaneously [154].
- As an emulator, it gives a lot of specifics in regard of the node’s hardware.

Lastly, the aforementioned features of Cooja, in particular, supporting the hardware-level simulations are the crucial reasons to consider it as a simulator for this study.

3.3.2 IoT Testbed Environment

For experiments, a distributed indoor testbed environment has been configured, using Tmote Sky/TelosB nodes designed by Berkeley [28] . One of the nodes is configured as a root of DODAG while the remaining nodes operated as the sender to send the generated packets towards the root node. All the nodes are running the Contiki operating system version 2.7 and forming a routing tree via using RPL routing protocol. The placement of the nodes/motes is fixed. The experiments were with a single root node for each configuration. It is worth to point that more than one term is used to refer to those devices such as "constrained nodes" and "LLNs nodes" [42].

The realistic indoor testbed environment contains 22 LLN Tmote Sky nodes distributed over the university building as shown in Figure.4.6. The IEEE 802.15.4 motes can measure temperature, relative humidity and light via embedded sensors as shown in Figure 4.7, and Contiki operating system compliant [132].

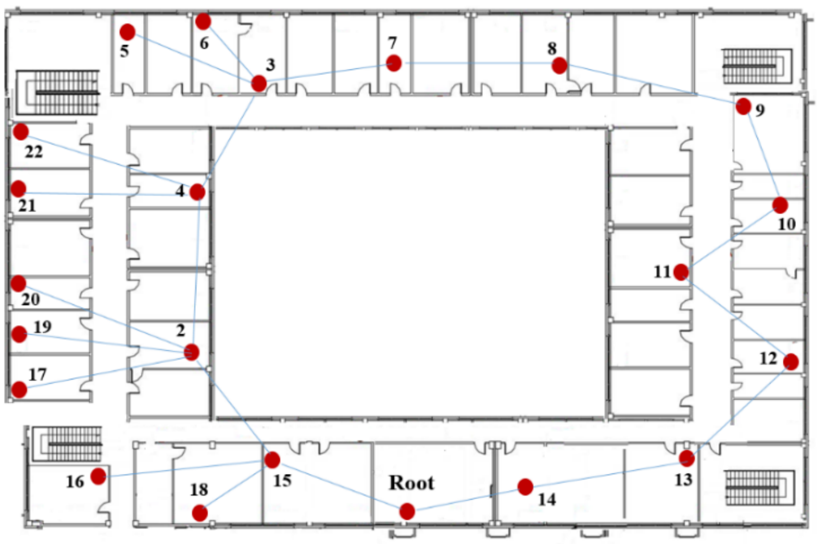


Figure 3.6 Testbed nodes topology in Merchiston campus premises

All the nodes used in the distribution are with a compliant Texas Instruments MSP430 processor with a transmission speed of 250 kbps, and a CC2420 radio working in the 2.4GHz ~ 2.485GHz band and an external +5 dBi SMA antenna. The transmission power within the range -25dBm ~ 0dBm will be fulfilled by 2xAA batteries to power each node as provided in Table4.1. USB support is also provided to host computer for configuration and data acquisition. The following Figures show the front and the back sides of the Temote Sky board.

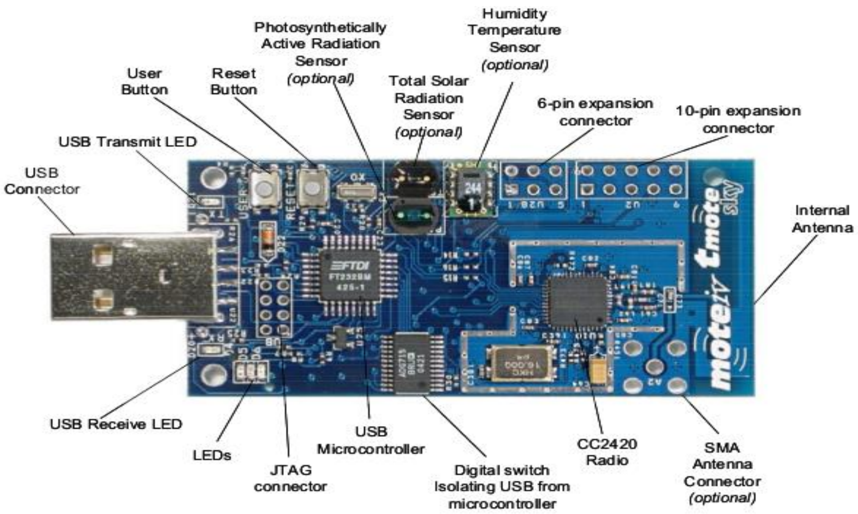


Figure 3.7 Tmote Sky Sensor Boards Front Side

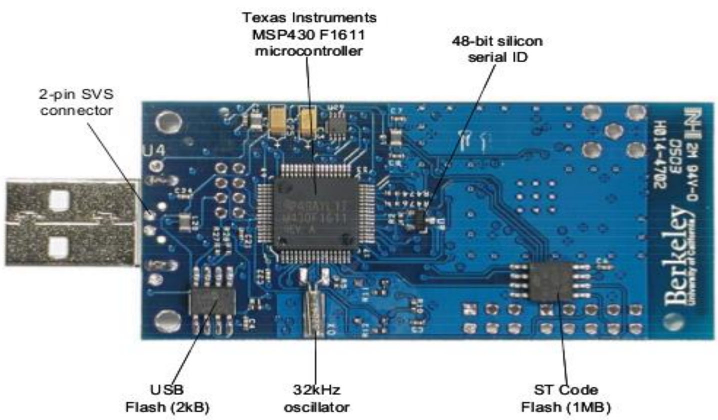


Figure 3.8 Tmote Sky Sensor Boards Back Side

Processor Model	Texas Instruments® MSP430F1611
Memory	48KB /10KB/ 1MB
ADC	12bit resolution
Interfaces	UART, SPI, I2C USB
RF Chip	Texas Instruments® CC2420
Frequency Band	2.4GHz ~ 2.485GHz
Transfer Rate	250Kbps
RF Power	-25dBm ~ 0dBm
Range	~300m(outdoor), 40~50m(indoor)
RF Power Supply	2.1V ~ 3.6V
Antenna	Dipole Antenna / PCB Antenna
Light 1	Hamamatsu® S1087 Series
Light 2	Hamamatsu® S1087 Series
Temperature & Humidity	Sensirion® SHT11
Power	3V (2xAA Battery Holder Provided)

Table 3.1 Specifications of Skymote [155]

The market has plenty and varies of motes as illustrated in Table 4.2. However, the Tmote Sky/TelosB nodes are the best platform that outfits the need of our research.

Architecture	Model	Radio-Chip	MCU	RAM	Flash
AVR	MicaZ	CC2420	ATmega128L	4 KB	128 KB
	Waspmote	8 radios	ATmega128L	8 KB	128 KB
MSP430	Skymote	CC2420	MSP430F1611	48KB 10KB 1MB	48 KB
	XM1000	CC2420	MSP430F268	8 KB	116 KB
ARM	CC2538	integrated 802.15.4 radio	ARM7MC13224	16 KB 32 KB	128 KB 256 KB 512 KB
	Econotag	integrated 802.15.4 radio	ARM7MC13224	96 KB	128 KB

Table 3.2 Specifications of typical constrained devices[21]

3.4 Summary

In this chapter, we showed the operating systems design thoughts for IoT and embedded systems. The focus was on the main features of three common OS systems alongside the basic concepts and compared the current operating systems. It can be concluded that the quandary of building a complete OS on resource-constrained sensor nodes is underlined by the fact that only a scarce have been built in the last few years of research on LLNs. The main features of the common IoT OS are summarised in one table. Then, this chapter presented some feature of the simulator Cooja with some feature that allowed to be the chosen simulator to conduct the experiments of this study. Furthermore, generic concepts about the hardware testbed devices have been given in this chapter, in addition to a description of the configured real environment that has been used in this study. The following chapter presents a new solution to balance the load traffic in RPL considering new metric and technique within the OF.

5

An Efficient Load Balancing Objective Function for RPL

4.1 Introduction

In this chapter, the load-balancing issue in RPL is identified and investigated. Accordingly, a new Load Balancing Objective Function for RPL is introduced to improve the balance of traffic via improving the fairness distribution in the number of child nodes among the overburdened parent nodes, to ensure network lifetime maximization. To implement the proposed objective function, the number of child nodes is injected in the sent DIO message to consider it within the Rank calculation. Furthermore, a new utilisation technique has been used to mitigate any potential extra overhead. Finally, a new RPL metric has been introduced to improve the balance of the traffic load over the LLNs nodes.

4.2 Background and Motivation

The routing protocol RPL has been designed with several robust features such as exiguous delay, quick configuration, loop-free topology, and self-healing to fulfil the needs of the constrained devices (things). However, the load imbalance is considered as a significant weakness in this protocol, as clearly presented in the literature review chapter, which affects the network reliability negatively.

More specifically, RPL is dealing with uniform distribution in large-scale LLNs in addition to the non-uniform ones, which leads to unequal data traffic. Consequently, it may result in significant load imbalance, and the energy of the overburdened nodes will be drained much faster than other nodes. Moreover, this problem has more harmful impacts if the overloaded node is a bottleneck node (i.e., those are with the first hop to the DODAG root) as depicted in Figure 5.1 for nodes 2 and 3.

Accordingly, the severely congested (overloaded) parent node drops many numbers of children packets due to the parent’s buffer limitation [82]. Moreover, the connection over the overloaded parent towards the DODAG root will be fragile, for instance, the entire traffic through node 3, as it is the only link (i.e., acts as a gate) to the DODAG root in such a scenario. This consequently leads to disconnecting this part of the network if the energy of node 3 is depleted.

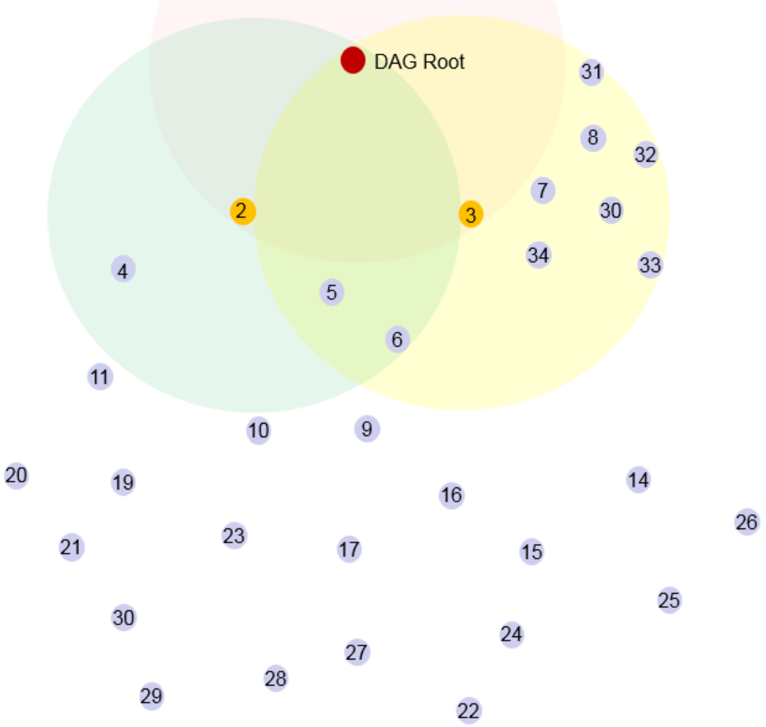


Figure 4.1 Random Topology

Generally, the load balancing is still a challenging issue in every tree-based topology. The reason behind that, typically, every single node has more than one decision to take, consequently, the parent node either has the entire traffic or nearly no traffic at all [131]. Due to the fact that RPL is tree-based protocol, this serious problem befalls due to the parent selection policy in DODAG construction as shown in Figure 5.2, which is only rely on the base of the first received DIO, regardless the received DIO comes from an overloaded parent node or not. Thus, this node (i.e. node 3) will polarise more nodes to choose it as preferred parent.

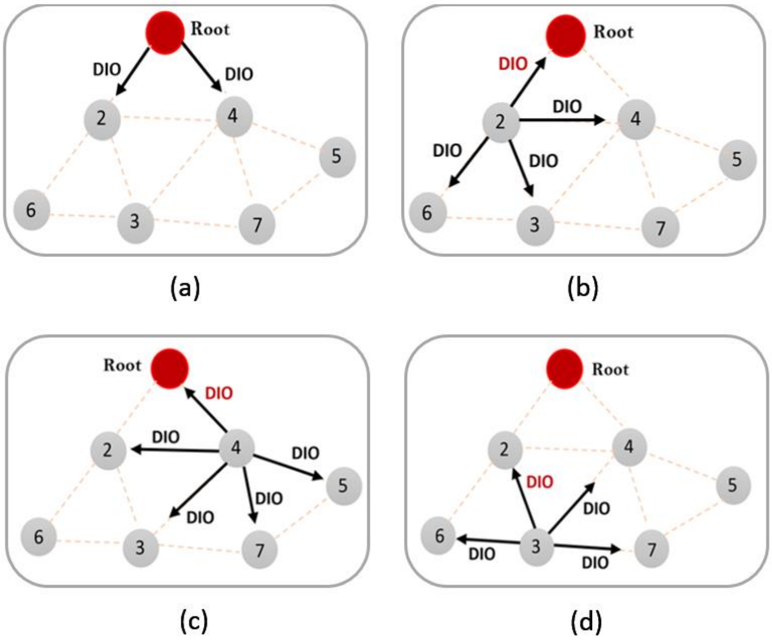


Figure 4.2 The sequence of DIO in DODAG construction in RPL

To this end, the child nodes persist with the current parent as the quality of the link (represented in ETX metric) influences the calculated rank of this parent node, although it deteriorates with more load (i.e. with more children). The ETX metric is calculated as follows:

$$ETX_{new} = \beta \cdot ETX_{old} + (1 - \beta) \cdot ETX_{packet} \quad (5.1)$$

where ETX_{packet} is the total number of transmissions of a packet before being successfully received or dropped and β is the learning ratio ($\beta = 0.9$ *in ContikiRPL*).

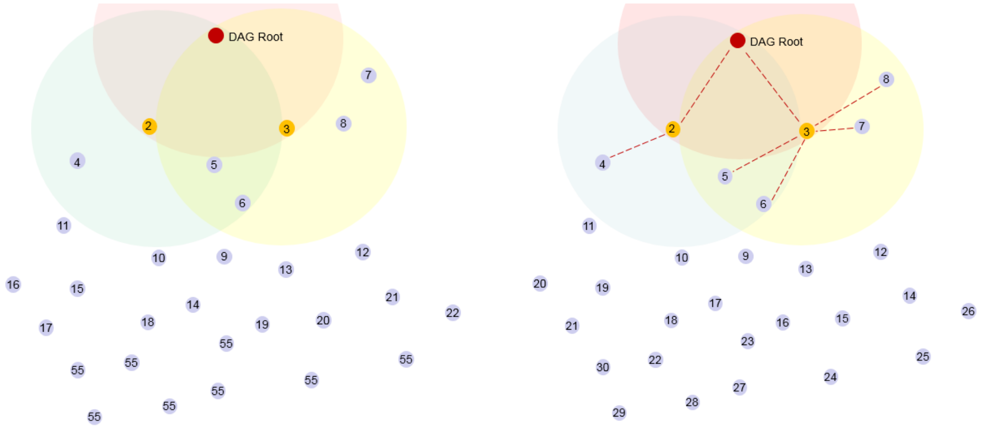
Also, Contiki-RPL normalizes the ETX using an exponentially weighted moving average (EWMA) [156] [157] filter to mitigate the counting of oscillations. Furthermore, it is hard to develop an estimator of the link quality in RPL as DIO messages are not sending periodically according to the Trickle Timer [158]. As aforementioned, ETX metric is clearly insufficient to recognise the load traffic. For that purpose, it is necessary for a new parent selection technique to tackle this problem.

Back to our example, the only conceivable scenarios to change the current parent (i.e. node 3) to another candidate parent (node 2) are as follow: first, if the current parent dies due to battery depletion. The second possibility, when the lossy percentage becomes higher than before, so no acknowledgement message can be heard from the preferred parent for a certain period of time.

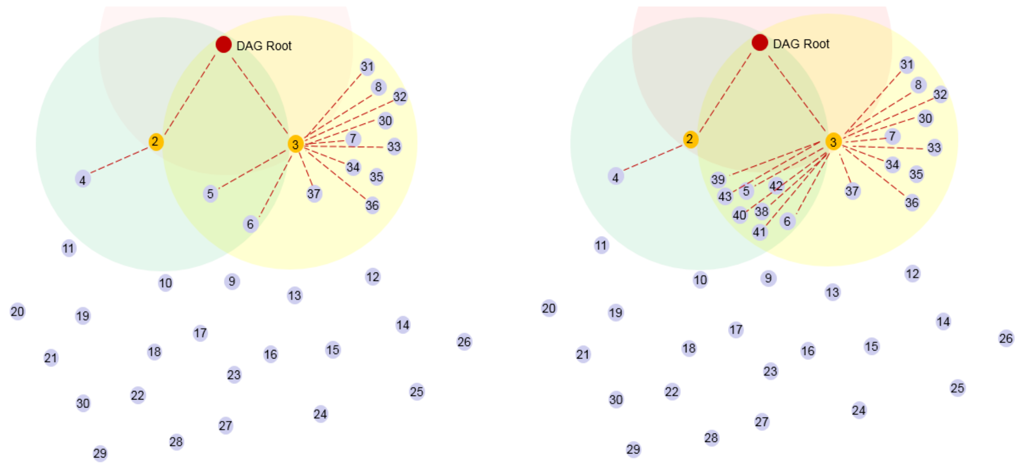
Figure 5.3, sheds more light on the problem, by identifying it in steps. In Figure 5.3 (a) nodes 5 and 6 are within the shared area of nodes 3 and 2. However, nodes 5 and 6 select node 3 as a preferred parent to have in total 4 child nodes and leave node 2 only with one child (node 4) as depicted in Figure 5.3 (b).

Furthermore, in attempt to hand over nodes 5 and 6 to node 2 and force them to select node 2 as a preferred parent, we increased the number of child nodes of node 3 from 4 to 12 children as shown in Figure 5.3 (c). Yet, the nodes 5 and 6 stick to node 3 as a preferred parent even if this parent deteriorates with more load (i.e. being a parent for more children) and despite that node 2 has only one child. A further step has been taken to investigate this issue by increasing the number of nodes within the shared area to up 8 nodes as shown in Figure 5.3 (d), where culminated node 3 by dominating 18 child nodes. Although node 2 has only one child node, node 3 still dominates all the nodes within the shared area (i.e., eight nodes) regardless the total number of the child nodes it has 18 child nodes as summarized in the table provided in Figure 5.3 (e).

It is notable that the connection of all nodes through node 3 is fragile as it is the only the link to the DODAG root with an overloaded bottleneck node, thus, disconnecting part of the network if node 3 dies. In particular, this serious problem occurs in RPL due to omitting the number of children in existing parent selection technique.



(a) (b)



(c) (d)

Parent Node	Number of Children	Children from the shared area
2	1	0
3	18	8

(e)

Figure 4.3 The unbalanced parent selection approach in RPL

This is a key sample of how the fairness of child nodes distribution can be used to achieve even better level of load balancing. As we proposed in the literature review chapter, the existing work is thriving to address the load balancing issue in RPL from various perspectives as shown in the load balancing solution summary in Section 3.7.

4.3 The Proposed Load Balanced OF

Our proposed load balanced objective function (LB-OF leverages the lifetime of the entire network routed, via balancing the data traffic by taking into account the number of children for each candidate parent. Basically, several steps have been put forward as follows:

(1) Amended the DIO option format, (2) A new utilisation technique for the amended DIO, (3) New RPL metric to balance the traffic load. A summary of LB-OF operation is described in Algorithm 1.

Algorithm 1: LB-OF ALGORITHM

Input: ReceivedDIO [InstantID, VersionNumber, Rank, ParentID]

begin
    for $Received \leftarrow DIO$ **do**
        $CurrentNode \leftarrow DIO$;
        if $Sender \notin ParentList$ **then**
            $ParentList \leftarrow Sender$;
        end if
        if $NodeID = ParentID$ **then**
            $Childset \leftarrow Childset + 1$;
        end if
        Calculate the rank considering the number of children;
        if $SendrRank > CurrentRank$ **then**
            Maintain the location in the DODAG;
        else
            Get the parent with lower rank and **Discard** the current parent;
        end if
        Broadcast the updated DIO;

(***1***) The amended DIO option format Typically, the DIO carries the RPL InstanceID, DODAG identifier, version number, Rank and the objective function that has been used to calculate the rank. We amended the DIO by injecting the chosen parent ID into the broadcasted DIO as illustrated in Figure 5.4.

(***2***) A new utilisation technique for the amended DIO: Generally, in the upward routes the root initiates the DODAG construction by sending the first DIO message as shown in Fig. 1 (a). Once other nodes receive this DIO, they select the sender as a preferred parent, and then they start calculating their own ranks based on the assigned objective function. After that, each node broadcasts its own DIO message (i.e. the updated DIO that contains the newly calculated rank value) to all neighbours including the chosen preferred parent who sent the original DIO message as Figure 5.4 (b) depicts.

In the standard objective functions, the preferred parent ignores the DIOs that come from its child. So here in our example, the root drops any DIO, whether this DIO message is sent by node 2 in Figure 5.4 (b) or sent by node 4 in Figure 5.4 (c). In this stage, we aim to allow each parent to count its number of children to normalize the load balance based on it. However, that is not possible in the upward routes, as the only control message that can be acknowledged by the destination is the Destination Advertisement Object (DAO) message in the downward routes.

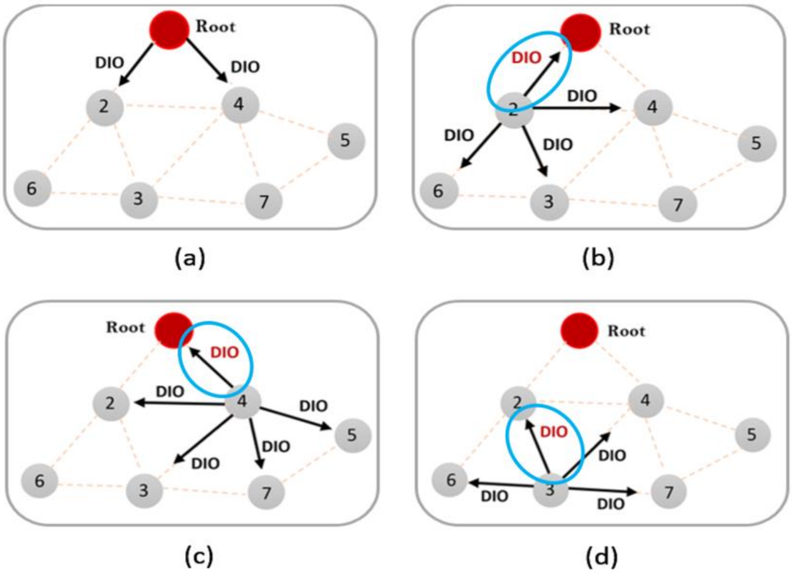


Figure 4.4 Utilising the ignored DIO

Moreover, the DAO messages can be only sent if the DODAG has been already constructed, and our aim is to recognise the number of children for each parent while the DODAG is under construction not after that (i.e. in the upwards routes not in the downwards routes). Alternatively, a handshaking mechanism between parent and children can be set up, but this also brings an extra overhead for the entire network and subsequently increases the power consumption massively. To overcome this problem, we propose LB-OF using a new technique as detailed below.

In the LB-OF algorithm, the received DIO from the child node is counted by the preferred parent node using a special buffer (set) created for this purpose. As mentioned in step (1) the amended DIO contains the ID of the chosen preferred parent. Thus, the node matches its own ID with the preferred parent ID that is inserted in the DIO message, then increments the number of children set by one for this node if there is a matching.

Hence, this technique evades increasing the overhead that can be caused by the handshaking process for the entire network. In addition, we utilise the coming DIO and allow each preferred parent to distinguish the number of its children during the DODAG construction stage. In this regard, it is worth to point why the DIO, in particular, has been considered to count the number of child nodes rather than the DAO messages. The explanation for this located in the RFC 6550 page 78, "*If the MOP is 0, indicating no Downward routing, nodes MUST NOT transmit DAO messages and MAY ignore DAO messages.*" [53]. Thus, the downward routes is an optional feature in RPL, the proposed idea targeting the RPL mode of operation without downward routes.

*(**3**)* Balancing the load traffic: As aforementioned, being a preferred parent for more children means more overhead and unbalanced load consequently drained its own energy much faster than other candidate parents did. To solve this problem, a new metric has been proposed. The children set created in step (2) provides each preferred parent how many children it has. Based on that, we consider the number of children in the rank calculation rather than only ETX.

In more specific words, the parent with the least number of children will be elected as preferred parent. To this end, the balance has been achieved by declining the number of children of the overloaded bottleneck node. As a result, the majority of children will choose another preferred parent according to the lower rank, and surely has less number of children. An illustrative example is given in Figure 5.6, where the unbalanced load traffic can be noticed in MRHOF, where node 3 has 16 children and node 2 has only one child. On the contrary, using LB-OF both bottleneck candidate parents have almost the same number of children, node 3 would have 9 children and node 2 would have 8 children.

It is worth to point that this metric (i.e. the number of children) is implemented within the option field to avoid any extra overhead.

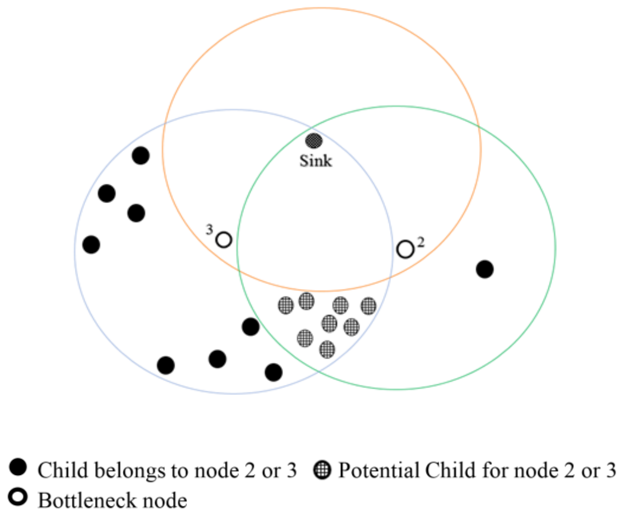


Figure 4.5 The bottleneck nodes

4.4 Experiments and Evaluation

The conducted experiments in this chapter are divided into different scenarios and different densities. These scenarios are designed for evaluating the conventional OFs (OF0 and MRHOF) in RPL which are commonly used for constructing the RPL DODAG in the natural environment for different IoT applications. Thus, we have considered the random distribution topology not the grid one, as the unbalance of traffic mostly comes from the unfairness distribution in particular among the tree-based oriented routing protocols as the case in RPL. Our focus in these scenarios is concentrated on the bottleneck nodes (i.e., the nodes within the first hop to the DODAG root) for the critical consequences of these nodes failure.

Moreover, due to the scarce energy within constrained nodes and the multi-hop manner in RPL, the load balancing consideration has a crucial impact on the power consumption and the network lifetime, especially for, which are typically battery-operated. For that reason, the most sufficient metrics to consider within the evaluation are packet delivery ratio (PDR), energy consumption balancing, and network lifetime. The purpose of these scenarios is to apply the newly proposed routing module, alongside a diverse number of parameters to evaluate the performance of the aforementioned metrics of the network.

4.4.1 Simulation Setup

To evaluate our objective function, we conduct simulation experiments using the Cooja simulator [136] using Contiki Operating System 2.7 [132] . The simulation parameters are given in Table 1.

Table 4.1 SIMULATION CONFIGURATIONS FOR EXPERIMENTS

Parameter	Value
Operating System	Contiki 2.7
Node Type	TMote sky [28]
Routing protocol	RPL
MAC/adaptation layer	ContikiMAC/ 6LowPAN
Radio Environment	Unit Disk Graph Medium (UDGM)
Number of nodes	18,50,100
Simulation duration	30, 80 min
Full battery	3000 mJ
Transmission range	100 m
Data packet timer	60 sec
RPL parameters	MinHopRankIncrease = 256
Mote start-up delay	1000 msec
Simulation area	400×300 m
DIO interval min	12 msec
DIO interval doublings	8 msec
Deployment mode	Random

4.4.2 Results Analysis

Figure 5.6 compares the power consumption between the two critical nodes (i.e. node 2 and node 3) using two different objective functions. In the conventional objective functions MRHOF and OF0, the traffic load is concentrated on the node 2 as it has more children than node 3, this is because node 2 is always sends its DIO before node 3. Consequently, this leads to a significant difference in power consumption. In contrast, in LB-OF, the load balancing has been achieved by taking into account the number of children for each node as a metric.

Clearly, the power consumption of node 2 is mitigated in LO-OF and then relatively stabilized. The main reason behind that is the regression in the number of children from 14 to only 8 children for node 2. On the other hand, node's 3 power consumption is increased as it became a parent for more children (i.e., 6 children) than before (1 child). Therefore, the gap between those two critical nodes has been massively declined using the new LB-OF objection function, and undoubtedly the traffic load is balanced.

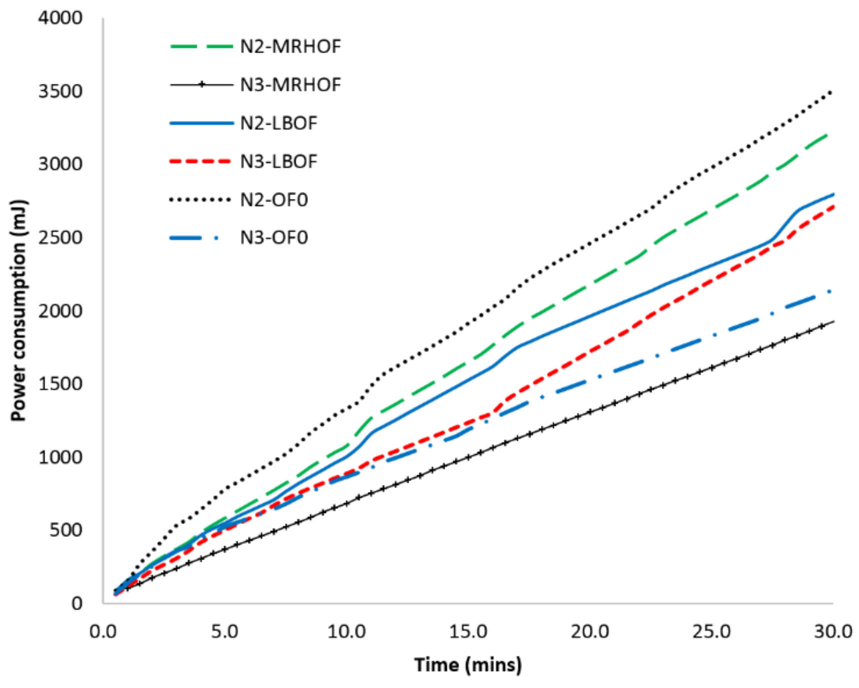


Figure 4.6 Power Consumption Balancing 18 nodes

Likewise, in Figure 5.7 we increased the density of network to 50 nodes rather than 17 as the previous scenario. Thus, we ended up with the same behaviour in Figure 5.6. However, the power consumption levels for both nodes using LB-OF has been raised compared to the results illustrated in Figure 5.6. The main reason behind this is the parent churn, i.e., the fluctuation in the number of children for each parent is quite frequent. That can be concluded from Figure 5.8, where the high fluctuation can be observed at the first 12 mins. Subsequently, node 2 and node 3 became parents for 8 nodes and 6 nodes respectively. Not surprisingly, the freely chosen of the preferred parent with less number of children comes with a cost. However, this churn stabilizes until the end of the second period (33-52) mins, and then fluctuates again but with less oscillation, and that expected, as the number of children for each preferred parent has not stabled yet. Since then, each parent sticks with its number of children with a churn equals to zero as clearly exhibited in Figure 5.8.

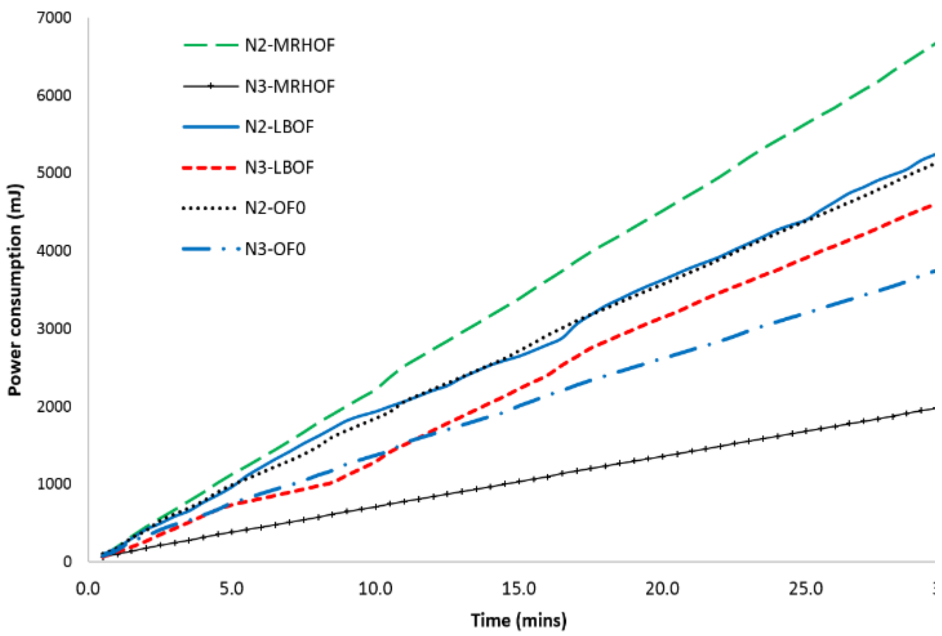


Figure 4.7 Power Consumption Balancing 50 nodes

The ultimate goal of this protocol is to guarantee better load balancing for the bottleneck nodes, in order to avoid the negative impact on the lifetime of one node at the expense of another node lifetime, subsequently, allows evading bottleneck nodes.

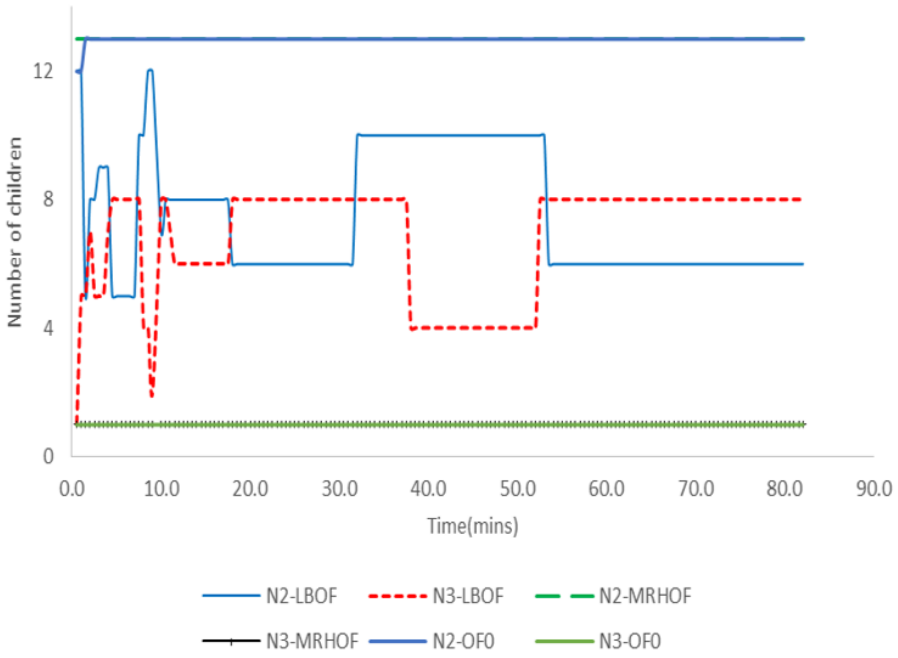


Figure 4.8 Number of children for N2 and N3 (Churn)

So, we evaluated the network lifetime where Figure 5.9 compares the lifetime between the two critical nodes. The unbalanced lifetime is obvious when MRHOF or OF0 is used.

Contrariwise, LB-OF balances both nodes which allow node 2 to sustain longer and minimizing the probability of disconnecting the fragile nodes. Overall, our objective function outperforms the standard RPL MRHOF and OF0 in terms of lifetime.

Figure 5.10 illustrates the packet delivery ratio (PDR) versus network density. The PDR is calculated as the ratio of the number of packets received by the border router and transmitted by each node. It can be noticed that the results of MRHOF, OF0 and LB-OF are nearly identical in terms of PDR when the network density is 18 and with a slight difference when 50 nodes. However, LB-OF presents a better PDR than MRHOF and OF0 when the density is 100. As the number of the lost packet is increased when one critical node serves the majority of nodes and acts as a hub to the root.

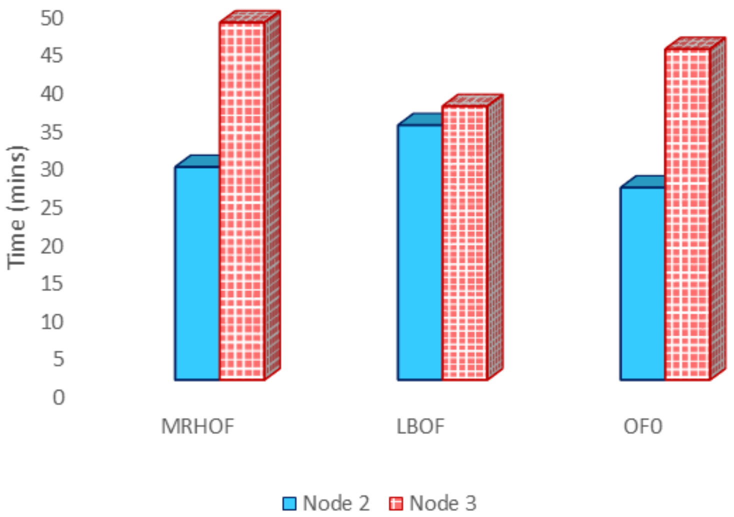


Figure 4.9 First node to die

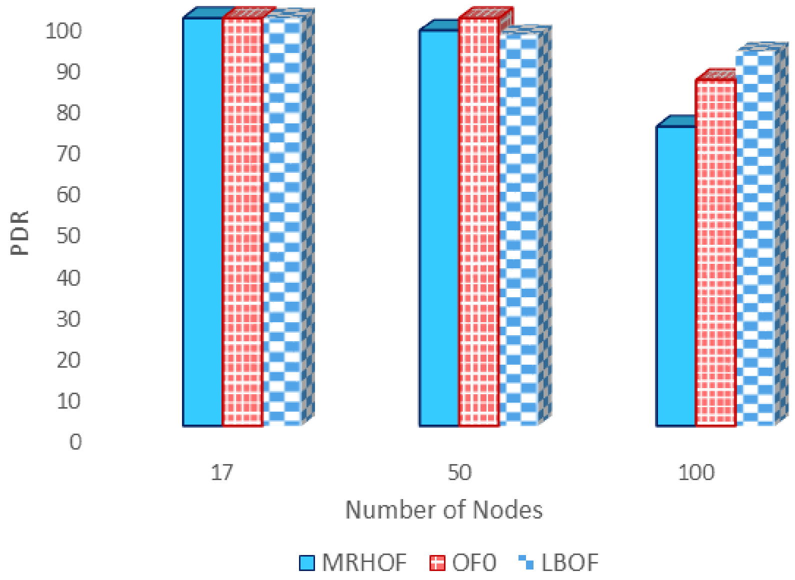


Figure 4.10 Network PDR

4.5 Conclusion

Load balancing is a key issue, but it was not properly addressed when designing existing objective functions in LLNs. In this chapter, we proposed a new load balanced objective function for RPL protocol to achieve a better workload distribution among all nodes in LLNs and battery lifetime thriving for a longer time to come, which in turn to better network reliability. A new technique has been used to count the number of children for each parent without generating any extra overhead and then injected in the DIO. Thus, better fairness of child nodes distribution has been achieved through RPL DODAG construction process. Simulation results proved that the proposed load balanced objective function is better than the MRHOF and OF0 in terms of balanced power consumption, packet delivery rate, balanced lifetime, and balanced number of children for each node.

The next chapter is addressing the antagonistic requirements issue along with the context awareness in RPL and how to optimise the selection decision of the routing path towards the root node according to the new context triggers, and taking into account the antagonistic requirements mitigation approach.

6

Context-adaptive RPL for Antagonistic Requirements of LLNs Applications

5.1 Background and Motivation

More IoT applications have secured a prominent part of our life and covered a wide range of fields starting from smart homes to smart cities, either in the ordinary or hazardous areas. Apparently, the requirements of every application are varied and depend on the specific needs of that application that be used to determine the routing metrics. However, some of these requirements may seem antagonistic but also pertinent and has to be within one application. In other words, the performance of the first metric can be affected or influenced on the expenses of the other metric(s) and, as such, impossible to fulfill them simultaneously [159] such as, but not limited to, network lifetime and minimum-delay requirements. For instance, the antagonistic impact in the forest fire detection system can affect the emergency responses, by increasing the delay, when a fire is detected. This case requires a specific attention to handle an emergency, and it would be more efficient to priorities the minimum end-to-end delay rather than the minimum power consumption.

Moreover, prioritising single metric or composite metric, without considering the new context, cannot pay off the antagonistic needs of the LLNs applications sufficiently. This is in contradiction with the fact that any delay in the emergency scenario should be mitigated to the minimum level. In this regard, it is worth pointing out that the precise category to define whether the application has critical tasks or not is out of this study scope.

Therefore, there is a need to design new routing mechanisms to mitigate the consequences of the antagonistic impact to speed up the delivery of critical data when an emergency occurs. In this chapter, we propose a dynamic context-aware Objective Function to facilitate information exchange between the application and network layers. Hence, feeding the RPL with the context information, to optimise the selection decision for the routing path, is highly recommended in emergency management. In fact, for the sake of efficiency, these types of applications should be smart enough to react adaptively via re-prioritising the application requirements according to the new context needs. To address this issue, we propose an adaptive OF to operate at the trade-off between the antagonistic requirements for the critical IoT applications. Doing so, this chapter provides the following main contributions:

1) A new category for the existing RPL OF approaches.

2) A new algorithm for OF lifetime optimisation based on the accumulated number of children to introduce less power consumption.

3) Adaptive OF based on the context that mitigates the impact of the antagonistic requirements during the emergency scenarios.

5.2 The problem statement

As observed in the related work, RPL is designed with several robust peculiarities such as exiguous delay, quick configuration, loop-free topology, and self-healing, in addition, the flexible design of the standard. In fact, RPL specification does not only separate the OFs, as aforementioned in Section 6.3, it also differentiates between the routing metrics themselves and the OF owing to foster the level of resilience and additivity.

Despite the numerous proposed algorithms for OFs, which have been exploiting a great deal with building RPL topology as outlined in Chapter 3. However, these efforts either focus on combining metrics algorithms or propose new metrics. Whereas the attention that has been paid to tackle the antagonistic requirements problem is erratic and with rigid boundaries in the literature. To the best of our knowledge, the majority of existing works [30,44,46,90,91,93-99,102-105,107,108,110-112] only provide different approaches to combining multiple metrics in a *STATIC* manner (unchangeable with the new context) as shown in Figure 6.1.

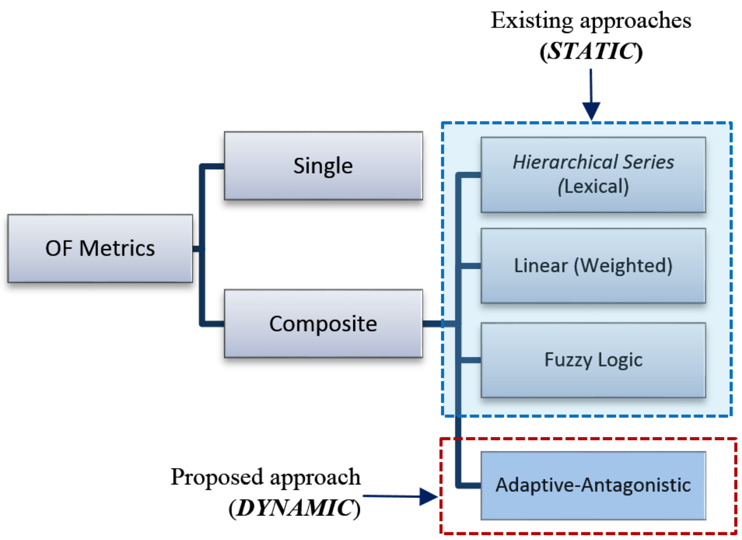


Figure 5.1 OF approaches for composite metrics

In short, this chapter highlights the problem of compromising the antagonistic requirements in two contradiction scenarios:

5.2.1 The impact of antagonistic requirements on an emergency scenario

Typically, the metrics that have been used to build the RPL topology reflect the needs of particular LLNs application in a specific environment. However, due to the dynamicity of the context in any environment prone the application needs to change particularly in the emergency scenarios. To this end, restricting the priority within specific routing metrics, to pay off the application needs for specific environment parameters, is not sufficient in IoT applications. More importantly, the antagonistic requirements problem has more harmful impacts in the emergency scenarios as the consequences in such cases become more critical. For instance, the coexist of the maximum lifetime and minutest delay requirements are impossible to achieve fully functional for both metrics. Those two trade-off metrics are needed within one application in some cases. A good example for this can be seen in the forest fire detection systems, wherein the lifetime is critical for these constrained devices as they powered by a limited battery, and they are in a position where their battery replacement is not an option.

Therefore, pushing the lifetime for the entire network, which could contain thousands of nodes, is truly necessary.

On the other hand, minimizing the delay in these types of applications is highly demanding, particularly within the emergency scenarios. In such a case, i.e. when the fire is detected the priority becomes to receive the notification of the sensing temperature (context), which has been triggered by sensing a fire, rather than considering the minimum power consumption as a priority in this case.

To this end, the current standards RPL OFs lacks to react effectively to adapt and re-shape the topology in such cases, due to omitting the context-awareness of the antagonistic requirements. Thus, the challenging question could be *how to compromise between the antagonistic requirements within one LLNs application?*

Based on the above thoughts, we think there is still a room for RPL enhancement to make it more reliable and adaptive to the context accordingly.

5.2.2 Survival time scenario

On the other hand, the scarcity of power is still a critical problem in LLNs applications, for those powered by a battery in particular analogous to our case, as stated in the previous chapter. Therefore, in the literature, many protocols have been proposed to address the lifetime problem.

5.3 The Proposed Solution

This section is devoted to clarify in details our enhancement for RPL in emergency scenarios as a result of new context. The aim of the proposed protocol is to help different antagonistic requirements to coexist within one LLNs application. Indeed, by the envisioned protocol, the reliability and the lifetime of the entire network routed by RPL will be influenced. Two scenarios have been put forward as follows:

5.3.1 Minimize the power consumption within the Lifetime Scenario

In this subsection, we extend the load balanced objective function (LB-OF) [103] [160] in attempt to relatively equalize the data traffic, and reduce the power consumption for the benefit of network lifetime. Unlike the previous chapter, we take into account the total number of the *accumulated child nodes* for each path towards the sink rather than only the direct child nodes for the preferred parent. Moreover, in the previous contribution, the focus was on the bottleneck (overloaded) nodes, however, this contribution targets all the nodes along the path towards the sink including the bottleneck nodes. Likewise, in the previous contribution, the Parent ID has been injected into the DIO which normally embraces the InstanceID, DODAG identifier, version number, Rank. However, to avoid increasing any overhead, in case the handshaking approach has been used, the ignored DIOs have been utilised instead as depicted in Figure 6.2 (a). Typically, any received DIO contains higher rank compares to the node's rank itself then it will be discarded.

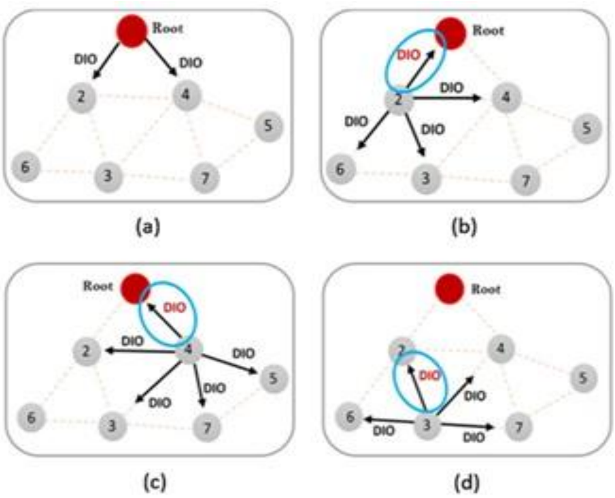


Figure 6.2 (a) Utilising the ignored DIOs

In respect of that, the chosen parent discards any DIO coming from its child as demonstrated in Figure 6.2 as detailed earlier in Section 5.3. This discarded DIO that contains the chosen parent ID will be employed to calculate the *holistic number* of children for each path towards the sink as proposed in equation 6.1.

$$\forall\, Senders \in Parent\ List$$

$$AcNC = \sum_{i=0}^{n} Pnc(i) \tag{6.1}$$

Where the *AcNC* represents the accumulated number of children from the sink to the current hop, *i* represents the hop count until the current node (n), and *Pnc* represents the number of children for one of the parents on the path towards the sink.

Algorithm 1: ACCUMULATED NUMBER OF CHILDREN ALGORITHM

input : Received DIO [InstantID, VersionNumber, Rank, ParentID]

output: Updated DIO

for $Received \leftarrow DIO$ **do**
    $CurrentNode \leftarrow DIO$;
    if $Sender \notin ParentList$ **then**
        $ParentList \leftarrow Sender$;
    end
    if $NodeID = ParentID$ **then**
        $ChildrenSet \leftarrow ChildrenSet + 1$;
    end
    $\forall$sender $\in ParentList$
    $AcNC = \sum_{i=0}^{n} P_{nc}(i)$
    Calculate the rank considering the number of children;
    if $SendrRank > CurrentRank$ **then**
        Maintain the location in the DODAG;
    else
        Get the parent with lower rank;
        Discard the current parent;
    end
    Broadcast the updated DIO;
end

By using this technique, each parent will be able to sum the accumulated children for the entire path along to the sink, in the upward routes approach, to be used to balance the load traffic. In more specific, the path with the least number of children will be selected via choosing the last (edge) node of this path as a preferred parent. This new metric will be used in the OF to optimise the best parent selection, in turn, that affects the network lifetime positively. For the sake of lucidity, Algorithm 1 shows a summary of the extended LB-OF operations along with the flowchart in Figure 6.2 (b).

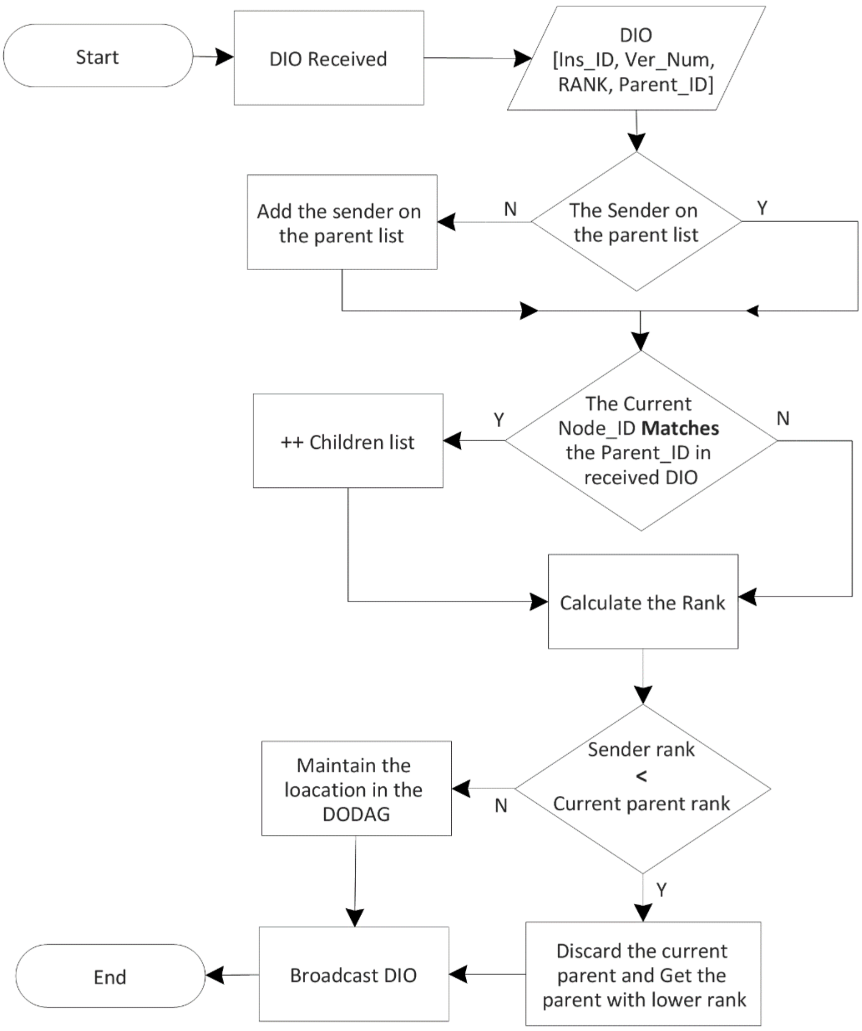


Figure 5.2 (b) The calculation of accumulative number of child nodes flowchart

5.3.2 The Emergency Context Scenario

In this subsection, we consider the forest fire detection system as a case study for the emergency scenario. However, this also can be applied in any real-time application such as emergency in the navigation systems, the medical response in WSNs disasters systems and plenty others. This type of topology is quite common in applications such as environment monitoring, where the sensor nodes are distributed in positions of interest capturing the required information [111]. As discussed, the reliable route needs to avoid overloaded traffic, which is indicated by more number of children, to boost the lifetime. That could mean an extra number of hops towards the DAG root, and that consequently augments the end-to-end delay, which is undesirable in such real-time applications as the consequences are so critical.

Therefore, it should be minimized as far as possible, even if that on the expenses of other requirements or metrics (i.e., lifetime in our case), which imposes a serious challenge in terms of the antagonistic requirements.

As aforementioned previously in Section 6.2, all the current algorithms focused on combining metrics in different manners statically and tried to tackle this problem by considering only the routing protocol itself. However, these approaches are not able to take the advantage of the context to get the full functionality of non-antagonistic metric(s).

In other words, the consequences of the antagonistic metrics degenerate to get the best potential benefits of specific metric(s) in a certain context. While, considering only the most beneficial non-antagonistic metric(s) together for this context apart from the rest metrics, to guarantee the best reaction for the network behaviour in such scenarios.

In the ordinary scenarios, the minimum power consumption is considered as the highest priority to prolong the network lifetime. In this case, as depicted in Figure 6.3, node 7 opted node 8 as a preferred parent because the path through node 8 has less number of accumulated children, compared to the path through node 3, which means less traffic and defiantly less power consumption.

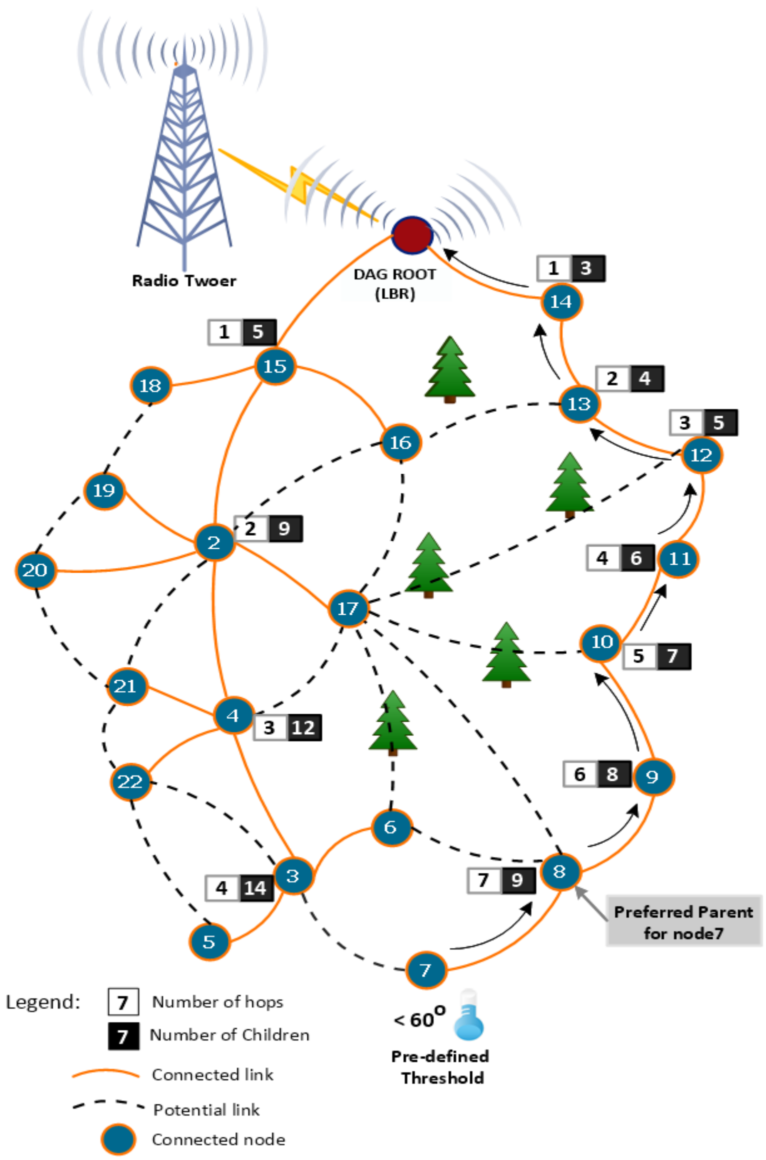


Figure 5.3 Regular (Life-time) scenario

However, when a fire is breaking out as shown in Figure 6.4, the temperature to the closest node will dramatically increase. As soon as the captured temperature reaches the predefined threshold, which has been fixed to 60 degrees, for instance, our algorithm re-prioritises the application requirements according to the new urgent context. Consequently, in our example, as shown in Fig 5, node 7 selects node 3 as a preferred parent despite the fact that node 3 has more accumulated children, however, this number is not a priority any longer in this case.

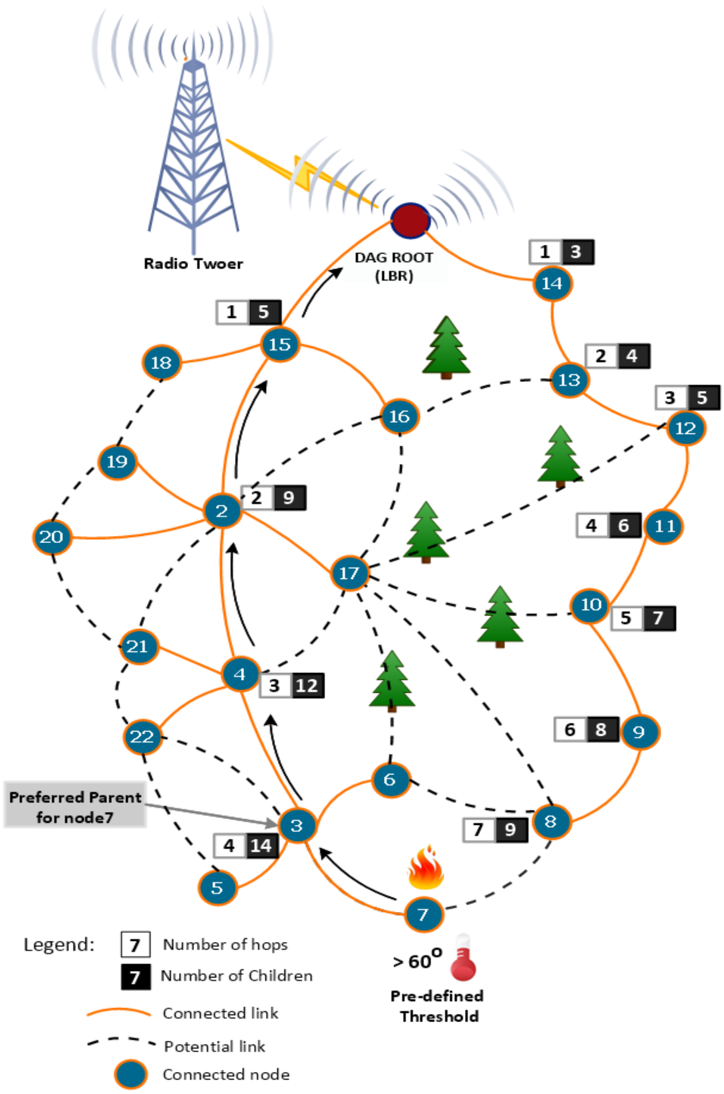


Figure 5.4 Emergency (Minimum delay) scenario

Henceforth, the selection decision becomes based on the less number of hops, which perceived from the antagonistic buffer as shown in Figure 6.4, to notify the DODAG root of the minimum end-to-end delay. In this regard, our aim is to recognise the number of children for each parent while the DODAG is under construction not after that (i.e. in the upwards routes not in the downwards routes). Alternatively, we could rely on the DAO messages to do so, however, as mentioned in the RFC [53, p. 6], *"If the MOP is 0, indicating no Downward routing, nodes MUST NOT transmit DAO messages and MAY ignore DAO messages.*". Thus, the downward routes are an optional feature in RPL, so it is more reliable to rely on a compulsory feature rather than optional to engage all types of applications.

As mentioned, being a preferred parent for more children means more overhead and unbalanced load, which consequently drains its own energy much faster than other candidate

Algorithm 2: ANTAGONISTIC METRIC ALGORITHM

input : Received DIO [InstantID, VersionNumber, Rank, ParentID]
output: Updated DIO

1 **for** $\forall node \in DAG$ **do**
2     $HC(Buffer) \leftarrow \sum_{i=0}^{n} H(i)$;   // H(i): next hop of node(i)
    $AcNC(Buffer) \leftarrow Algorithm1$;
3   **if** $Temp >= Threshold$ **then**
4         $minHC \leftarrow HC(Buffer)$;
5   **Select** the preferred parent with $minHC$;
6   **Discard** the current parent;
7   **end**
8   **else**
9         $minAcNC \leftarrow AcNC(buffer)$;
10   **Select** the preferred parent with $minAcNC$;
11   **Discard** the current parent;
12   **end**
13   **Broadcast** the updated DIO;
14 **end**

parents did. To solve this problem, a new metric has been proposed. The children set created in step (2) provides each preferred parent how many children it has. Based on that, we consider the number of children in the rank calculation formula (1) rather than using only ETX. A summary of the antagonistic metric operations are described in Algorithm 2.

It is worth to point that both metrics of the proposed OF (i.e the accumulated number of children and the number of hops) are implemented within the metric container to pass these additive values for every node within the DODAG as shown in Figure 6.5.

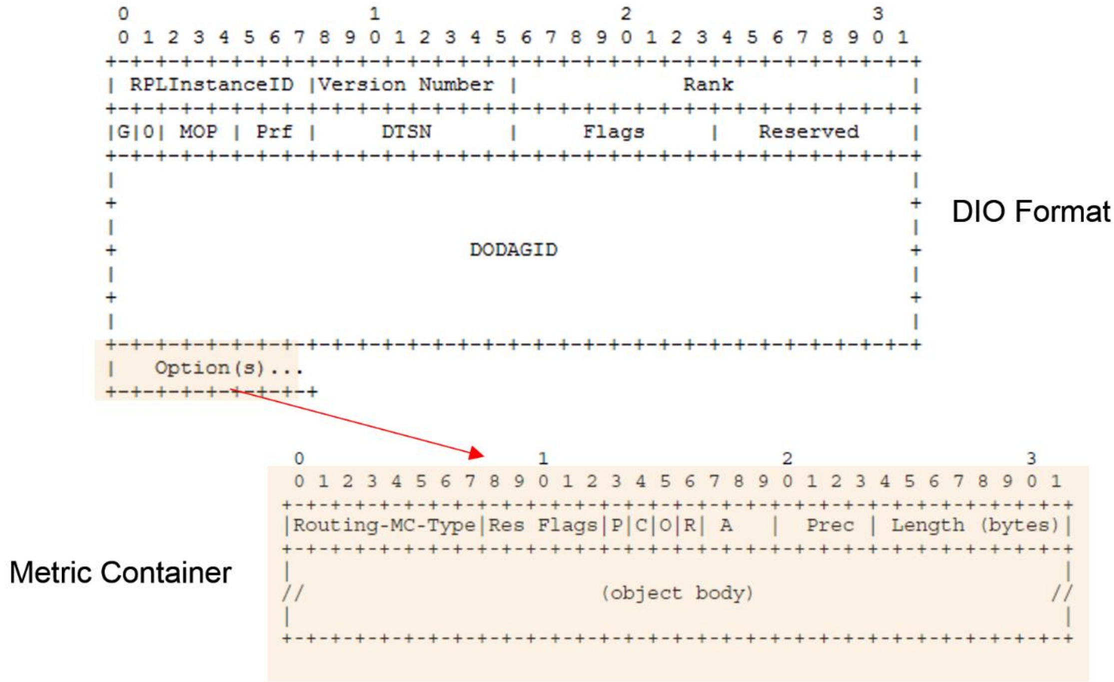


Figure 5.5 DODAG Metric Container Format

5.4 IPv6 Traffic Class (Tagging)

Once the triggered node (i.e., node 7 in Figure 6.6) selects the path with a minimum number of hopes, to mitigate the end-to-end delay, possibly this route could be congested with more load traffic, which may cause more delay towards the DODAG root. This unwanted delay cannot be neglected especially in the emergency cases. To overcome this problem, the IPv6 traffic class (Tagging)[161] [162] feature has been considered in our solution. Enabling IPv6 traffic class via tagging any packet comes from the triggered node gives this packet the priority to pass over the non-tagged packets, without taking the queue (i.e., by injecting the tagged packet in front of the queue). To do so, the queueing in the MAC layer has been considered to react to the context scenario, in collaboration with IPv6 protocol in a cross-layer manner.

The 8-bit Traffic Class field in the IPv6 header, as Figure 6.6 presents, is designed to differentiate between services and overt congestion.

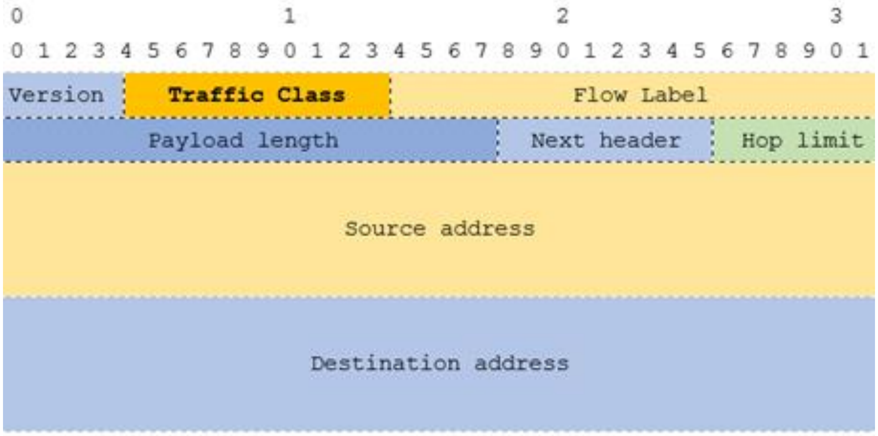


Figure 5.6 IPv6 Header format

In this sense, it is worth stating that RPL RFC points to the tagging feature to use it in the RPL Target Descriptor Option for the DAO message according to the format shown in Figure 6.7.

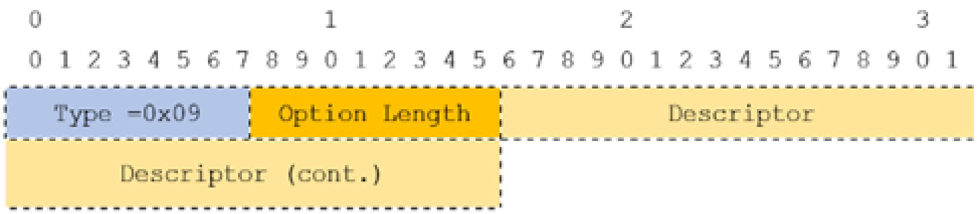


Figure 5.7 RPL Target Descriptor Option Format

Apparently, the path in Figure 6.4 is not a congested path, so the delay cannot be recognised due to the limited traffic. Thus, to investigate the tagging technique in our solution, the illustrated topology in Figure 6.8 is considered. Therefore, the number of nodes is increased to reach 52 nodes to raise the amount of traffic and challenge the congested path over 12 hopes from the alarmed node 7 towards the DODAG root. Thus, in our OF the information of the context goes beyond the network layer towards the queueing in the mac layer where the implementation took a place.

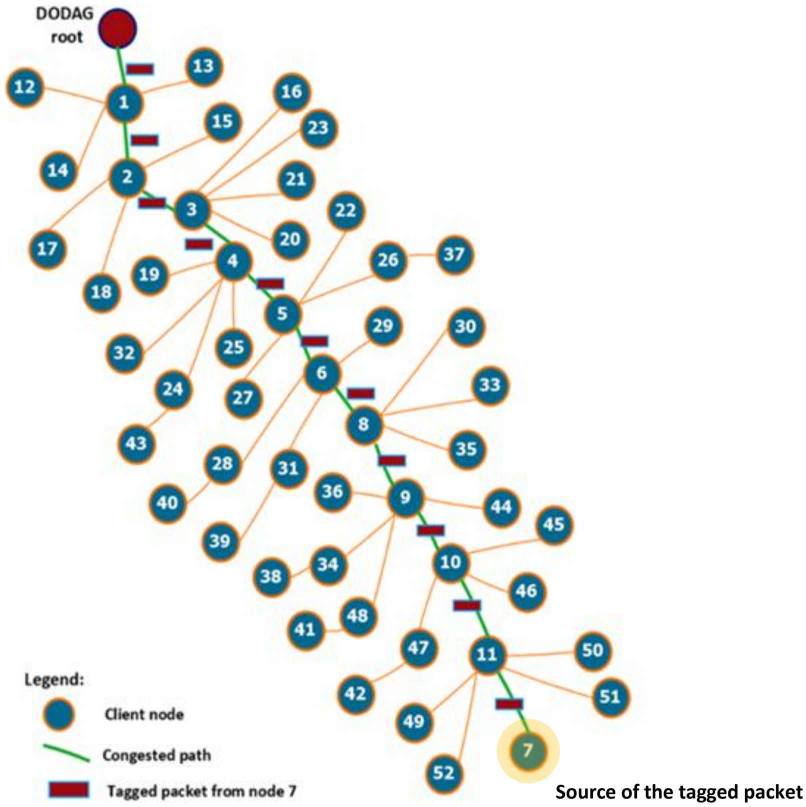
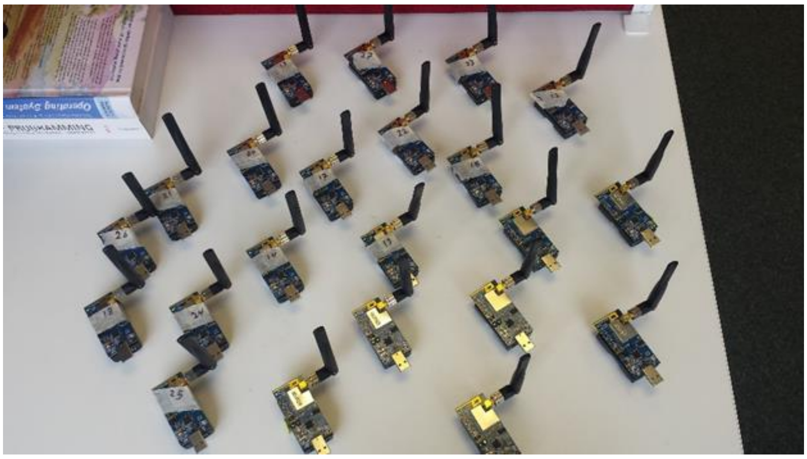


Figure 5.8 Tagging Packet in a Congested Path

5.5 Experiments and Evaluation

To validate the performance of the proposed OF in multihop LLNs, we configured a realistic indoor testbed environment, as shown in Fig.7. This testbed contains 22 LLN nodes likewise in [163] [164] [8]. The type of the used nodes is Tmote Sky/TelosB which is designed by Berkeley [165]. The IEEE 802.15.4 motes can capture temperature, relative humidity and light via embedded sensors and compatible with Contiki operating system [132]. All the nodes used in the distribution are with an MSP430 processor, a CC2420 radio working in the 2.4GHz ~ 2.485GHz band and an external +5 dBi SMA antenna. The transmission power within the range -25dBm ~ 0dBm will be fulfilled by 2xAA batteries to power each node as depicted in Figure 6.9.



(a) Client nodes

(b) The DODAG root node

(c) Nodes deployment

Figure 5.9 The testbed nodes

The purpose of this environment is to investigate the impact of the dynamicity on the RPL routing performance, to optimise the power saving and achieve the minimum delay in the emergency cases, in attempt to mitigate the impact of the antagonistic metrics. We consider

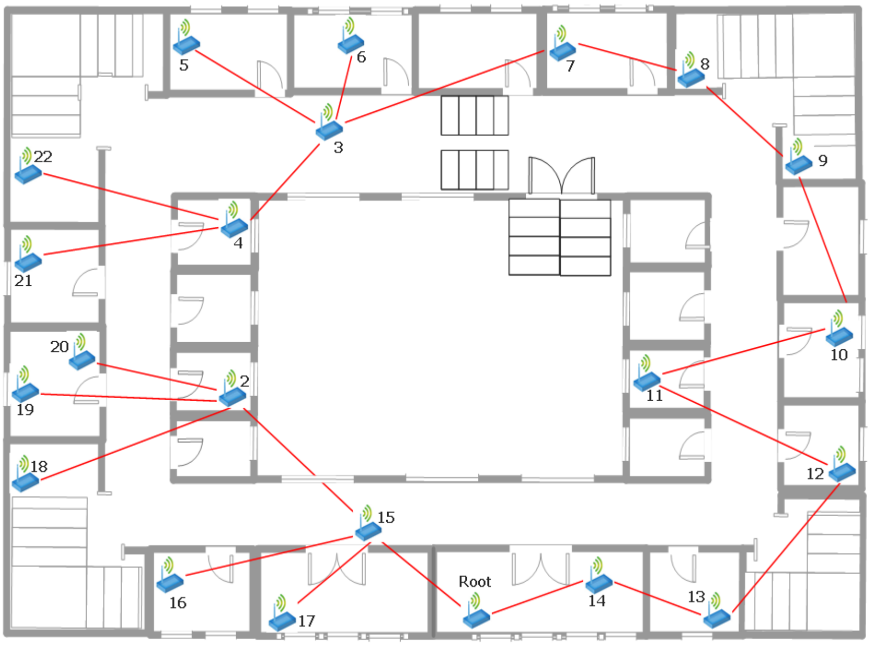


Figure 5.10 Network deployed in Merchiston campus premises, Edinburgh Napier University

different densities scenarios that are best for evaluating the conventional OFs in RPL (i.e., OF0 and MRHOF), these OFs are commonly used for constructing the RPL DODAG in the natural environment for different IoT applications.

In terms of the distribution, the random fixed topology has been considered due to it is reconcile with our case study that we investigate. In fact, in the wood fire detection application ,and in similar applications, the grid distribution is not an option due to the massive geographical area that needs to cover [166] (for instance 347 million hectares in Canada [167]), by the sensors which makes the grid is inapplicable choice.

The findings are averaged over seven iterations until the results showed a stability. These findings are carried out from the distributed testbed nodes in our campus premises of Edinburgh Napier University as illustrated in Figure 6.10. In these experiments, we considered an infinite buffer size for the DODAG root, and configured the temperature threshold to 32 °C. Moreover, in these findings, we considered an infinite buffer size and configured the temperature threshold to 32 °C.

5.5.1 Testbed-Lossy Network

Figure 6.11 compares the end-to-end delay for the conventional objective functions (i.e. MRHOF and OF0) along with the proposed algorithm. The delay measurements carried out only for node7 (shown in Fig 6.10) when the captured temperature threshold has been reached. Indeed, in this case, the alarmed node (i.e., node7) switches its preferred parent to take the minimum delay path towards the DODAG root. As can be observed, the delay in the proposed algorithm starts to decline, after reaching the threshold, and then switching the parent from node 8 to become node 3. Consequently, the path with a new parent has less number of hops and less delay with 7% than the conventional objective functions. Likewise, in Figure 6.12, the delay drops down in the hour 7 (where the temperature became 32°C) in the proposed algorithm and then stabilises from hour 8 onward to the end of experiment duration length (i.e. for 14 hours). The gain in delay can be clearly explained from Figure 6.13, which shows the average delay of using each single or composite metric in our scenario.

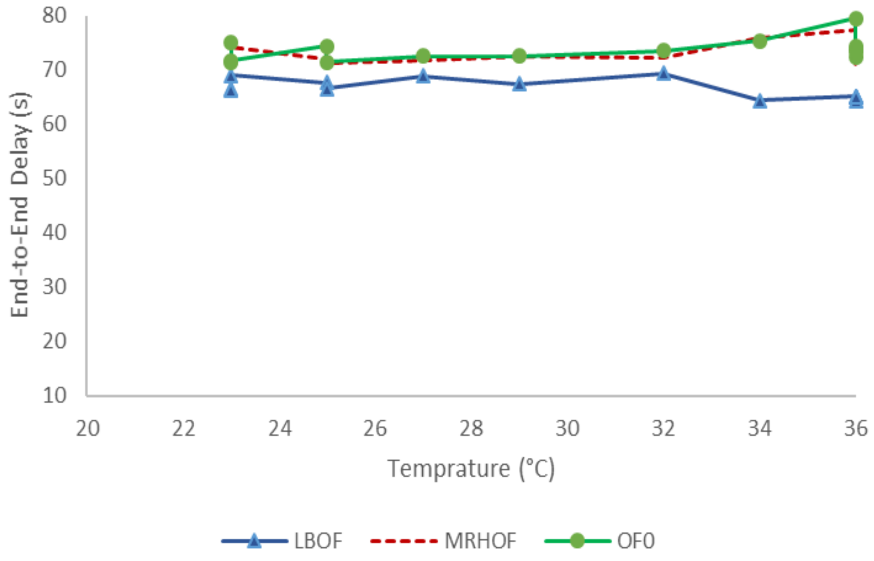


Figure 5.11 The temperature's impact on delay

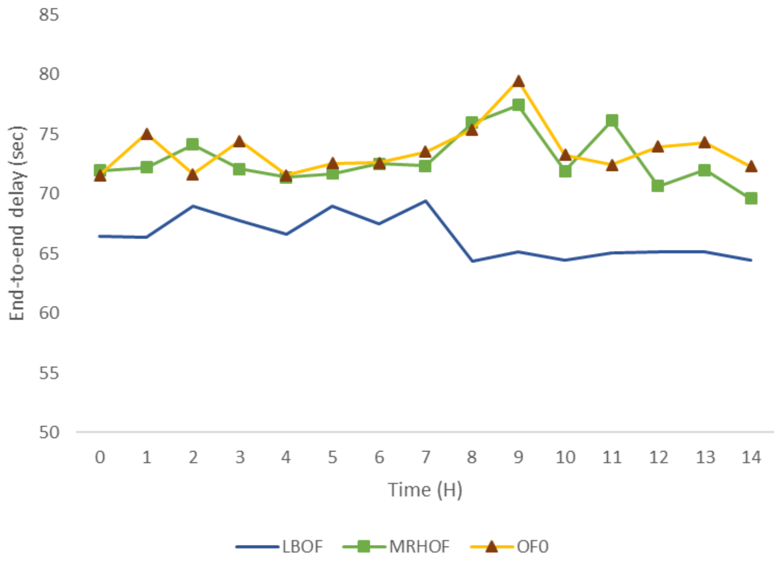


Figure 5.12 Delay over the time

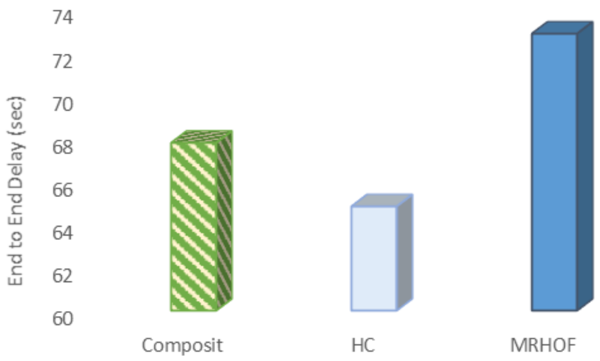


Figure 5.13 Average delay for each metric

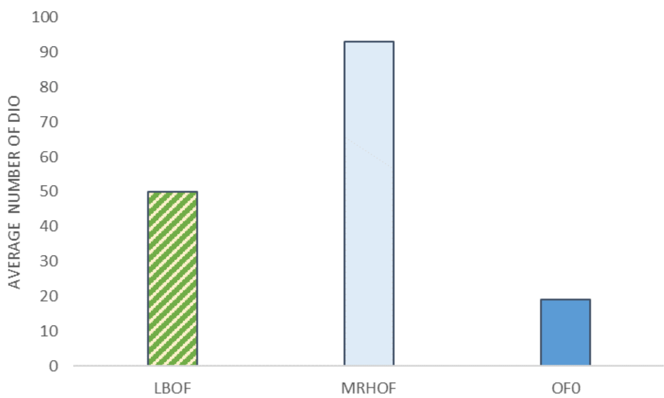


Figure 5.14. Number of DIOs

In Figure 6.14, the MRHOF incurs the largest number of DIO overhead, unlike the OF0, has least DIO overhead due to the minimum churn. However, as the proposed OF uses a composite metric before reaching the configured threshold which generates more churn before stabilizing the number of children for each node.

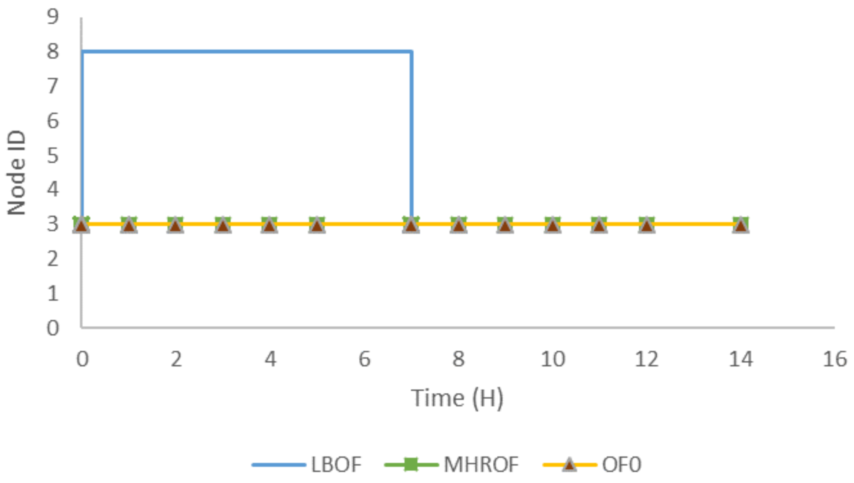


Figure 5.15 The churn in node7

In terms of the churn (i.e., the frequency of changing the preferred parent), the largest number of changes can be observed in the proposed algorithm as depicted in Figure 6.15. This oscillation due to the unstable number of children at the early of the first stage of building the DODAG, hence, the node 7 keeps changing its preferred parent between node3 and node 8. Afterward, node 8 will be selected as a preferred parent until the temperature triggers the node 7 to switch its preferred parent during the seventh hour.

Figure 6.16 compares the power consumption for node8 among the aforementioned three different objective functions. It can be noticed that the power consumption in the conventional objective functions is steady due to the fact that the churn is in the minimum level, and node 7 adhered with the selected parent along with all the experiment duration length (36 hours).

However, in LBOF the power consumption begins with a high level, in resulting of the churn within the first hour. Then, the power consumption clearly becomes less when node 7 reaching the threshold (i.e. unlike the cases in OF0 and MRHOF, the traffic of node 7 in LBOF has been handed over to a new preferred parent (i.e. node 3), and left node 8 (the previous preferred parent) with less passing traffic through and definitely less power consumption. It is worth to point that with the continual sloping of the LBOF behaviour the superiority of OF0 will be diminished and that is due to using the hop-count as a metric in both OF0 and LBOF after reaching the threshold.

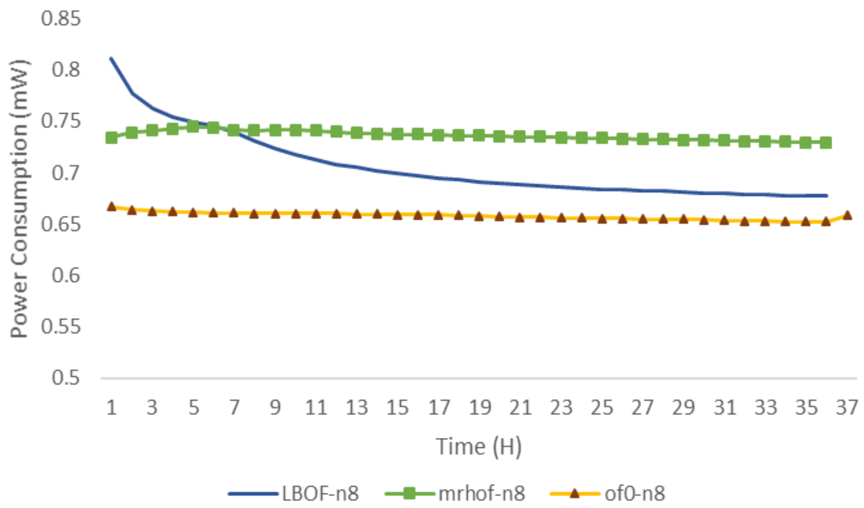


Figure 5.16 Power consumption for node 8

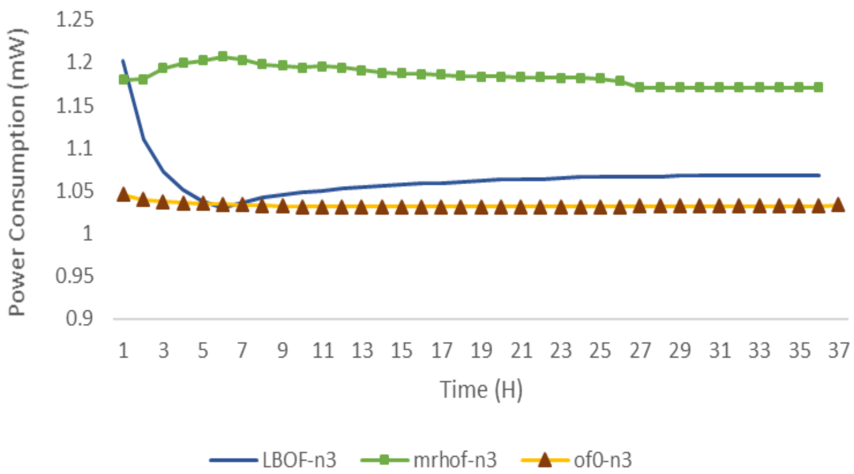


Figure 5.17 Power consumption for node 3

In the same context, node 3 in Figure 6.17 performs in a contrary way compares to node 8 as node 7 has been handed over to it and became the preferred parent, therefore, the power consumption will slightly increase to pass the traffic of the new child. However, before switching to another preferred parent (i.e., reaching the threshold), the power consumption was declining, as Figure 6.18 clearly shows, which compares the three nodes in every investigated OF.

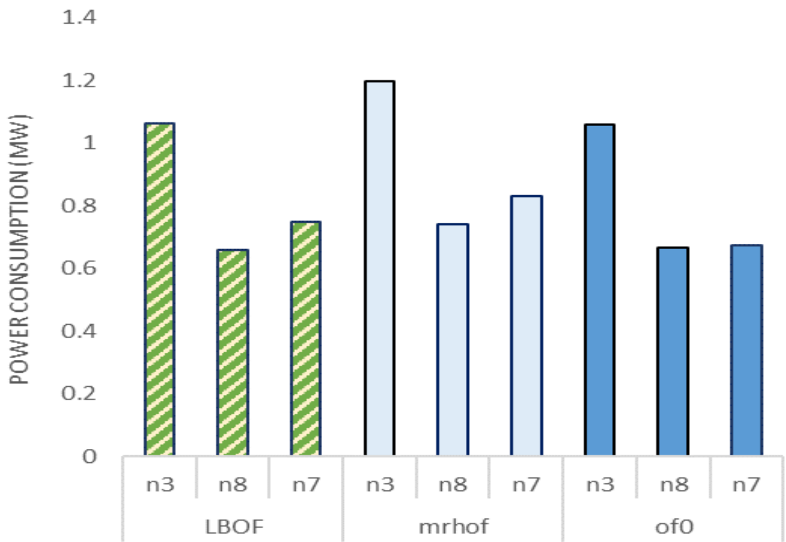


Figure 5.18 Power Consumption for the three nodes in each OF

5.5.2 Fixed Lossy network by the simulator

As the lossy ratio within the testbed nodes is out of our control and we cannot fix it on a particular ratio, we used the Cooja simulator to configure a fixed lossy ratio to 50% in all three compared OFs to investigate the impact of increasing the lossy ratio on the proposed OF. The study of this scenario would provide a unique lens to look at the performance of the proposed protocol in such a very lossy environment.The well-known Cooja simulator is based on Instant Contiki 3.0 operating system [132], which is considered as a lightweight, highly portable open source operating system and dedicated for WSNs and IoT. The simulation parameters are given in Table 1.

Table 5.1 Simulation Parameters

Parameter	Value
Operating System	Contiki 3.0
Node Type	TMote sky
Routing protocol	RPL
MAC/adaptation	ContikiMAC, NullRDC/ 6LowPAN
Radio Environment	Unit Disk Graph Medium (UDGM)
Number of nodes	22-53
Simulation time	15-36 hours
Transmission range	50 (m)
Data packet timer	5,10,15,20,30,60 sec
Mote start-up delay	1000 msec
Simulation area	600×500 m
DIO interval min	12 msec
DIO interval doublings	8 msec
Deployment mode	Random

Figure 6.19 compares the three aforementioned OFs in terms of delay. Verily, the proposed protocol performs much better than the conventional objective functions (i.e. MRHOF and OF0). The delay in all OFs is much higher than in previous scenario i.e. in Fig.11, indeed, that is expected as the lossy ratio has been increased.

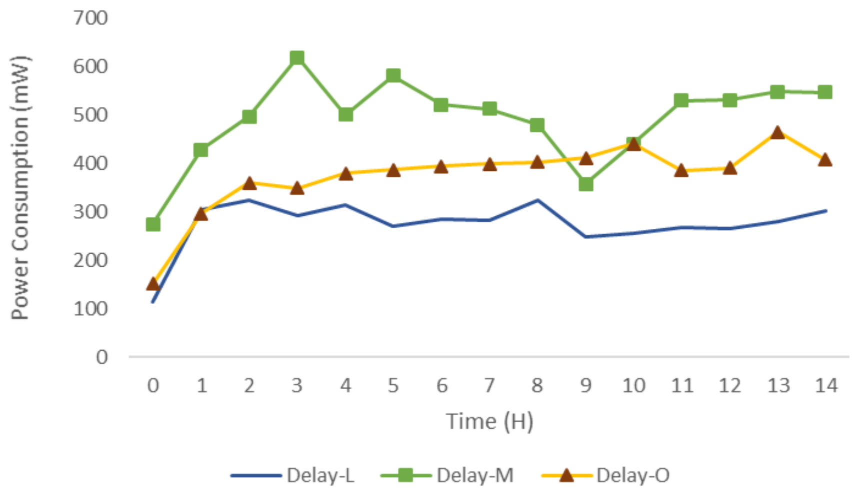


Figure 5.19 Delay with fixed lossy ratio

Likewise, the number of DIO increased as shown in Figure 6.20, the most incurred OF is the MRHOF and with a small difference between OF0 and the proposed OF. The slight difference in these results is partially due to the changing parent in the proposed algorithm.

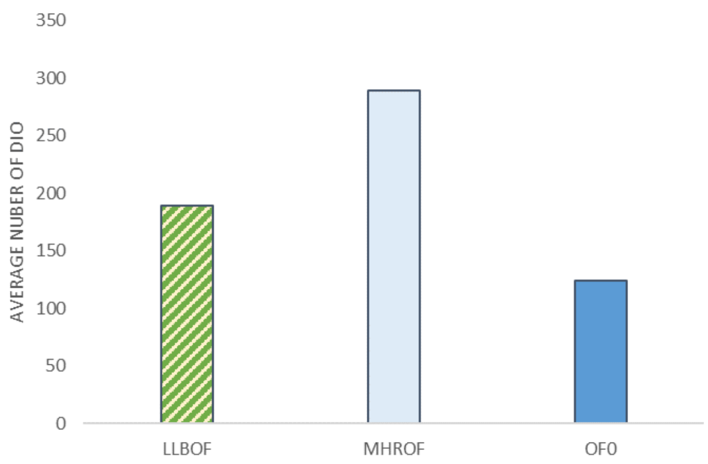


Figure 5.20 Average DIO

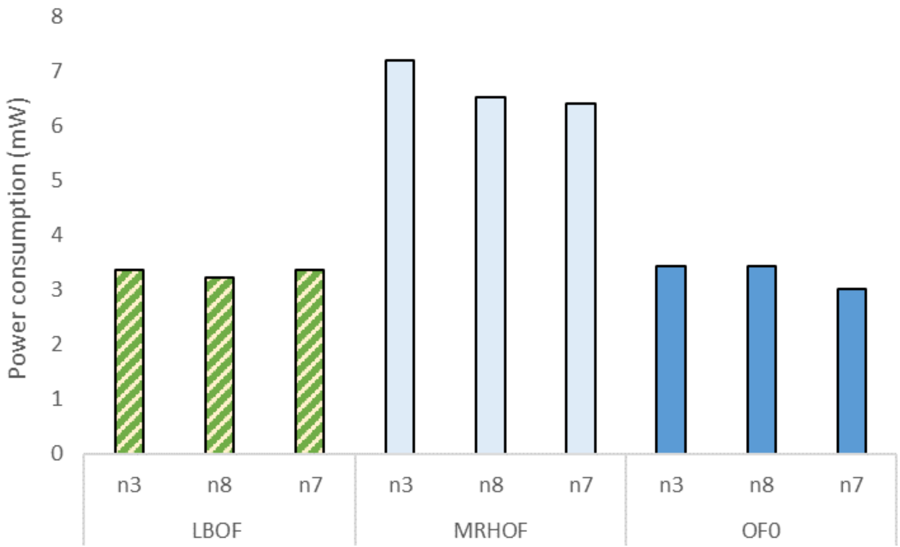


Figure 5.21 Power Consumption for node 3, 7, and 8 in each OF

In terms of power consumption, as can be noticed in Figure 6.21, it has been increased, as expected, in all OFs compares to the measurements in Figure 6.18 due to the extra lossy ratio. The proposed OF in both node 3 and node 8 clearly overcomes the other OFs as can be observed in Figure 6.22 and Figure 6.23 respectively. However, node 7 the OF0 copes the rest OFs due to the aforementioned churn.

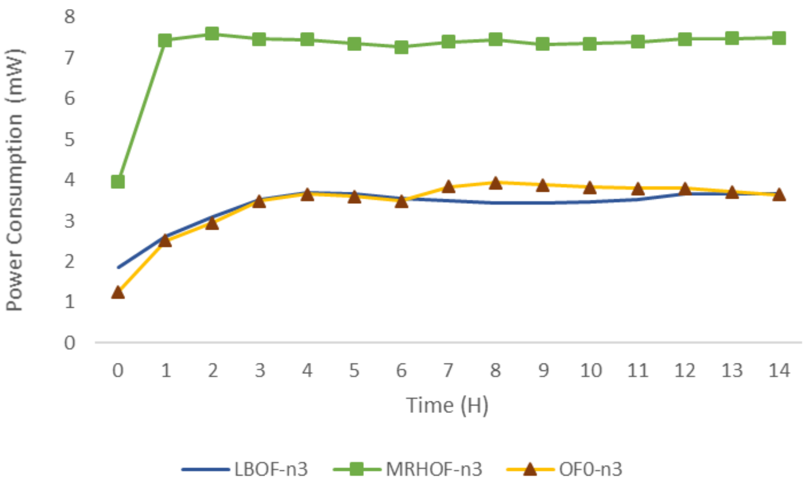


Figure 5.22 Power Consumption for node 3

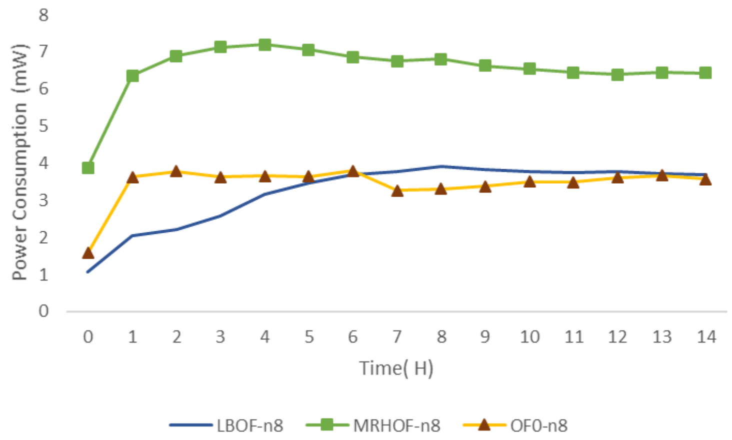


Figure 5.23 Power Consumption for node 8

5.5.3 Tagging scenario

As mentioned in Section 5.3, to comprise any extra delay caused by more traffic on the path towards the DODAG root, especially if the case is emergency. The illustrated topology in Fig. 6.8 has been used to investigate the efficiency of tagging technique to overcome the delay problem over the congested path. The findings in Figure 6.24 show the end-to-end delay of the tagged packet has been mitigated compared to the delay of the non-tagged packets. Four different timers for the data packets are considered (5 -20 sec) and all show that the delay with the tagged packet is less than the non-tagged one.

In terms of the packet delivery ratio (PDR), the superiority of the tagged packet is obvious as Figure 6.25 illustrates. The reason behind the improvement of PDR of the tagged packet that the number of the lost packet is decreased as shown in Figure 6.26. It is notable when the sending rate is 10 the lost packets are more and that due to the limited memory capacity of the node, which cannot accommodate the huge amount of data traffic, especially for the node with less number of hops from the DODAG root.

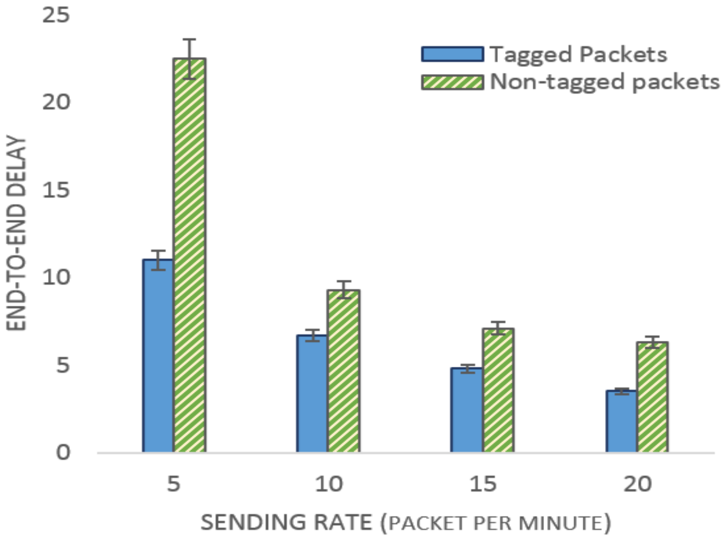


Figure 5.25 End-to-end delay for tagged and non-tagged packets

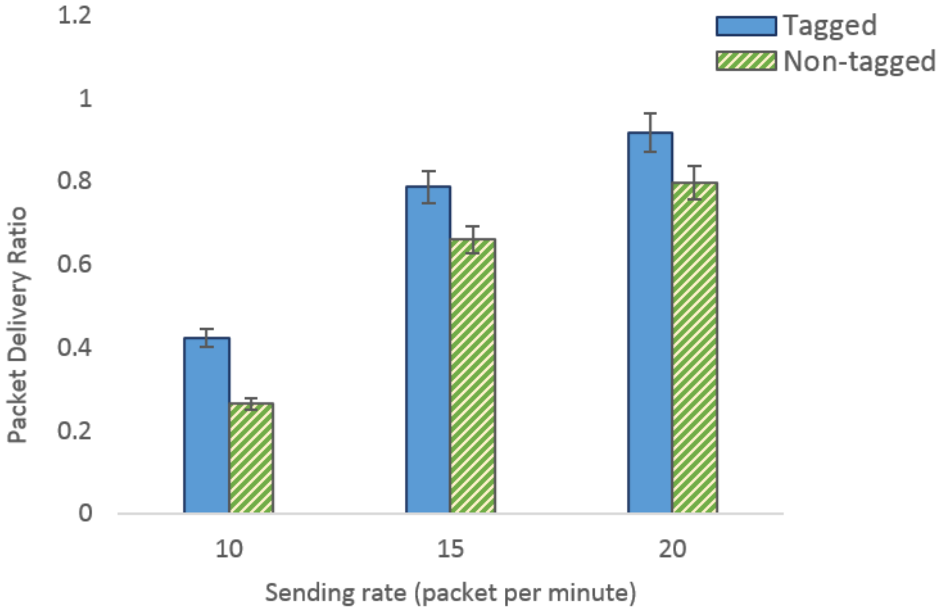


Figure 5.24 PDR for tagged and non-tagged packets

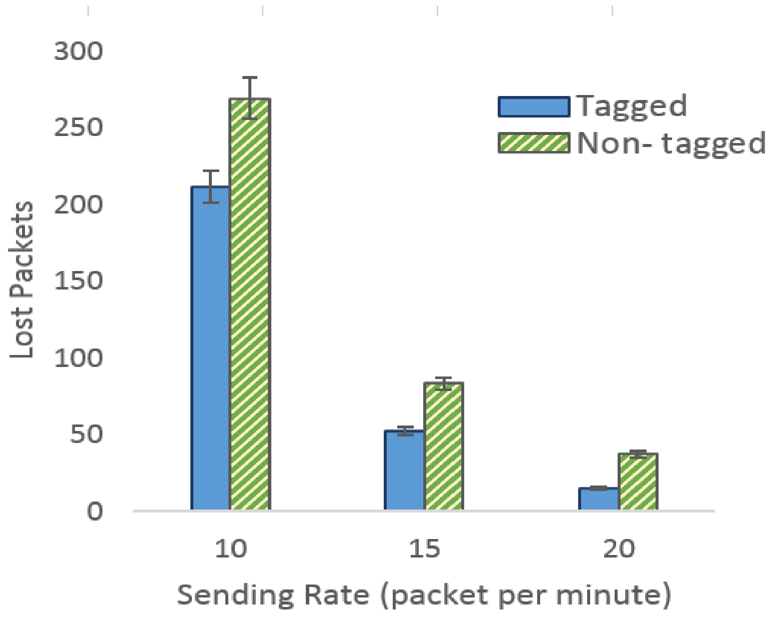


Figure 5.26 Lost packet for tagged and non-tagged

5.6 Conclusion

This Chapter investigates the impact of the antagonistic metrics dynamicity on the RPL routing performance on the emergency scenarios. We propose a context-adaptive OF to optimise the selection decision of the routing path, in a dynamic manner, to achieve better low-latency and maximize the lifetime of the network. The proposed OF endorses the robustness and elasticity of the current OFs and re-prioritises the metrics adaptively. The results, which have been conducted using real testbed devices, show that the proposed OF improves the network reliability, during the emergency situation, by reducing the end to end delay up to 15%. Moreover, IPv6 traffic class (Tagging) approach is considered to mitigate any possible delay, which attains about 40% over the non-tagged one, and up to 14% in terms of PDR. For the other scenario, i.e. in terms of the lifetime, our OF performs better than the other OFs with up to 30% during the non-emergency cases.

Thus, a trade-off will have to be made between maintaing the minimum delay in the emergency scenarios and the minimum power consumption in the regular scenarios within the same application. Next chapter presents the overall conclusion of this thesis and the future directions.

7

Conclusion and Future Work

6.1 Introduction

This chapter concludes the contributions of this thesis. It briefly presents how the previous chapters address the raised problems in chapter 1, the key findings along with the potential avenues for future research.

Given the fact that the routing protocol is one of the main pillars of networking architecture, and confidently foreseeing its necessity for Low Power and Lossy Networks (LLNs), IPv6 Routing Protocol for LLNs (RPL) has swiftly become the de-facto routing protocol for IoT. Hence, RPL protocol tends to play a key role towards underpinning the infrastructure for a tremendous number of potential IoT applications in the near future, in both regular and harsh environments including the real-time applications in inaccessible areas with uniform and non-uniform distribution in large-scale networks. RPL is formulated with several robust landscapes, which tend to lid the limitations of constrained devices, such as exiguous delay, quick configuration, loop-free topology, and self-healing. However, RPL still suffers from some problems that can cause serious consequences on the network reliability, such as the unbalanced load traffic and the context awareness of the antagonistic requirements.

6.2 Thesis Summary

The main goal of this thesis is to enhance the performance of the current standard routing protocols of IoT, namely, RPL protocol. Taking into account the acute contest growth of protocols and applications within the IoT paradigm. However, among these protocols, the routing protocol RPL was in particular the focus of this thesis, as it is the main de facto protocol in low power and lossy networks and machine-to machine networks, which are the backbone of the IoT paradigm. Subsequently, the IoT connotation, applications, and the IoT stack evolutionary have been discussed. We came across all protocols of IoT stack to show how the accumulated efforts contributed gradually to the LLNs protocols to end up with this stack. As RPL protocol is the main theme of this thesis, more details and discussion have been given to usher the essential background for better understanding of the research problems within this thesis.

Accordingly, a variety of studies and solutions for RPL load balancing with the attributed metrics and context-awareness are surveyed, revealing the significant consequences of these problems on RPL, and the potential avenues for more solutions in this regards. As part of mastering the state of the art, we proposed a new category for the current solutions of RPL load balancing and context-awareness, along with a summary comparison of the current works to shed the light on some unsolved problematic challenges.

The current IoT tools and methods are discussed, highlighting the characteristics and the main features of: firstly, the details of the common IoT operating systems are summarised, secondly, the real testbed devices alongside the widely used simulator so-called Cooja features that allowed being the chosen simulator to conduct the experiments of this study.

This thesis has contributed a trivial step towards the load balancing problem in RPL, which is in line with the parent selection policy in the OFs within this protocol. This policy was investigated extensively through different scenarios. Thus, a new load balanced OF for RPL protocol is proposed to achieve a better workload distribution among all nodes in LLNs and battery lifetime thriving for a longer survival. A novel technique has been used to count the number of children for each parent without generating any extra overhead, and then injected in the DIO message. Thus, the better fairness of child nodes distribution has been attained through the proposed OF DODAG construction process, and balanced lifetime. Hence, both first and second research questions presented in Chapter 1.

The synergy between the context parameters and RPL routing protocol is explored, which further alleviates risks caused by the antagonistic requirements impact. Thus, an adaptive OF is proposed based on the context that mitigates the impact of the antagonistic needs throughout the emergency scenarios. On the other hand, to optimise the lifetime, during the non-emergency scenarios, the accumulated number of children towards the DODAG root is considered to introduce less power consumption. The implemented OF has been tested using real testbed measurements of a multi-hop LLNs nodes in conjunction with simulator. Accordingly, this contribution satisfied the third and fourth research question that introduced in Chapter 1.

6.3 Summary of Contributions

According to the overall depiction of the research conducted in this thesis, the key outcomes of the contributions are as follow:

The key outcomes of the first contribution:

(i) A new classification of the existing load-balancing RPL OF approaches is introduced. This classification places a great deal of emphasise on clarifying the unsolved issues in this regards, and revealing the direction of the research conducted in this thesis.

(ii) A new utilisation technique for the DIO message has been considered. Typically, in the standard OFs, the preferred parent ignores the DIOs that come from its child. In the proposed LB-OF algorithm, the preferred parent node counts the received DIO from the child node after injecting the DIO with the selected preferred parent ID in the options field. This technique avoids increasing any extra overhead and allows each preferred parent to distinguish the number of its children during the DODAG construction stage. Thus, a better fairness distribution in the number of direct children among the candidate parents is achieved.

(iii) Simulation results show that the proposed load balanced objective function is better than the current standards OFs in terms of balanced power consumption, packet delivery rate, balanced lifetime, and a number of children for each node.

The key outcomes of the second contribution:

(i) A new classification of the existing RPL OF approaches in terms of the used metric(s) is presented. This classification introduces the techniques that have been used to comprise more than one metric.

(ii) The distribution in the number of children has been widen using new technique to accommodate the entire path towards the root rather than the number of direct children only. This technique optimises the routing decision overall path.

(iii) As the requirements in IoT are generally application specific, then the impact of the antagonistic requirements may degenerate the network reliability in emergency situations. Thus, the protocol is improved to cater for emergency scenarios, and a context-adaptive OF has been proposed to ensure the co-existence of the emergency and non-emergency scenarios within one application. In fact, this new OF facilitates exchanging critical information between application and network layers to mitigate the impact of the antagonistic requirements based on context parameters (such as temperature) to rationalize the selection decision of the routing path towards the root node.

(iv) The performance of the proposed OF has been examined using real testbed measurements of a multi-hop LLNs nodes in conjunction with simulation experiments. The obtained results confirm the superiority of our solution over the existing ones with respect to network lifetime, end-to-end delay with up to 15%, and 14% in terms of packet delivery ratio.

6.4 Border impact

The impact and applicability of the mechanisms and techniques developed in this work are expected to go beyond academia theory and expected to be adopted within the standards body IETF. Thus, the current standard protocol reliability will be optimised and enhanced to accommodate the plethora of the coming IoT applications. Furthermore, emphasising RPL for real-time and emergency IoT applications is crucial and might open the door to its fullest extent for unpresented applications in many disciplines. Moreover, this research not only make new contributions to the research community but also open up promising doors towards future research.

6.5 Future Directions

Despite the enhancements that have been conducted in the thesis, however, we believe there are further enhancements are envisaged as future directions. Thus, the contributions of this thesis can be enriched in a number of avenues:

Binary modes: Utilising the binary modes in RPL (i.e., storing and non-storing modes) to count the number of children to minimise the overload that can be caused by injecting the preferred parent IP address in the DIO message as detailed in the standard draft [160]. In the storing mode, DAO message can be utilised for child nodes registration while no-Path DAO can be used for de-registration, and thus the number of child nodes will be counted. Hence, to minimalise traffic load, there no need to add the Parent IP Address in the DIO message in the storing mode. While, in the non-storing mode, the parent address should be considered within the DIO along with the hysteresis threshold for the number of children to switch from parent to another, the selected threshold depends on the application requirements. Overall, this will be determined according to the application requirements. Additionally, using the number of children along with another metric(s) (e.g. ETX, number of hops, energy, etc., according to the application requirements).to reduce the oscillation of changing the preferred parent.

Mobility: Furthermore, the mobility feature within RPL is another option to investigate, considering the proliferation of IoT applications that need to be accommodated by RPL. Being the considered topologies in this thesis were fixed, we believe that the mobility will generate more churn and changing the parent address as result of topology dynamicity. Thus, the instability situation will result in more power draining causing less reliable network.

Heterogeneity: The umbrella of IoT tends to lid different technologies together containing heterogeneous nodes, in terms of resources and technologies such as ZigBee, LoRa, Sigfox, 4G nodes, and so forth. These heterogeneous nodes differ from routing in a network with homogeneous nodes. Hence, with this envision the need to use the hybrid networks that can accommodate more than one technology to operate according to the needs and resources availability. Thus, these type of networks can underpin the infrastructure for a wide range of IoT applications.

References

[1] D. Uckelmann, M. Harrison, and F. Michahelles, “An Architectural Approach Towards the Future Internet of Things,” in *Architecting the Internet of Things*, D. Uckelmann, M. Harrison, and F. Michahelles, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 1–24.

[2] J.-P. Vasseur and A. Dunkels, *Interconnecting smart objects with IP: the next Internet*. Burlington, MA: Morgan Kaufmann Publishers/Elsevier, 2010.

[3] M. R. Palattella *et al.*, “Standardized Protocol Stack for the Internet of (Important) Things,” *IEEE Commun. Surv. Tutor.*, vol. 15, no. 3, pp. 1389–1406, 2013.

[4] *Vision and challenges for realising the Internet of Things*. Luxembourg: Publications Office of the European Union, 2010.

[5] J. Pan, S. Paul, and R. Jain, “A survey of the research on future internet architectures,” *IEEE Commun. Mag.*, vol. 49, no. 7, 2011.

[6] A. J. Jara, L. Ladid, and A. F. Gómez-Skarmeta, “The Internet of Everything through IPv6: An Analysis of Challenges, Solutions and Opportunities.,” *JoWua*, vol. 4, no. 3, pp. 97–118, 2013.

[7] H.-S. Kim, J. Ko, D. E. Culler, and J. Paek, “Challenging the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL): A Survey,” *IEEE Commun. Surv. Tutor.*, pp. 1–1, 2017.

[8] H.-S. Kim, H. Kim, J. Paek, and S. Bahk, “Load Balancing under Heavy Traffic in RPL Routing Protocol for Low Power and Lossy Networks,” *IEEE Trans. Mob. Comput.*, pp. 1–1, 2016.

[9] O. Iova, F. Theoleyre, and T. Noel, “Using multiparent routing in RPL to increase the stability and the lifetime of the network,” *Ad Hoc Netw.*, vol. 29, pp. 45–62, Jun. 2015.

[10] T. B. Oliveira, P. H. Gomes, D. G. Gomes, and B. Krishnamachari, “ALABAMO: A LoAd BAlancing MOdel for RPL,” in *Proc. Brazil. Symp. Comput. Netw. Distrib. Syst.(SBRC)*, 2016.

[11] E. C. Whitman, “Sosus: The ‘secret weapon’ of undersea surveillance,” *Undersea Warf.*, vol. 7, no. 2, p. 256, 2005.

[12] V. S. Cronin and K. A. Sverdrup, “Defining static correction for jointly relocated earthquakes along the Blanco Transform Fault Zone based on SOSUS hydrophone data,” in *OCEANS 2003. Proceedings*, 2003, vol. 5, pp. P2721–P2726.

[13] Chee-Yee Chong and S. P. Kumar, “Sensor networks: Evolution, opportunities, and challenges,” *Proc. IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[14] B. Djamaa and A. Yachir, “A Proactive Trickle-based Mechanism for Discovering CoRE Resource Directories,” *Procedia Comput. Sci.*, vol. 83, pp. 115–122, 2016.

[15] JP. Vasseur, "Terms Used in Routing for Low-Power and Lossy Networks," IETF, RFC7102, Jan. 2014.

[16] C. Bormann, Mehmet Ersue, and A. Keranen, "Terminology for constrained-node networks," IETF, RFC 7228, 2014.

[17] Z. Shelby and C. Bormann, *6LoWPAN: the wireless embedded internet*. Chichester, U.K: J. Wiley, 2009.

[18] H. Fotouhi, "Reliable Mobility Support in Low-Power Wireless Networks," University of Porto, 2015.

[19] L.-O. Varga, "Multi-hop energy harvesting wireless sensor networks: routing and low duty-cycle link layer," Université Grenoble Alpes, 2015.

[20] G.-Z. Yang, Ed., *Body Sensor Networks*, 2nd ed. Imperial College London: Springer London, 2014.

[21] Badis Djamaa, "Pervasive Service Discovery in Low-power and Lossy Networks," PhD Thesis, Cranfield University, UK, 2015.

[22] "Auto-ID Labs," *Auto-ID Labs*, 01-May-2015. [Online]. Available: http://www.autoidlabs.org.

[23] K. Ashton, "That 'internet of things' thing," *RFiD J.*, vol. 22, no. 7, 2011.

[24] B. C. Villaverde, R. De Paz Alberola, A. J. Jara, S. Fedor, S. K. Das, and D. Pesch, "Service Discovery Protocols for Constrained Machine-to-Machine Communications," *IEEE Commun. Surv. Tutor.*, vol. 16, no. 1, pp. 41–60, 2014.

[25] L. Yang, S.-H. Yang, and L. Plotnick, "How the internet of things technology enhances emergency response operations," *Technol. Forecast. Soc. Change*, vol. 80, no. 9, pp. 1854–1867, 2013.

[26] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[27] D. D. Guinard and V. M. Trifa, *Building the Web of Things with Examples in Node.Js Aand Raspberry Pi*. Manning Publications, 2016.

[28] N. Cuartero Moya and S. Quintana Alcaraz, "Tmote Sky Datasheet for WSNs," Universitat Politècnica de Catalunya, Technical data-sheet, 2011.

[29] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols and Applications," *IEEE Commun. Surv. Tutor.*, pp. 1–1, 2015.

[30] N. Alhakbani, M. M. Hassan, and M. Ykhlef, "An Effective Semantic Event Matching System in the Internet of Things (IoT) Environment," *Sensors*, vol. 17, no. 12, p. 2014, Sep. 2017.

[31] Y. Krytska, I. Skarga-Bandurova, and A. Velykzhanin, “IoT-based situation awareness support system for real-time emergency management,” 2017, pp. 955–960.

[32] T. Winchcomb and S. Massey, “Review of latest developments in the Internet of Things,” Ofcom, P2952–R–001 v4.8, Mar. 2017.

[33] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Comput. Netw.*, vol. 38, no. 4, pp. 393–422, 2002.

[34] Z. Sheng, S. Yang, Y. Yu, A. Vasilakos, J. Mccann, and K. Leung, “A survey on the ietf protocol suite for the internet of things: Standards, challenges, and opportunities,” *IEEE Wirel. Commun.*, vol. 20, no. 6, pp. 91–98, 2013.

[35] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of IPv6 packets over IEEE 802.15. 4 networks,” 2007.

[36] IEEE Computer Society, LAN/MAN Standards Committee, Institute of Electrical and Electronics Engineers, and IEEE-SA Standards Board, *802.15.4e-2012: IEEE Standard for Local and Metropolitan Area Networks — Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC Sublayer*. New York: Institute of Electrical and Electronics Engineers, 2012.

[37] D. Dujovne, T. Watteyne, X. Vilajosana, and P. Thubert, “6TiSCH: deterministic IP-enabled industrial internet (of things),” *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 36–41, 2014.

[38] “WirelessHART Specification 75: TDMA Data-Link Layer,” HART Communication Foundation Std., Rev. 1.1, 2008, hCF_SPEC-75.”

[39] D. Dujovne, T. Watteyne, X. Vilajosana, and P. Thubert, “6TiSCH: deterministic IP-enabled industrial internet (of things),” *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 36–41, 2014.

[40] D. De Guglielmo, S. Brienza, and G. Anastasi, “IEEE 802.15.4e: A survey,” *Comput. Commun.*, vol. 88, pp. 1–24, Aug. 2016.

[41] 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 Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1573–1588, Feb. 2017.

[42] T. Watteyne, M. Palattella, and L. Grieco, “Using IEEE 802.15. 4e time-slotted channel hopping (TSCH) in the internet of things (IoT): Problem statement,” 2015.

[43] T. Watteyne, Q. Wang, and P. Thubert, “Terminology in IPv6 over the TSCH mode of IEEE 802.15.4e draft-ietf-6tisch-terminology-09,” *IETF*, Dec. 2017.

[44] S. Duquennoy, A. Elsts, B. A. Nahas, and G. Oikonomo, “TSCH and 6TiSCH for Contiki: Challenges, Design and Evaluation,” 2017, pp. 11–18.

[45] “ISA100 standard, Architecture for Industrial Internet of Things IPv6 Industrial Wireless Network Universal, Reliable, Scalable,” Automation Standards Compliance Institute, IEC 62734, 2014.

[46] X. Vilajosana, K. Pister, and T. Watteyne, “Minimal IPv6 over the TSCH Mode of IEEE 802.15. 4e (6TiSCH) configuration,” 2017.

[47] “6top Protocol (6P) Q. Wang, Ed. Internet-Draft Univ. of Sci. and Tech. Beijing Intended status: Standards Track X. Vilajosana Expires: April 28, 2018,” 2017.

[48] J. Ko, A. Terzis, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, and P. Levis, “Connecting low-power and lossy networks to the internet,” *IEEE Commun. Mag.*, vol. 49, no. 4, 2011.

[49] B. Pavkovic and M. Batic, “The Importance of Cross-layer Considerations in a Standardized WSN Protocol Stack Aiming for IoT,” *Mag. Ubiquity ACM*, 2015.

[50] P. Thubert and J. W. Hui, “Compression format for IPv6 datagrams over IEEE 802.15. 4-based networks,” 2011.

[51] O. Iova, P. Picco, T. Istomin, and C. Kiraly, “RPL: The Routing Standard for the Internet of Things... Or Is It?,” *IEEE Commun. Mag.*, vol. 54, no. 12, pp. 16–22, Dec. 2016.

[52] A. H. Chowdhury *et al.*, “Route-over vs Mesh-under Routing in 6LoWPAN,” in *Proceedings of the 2009 international conference on wireless communications and mobile computing: Connecting the world wirelessly*, 2009, pp. 1208–1212.

[53] T. Winter, “RPL: IPv6 routing protocol for low-power and lossy networks,” 2012.

[54] P. Samar, M. R. Pearlman, and Z. J. Haas, “Independent Zone Routing: An Adaptive Hybrid Routing Framework for Ad Hoc Wireless Networks,” *IEEEACM Trans. Netw.*, vol. 12, no. 4, pp. 595–608, Aug. 2004.

[55] L. Villasenor-Gonzalez, Y. Ge, and L. Lament, “HOLSR: a hierarchical proactive routing mechanism for mobile ad hoc networks,” *IEEE Commun. Mag.*, vol. 43, no. 7, pp. 118–125, 2005.

[56] N. Kim and O. Gnawali, “Networking Working Group J. Ko Internet-Draft J. Jeong Intended status: Standards Track J. Park Expires: February 18, 2013 J. Jun,” 2012.

[57] A. Conta and M. Gupta, “Internet control message protocol (icmpv6) for the internet protocol version 6 (ipv6) specification,” 2006.

[58] O. Gaddour and A. Koubâa, “RPL in a nutshell: A survey,” *Comput. Netw.*, vol. 56, no. 14, pp. 3163–3178, Sep. 2012.

[59] O. Gnawali, “The minimum rank with hysteresis objective function,” 2012.

[60] P. Thubert, “Objective function zero for the routing protocol for low-power and lossy networks (RPL),” 2012.

[61] JP. Vasseur, Ed. M. Kim, Ed., “Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks,” IETF, RFC 6551, Mar. 2012.

[62] D. S. J. De Couto, D. Aguayo, J. Bicke, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” in *9th Annu*, 2003, pp. 419–434.

[63] K. S. Hong and L. Choi, "DAG-based multipath routing for mobile sensor networks," in *ICTC 2011*, 2011, pp. 261–266.

[64] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," RFC 3561, 2003.

[65] U. S. Kushwaha, P. K. Gupta, and S. P. Ghrera, "Performance evaluation of AOMDV routing algorithm with local repair for wireless mesh networks," *CSI Trans. ICT*, vol. 2, no. 4, pp. 253–260, Jan. 2015.

[66] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," Internet Engineering Task Force (IETF), Standards Track rfc 7252, Jun. 2014.

[67] B. Shala, P. Wacht, U. Trick, A. Lehmann, B. Ghita, and S. Shiaeles, "Framework for automated functional testing of P2P-based M2M applications," in *Ubiquitous and Future Networks (ICUFN), 2017 Ninth International Conference on*, 2017, pp. 916–921.

[68] J. Fabini and T. Zseby, "M2M communication delay challenges: Application and measurement perspectives," in *Instrumentation and Measurement Technology Conference (I2MTC), 2015 IEEE International*, 2015, pp. 1859–1864.

[69] B. C. Villaverde, D. Pesch, R. De Paz Alberola, S. Fedor, and M. Boubekeur, "Constrained Application Protocol for Low Power Embedded Networks: A Survey," 2012, pp. 702–707.

[70] C. H. Tseng, "Multipath Load Balancing Routing for Internet of Things," *J. Sens.*, vol. 2016, pp. 1–8, 2016.

[71] G. Gupta and M. Younis, "Performance evaluation of load-balanced clustering of wireless sensor networks," in *10th International Conference on Telecommunications, 2003. ICT 2003*, 2003, vol. 2, pp. 1577–1583 vol.2.

[72] V. Kumar, S. Jain, S. Tiwari, and others, "Energy efficient clustering algorithms in wireless sensor networks: A survey," *IJCSI Int. J. Comput. Sci. Issues*, vol. 8, no. 5, pp. 1694–814, 2011.

[73] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 325–349, May 2005.

[74] Y. Chen and N. Nasser, "Energy-Balancing Multipath Routing Protocol for Wireless Sensor Networks," presented at the The Third International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, 2006.

[75] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey," *IEEE Commun. Surv. Tutor.*, vol. 15, no. 2, pp. 551–591, Second 2013.

[76] Z. Wang, L. Zhang, Z. Zheng, and J. Wang, "Energy balancing RPL protocol with multipath for wireless sensor networks," *Peer--Peer Netw. Appl.*, Jul. 2017.

[77] L. Lassouaoui, S. Rovedakis, F. Sailhan, and A. Wei, “Evaluation of energy aware routing metrics for RPL,” in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2016 IEEE 12th International Conference on*, 2016, pp. 1–8.

[78] M. Nassiri, M. Boujari, and S. V. Azhari, “Energy-aware and load-balanced parent selection in RPL routing for wireless sensor networks,” *Int. J. Wirel. Mob. Comput.*, vol. 9, no. 3, p. 231, 2015.

[79] C. Abreu, M. Ricardo, and P. M. Mendes, “Energy-aware routing for biomedical wireless sensor networks,” *J. Netw. Comput. Appl.*, vol. 40, pp. 270–278, Oct. 2013.

[80] J. Nurmio, E. Nigussie, and C. Poellabauer, “Equalizing Energy Distribution in Sensor Nodes through Optimization of RPL,” 2015, pp. 83–91.

[81] H.-S. Kim, J. Paek, and S. Bahk, “QU-RPL: Queue Utilization based RPL for Load Balancing in Large Scale Industrial Applications.”

[82] R. T. Chekka, T. Miao, and K.-H. Kim, “Implementation of adaptive binary exponential backoff (ABEB) algorithm with dynamical sizing buffer for load-balanced RPL,” in *Ubiquitous and Future Networks (ICUFN), 2014 Sixth International Conf on*, 2014, pp. 562–564.

[83] M. Zhao, A. Kumar, P. H. Joo Chong, and R. Lu, “A comprehensive study of RPL and P2P-RPL routing protocols: Implementation, challenges and opportunities,” *Peer--Peer Netw. Appl.*, Jul. 2016.

[84] P. Di Marco, G. Athanasiou, P.-V. Mekikis, and C. Fischione, “MAC-aware routing metrics for the internet of things,” *Comput. Commun.*, vol. 74, pp. 77–86, May 2015.

[85] P. D. Marco, C. Fischione, G. Athanasiou, and P. V. Mekikis, “MAC-aware routing metrics for low power and lossy networks,” in *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2013, pp. 79–80.

[86] P. Di Marco, C. Fischione, G. Athanasiou, and P.-V. Mekikis, “Harmonizing mac and routing in low power and lossy networks,” in *Global Communications Conference (GLOBECOM), 2013 IEEE*, 2013, pp. 231–236.

[87] H. Tall, G. Chalhoub, and M. Misson, “M-CoLBA: Multichannel Collaborative Load Balancing Algorithm with queue overflow avoidance in WSNs,” in *Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th International*, 2017, pp. 2127–2133.

[88] A. Ghaffari, “Congestion control mechanisms in wireless sensor networks: A survey,” *J. Netw. Comput. Appl.*, vol. 52, pp. 101–115, Jun. 2015.

[89] M. R. Mundada and P. B. Desai, “A survey of congestion in Wireless Sensor Networks,” in *Advances in Human Machine Interaction (HMI), 2016 International Conference on*, 2016, pp. 1–5.

[90] W. Tang, X. Ma, J. Huang, and J. Wei, “Toward Improved RPL: A Congestion Avoidance Multipath Routing Protocol with Time Factor for Wireless Sensor Networks,” *J. Sens.*, vol. 501, p. 264982, 2015.

[91] Y. Jin, S. Gormus, P. Kulkarni, and M. Sooriyabandara, “Content centric routing in IoT networks and its integration in RPL,” *Comput. Commun.*, pp. 87–104.

[92] H. A. Al-Kashoash, Y. Al-Nidawi, and A. H. Kemp, “Congestion-aware RPL for 6L0WPAN networks,” in *Wireless Telecommunications Symposium (WTS), 2016*, 2016, pp. 1–6.

[93] R. Ullah, Y. Faheem, and B. S. Kim, “Energy and Congestion-Aware Routing Metric for Smart Grid AMI Networks in Smart City,” *IEEE Access*, pp. 1–1, 2017.

[94] K. Sha, J. Gehlot, and R. Greve, “Multipath Routing Techniques in Wireless Sensor Networks: A Survey,” *Wirel. Pers. Commun.*, vol. 70, no. 2, pp. 807–829, May 2013.

[95] O. Iova, F. Theoleyre, and T. Noel, “Exploiting multiple parents in RPL to improve both the network lifetime and its stability,” in *2015 IEEE International Conference on Communications (ICC)*, 2015, pp. 610–616.

[96] A. Fraboulet, G. Chelius, and E. Fleury, “Worldsens: development and prototyping tools for application specific wireless sensors networks,” in *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, 2007, pp. 176–185.

[97] B. Pavković, F. Theoleyre, and A. Duda, “Multipath opportunistic RPL routing over IEEE 802.15. 4,” in *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, 2011, pp. 179–186.

[98] Q. Le, T. Ngo-Quynh, and T. Magedanz, “RPL-based multipath Routing Protocols for Internet of Things on Wireless Sensor Networks,” in *2014 International Conference on Advanced Technologies for Communications (ATC 2014)*, 2014, pp. 424–429.

[99] L. Zhu, R. Wang, and H. Yang, “Multi-Path Data Distribution Mechanism Based on RPL for Energy Consumption and Time Delay,” *Information*, vol. 8, no. 4, p. 124, Oct. 2017.

[100] S. Balandin, S. Andreev, and Y. Koucheryavy, Eds., *Parent-aware routing for iot networks*, vol. 9247. Cham: Springer International Publishing, 2015.

[101] A. K. Dey, “Providing architectural support for building context-aware applications,” Georgia Institute of Technology, 2000.

[102] B.-L. Wenning, *Context-based routing in dynamic networks*, 1. ed. Wiesbaden: Springer, 2010.

[103] M. Qasem, A. Al-Dubai, I. Romdhani, B. Ghaleb, and W. Gharibi, “A new efficient objective function for routing in Internet of Things paradigm,” in *Standards for Communications and Networking (CSCN), 2016 IEEE Conference on*, 2016, pp. 1–6.

[104] M. T. Nöel, “Going towards the future Internet of Things through a cross-layer optimization of the standard protocol suite,” Citeseer, 2013.

[105] P. Karkazis *et al.*, “Design of primary and composite routing metrics for RPL-compliant Wireless Sensor Networks,” in *Telecommunications and Multimedia (TEMU), 2012 International Conference on*, 2012, pp. 13–18.

[106] 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.

[107] A. Kalmar, R. Vida, and M. Maliosz, “CAEsAR: A Context-Aware Addressing and Routing scheme for RPL networks,” in *2015 IEEE International Conference on Communications (ICC)*, 2015, pp. 635–641.

[108] A. Kalmar, R. Vida, and M. Maliosz, “Context-aware addressing in the Internet of things using bloom filters,” in *Cognitive Infocommunications (CogInfoCom), 2013 IEEE 4th International Conference on*, 2013, pp. 487–492.

[109] T. A. Butt, I. Phillips, L. Guan, and G. Oikonomou, “TRENDY: An adaptive and context-aware service discovery protocol for 6LoWPANs,” in *Proceedings of the third international workshop on the web of things*, 2012, p. 2.

[110] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, “MQTT-S—A publish/subscribe protocol for Wireless Sensor Networks,” in *Communication systems software and middleware and workshops, 2008. comsware 2008. 3rd international conference on*, 2008, pp. 791–798.

[111] B. Sharkawy, A. Khattab, and K. M. F. Elsayed, “Fault-tolerant RPL through context awareness,” in *2014 IEEE World Forum on Internet of Things (WF-IoT)*, 2014, pp. 437–441.

[112] M. Musolesi and C. Mascolo, “CAR: Context-Aware Adaptive Routing for Delay-Tolerant Mobile Networks,” *IEEE Trans. Mob. Comput.*, vol. 8, no. 2, pp. 246–260, Feb. 2009.

[113] C. Mascolo and M. Musolesi, “SCAR: context-aware adaptive routing in delay tolerant mobile sensor networks,” in *Proceedings of the 2006 international conference on Wireless communications and mobile computing*, 2006, pp. 533–538.

[114] S. Taghizadeh, H. Bobarshad, and H. Elbiaze, “CLRPL: Context-Aware and Load Balancing RPL for IoT Networks under Heavy and Highly Dynamic load,” *IEEE Access*, pp. 1–1, 2018.

[115] L. A. Zadeh, “Fuzzy logic,” *Computer*, vol. 21, no. 4, pp. 83–93, 1988.

[116] A. M. Ortiz, F. Royo, T. Olivares, N. Crespi, and L. Orozco-Barbosa, “Smart wireless design scheme: fuzzy-logic routing and TDMA MAC protocol integration,” in *Proceedings of the 11th ACM international symposium on Mobility management and wireless access*, 2013, pp. 81–88.

[117] S. V. Que, N. T. Tri, and T. L. P. Nguyen, “Context-aware monitoring model using fuzzy logic for Wireless Sensor Networks in logistics,” in *2014 International Conference on Advanced Technologies for Communications (ATC 2014)*, 2014, pp. 78–83.

[118] S. Gajjar, M. Sarkar, and K. Dasgupta, “FAMACRO: Fuzzy and Ant Colony Optimization Based MAC/Routing Cross-layer Protocol for Wireless Sensor Networks,” *Procedia Comput. Sci.*, vol. 46, pp. 1014–1021, 2015.

[119] N. Khan, M. G. Martini, and D. Staehle, “QoS-aware composite scheduling using fuzzy proactive and reactive controllers,” *EURASIP J. Wirel. Commun. Netw.*, vol. 2014, no. 1, p. 138, 2014.

[120] E. C. C. Tsang and D. S. Yeung, “Optimizing fuzzy knowledge base by genetic algorithms and neural networks,” in *1999 IEEE International Conference on Systems, Man, and Cybernetics, 1999. IEEE SMC ’99 Conference Proceedings*, 1999, vol. 3, pp. 367–371 vol.3.

[121] O. Gaddour, A. Koubaa, N. Baccour, and M. Abid, “Of-fl: Qos-aware fuzzy logic objective function for the rpl routing protocol,” in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2014 12th International Symposium on*, 2014, pp. 365–372.

[122] H. Araújo *et al.*, “A Proposal for IoT Dynamic Routes Selection Based on Contextual Information,” *Sensors*, vol. 18, no. 2, p. 353, Jan. 2018.

[123] N. Accettura, L. A. Grieco, G. Boggia, and P. Camarda, “Performance analysis of the RPL routing protocol,” in *Mechatronics (ICM), 2011 IEEE International Conference on*, 2011, pp. 767–772.

[124] O. Gaddour, A. Koubâa, S. Chaudhry, M. Tezeghdanti, R. Chaari, and M. Abid, “Simulation and performance evaluation of DAG construction with RPL,” in *Third International Conference on Communications and Networking*, 2012, pp. 1–8.

[125] M. Vucinic, B. Tourancheau, and A. Duda, “Performance comparison of the rpl and loadng routing protocols in a home automation scenario,” in *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, 2013, pp. 1974–1979.

[126] S. Liu, M. Wu, C. Chen, B. Lv, and S. Li, “A high-throughput routing metric for multi-hop Ad hoc networks based on real time testbed,” in *TENCON 2013-2013 IEEE Region 10 Conference (31194)*, 2013, pp. 1–4.

[127] P. Gonizzi, R. Monica, and G. Ferrari, “Design and evaluation of a delay-efficient RPL routing metric,” in *Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International*, 2013, pp. 1573–1577.

[128] S. S. Rao and N. C. K. Reddy, “An energy efficient improved RPL routing protocol,” in *Intelligent Computing and Control (I2C2), 2017 International Conference on*, 2017, pp. 1–7.

[129] N. Tsiftes, J. Eriksson, N. Finne, F. Österlind, J. Höglund, and A. Dunkels, “A framework for low-power IPv6 routing simulation, experimentation, and evaluation,” in *ACM SIGCOMM Computer Communication Review*, 2010, vol. 40, pp. 479–480.

[130] M. Nuvolone, “Stability analysis of the delays of the routing protocol over low power and lossy networks,” KTH Royal Institute of Technology, 2010.

[131] Oana Iova, “Standards Optimization and Network Lifetime Maximization for Wireless Sensor Networks in the Internet of Things,” PhD Thesis, Universite de Strasbourg, 2015.

[132] 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*, 2004, pp. 455–462.

[133] A. Dunkels, O. Schmidt, T. Voigt, and M. Ali, “Protothreads: simplifying event-driven programming of memory-constrained embedded systems,” in *Proceedings of the 4th international conference on Embedded networked sensor systems*, 2006, pp. 29–42.

[134] N. Tsiftes, A. Dunkels, Z. He, and T. Voigt, “Enabling large-scale storage in sensor networks with the coffee file system,” in *Information Processing in Sensor Networks, 2009. IPSN 2009. International Conference on*, 2009, pp. 349–360.

[135] A. K. Dwivedi, M. K. Tiwari, and O. P. Vyas, “Operating systems for tiny networked sensors: A survey,” *Int. J. Recent Trends Eng.*, vol. 1, no. 2, pp. 152–157, 2009.

[136] “Contiki OS,” *Contiki-OS*. [Online]. Available: https://www.contiki-os.org/. [Accessed: 30-Aug-2017].

[137] “Contiki-ng,” *Contiki-NG*. [Online]. Available: https://www.contiki-ng.org/. [Accessed: 15-Jan-2018].

[138] P. Levis and D. Gay, *TinyOS Programming 1st*, 1st ed. New York, USA: Cambridge University Press, 2009.

[139] “TinyOS Documentation Wiki,” *TinyOS Documentation Wiki*. [Online]. Available: http://tinyos.stanford.edu/tinyos-wiki/index.php/Main_Page. [Accessed: 07-Mar-2017].

[140] M. O. Farooq and T. Kunz, “Operating Systems for Wireless Sensor Networks: A Survey,” *Sensors*, vol. 11, no. 12, pp. 5900–5930, May 2011.

[141] P. Levis *et al.*, “TinyOS: An operating system for sensor networks,” *Ambient Intell.*, vol. 35, pp. 115–148, 2005.

[142] H. Will, K. Schleiser, and J. Schiller, “A real-time kernel for wireless sensor networks employed in rescue scenarios,” in *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on*, 2009, pp. 834–841.

[143] “tinyos.stanford.edu.” [Online]. Available: http://tinyos.stanford.edu. [Accessed: 30-Jul-2017].

[144] E. Baccelli, O. Hahm, M. Wählisch, M. Günes, and T. Schmidt, “RIOT: one OS to rule them all in the IoT,” INRIA, 2012.

[145] “RIOT OS.” [Online]. Available: https://riot-os.org/. [Accessed: 07-Jun-2017].

[146] E. Baccelli, O. Hahm, M. Gunes, M. Wahlisch, and T. C. Schmidt, “RIOT OS: Towards an OS for the Internet of Things,” in *Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on*, 2013, pp. 79–80.

[147] “IoT Standards and Protocols,” *postscapes*, 24-Jan-2018. [Online]. Available: https://www.postscapes.com/internet-of-things-protocols/#protocols.

[148] Y. Yao, L. Wan, and Q. Cao, “System Architecture and Operating Systems,” in *The Art of Wireless Sensor Networks*, H. M. Ammari, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 697–738.

[149] F. Osterlind, “A sensor network simulator for the Contiki OS,” SICS, T2006:05 ISRN:SICS-T2006/05-SE, 2006.

[150] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, “Cross-level sensor network simulation with cooja,” in *Local computer networks, proceedings 2006 31st IEEE conference on*, 2006, pp. 641–648.

[151] J. Eriksson *et al.*, “COOJA/MSPSim: interoperability testing for wireless sensor networks,” 2009.

[152] E. Ancillotti, R. Bruno, and M. Conti, “RPL routing protocol in advanced metering infrastructures: an analysis of the unreliability problems,” in *Sustainable Internet and ICT for Sustainability (SustainIT), 2012*, 2012, pp. 1–10.

[153] H. Sundani, H. Li, V. Devabhaktuni, M. Alam, and P. Bhattacharya, “Wireless sensor network simulators a survey and comparisons,” *Int. J. Comput. Netw.*, vol. 2, no. 5, pp. 249–265, 2011.

[154] M. Jevtić, N. Zogović, and G. Dimić, “Evaluation of wireless sensor network simulators,” in *Proceedings of the 17th Telecommunications Forum (TELFOR 2009), Belgrade, Serbia*, 2009, pp. 1303–1306.

[155] “advanticsys,” *advanticsys*. [Online]. Available: http://www.advanticsys.com/shop/mtmcm5000msp-p-14.html. [Accessed: 01-May-2015].

[156] J. . HUNTER, “The exponentially weighted moving average,” *Jal Qual. Technol.*, vol. 18, pp. 203–210, 1986.

[157] C. A. Lowry, W. H. Woodall, C. W. Champ, and S. E. Rigdon, “A Multivariate Exponentially Weighted Moving Average Control Chart,” *Technometrics*, vol. 34, no. 1, pp. 46–53, Feb. 1992.

[158] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko, “The trickle algorithm,” Internet Engineering Task Force (IETF), RFC 6206, 2011.

[159] K. Pister, P. Thubert, S. Dwars, and T. Phinney, “Industrial routing requirements in low-power and lossy networks,” Internet Engineering Task Force (IETF), RFC 5673, 2009.

[160] M. Qasem, A. Al-Dubai, I. Romdhani, and B. Ghaleb, “Load Balancing Objective Function in RPL,” IETF, Standards Draft 1, Aug. 2017.

[161] S. E. Deering, “Internet protocol, version 6 (IPv6) specification,” 1998.

[162] A. Friday, M. Wu, J. Finney, S. Schmid, K. Cheverst, and N. Davies, “Network layer access control for context-aware IPv6 applications,” *Wirel. Netw.*, vol. 9, no. 4, pp. 299–309, 2003.

[163] E. Ancillotti, R. Bruno, and M. Conti, “Reliable Data Delivery With the IETF Routing Protocol for Low-Power and Lossy Networks,” *IEEE Trans. Ind. Inform.*, vol. 10, no. 3, pp. 1864–1877, Aug. 2014.

[164] J. Ko, J. Jeong, J. Park, J. A. Jun, O. Gnawali, and J. Paek, “DualMOP-RPL: Supporting Multiple Modes of Downward Routing in a Single RPL Network,” *ACM Trans. Sens. Netw.*, vol. 11, no. 2, pp. 1–20, Mar. 2015.

[165] J. Polastre, R. Szewczyk, and D. Culler, “Telos: enabling ultra-low power wireless research,” in *Proceedings of the 4th international symposium on Information processing in sensor networks*, 2005, p. 48.

[166] M. Hefeeda and M. Bagheri, “Forest Fire Modeling and Early Detection using Wireless Sensor Networks.,” *Ad Hoc Sens. Wirel. Netw.*, vol. 7, no. 3–4, pp. 169–224, 2009.

[167] “How much forest does Canada have?,” *Natural Resources Canada*, Mar-2018. [Online]. Available: https://www.nrcan.gc.ca/forests/report/area/17601. [Accessed: 01-Apr-2018].