



HAL
open science

Toward reliable and bounded latency for internet of things

Tomas Lagos Jenschke

► **To cite this version:**

Tomas Lagos Jenschke. Toward reliable and bounded latency for internet of things. Networking and Internet Architecture [cs.NI]. Ecole nationale supérieure Mines-Télécom Atlantique, 2020. English. NNT : 2020IMTA0222 . tel-03135558

HAL Id: tel-03135558

<https://theses.hal.science/tel-03135558>

Submitted on 9 Feb 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPERIEURE MINES-TELECOM ATLANTIQUE
BRETAGNE PAYS DE LA LOIRE - IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies
de l'Information et de la Communication*
Spécialité : *informatique*

Par

Tomás LAGOS JENSCHKE

Toward Reliable and Bounded Latency for Internet of Things

Thèse présentée et soutenue à IMT Atlantique, Campus Rennes, France, le 04/12/2020
Unité de recherche : Département systèmes réseaux, cybersécurité et droit du numérique (SRCD)
Thèse N° : 2020IMTA0222

Rapporteurs avant soutenance :

MITTON Nathalie Directrice de Recherche, Inria Lille-Nord Europe, France
GALLAIS Antoine Professeur, INSA Hauts-de-France, France

Composition du Jury :

Président :	GALLAIS Antoine	Professeur, INSA Hauts-de-France, France
Examineurs :	DUJOVNE Diego	Maitre de conférences, Universidad Diego Portales, Chili
	MITTON Nathalie	Directrice de Recherche, Inria Lille-Nord Europe, France
	PAPADOPOULOS Georgios	Maitre de conférences, IMT Atlantique, France
	QUOITIN Bruno	Maitre de conférences, Université de Mons, Belgique
Dir. de thèse :	MONTAVONT Nicolas	Professeur, IMT Atlantique, France

Invité(s) :

THUBERT Pascal Ingénieur en chef, Cisco, France

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgments.

Rennes, December 04, 2020

Tomás LAGOS JENSCHKE

This thesis is dedicated to all the people who encouraged me to start, persevere, and finish this great journey.

ACKNOWLEDGEMENT

I want to thank all the people involved in this great journey. Starting with the members of my team, my thesis supervisor, Georgios Papadopoulos, was the one who pushed me to keep going in this thesis work throughout these 3 years. Without him and his ideas, I would not have been able to get to where I am now. In addition, my thesis supervisor, Nicolas Montavont, led me to new ideas that broadened my view of the problems. Finally, I thank Remous-Aris Koutsiamanis, who worked with me during my thesis.

I would like to thank my thesis jury members Nathalie Mitton, Antoine Gallais, Diego Dujovne, Bruno Quoitin, and Pascal Thubert for their corrections, observations, and discussions we had during the thesis defense.

Returning to the IMT Atlantique team, I would like to thank all the members of the OCIF team for allowing me to discover new research and interesting ideas. I would also like to thank my colleagues and friends who transformed this stage into a very unique one.

Finally, I would like to thank my family, girlfriend, and friends who encouraged me to make and finish this great experience.

LIST OF PUBLICATIONS

Journal Article

- [Lag+21] T. Lagos Jenschke, R.-A. Koutsiamanis, G. Z. Papadopoulos, and N. Montavont, « ODeSe: On-Demand Selection for Multi-path RPL Networks », *in: Ad Hoc Networks* (2021).

Conference Proceedings

- [Lag+20] T. Lagos Jenschke, G. Papadopoulos, R.-A. Koutsiamanis, and N. Montavont, « Stratégies d'ancêtre commun pour les réseaux RPL multi-chemins », *in: Proceedings of the Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication (CoRes)*, Lyon, France, Sept. 2020.
- [Cza+20] A. Czarnitzki Estrin, T. Lagos Jenschke, G. Z. Papadopoulos, I Alvarez-Hamelin, and N. Montavont, « Thorough Investigation of multipath Techniques in RPL based Wireless Networks », *in: Proceedings of the IEEE Symposium on Computers and Communications (ISCC)*, Rennes, France, July 2020.
- [Kou+20] R.-A. Koutsiamanis, G. Z. Papadopoulos, T. Lagos Jenschke, P. Thubert, and N. Montavont, « Meet the PAREO Functions: Towards Reliable and Available Wireless Networks », *in: Proceedings of the IEEE International Conference on Communications (ICC)*, Dublin, Ireland, June 2020.
- [Lag+19] T. Lagos Jenschke, G. Z. Papadopoulos, R.-A. Koutsiamanis, and N. Montavont, « Alternative Parent Selection for Multi-Path RPL Networks », *in: Proceedings of the IEEE World Forum on the Internet of Things (WF-IoT)*, Limerick, Ireland, Apr. 2019.

- [Lag+18] T. Lagos Jenschke, R.-A. Koutsiamanis, G. Z. Papadopoulos, and N. Montavont, « Multi-path Selection in RPL Based on Replication and Elimination », *in: Proceedings of the International Conference on Ad-Hoc Networks and Wireless*, St. Malo, France, Aug. 2018.

ABBREVIATIONS

6LoRH	6LoWPAN Routing Header.
ACK	acknowledgment.
AODV	Ad-hoc On Demand Distance Vector.
AP	Alternative Parent.
ARQ	Automatic Repeat reQuest.
ASN	Absolute Slot Number.
CA	Common Ancestor.
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance.
CTS	Clear to Send.
DAG	Directed Acyclic Graph.
DAO	Destination Advertisement Object.
DIO	DODAG Information Object.
DIS	DODAG Informational Solicitation.
DODAG	Destination Oriented Directed Acyclic Graph.
ETX	Expected Transmission Count.
FDMA	Frequency Division Multiple Access.
FHSS	Frequency Hopping Spread Spectrum.
IETF	Internet Engineering Task Force.
IoT	Internet of Things.
LANET	Visible-light Ad-hoc network.
LIFO	Last In, First Out.

LLN	Low Power and Lossy Network.
LOADng	Lightweight On-Demand Ad hoc Distance Vector Routing Protocol-Next Generation.
LPWAN	Low-Power Wide Area Network.
LR-WPAN	Low-Rate Wireless Personal Area Network.
LRU	Least Recently Used.
MAC	Medium Access Control.
MANET	Mobile Ad-hoc Networking.
MC	Metric Container.
MRHOF	Minimum Rank with Hysteresis Objective Function.
NSA	Node State and Attribute.
ODeSe	On-Demand Selection.
OF	Objective Function.
OF0	Objective Function Zero.
OH	Overhearing.
PAREO	Packet Automatic Repeat reQuest, Replication and Elimination, and Overhearing.
PDR	Packet Delivery Ratio.
PER	Packet Error Ratio.
PP	Preferred Parent.
PS	Parent Set.
QoS	Quality of Service.
RE	Replication and Elimination.
RPL	IPv6 Routing Protocol for Low-Power and Lossy Networks.
RTS	Request to Send.
RTX	Retransmission.

SRH	Source Routing Header.
TDMA	Time Division Multiple Access.
TSCH	Timeslotted Channel Hopping.
VANET	Vehicular Ad-hoc Network.
WSN	Wireless Sensor Network.

TABLE OF CONTENTS

List of Figures	vi
List of Tables	vii
1 Introduction	1
1.1 General Context	1
1.2 Thesis background	3
1.3 Motivation	4
1.4 Thesis contributions	5
1.5 Thesis structure	6
2 State-of-the-art	9
2.1 IEEE std 802.15.4	9
2.1.1 Channel access methods	10
2.1.2 TSCH	11
2.2 Ad-Hoc Networks	13
2.2.1 Routing - A problem statement	13
2.2.2 Discussion of an existing debate	16
2.2.3 A different approach	17
2.3 RPL	18
2.3.1 Instance	18
2.3.2 Objective Function (OF)	19
2.3.3 Control Packets	20
2.3.4 DODAG Formation	22
2.3.5 Trickle Timer	23
2.4 Multi-path routing	23
2.5 Conclusion	29
3 Patterns and Replication functions	31
3.1 Introduction	31

TABLE OF CONTENTS

3.2	Redudant patterns	32
3.2.1	Disjoint pattern	34
3.2.2	Triangle pattern	34
3.2.3	Braided pattern	35
3.2.4	Patterns reliability evaluation	35
3.3	PAREO functions	36
3.3.1	Automatic Repeat reQuest function	36
3.3.2	Replication and Elimination function	37
3.3.3	Overhearing	38
3.3.4	Topology	40
3.3.5	Performance Evaluation	41
3.4	Conclusions	44
4	N-disjoint	47
4.1	Introduction	47
4.2	RE and ARQ	48
4.3	N-Disjoint Strategies	48
4.3.1	Basic Strategy	49
4.3.2	Advanced Strategy	50
4.4	Performance evaluation	51
4.4.1	Simulation Setup	51
4.4.2	Simulation Results	52
4.5	Conclusions	55
5	Common Ancestor selection	57
5.1	Introduction	57
5.2	Selection models	57
5.2.1	Strict CA	58
5.2.2	Medium CA	59
5.2.3	Soft CA	60
5.2.4	Discussion	60
5.2.5	Implementation Aspects	61
5.3	Performance evaluation	62
5.3.1	Simulation setup	62
5.3.2	Simulation Results	63

5.4	Conclusions	68
6	On-Demand Selection	71
6.1	Introduction	71
6.2	CA Trade-offs	72
6.3	ODeSe	73
6.4	Performance evaluation	76
6.4.1	Number of relays	76
6.4.2	Simulation setup	78
6.4.3	Simulation results	79
6.5	Conclusions	82
7	Conclusion and future work	83
7.1	Conclusion	83
7.2	Perspectives	85
7.2.1	MAC layer and scheduling protocols	85
7.2.2	PAREO functions in reactive routing protocols	85
7.2.3	Topology and experimenting environments	86
7.2.4	Cooja simulator and multi-path RPL	86
	Appendices	87
A	Résumé en français:	
	Vers une latence fiable et limitée pour l'Internet des objets	88
A.1	Contexte général	88
A.2	Contexte de la thèse	90
A.3	Contributions de thèse	92
A.4	Modèles de sélection	93
A.5	Algorithme N-Disjoint	93
A.5.1	Stratégie de base	94
A.5.2	Stratégie avancée	94
A.6	Les algorithmes d'ancêtres communs (CA)	94
A.6.1	Strict CA	95
A.6.2	Medium CA	95
A.6.3	Soft CA	95
A.7	Algorithme de sélection on-demand (ODeSe)	95

TABLE OF CONTENTS

A.8 Conclusions	96
Bibliography	99

LIST OF FIGURES

Figure 2.1	Time Slot Channel Hopping (TSCH) schedule example.	11
Figure 2.2	DODAG Information Option (DIO) Base Object.	20
Figure 2.3	Destination Advertisement Option (DAO) Base Object.	21
Figure 2.4	DODAG Informational Solicitation (DIS) Base Object.	21
Figure 2.5	Destination Oriented Directed Acyclic Graph (DODAG) example. .	22
Figure 3.1	Transmissions attempts required for a single hop delivery.	33
Figure 3.2	Redundant patterns for TSCH networks.	34
Figure 3.3	Automatic Repeat reQuest (ARQ) example with 1 transmission and 3 RTX.	37
Figure 3.4	Multi-path transmission by Replication and Elimination (RE). . . .	38
Figure 3.5	Overhearing (OH) functionality representation.	39
Figure 3.6	Experimental mesh topology illustration.	40
Figure 3.7	PDR obtained by using the PAREO function.	42
Figure 3.8	Delay and Jitter (Percentiles of Delay) obtained by using the PAREO function.	43
Figure 3.9	PAREO power consumption per node per slotframe due to radio operation.	44
Figure 4.1	Potential N-disjoint scenarios.	49
Figure 4.2	N-disjoint latency using different replicas and RTX.	52
Figure 4.3	N-disjoint PDR using different replicas and RTX.	53
Figure 4.4	N-disjoint energy performance.	54
Figure 4.5	Average relay nodes per transmission using N-disjoint.	55
Figure 5.1	Algorithm list for Common Ancestor (CA) selection.	59
Figure 5.2	Probability of obtaining an AP according to the algorithm.	61
Figure 5.3	DAG Metric Container (MC) with nested Node State and Attribute (NSA) object.	62
Figure 5.4	PDR obtained by each CA algorithm.	64

LIST OF FIGURES

Figure 5.5 Latency obtained by each CA algorithm. 65

Figure 5.6 average number of copies per packet (a), and average number of
 relay nodes (b) per simulated environment. 66

Figure 5.7 Energy expenditure of each CA algorithm. 68

Figure 6.1 Flooding using the strict CA algorithm. 72

Figure 6.2 On-Demand Selection (ODeSe) functions for parent selection 73

Figure 6.3 Hop-by-Hop option fields (RPL option and ODeSe field). 76

Figure 6.4 Average number of relay nodes used per layer. 77

Figure 6.5 CA algorithm PDR compared to ODeSe PDR. 79

Figure 6.6 (a) Average power consumption by CA and ODeSE algorithms and
 (b) Relationship between average power consumption and PER by algo-
 rithm. The line represents the Pareto’s frontier. 80

Figure 6.7 (a) Average number of traverse nodes per transmission by CA and
 ODeSE algorithms and (b) Relationship between average power consump-
 tion and traverse nodes per transmission by algorithm. The line represents
 the Pareto’s frontier. 81

LIST OF TABLES

Table 2.1	Comparative table of multi-path RPL algorithms.	28
Table 3.1	Simulation setup PAREO.	41
Table 4.1	Simulation setup for N-disjoint.	51
Table 5.1	Simulation setup for CA algorithms.	63
Table 6.1	Simulation setup for ODeSe algorithm.	78

INTRODUCTION

1.1 General Context

The Internet of Things (IoT) [Gub+13] is a set of standards and technologies developed to connect integrated and intelligent objects to the Internet. The technologies covered are characterized by their low production cost, their great adaptability to different environments, and their ease develop. Moreover, IoT technologies have been massively integrated into various systems and applications such as smart grid or industry thanks to these features. In response to that, they have been required to become flexible to any environment while providing rapid responses to failures. Thus, in the last decade, there has been a constant migration towards wireless technologies since they allow flexible positional installation [Can+18].

The industry has recently begun to adopt the IoTs in its production chains, evolving towards Industry 4.0 [Lu17]. Such a concept consists of incorporating devices within a network, both wired and wireless, in order to maintain the industry's different systems autonomously. All these thanks to the consistency of the communication that a device has with the other nodes within its working area. Furthermore, it allows for potential centralization of the production chain since an IoT node can act as an administrator for other devices. As a result, the production complexity tends to be rather automatic and, thus, less expensive.

Taking the approach of Industry 4.0, we cover different concepts that belong to the IoT, being the Wireless Sensor Network (WSN) one of the predominant in this market. These networks are made up of hundreds of nodes equipped with multiple sensors to monitor physical or environmental conditions. Their distribution deployment is towards a centralized node that is called the gateway or the sink. As a result, these nodes operate cooperatively to transfer their data to the sink node. However, the problem encountered with these technologies is their limited capabilities in terms of CPU and memory, as well as their limitations of wireless technologies.

It is known that wired networks maintain stable communication among the sensor devices, while the wireless networks are exposed to environmental conditions that result in communication interference and collisions. These limitations are currently mitigated by predictive systems, in addition to intelligent processes that provide communication robustness. Unfortunately, in order to make them operate properly, systems with a high level of processing are required. Moreover, due to the mobility feature that wireless technologies allow, many of the devices are battery-powered, resulting in a limited lifetime.

Several approaches have been made to support these technologies, being the manipulation of routing one of them. In fact, this approach consists of providing a path according to the requirements of the application, which will later be used to route the transmission of a packet. Nevertheless, to make this possible, different strategies and decisions must be considered.

As mentioned in [Pis+09], the industrial topology can constitute approximately 10 to 200 interconnected devices. Moreover, in smart grid technologies, the topology can be made up of thousands of devices [MZX18]. This may have repercussions on the routing algorithms' decisions since as more paths exist towards the destination, the more information will be processed. Besides, given the variability of wireless networks, some nodes can relay to invalid addresses, making it necessary to detect those nodes capable of providing the corresponding routing. For this reason, a routing system must guarantee fast and without disturbances communication between the different nodes. Furthermore, given that some systems require rapid and precise communication exchanges, the routing protocol must guarantee transmission reliability since consecutive packet losses can lead to the malfunctioning of these systems.

This dramatically affects the industrial networks, as they require that transmissions maintain a minimum quality standard. In fact, the industry's Quality of Service (QoS) consists of achieving high transmission reliability, low latency fluctuation, and optimized energy expenditure.

Fortunately, these requirements can be addressed in different ways as shown in [KK20]. Here the authors explain that there are several strategic mechanisms to address the QoS of industrial use cases. However, since each mechanism has different behaviors, their environmental conditions are also different, requiring varying trade-offs. To clarify this, we will take as an example of the selective mechanisms of differentiation and grouping of services.

Service differentiation consists of effectively balancing a limited network of resources

based on one or more criteria. However, it requires a mechanism that detects the dynamic allocation of priorities and the dynamic conditions of the network.

Clustering, on the other hand, divides the network into small clusters, each with a cluster head. Consequently, the nodes will only focus their transmission on their cluster heads, while the cluster heads will focus their transmission efforts on the sink. However, given the amount of traffic that must pass through the cluster heads, the clusters exhaust their residual energy more quickly.

Both examples can be used to improve the same industry requirement, but their different strategies indicate that their trade-offs will be different. Therefore, it is necessary to implement a strategy that adapts to the needs of a system without causing a considerable impact on network resources. In this manuscript, we intend to address the following requirements to guarantee the QoS of the industry:

- Ensure high-level of network reliability;
- Keep low delay variation (i.e., jitter);
- Optimize the network latency;
- Maintain a low and balanced power consumption.

1.2 Thesis background

The Low Power and Lossy Network (LLN) [Gha+18] refers to a class of wired and wireless networks in which the nodes are very limited in resources and communication technologies. Their resource limitations include restricted power reserves, processing, and storage capabilities. They also have a low data rate, highly asymmetric link characteristics, and short communication distances. However, their low cost makes them attractive to the market. One of the components that characterize the LLN is the hops, which represent the different connections of a route between devices in a network. There are LLNs with multi-hop structures like WSNs or single-hop structures like the Low-Power Wide Area Networks (LPWANs) [Far18], as well as encompassing other IoT technologies.

Because of this, the Internet Engineering Task Force (IETF) defined the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL). This routing protocol is characterized by not depending on the other layers of the network attached to it, as well as by providing a routing method adaptable to different network environments. Besides, it has a route repair mechanism that consists of redirecting the traffic to another valid node if a node of the original route stops working or because the new node satisfies the established routing needs

more efficiently. As a result, RPL has become one of the most popular routing protocols for LLNs. However, because the default implementation of this protocol is single-route routing, end-to-end reliability is usually low for industrial networks.

Within an LLN, transmission reliability is generally directly related to the total hop distance that a packet must traverse. The reason for this is that as the source is located further away from the sink, the packet is immersed in a greater number of hops, which in turn are subject to wireless conditions. Furthermore, if one of the relay nodes suddenly crashes or fails, the RPL repair procedure can cause a significant increase in the delay since it must find another route.

A standard solution to this problem is to set up multiple transmission retries [FW02], providing more opportunities for a packet to perform a successful transmission. However, the more transmission retries required, the greater the delay and performance fluctuations.

Another possible solution is the use of multiple routes to transmit a single packet. More specifically, broadcast a packet by multi-path instead of single-path. However, uncontrolled broadcasting can lead to traffic overload and flooding. Moreover, as more devices are used, the shorter their life expectancy will be. Therefore, it is required that the broadcasting must be done to strategically targeted parents.

In this manuscript, we focus on different tools and methodologies to extend RPL with multi-path to ensure transmission reliability. All this while maintaining an acceptable power consumption. Additionally, we provide a study and analysis of the related work in the state of the art to provide an overview of the problem.

1.3 Motivation

As previously stated, LLNs have low transmission reliability due to their characteristics. Consequently, this behavior has negatively affected their migration to industry 4.0 since it requires a QoS that LLN technologies do not offer. As a result, this has led to the development of several strategies to increase this reliability. However, for each improvement achieved, a trade-off is required on a different resource.

An important part of ensuring the network reliability of packet transmission is routing. This is since the protocols in charge must elaborate strategic paths to satisfy the system's domain. As a result, the IETF specifies the RPL protocol, which focuses on LLN. Given the flexibility and behavior that RPL delivers to perform its routing, it has opened multiple possibilities to perform reliability strategies.

One of these approaches is multi-path routing, which consists of selecting different paths from a node. This routing can be used for various objectives, such as for load balancing or for network reliability. However, given its expansive behavior, uncontrolled multi-path routing can lead to flooding, traffic overload, or unnecessary energy consumption. Consequently, identifying an optimal algorithm to extend RPL with multi-path has been challenging.

Therefore, it is essential to provide new multi-path algorithms applied for RPL, and not only, but that also ensure network reliability while meeting the further requirements of the system.

1.4 Thesis contributions

This manuscript intends to provide redundant routing functions and algorithms to extend the multi-path RPL routing protocol. Our goal is to generate routing algorithms for LLNs that ensure transmission reliability, stable latency, and balanced power consumption for use in the industry 4.0 environment.

Commencing with the Packet Automatic Repeat reQuest, Replication and Elimination, and Overhearing (PAREO) functions, its goal is to provide the necessary tools to support multi-path in RPL. Its Replication and Elimination (RE) function consists of allowing a source node to make multiple copies of a packet or replicas to the different neighboring nodes that the source selected during its routing to make a transmission. The Automatic Repeat reQuest (ARQ) function, on the other hand, provides a maximum number of retransmissions to a node in order to use them every time a packet cannot make a hop. Finally, the Overhearing (OH) function allows a neighboring node to receive the same message from a receiving node if this neighboring node is within the range of the transmitting node.

As for multi-path algorithms, we have the N-Disjoint algorithm which consists of exploiting the bases of a disjoint pattern using the functions of PAREO RE and ARQ. Its goal is to ensure a reliable transmission by maintaining a balance between the number of retransmissions and the number of replicas.

The Common Ancestor (CA) algorithm involves the elaboration of multiple redundant paths, which are merged repeatedly along with the routing in order to concentrate a large number of retransmission attempts towards a hop. However, to achieve this a node must forward its transmission to strategic nodes in order to maintain a high number of path

merges. Given this, there is a probability of not getting a suitable parent. Consequently, CA establishes three algorithms: Strict CA, Medium CA, and Soft CA, each of which has a different selection policy based on the probability of finding a suitable node for the algorithm. The goal of the CA algorithms is to ensure transmission reliability by maintaining a minimum number of retransmissions.

Finally, the On-Demand Selection (ODeSe) algorithm adopts the same principles of the CA algorithm but with the difference that it extends them in an on-demand way to each node where a packet transits. The goal of the ODeSe algorithm is to ensure transmission reliability by maintaining a minimum number of retransmissions and maintaining an optimized energy consumption.

1.5 Thesis structure

This thesis is organized into 7 chapters. Chapter 1 is an introductory chapter. Here the problem statement is introduced, as well as a summary of the chapters that constitute this research.

Chapter 2 consists of the current state of the art of research. It introduces the main components of the IEEE std 802.15.4 standard. Additionally, we discuss the Ad-Hoc network and routing in Section 2.2. We also present RPL and its components in Section 2.3. Finally, a related work on multi-path RPL researches is detailed in Section 2.4, and a conclusion in Section 2.4.

Chapter 3 is divided into two parts. The first consists of a series of redundant patterns presented in Section 3.2. Then, the functions of PAREO are presented in Section 3.3 and evaluated in Section 3.3.5.

In Chapter 4, we provide the multi-path N-disjoint algorithm. The functions used within the disjoint pattern are described in Section 4.2, as well as its strategies in Section 4.3. Additionally, a performance evaluation is provided in Section 4.4.

In Chapter 5 we introduce the CA selection method were in Section 5.2, where three algorithms are presented. Moreover, the performance evaluation of the proposed algorithms is explained in Section 5.3.

In Chapter 6, we present our last contribution, the ODeSe algorithm. A discussion on the CA selection policy is given in Section 6.2, as well as the functionality of ODeSe in Section 6.3, while the performance evaluation is presented in Section 6.4 as well as a conclusion in Section 6.5.

Finally, Chapter 7 presents the conclusion of the thesis and the perspectives of this research.

STATE-OF-THE-ART

2.1 IEEE std 802.15.4

The Low-Rate Wireless Personal Area Network (LR-WPAN) is a standard for communication between different types of wireless devices with low data rates and located near a sink or border router. The mission of this standard originates from the need to manage several areas in an individual workspace. As a result, the distributive complexity and costs of the process are reduced due to individualization.

Within this standard, IEEE std 802.15.4 [20] is responsible for defining the physical layer and Medium Access Control (MAC) layer of the participating devices. Therefore, it provides the basis for enabling communication.

The properties that characterize it are:

- Communication synchronization;
- Collision anticipation;
- Communication security support.

There are also power consumption monitoring functions, such as power quality links and power detection.

At the physical level, IEEE std 802.15.4 operates in one of three possible unregulated frequency bands.

- 868-868.6 MHz operated in Europe with one channel;
- 902-928 MHz operated in North America with 13 channels;
- 2400-2483.5MHz operated worldwide with 16 channels.

Finally, this standard comprises node networks based on star and mesh topology. In this chapter, we proceed to explain some of the most relevant communication standards of IEEE std 802.15.4 to conclude with the Timeslotted Channel Hopping (TSCH) algorithm.

2.1.1 Channel access methods

At the MAC layer, channel access methods allow more than one device to use the same communication medium for transmission and reception. These methods use multiplexing, which consists of letting several links share the same communication channel.

This section describes some of the most commonly used channel access methods in IEEE std 802.15.4.

CSMA/CA

The Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol consists of determining whether the transmission medium is available to avoid collisions. To achieve this, each device will communicate a Request to Send (RTS) packet to its access point.

In the event of a collision, a node will establish a random timeout to retry sending the message again. Otherwise, the access point will send a Clear to Send (CTS) packet which informs the node that the medium is available for transmission. However, if the CTS packet does not belong to that node, it is inhibited for a certain period.

Finally, when a node completes its transmission, the access point sends an acknowledgment (ACK) packet informing that the transmission medium is clear.

TDMA

The Time Division Multiple Access (TDMA) consists of mapping communication between two or more devices at specific time intervals called *TimeSlots*. The purpose is to synchronize the sending of a packet in specific instances. As a result, the activity of the radio links decreases as they do not require being constantly awake. However, this methodology tends to suffer from more collisions as more nodes are incorporated into the network. Therefore, TDMA schedules transmissions on different *TimeSlots* overtime to mitigate them.

Since there are mobile nodes in wireless networks, the structure of the topology changes. As a result, TDMA scheduling must be adapted to different conditions, resulting in different solutions. This can be observed in [SVV15], where the authors have taken an in-depth look at this problem by preparing a survey of different proposals.

Finally, TDMA is distinguished in IEEE std 802.15.4 mesh networks due to its flexibility and easy implementation.

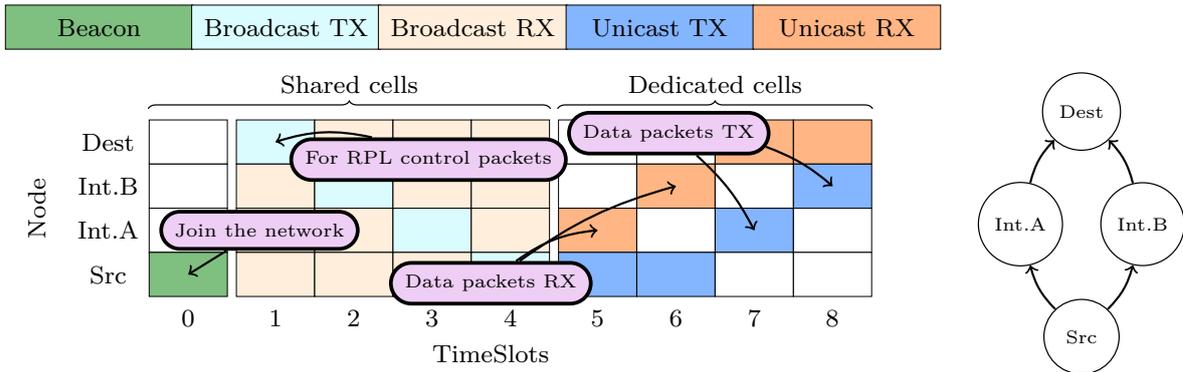


Figure 2.1 – Time Slot Channel Hopping (TSCH) schedule example.

FDMA

The Frequency Division Multiple Access (FDMA) is a MAC mechanism used in digital and analog communication protocols to divide the available spectrum into channels, representing different frequency ranges. These channels are assigned to several nodes so that they can communicate without interfering with each other. It should be noted that to establish communication between two devices, both nodes must be situated on the same channel.

2.1.2 TSCH

IEEE std 802.15.4-2015 TSCH [WPG15] is a wireless MAC protocol that provides predictable medium access and mitigates potential multi-path fading. TSCH allows nodes in the network to coordinate through a communication scheduler which uses TDMA for the time dimension and FDMA for the frequency dimension.

More specifically, time is divided into *timeslots* and the available frequency range into multiple non-overlapping physical radio *channels*. We call *cell* a (timeslot, channel offset) pair, which is the representation of a communication opportunity at a specific point in time and frequency. Frequency Hopping Spread Spectrum (FHSS) is employed to mitigate multi-path fading and for this reason, the cells in the schedule do not use the physical radio channel values directly but a *channel offset*, which is mapped at each timeslot to a different but deterministically computed physical channel. Finally, a set of timeslots are organized into a *slotframe* of length equal to the repetition period (in timeslots). IEEE std 802.15.4-2015 describes how TSCH should operate, but the definition of the schedule

is left to specific implementations based on the needs of the application domain.

Cell types

As shown in Figure 2.1, it is possible to classify cells into two types.

- Shared cells: these are the contention-based cells, i.e. CSMA/CA, and they are used for control plane traffic.
- Dedicated cells: these are the contention-free cells. Only a specific pair of nodes can communicate concurrently so collisions are avoided within a given TSCH network. These cells are commonly used for application traffic.

Channel offset mapping (on hold)

To support FHSS, a channel offset is mapped to a physical channel using the following Equation 2.1:

$$CH = F[(ASN + ChannelOffset) \bmod n_{CH}] \quad (2.1)$$

where n_{CH} is the number of existing physical channels, Absolute Slot Number (ASN) is the timeslot counter for the lifetime of the network, and F is a lookup table that maps the results from the equation to the radio channel CH .

Scheduling

While TSCH defines the general structure and operation of the schedule, the specific contents of the schedule (which cells will be employed) as well as the processes of creating and distributing the schedule between nodes are left to the implementation. These latter processes can be classified into two broad categories:

- Centralized scheduling: a schedule is computed in a single central place, taking into account the needs of all the nodes of the network, and is distributed to the nodes afterwards.
- Decentralized scheduling: nodes negotiate and create their schedule with the other nodes in their neighborhood. Thus, different parts of the whole network use different local schedules.

2.2 Ad-Hoc Networks

To standardize IP routing over wireless topologies with dynamism, IETF established the Mobile Ad-hoc Networking (MANET) [CM99] working group. This group is characterized by encompassing networks with both static and dynamic topology, which have limited bandwidth and employ devices with restricted power consumption. Furthermore, MANETs are designed for autonomous mobile systems that can be distributed as mesh or cellular networks.

Within this working group are LLNs [Pis+09], which are characterized by being composed of several interconnected nodes with limited power, memory, and processing. However, due to their simplicity of installation and low costs they have been studied within the industry. As a result, the following goals have been set for these networks:

- High reliability of data transmission.
- Low power consumption.
- Low temporal variability during transmission.

Furthermore, these systems require sustainable routing systems, with a high propagation speed and quick responses to anomalies and repairs.

In this section, we proceed to discuss the routing of LLNs with emphasis on their requirements and characteristics. For this, we investigate the multi-hop topology to study the reliability it requires. Also, we will analyze how routing should be adapted according to different scenarios, and finally, discuss how routing can also be migrated to different approaches.

2.2.1 Routing - A problem statement

Within telecommunications, a network consists of the interconnection of several nodes to allow communication between them. They are represented in graphs in which each path to the next node is called a link. Routing, on the other hand, deals with strategy and decisions on how to effect communication between two or more nodes in a network. This set of decisions will depend on the needs of a network and may be affected by the environment in which they are located. An example of this is network topologies since it is the physical representation of the location of the nodes.

Considering a LLN, its topology will be constituted by a set of nodes connected by links (mesh topology). This makes the routing more complex since there are several alternatives to address a message. Besides, not all routes will have the same performance since relay

nodes and their links may be subject to change.

This is a challenge for production, as routing in the industry must ensure reliability as a low variability of transmission time. Also, the more nodes used, the more energy will be spent. This section will explain the basic features required by an industrial routing protocol based on mesh topologies.

Reliable routing

In an industrial network, transmission reliability must be guaranteed [OG20]. After all, production lines and industrial processes require great precision between their parts.

This is challenging for LLN since its performance is oriented to the best effort. As a result, IETF network working group established the following requirements [Pis+09] to ensure reliability.

- Maximum outage time: For industrial networks, routing must guarantee a link repair time that does not exceed a maximum outage time. As result, transmitted packet will not be affected by broken links or dead nodes.
- Route maintenance: To ensure network reliability, a routing protocol must keep nodes informed of their link statistics. In this way, the most appropriate route for the packet transmission can be decided.
- Bounded latency: Availability in industrial applications is subject to the transmission of packets within a restricted latency. Therefore, it is important to ensure the maximum delivery time.
- Path diversity: Because LLNs tend to have low link stability, it is required to provide diversity of paths in order to give a node more alternatives to transmit its message.
- LLN Border Router Redundancy: Since the information is processed in the border router and redirected to the plant applications, these access points are required to have redundant routes to them.

In summary, to provide reliability to a network in terms of routing, the algorithm must provide diversification and redundancy when it computes a route. In addition, it must have fast-acting detection and repair systems to prevent traffic obstruction.

Metrics

Within wireless networks, packet routing has to satisfy the corresponding requirements of a network. To do so, it uses network statistics defined as metrics to compute the most

convenient route for transmission.

From this, the authors in [YWK05] classify the metrics into two groups, load-sensitive, and topology-dependent metrics. Metrics within the load-sensitive group consist of the statistics used to attribute a weight to a route based on traffic load. The topology-dependent metrics group, on the other hand, includes the metrics used to assign a weight to a route according to the logical properties of the topology. Also, they identify four conditions for developing a metric, which are:

- Maintain route stability: Routing depends on the stability provided by its links. If a link changes regularly, either because of the quality of the link or because of the instability of its participants, then this results in a higher probability of packet losses. Therefore, it is important that the metrics used for each routing protocol satisfy the requirements of the protocol given that each routing protocol has a different tolerance for route instability.
- Good performance for minimum weight paths: A metric should allow the routing protocol to generate routes using device resources efficiently. Targeting high performance with low packet delay, the authors establish four possible requirements. The first one consists in select the path with the shortest distance. The second one, choose the roads that have the highest traffic density capacity. The third one selects the roads with the least losses statistically. Finally, the fourth one chooses the nodes with less interference as shared bandwidth between them.
- Ensure algorithms to calculate minimum weight paths: A metric must ensure that the weight order of two routes is preserved if a third common route is added (isotonicity).
- Ensuring loop-free routing: A routing protocol must maintain a free loop route since it is a required feature to choose the best path. Metrics with isotonic properties help to maintain that route structure.

As long as a metric does not meet these conditions, the algorithm may not be able to find the optimal path.

Lastly, they compared four metrics (hop count, ETT, MIC, and WCETT) with the defined conditions, concluding that a metric has to satisfy them to achieve an optimal performance.

In [Vu +19], the authors studied several metrics in MANETs networks focusing on overall performance, quality of service, energy, and security. Using this approach, they categorized the metrics into four groups.

- Traffic-based: It corresponds to the metrics used by the MANETs that require QoS in traffic. Within this group are the metrics based on the quality of the link, latency, throughput, etc.
- Radio information: This group includes all the metrics that represent the physical information of a radio frequency. In this group, we have a Signal strength metric, Signal-to-Noise metric, and Bit Error Rate metric.
- Location and mobility: MANETs are networks that are characterized by having mobile nodes. As well, routing based on geolocation requires metrics that deliver the position of their devices. In this group, we find metrics such as location or mobility.
- Energy: Since some MANETs use batteries due to their autonomy and mobility, it is important to control the energy expenditure of the devices they have in charge. Within this group are included the metrics that provide information to minimize the total energy consumption of the network and also for optimizing the lifetime of the nodes.

As a result of the analysis, they provide a list of various routing metrics that should be focused on. Further, they discuss that while routing metrics are designed to satisfy MANET applications, there are still cases that have not been studied.

Based on the arguments studied, routing metrics must be selected according to the needs of the application being pursued. This to choose the most suitable route for a specific problem.

2.2.2 Discussion of an existing debate

In multi-hop networks, there are several ways to generate a route that allows communication from a source to a target node, and depending on their behavior and structure, they can be classified as either reactive or proactive. A proactive protocol [SK14] is defined as a policy that maintains a routing table based on periodic information provided by the nodes. This information covers the constant updates that a network experiences throughout its operational lifetime. A reactive protocol [Pat+14], on the other hand, establishes an on-demand route between the source and target nodes. These routes have an expiration time since they may no longer be used or due to the existence of broken links.

Although these types of routing protocols follow different philosophies, it is known that both reactive and proactive protocols are equally valid for solving a routing problem. This will only depend on the features and requirements of a given network. For example

in [Fot+07], the authors analyze the energy performance between DSR (reactive candidate) and OLSR (proactive candidate) showing that reactive networks take advantage of routing policy when the traffic load is low. While in networks with high traffic rates, proactive networks tend to perform better.

In [Moh+10], the authors compare various reactive and proactive protocols focusing on their characteristics and differences, concluding that there is no clear algorithm for all types of scenarios. Also, they mention some methods that could reduce the disadvantages of these protocols, such as the use of conditional rather than periodic updates to decrease overhead in proactive routing or monitoring Route discovery and route maintenance to improve the reactive routing scalability.

Finally, in [TDV14] the authors evaluate the performance of the proactive protocol RPL [Win+12] and its reactive counterpart the Lightweight On-Demand Ad hoc Distance Vector Routing Protocol-Next Generation (LOADng) [CYH17]. They conclude that the LOADng complexity compared to RPL is lower and additionally, LOADng prosper in networks that are not heavily loaded with traffic. However, its performance declines compared to RPL in larger-scale networks.

2.2.3 A different approach

In addition to MANETs, routing protocols are also used in other areas and can have different behaviors and characteristics. An example of these are the Visible-light Ad-hoc networks (LANETs) [Cen+19] which aim to take advantage of the characteristics of optical communications belonging to optical wireless communications. These networks differ from radio-frequency networks because they do not have electromagnetic interference and because they have a fast transmission speed. On the other hand, it has a short range of coverage due to its short-wavelength; the devices it covers must be aligned in such a way that they can communicate; they can be affected and interfered with by other types of lights such as sunlight or other LEDs and lastly because they do not have a well-established channel model.

Anothe example is in [JM19], where authors propose a solution to improve the reliability of communication between devices in LANET networks. For this, they define a metric that they call RRS. The RRS is based on the estimation of the hop reliability towards a sink using the neighbors of the node that carries the packet. In conjunction with the use of an opportunistic MAC protocol, they demonstrate improved reliability performance over a geographic and a greedy routing protocol.

The Vehicular Ad-hoc Network (VANET) [DB17] is another example. These networks share some of the characteristics of MANETs such as node mobility, limited bandwidth, and so on. However, the energy limitation is not a problem because the vehicles are equipped with an alternator. Therefore, the routing protocols are mainly focused on ensuring the reliability of communication between the nodes.

Finally, in [GS19], the authors propose the combination of the Ad-hoc On Demand Distance Vector (AODV) routing protocol with the ant colony optimization policy. Additionally, they used the ant colony optimization policies in conjunction with Particle Swarm Optimization to provide optimal TDMA MAC scheduling. As a result, the authors demonstrate an improvement in Packet Delivery Ratio (PDR), latency, and throughput.

In conclusion, the routing protocols are adaptable and can be used in a variety of applications and environments. Therefore, as there are many ways to address a problem, one solution can also be used in different applications.

2.3 RPL

RPL [Win+12][GK12] is a multi-hop distance vector protocol used for LLNs. The goal of this protocol is to point multiple Directed Acyclic Graphs (DAGs) to one or more sinks, thus generating one or more Destination Oriented Directed Acyclic Graph (DODAG)(s) The selection of the receivers who carry out this protocol is governed by an Objective Function (OF), whose objective is to find the most suitable node according to the requirements of the network.

This protocol has been designed to integrate LLNs into industry 4.0 and is currently recognized by IETF as the industry standard for smart grid routing. For this reason, we have determined to use RPL for our research to improve its performance in terms of reliability, while maintaining a low delay variability. This section will present the functionality and features that make up this protocol.

2.3.1 Instance

In RPL, the instances are in charge of generating and maintaining the DODAGs. These can have one or more sinks defined as *DODAG Roots*, and each *DODAG Root* has its own DODAG. Additionally, the instances are defined with an RPLInstanceID and can encompass the following conditions.

-
- Single Instance, Single *DODAG Root*, e.g., a DODAG based on star topology.
 - Single Instance, Multiple *DODAG Roots*, e.g., smart grids which are separated from each other.
 - Single Instance, Multiple virtual *DODAG Roots*, e.g., networks that lead to multiple border routers to improve the reliability of the link between the DODAG and the *DODAG Root*.

2.3.2 Objective Function (OF)

The OF is the entity responsible for selecting the most appropriate route for transmission within a DODAG. To achieve this, it receives different metrics from the set of neighbors of a node. It then computes them to rate each neighbor with a score called RANK. Finally, the nodes classified with RANKs lower than the host node are selected as possible receivers.

The metrics used to decide the routes selected by the OF [Vas+12] can be grouped into two groups, node metrics, and link metrics. From the node metrics, some of them are list as follows.

- Node residual energy: This metric consists of providing information about the energy state of a node. This can be used, e.g., to avoid traffic to routers with low residual energy. The problem that can be found when using this metric is that the route to be used can be considerably longer, so constraint-based routing is also required. The simple way that RPL uses to span multiple applications is to encode three types of power sources: powered, battery, and scavenger.
- Hop Counting: This metric is used to report the distance between a node and its sink by counting the total number of traversed nodes between them.

While some of the link metrics are show as follows.

- Link Quality Level: It is a metric that provides a grade between 0 to 7 for link reliability, with grade 0 being corresponding to the unknown state and 1 to the most reliable link state.
- Expected Transmission Count (ETX): This metric measures a route's quality between two nodes to select the most reliable path that a packet can take. Since its value is calculated, this metric requires compliance with constraints to avoid routes with high transmission costs.

For every instance, there is one OF. Furthermore, OF must be defined so that RPL can meet the requirements of the specific needs of the application domain. Some of the

existing proposed objective functions are shown below.

1. The Objective Function Zero (OF0): This OF [Thu12] aims to provide the shortest route by using the hop counter metric. To accelerate link repairs, this OF selects an alternative route as a backup in case the main route fails.
2. The Minimum Rank with Hysteresis Objective Function (MRHOF): MRHOF [GL12] aims to select the most reliable route with the shortest distance. To achieve this, this OF selects the receiver node with the best reliability statistics (ETX) and then compares its distance with the other available neighbors. If a different candidate is found to be closer to the sink, then the receiver is replaced by the new candidate. In addition, MRHOF relies on the hysteresis principle to avoid too frequent route changes.

2.3.3 Control Packets

To keep the nodes up-to-date with the necessary information to be able to join, stay and be able to repair their routes when necessary within a DODAG, this protocol uses three types of control packets.

DODAG Information Object (DIO)

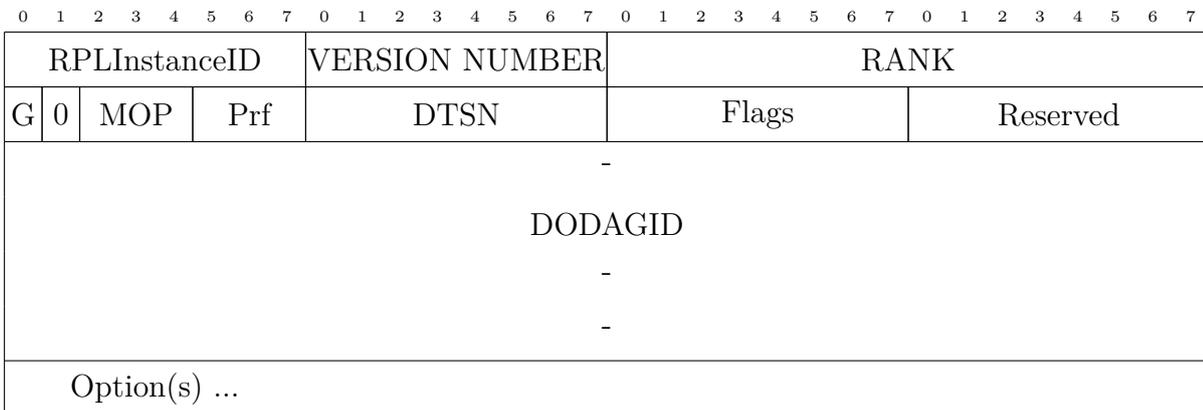


Figure 2.2 – DODAG Information Option (DIO) Base Object.

This control packet contains all the necessary information regarding an RPL instance, allowing a receiver node to connect into a DODAG. Given its nature, it is usually trans-

mitted as multi-cast and its intended to provide current DODAG information. The fields belonging to this packet are shown in Figure 2.2 and described in [Win+12].

Destination Advertisement Object (DAO)

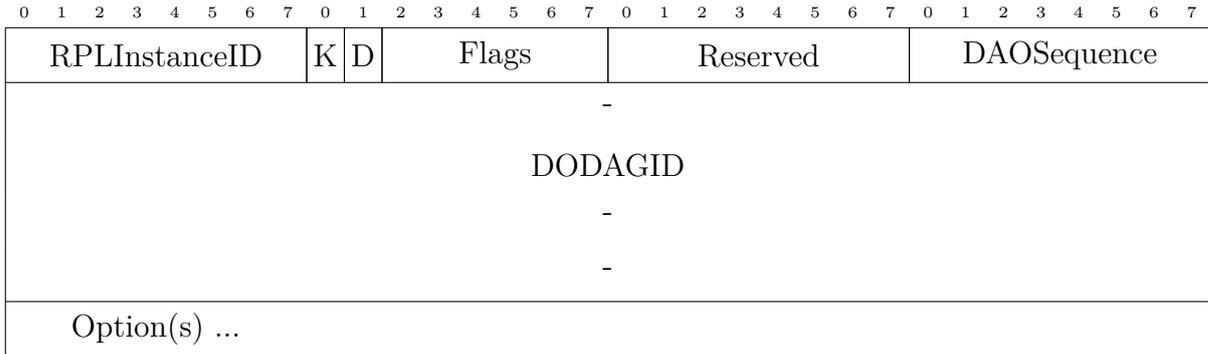


Figure 2.3 – Destination Advertisement Option (DAO) Base Object.

This control packet propagates reverse routing information every time a node joins a DODAG. At the time this packet is generated, it must decide where it should be transmitted. In case the routing depends on the nodes (storing mode), the packet will be sent to the receiver parent that best suits the node. In case the *DODAG Root* is the routing encoder (non-storing mode), the packet is sent to it. The fields belonging to this packet are shown in Figure 2.3 and described in [Win+12].

DODAG Informational Solicitation (DIS)

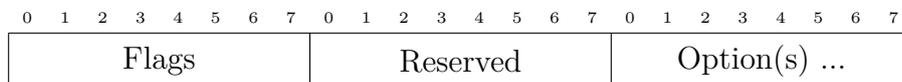


Figure 2.4 – DODAG Informational Solicitation (DIS) Base Object.

The goal of this control packet is to obtain the network status of a node’s neighborhood by broadcasting this packet. In this way, the receiver nodes will send a unicast DIO in response. The fields belonging to this packet are shown in Figure 2.4 and described in [Win+12].

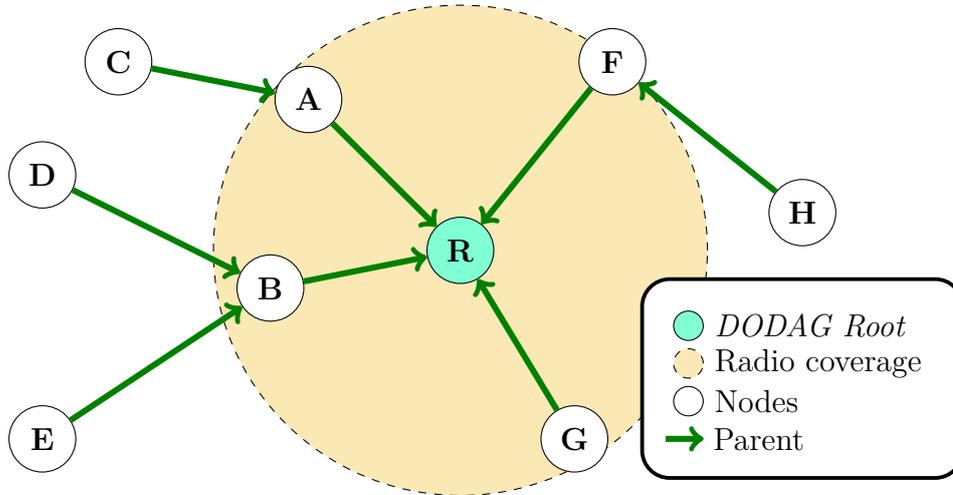


Figure 2.5 – Destination Oriented Directed Acyclic Graph (DODAG) example.

2.3.4 DODAG Formation

In RPL, a DODAG belongs to a *DODAG Root*, while an RPL instance can hold several *DODAG Roots*. When an RPL instance is defined, its *DODAG Roots* will set the information of that instance and their parameters into a DODAG Information Object (DIO) control packet, e.g., RPLInstanceID, RANK, DODAG ID and so on. The purpose of this is to inform the surrounding nodes about the existence of this RPL Instance.

As soon as a DIO control packet is formed, it is sent to the nodes within the coverage radius of the sink (broadcast transmission). This can be seen in Figure 2.5, where the *DODAG Root* R sends this information to nodes A, B, F, and G.

When the neighboring nodes receive this DIO packet, they will decide if it is convenient for them to join that DODAG. If they do not belong to any DODAG, they will join the first one they find, otherwise, they will join the one that best fits the OF, only if these DODAGs belong to the same RPL Instance. For this, the node will compare its RANK with the RANK delivered by the DIO packet. In case a node receives a DIO without belonging to a DODAG, the node will be incorporated into it by establishing a link to the packet transmitter. Otherwise, it will only take into account the DIO messages of nodes with a lower RANK than its own.

These actions will be performed by all the nodes in order to build a DODAG that meets the requirements of the instance.

2.3.5 Trickle Timer

Because of the characteristics that make up LLNs, it is essential to avoid traffic congestion. This is a difficult task in terms of control packets since they are transmitted periodically given their functionality. Trickle timer [Lev+11] reduces congestion by regulating the frequency of the packets as they normalize over time. In the event of an anomaly, the trickle timer will reset its timer by restarting the cycle. In RPL, this functionality is used on DIO packets since their data becomes redundant as DODAG stabilizes. If there are any inconsistencies between the nodes or they simply disagree, then the transmitting frequency is reset. Otherwise, the packet will be transmitted in exponential times.

2.4 Multi-path routing

In a wireless environment, one-way transmissions in the routing protocols can be affected for different reasons such as link quality, transmission distance, desynchronization between nodes, etc. To mitigate these disadvantages, several solutions have been developed or proposed over time. In particular with multi-path routing, the route can be modeled in different ways resulting in different transmission strategies [HF08] [Arm+16].

RPL is a protocol focused on LLN routing. However, it has several complications in satisfying the requirements of the industry. Therefore it is essential to enable multi-path over this protocol. In this section we will list different proposals based on multi-path published over time, explaining their operation, advantages, and disadvantages. Additionally, Table 2.1 will provide details of each of the proposals.

Finally, the following terms will be used to understand multi-path over RPL.

- Preferred Parent (PP): Default parent of the node. Like a single-path, this parent corresponds to the most suitable for a reception.
- Alternative Parent (AP): Corresponds to the parent chosen through the multi-path selection. This parent may not necessarily exist and can be used either as a backup or as an active parent.
- Neighbors: Corresponds to the nodes located within the transmission radius of the selected node.
- Parent Set (PS): List of neighbor nodes capable of being parents. These nodes satisfy the requirements needed to receive or relay a transmission.
- Node State and Attribute (NSA): List of IP addresses transported within a DIO's

Metric Container (MC). Here a limited PS of a node is transported, sorted in decreasing order by RANK. As a result, the node's PP and AP are transported. The purpose of this object is to inform the node's neighborhood about its PP and AP.

In [Hwa+20], multi-path is used to provide routes to different receivers to send broadcast packets over a DODAG. For this, the authors adopt Steiner tree optimization to maximize transmission paths, being the source node the root of the tree. This is achieved by providing routing information to the Destination Advertisement Object (DAO) control packets. At the same time, RPL must operate in storing mode so that the nodes can generate their routing tables. What is remarkable about this proposal is that is an efficient method to provide the information to different stations, however, its reliability is subject to transmission retries since its routes behave as single-path.

In [Lod+15], the authors propose to use multi-path to reduce the congestion load of a link. However, this will only be done if the link exceeds a given congestion threshold. To achieve this, the authors established two steps. The first, called congestion detection, consists of traffic detection by the PP. Here this parent counts the number of packets successfully received within a period of time to compare them with an expected number of satisfactory packets. If the number of packets received is less than the allowed threshold, the PP node notifies its children. The second stage consists of congestion mitigation. Its purpose is to authorize the child nodes to select an AP to distribute their traffic. What is remarkable about this methodology is the improved reliability in terms of transmission since fewer packets are dropped due to congestion. However, there is no opportunistic criterion for selecting the auxiliary parent.

The authors in [DLV13] propose an opportunistic upward and downward routing through anycast propagation. To achieve this, they use RPL in storing mode to ensure that each node has its neighborhood routing table. In terms of upstream transmission, the direction is determined by the RPL rank of each neighbor, directing its transmission to the PP. Meanwhile, the downstream transmission is oriented by the routing tables. This implementation improves reliability and allows dual communication. However, it encourages loops since the nodes can be affected by desynchronization.

In [ITN15], the authors use multi-path to argue that load balancing should be used in a manner that improves the life expectancy of a network. To this end, they propose the "Expected Lifetime" metric, which delivers the residual energy of a node. The objective of this metric is to provide additional information to the nodes to select an AP without this

being the second-best option of the default metric. In this way, both the PP and the AP can equally distribute the load of both transmission and power consumption. It should be noted that the parents selected are not permanent so multiple corrections may occur. The advantage of this implementation is its equity in terms of energy expenditure. As for disadvantages, the combination of nodes with bad links but high energy residue affects negatively the transmission performance.

In [Kou+18], the routing of multiple redundant paths is proposed to provide high reliability of transmission ensuring low variability of reception delays. This is achieved through two principles. The first consists of sending a copy of the packet to the AP while the original copy is sent to the PP, being mapped by a TSCH schedule. The second principle consists in selecting an AP if that node has as its ancestor the PP's PP. Additionally, the authors postulate the use of overhearing to provide the parents of an extra attempt to receive a data packet. The advantages of this proposal are the improvement of network reliability while keeping a low jitter. Among the disadvantages, there is an increase in energy expenditure compared to the single-path. There may also be a failure to have a AP due to node desynchronization or flooding if there are two or more possible parents on the parent list.

The authors in [LNM14], propose three types of algorithms to select an AP: Energy-awareness Load Balancing, Rapid Local Repair, and the combination of both. Starting with load balancing, the authors propose a protocol that has the function of selecting an AP based on two types of metrics. The hop counting metric in combination with the residual energy metric. As a result, the PP and AP will tend to be the closest parents with the highest residual energy. Also, the AP is reserved as a backup parent in case the PP fails. Rapid Local Repair, on the other hand, has as mission to reduce the number of local repairs. This is achieved by taking into consideration neighbors with the same rank as him (sibling nodes). Finally, the combination of the two algorithms results in reduced overheads and delays while maintaining a well-balanced network. However, like [ITN15], poor link quality impairs reliability. Besides, transmission loops can be caused since quick repairs depend on the consideration of sibling nodes as parents.

In [MT14], the authors propose to produce multiple ascending braided routes to give a different alternative to a node to select an AP. Then, they use a load distribution mechanism that obtains the traffic rate of each link. The reason for this is to equalize the load of the nodes that are at the same depth. However, although they demonstrate an improvement in throughput, the lifetime of the devices is not determined. All this, taking

into consideration that a parent can have many children. Consequently, the residual energy of a node can fall quickly.

In [MNT19], the authors propose the use of multiple routes by selecting an AP using the braided pattern. This proposal aims to use the AP if the PP cannot be reached. Furthermore, they propose the use of a centralized TSCH scheduler that is constructed in such a way that the nodes can perform an end-to-end transmission from the source to the sink in a single slotframe. To take advantage of the isolation that exists between the PP and AP nodes, the authors propose to assign the RX of the nodes of the same layer in a shared cell and the TX of the nodes that will transmit to them in dedicated cells. In this way, if there is a common receiver, the transmission from a PP or an AP will use the same timeslot. What is not taken into account is the existence of desynchronization in the reception of the packet. If we consider the transmission of a received data packet to be successful, an ACK is expected to verify its delivery. Since these packets can get lost, this can result in the AP being used for transmitting as well, thus generating collisions.

In [Tan+16], the authors analyze the difficulties associated with LLNs in emergency applications in terms of speed and reliability. To solve this difficulty, the authors propose a protocol designed for both traffic load balancing and transmission speed. To achieve this, they propose the use of three metrics. The first one consists in informing the proportion of received packets against the total number of expected packets in a defined time range. The second, ETX, used to report the number of re-transmissions required per link. And finally, DELAY_ROOT, which delivers the duty cycle of the nodes found in the neighborhood. All this to replace the PP by the PS with closest radio-link available. However, one of the disadvantages found is potential collisions. The reason for that is because several nodes may have several parents in common.

The authors in [ZWY17] propose a new extension of RPL to maximize the life of the network. To do so, they implemented an algorithm that allows adjusting the load by multiple parents taking into account the network requirements. This is achieved by combining the Newton method with the steepest descent method. As a result, the PS with the most weight to transmit is obtained. Regarding the improvement of the end-to-end delay, the algorithm adopts a greedy data forwarding model to compute the parent listening time. However, the nodes around the sink will not be able to generate load balancing, which will consequently lead to network failure.

	Mission	Algorithm functionality	Rel ¹	T.L ²	Fdg ³	SRx ⁴	Red ⁵
[Hwa+20]	Enable broadcast communication to multiple targets.	A Steiner tree is adopted from the source node to generate multi-paths towards every node within a DODAG.	✓		✓		
[Lod+15]	Allows a node to distribute its traffic load through an AP if its PP exceeds its reception threshold.	When the PP exceeds its threshold load, it alerts its children so that they can select an AP for traffic distribution.		✓	✓		
[DLV13]	Allow upstream and downstream anycast transmission within a DODAG.	Upward transmission is performed to the nodes with RANK lower than the transmitter RANK. Downward transmission is performed following the routing tables provided by the nodes (Storing mode).	✓		✓		
[ITN15]	Extend the longevity of a DODAG through traffic distribution using multi-path.	To obtain a PP and an AP with similar loads, the "Expected Lifetime" metric is computed in conjunction with the ETX metric to obtain the parents with the highest residual energy from reliable routes.		✓	✓		
[Kou+18]	The routing of multiple redundant paths is proposed to provide high reliability of transmission ensuring low variability of reception delays.	A node selects a node as AP if it has in its ancestor the PP's PP. This is followed by the transmission of a packet with its copy to the selected parents at scheduled times, while taking advantage of overhearing.	✓		✓	✓	✓

-
1. Ensures reliability.
 2. Traffic load control
 3. Generates network flooding.
 4. Ensures scheduled reception.
 5. Route redundancy.

[LNM14]	Maintain a network where its nodes can balance their traffic load through the use of multi-path as well as ensure fast response for repairs.	Two algorithms are proposed. The first consists in the selection of an AP to distribute its traffic load. The selected node will be the one with the highest residual energy at the shortest distance from the sink. The second algorithm allows us to choose as parents the nodes with equal RANKs to the current node. Finally, the two algorithms work together to obtain a third one.		✓	✓		
[MT14]	Level the load of the nodes that are at the same depth.	Multiple braided upward routes are produced to give different alternatives to a node for selecting an AP. The node that has similar conditions to the PP is selected as the AP according to a load distribution mechanism.		✓	✓		✓
[MNT19]	Use an AP as an auxiliary parent to replace the PP if it fails. All this without delaying the data packet for possible parent swaps.	An AP is provided if it has within its parent list the PP of the PP. Then, using a centralized TSCH schedule, Tx instances are placed in dedicated cells, while Rx instances are placed in shared cells.	✓			✓	✓
[Tan+16]	The aim is to achieve the transmission of a message reliably in the shortest possible time.	Through the ETX metric, the parent traffic load volume handled by the nodes, and the duty cycle report of each neighbor, this feature obtains the set of adequate nodes with the closest availability to be used for transmission.	✓	✓	✓	✓	
[ZWY17]	Extending the life of the network through a new RPL extension.	The Newton method is combined with the steepest descend method to select the most suitable set of parents to perform the load balancing.	✓	✓	✓		

Table 2.1 – Comparative table of multi-path RPL algorithms.

2.5 Conclusion

This chapter examines both Ad-Hoc networks and the explanation of the need for routing systems in these networks. The concepts and characteristics of these routing systems to ensure transmission reliability are also discussed. From here we can note that routing systems are subject to several factors, on which they depend for their improvement.

Focusing on the industry, these require that the network ensure transmission reliability as well as speed and stability. Additionally, if these devices have batteries, maintaining a minimum consumption of them is essential. Knowing that these devices belong to LLNs, it is fundamental to develop an appropriate routing system that satisfies these requirements.

This is why we use RPL since it provides tools that allow us to obtain the route that best satisfies the requirements of a system while offering a simple implementation and previous knowledge of how its selection will act. However, as we have learned throughout this chapter, its selection by single-path does not satisfy part of the industry requirements.

Given this, we have opted for the multi-path extension to RPL, which has proven to be an adaptable function for many use cases, but at the same time depends on a larger number of participants in a network for its use. In addition to requiring compensation from other resources to improve other areas.

However, in terms of transmission quality, we have identified that redundant selection systems allow for a better distribution of opportunities to deliver a packet, as well as not relying on just one link.

In terms of life expectancy, we have identified that load balancing extends the life of a network by distributing the load equally to the nodes with the highest residual energy.

In the following chapter, we will address different redundant routing patterns that will be used throughout this research. Also, a series of functions will be presented which allow the use of multi-path in RPL.

PATTERNS AND REPLICATION FUNCTIONS

3.1 Introduction

As we mentioned in the previous chapter, one of the crucial challenges in industrial sector is to ensure reliable communication among the devices in LLNs. However, given the limitations that these networks have, it has been difficult to incorporate them into the industry. To meet the demands of the industry, a QoS for their communications must be maintained. This means that for every packet transmitted, the network must ensure transmission success while providing a rapid response to changes and keeping its delay performance at low levels.

Given the above, several routing protocols and algorithms have been designed to meet these demands. However, with each improvement achieved in some of its parts, others will be negatively affected, since a trade-off in resources is required. An example of this can be seen when another parent is used for load balancing since the new parent will have to spend some of its residual energy to decrease the traffic of the original parent.

One way to optimize this trade-off is to establish a path selection strategy. More precisely, on how the routing protocol will behave for the selection of its routes. Considering the RPL protocol, we know that this protocol consists of elaborating its routes by a single-path policy. Moreover, the decisions regarding which route to take depends on the employed objective function. The most common strategy is the one of following the most reliable path using the ETX metric or the one of following the shortest path using the hop counting metric. (see Section 2.3.2) However, both strategies do not offer adequate transmission reliability when faced with distances with a large number of hops and/or when passing through low-quality links. Because of this, a popular strategy is the implementation of a limited number of transmission retries, allowing a packet to have more opportunities to a successful transmission. Consequently, delay and jitter metrics

are affected since it will take longer time for a packet to reach its destination.

Another strategy is to extend RPL with multi-path to send multiple copies of a single packet throughout the network and, thus, to enhance the reliability of the transmission. However, this methodology implies the generation of network flooding and an increase in the traffic overhead. Given this, we propose to use controlled paths to minimize the consequences of these flooding. More precisely, link paths that merge into a common node, i.e., redundant path patterns to take advantage of the wireless properties and characteristics of the topology. The goal of these redundant patterns is to focus several forwarding attempts on certain nodes, increasing their chances of receiving the packet.

In this chapter, we will focus on the study of some of these patterns, analyzing their advantages, complexities, and disadvantages. We will also introduce and explain the PAREO functions, which allow the integration of a reliable and optimized multi-path transmission in RPL.

3.2 Redudant patterns

In multi-hop networks, each transmitted packet will be subject to the transition probability of each relay link. In addition, if a relay node dies or becomes unsynchronized, all links that depend on it will either disappear or be subject to a local RPL repair period.

This behavior directly affects single-path RPL since the transmission is forced to take place over a single route. Given this, single-path networking must be supported with additional functions to reinforce the transmission success.

Considering Equation 3.1, in which q represents the link quality (i.e., the probability that a single transmission will be successful), and r is the maximum number of transmission attempts, it is observed that retransmissions improve the reliability of the link since a packet could be forwarded with 0 to $r - 1$ failed transmissions followed by a single successful transmission (see Figure 3.1). However, this action directly affects the network latency and the lifetime of the devices [Usm18].

$$P_{\text{success}}(q, r) = \sum_{k=0}^{r-1} (1 - q)^k q \tag{3.1}$$

For every transmission attempt, there is additional time cost since each transmission requires extra timeslots to reach its destination. Furthermore, because the transmitter radio is used more often, the power consumption is higher. Thus, trade-offs in reliability

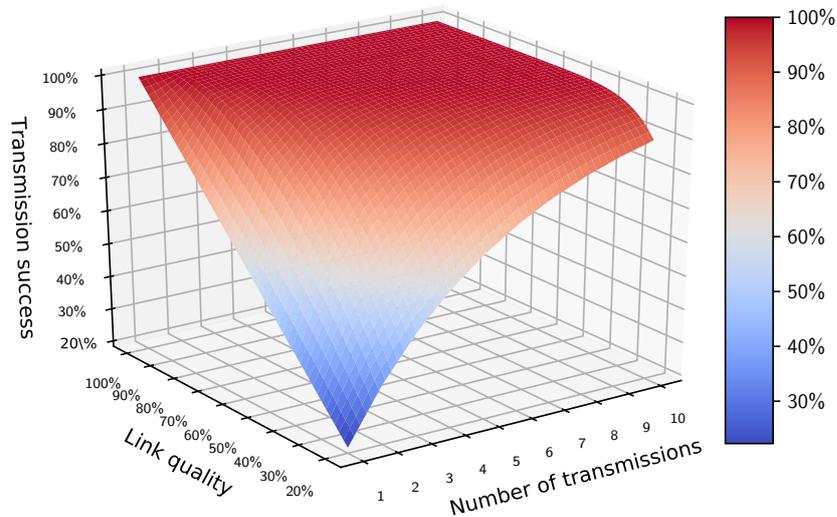


Figure 3.1 – Transmissions attempts required for a single hop delivery.

are determined by higher latency and energy.

In the case of the multi-path RPL approach, the transmission conditions change, since the transmission will not depend on one but several paths. Therefore, routing is exposed to other factors. For example, a multi-path routing can cover a larger extension of nodes than a single-path in a network, since multi-path can spread the routing over the topology. This factor guarantees reliability since more nodes forward the packet to the root, but at the same time leads to overuse of the nodes, since not all of them are required to transmit the packet.

One approach to mitigate this load flooding would be a transitional route redundancy to concentrate efforts towards a common direction. However, such redundancies would require selection strategies to realize them. An important point to keep in mind is that not all the neighbors of a node can be parents. This is because the nodes do not have a constant forward direction, since their redirection is based on the RANK values given by RPL (see Section 2.3.2). Therefore, these redundant paths must form following the taxonomy of the network.

In this section, we will present on three types of redundant patterns [MKL17], which can be adapted to TSCH. They are designed to provide link redundancy in order to improve reliability and to minimize the number of nodes used per transmission.

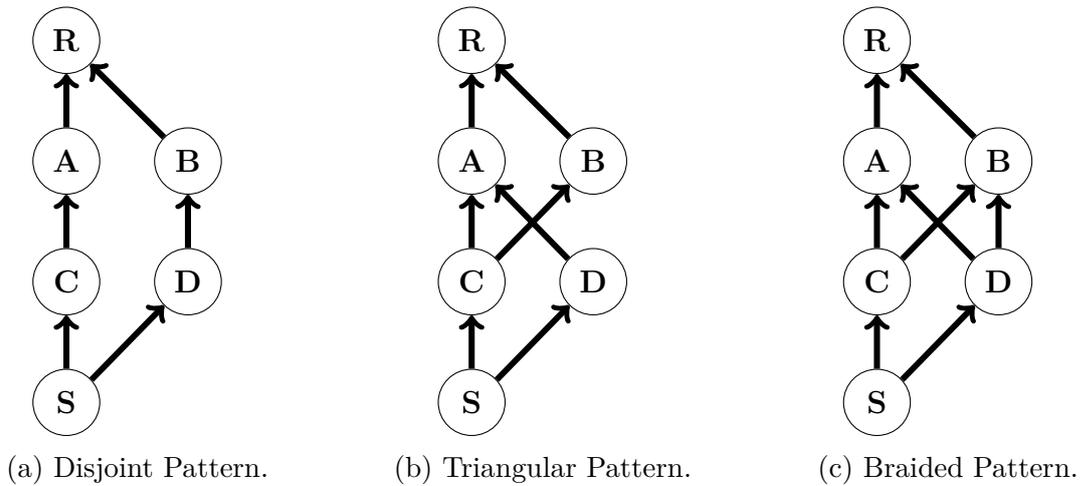


Figure 3.2 – Redundant patterns for TSCH networks.

3.2.1 Disjoint pattern

This pattern is characterized by two or more disjoint paths originating from the source node and merging with the destination node. It is represented in Figure 3.2a, and its purpose is to generate as many links to the sink node as possible to concentrate several forwarding transmissions to that node. Its complexity lies in the source since it is in charge of selecting the multiple parents to carry out the transmission. On contrary, forwarding nodes can only select one parent.

As an advantage, it is the pattern with the lowest energy consumption since each node, except the source, transmits in single-path. Regarding the disadvantage, its stability depends on the distance between the source and the sink since each path is subject to the state of each link that composes them. Finally, although it provides an improvement in reliability compared to single-path, this reliability will depend on the number of sink node children that can be provided a packet.

3.2.2 Triangle pattern

Unlike the disjoint pattern, this pattern prioritizes the differentiation of a PP from an AP. The selection consists of allowing a node to select an AP if this node has the same parent as the PP. It should be noted that an AP cannot select another AP. Figure 3.2b demonstrates this triangular chain of paths starting from the source and ending at the sink.

Due to the selection conditions of this pattern, the complexity lies in two factors.

- The source and the PPs can choose multiple parents.
- Each node must be able to recognize if it is PP or AP.

As an advantage, this pattern tends to have better reliability than the disjointed pattern since each common ancestor is provided with two attempts. Its disadvantage, results in higher power consumption as more links are used. Finally, due to its selection methodology, this pattern will not generate flooding since its alternative paths will depend only on the main PP route.

3.2.3 Braided pattern

This pattern uses the same methodology for path selection as the triangular pattern, with an extension that APs can also have its own APs. As a result, it provides better reliability than the triangular pattern.

An advantage that can be observed is an increase of common ancestors per hop. As a result, there is a greater chance of having an AP considering that due to irregularities or desynchronization between the nodes, there is the risk of an in-existence common parent. Additionally, if we count the number of transmission opportunities per hop, these are at least double compared to the triangular pattern. On the other hand, given its selection methodology, this pattern uses more energy than the previous patterns.

3.2.4 Patterns reliability evaluation

In order to evaluate the performance of each redundant pattern in terms of transmission reliability, we will rely on the results obtained by the authors in [MKL17]. Here they evaluated 5 different environments, however, we will focus on the first two, which are

- All links, for both PP and AP, have a 90% link quality.
- All links directing to a PP have a link quality of 90%, while all links directing to an AP have a link quality of 70%.

From these evaluations, the authors in [MKL17] show that the braided pattern has the best performance in terms of transmission reliability, with a PDR of 98.66% in the first case study, and 92.89% in the second case study. In the case of the triangular pattern, they obtained a PDR of 97.1% in the first case study, and 89.17% in the second case study. Finally, the disjoint pattern has the worst performance in terms of reliability with a PDR of 88.17% in the first case study, and 73.87% in the second case study.

3.3 PAREO functions

Considering the different transmission media of a network, a radio interface supports short, medium, or long-range communications with relatively good reliability. However, not as good as wired solutions can provide. In addition, since LLN devices are restricted to limited resources, these devices may be temporarily inaccessible.

Despite this, LLN play an important role, as they allow process automation in industrial environments at low cost. However, given the characteristics of these technologies, the main problem to be addressed is reliability and predictability, as these are dispensable requirements in the industry.

One way to address these requirements is by using multi-path routing across the network. More specifically, to transmit multiple packets with the same information over different paths to increase the chances of being received. However, RPL as a routing protocol does not have these features by default.

Given the above, this protocol requires different additional functions to perform multi-path. Moreover, given the characteristics of LLN, these functions must optimize the consumption of resources. In this chapter, we will focus on this requirement by introducing and analyzing the PAREO functions.

3.3.1 Automatic Repeat reQuest function

The ARQ function performs the re-transmission of data packets when a previous transmission failed [FW02]. In our context, we employ a link layer ARQ so the decision for re-transmission is local to the transmitting node. This function requires the use of an ACK control packet for each data transmission. TSCH allows cells to be configured to require an ACK, which is transmitted from the receiver to the sender within the same cell after the data transmission.

Since in our context only one packet will be in transit at any one point in time in a given cell, we can use the Stop-and-Wait variation of ARQ, as described below. The function starts by sending a packet and setting a short timeout to await an ACK (within the same cell). If the ACK is received from the receiver, the transmission is marked as successful. If no ACK is received, then the transmission is marked as temporarily failed and is re-scheduled again in the future. For the same data packet, each time a Retransmission (RTX) fails, a counter is incremented. If the counter reaches a threshold, named the RTX count, then the packet is marked as permanently failed and it is not scheduled again.

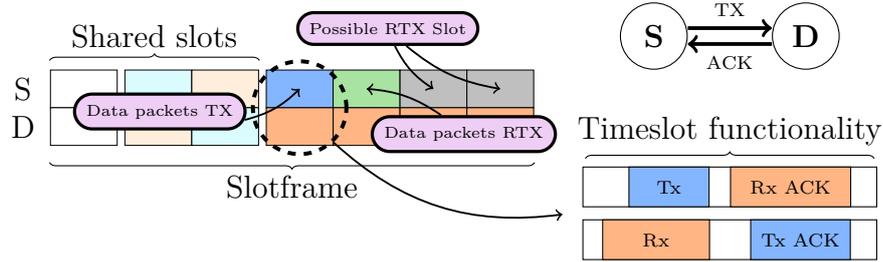


Figure 3.3 – Automatic Repeat reQuest (ARQ) example with 1 transmission and 3 RTX.

Figure 3.3 shows an example of this where node S attempts a transmission to node D. Here ARQ is configured with a threshold of three RTXs being carried out in a single slotframe. In the blue cell, the transmission is executed by default. If it fails, the green cell will proceed to perform the first RTX. The grey cells will perform the rest of the RTXs if the packet fails to report its reception until the established threshold is reached.

The scheduling of the re-transmission is tightly coupled with the structure of the TSCH schedule. Depending on the schedule, the timing of re-transmission attempts can vary significantly: from immediately after a temporary failure to later during the next repetition of the slotframe. While it is possible to include almost any number of re-transmission attempts for each pair of nodes in one slotframe, the more reserved cells for potential re-transmissions, the largest the slotframe size.

3.3.2 Replication and Elimination function

The RE function modifies packet forwarding to send a packet not only to the PP of a node but also to other nodes in the PS [Pap+19]. In the default implementations of RPL, each node only uses one node to forward packets (i.e., the PP). The problem with that approach is that if the PP fails for any reason, all packets from its children will be discarded until a new parent is selected. However, selecting a new parent requires one of two time-consuming processes, either local or global repair, during this period connectivity is not available. If the queue fills up, and the connectivity is not restored, the data packets will start being dropped. RE does not avoid the local or the global repair to select a new PP. However, having established two or more parents, the connectivity is maintained and, thus, packet drops are avoided due to an inaccessible parent node.

In order to make this possible, a packet is cloned and each copy is forwarded over a different parent node, called an AP, in the PS. Performing packet replication operation creates multiple copies of the same packet, which traverse independently through the

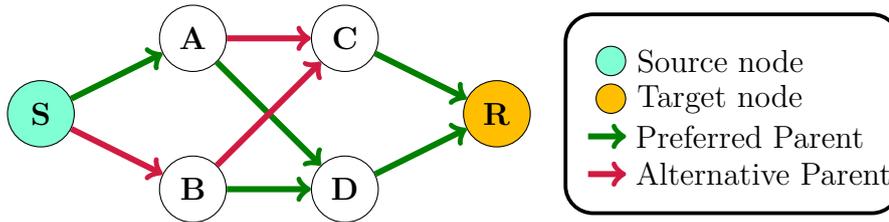


Figure 3.4 – Multi-path transmission by Replication and Elimination (RE).

network. In order to avoid a situation where each received copy results in a further independent set of packet copies being forwarded, resulting in a flooding, a packet elimination function is implemented.

To perform the above, a unique ID is assigned for each new packet. Thus, every time a node receives a packet, it comes with the node’s ID. As a result, if another packet with the same ID is received, the node will drop it, since it has already received an earlier copy.

Figure 3.4 shows the procedure, with node S representing the source and node R the target. Each transmission will depend on three hops, where the green links are the ones that point to the PP and the red ones to the AP. When S starts transmitting, the packet will be sent to the selected PP and AP. These nodes then perform the same action and so on, until the packet reaches node R.

Packet elimination uses a unique 32-bit packet ID added in the header of each packet sent (stored in the IPv6 RPL Option extension header in our implementation) to identify packet copies.

The unique ID of the last k packets forwarded are recorded in a Least Recently Used (LRU) or Last In, First Out (LIFO) unique-id-history cache in each node, so when a new packet arrives for forwarding, the presence of its unique ID is checked in the unique-id-history cache. If it is found, the packet is dropped without forwarding since this means that a copy has been previously forwarded. Otherwise, the unique ID is added in the cache, potentially evicting another entry. The size k of the unique-id-history cache can be set based on the maximum data traffic possible in the maximum end-to-end delay.

The interaction of RE with the TSCH schedule only concerns the required existence of cell(s) for transmitting packets from a node not only to its PP but also to the AP.

3.3.3 Overhearing

The OH function [LH09] can be used to take advantage of the shared nature of the wireless medium in order to increase reliability. This function allows a node to simulta-

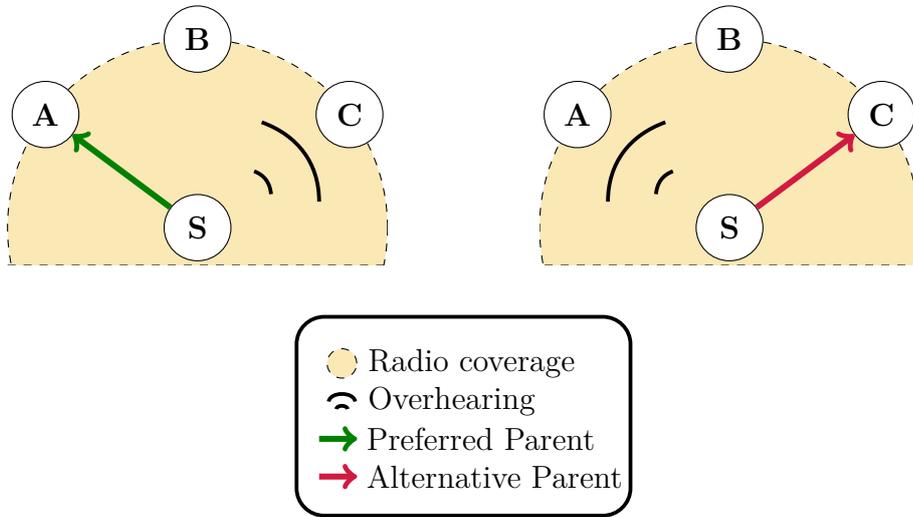


Figure 3.5 – Overhearing (OH) functionality representation.

neously forward a packet to multiple parent nodes, which by virtue of being in the PS are assumed to be within radio range (Figure 3.5). OH only makes sense when RE is also used, working in addition to it to increase reliability without sacrificing transmission latency.

Since with this function the sender effectively broadcasts the packet and receiving nodes disable their MAC filter, two issues are required to be addressed. Firstly, promiscuous OH is performed for data packets which are typically forwarded with ACK-ed unicast transmissions. However, multiple parents may receive the broadcast and attempt to respond with an ACK, potentially resulting in a collision if multiple nodes attempt at the same time. This would lead the transmitting node to consider the transmission to have failed. Thus, it is important that only one node, i.e., the recipient indicated in the destination address field, responds with an ACK, while all other nodes silently receive the packet without sending an ACK. This capability is neither present by default in TSCH nor implemented in Contiki OS. We provided a solution by introducing OH cells, which accept packets sent to any destination (by disabling MAC filtering) but which suppress responding with an ACK even when one is requested.

Secondly, it is desirable to control the replication of packets via OH so that only the AP node forwards the packet and not any node that happens to overhear it. This is not supported in TSCH or RPL by default, thus we have evaluated two options to implement it. One way to do this is via the TSCH schedule, configuring only the AP to overhear

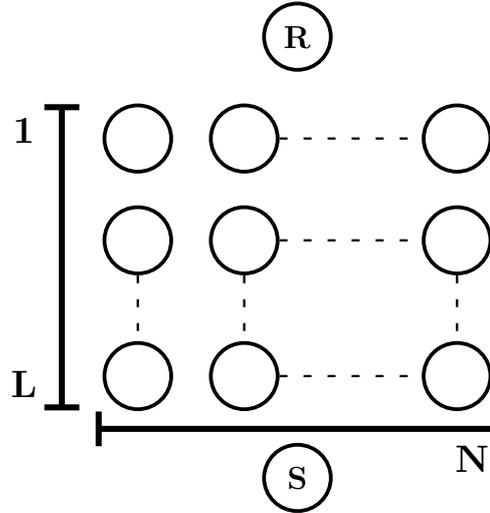


Figure 3.6 – Experimental mesh topology illustration.

when the node transmits. Another is by storing the address of the AP in the packet so even if other nodes receive it, they can check the stored AP address against their own and avoid forwarding. In our implementation we have opted for the latter since it simplifies the TSCH schedule management.

The interaction of OH with the TSCH schedule only concerns the required existence of receiver cell(s) for OH packets from a node to its AP.

3.3.4 Topology

To reduce the effects of randomness and to aid the analytical description of the behavior of the network, we chose a topology as illustrated in Figure 3.6. The network consists of a source node S , a root R , and L layers of nodes with N nodes each, i.e., $|\mathcal{L}(i)| = N, i \in \{1, 2, \dots, L\}$ where $\mathcal{L}(i)$ is the set of nodes in the i^{th} layer. The S node and the nodes in layers $2 \dots L$ each have all the N nodes in the layer above them in their PS, i.e., $PS(S) = \mathcal{L}(L), |PS(S)| = N$ and $\forall i \in \{2, \dots, L\}. \forall n \in \mathcal{L}(i). PS(n) = \mathcal{L}(i-1), |PS(n)| = N$. The nodes in layer 1 only have the root node R in their parent set. Finally, the nodes report a subset of their parent set in the PS extension [Kou+20] of the NSA object in the MC of DIO messages (referred to as the PS_{MC} parent set here on). This PS_{MC} extension contains a fixed number of addresses M , with $1 \leq M \leq N$.

Simulation		Topology	
Duration	Until 5000 pkts	Topology	Mesh topology
Data traffic	1 pkt/ 15 sec	N ^o of nodes	32
Routing protocol	RPL	N ^o of layers (L)	5
Parent set size (N)	6	N ^o of sources	1
PS_{MC} size (M)	6	Link Quality	50%, 75%
TSCH	Single-path	Multi-path	
Scheduling	Centralized	Centralized	
Timeslot length	10 ms	10 ms	
Slotframe length	357 Timeslots	357 Timeslots	
N ^o of channels	1	1	
N ^o of RTX	0 RTX, 1 RTX 3, RTX, 7 RTX	1 RTX	

Table 3.1 – Simulation setup PAREO.

3.3.5 Performance Evaluation

Simulation Setup

We performed extensive experiments to evaluate the trade-offs presented by the use of the RE, ARQ, and OH functions. More specifically, we evaluated with a network topology (shown in Figure 3.6) that consists of 32 nodes, out of which, one source node (S), one root node (R) and 30 relay nodes. In addition, we established that the multi-path RPL routing follows a braided pattern since it is the pattern with the most redundant links, thereby providing full exploitation of the PAREO functions. To do so, we established that a node can be selected as an AP if the PP of the PP belongs to the PS of the AP.

The relay nodes form 5 layers ($L = 5$) of 6 nodes ($N = 6$) and every node in the network, except for the root, is within the radio communication range of all the nodes in the layers immediately above and below it. We evaluated a total of 16 scenarios which include the standard single-path forwarding mechanism, with different values for the number of MAC layer RTX: 0, 1, 3, and 7 (for a total of 1, 2, 4, and 8 transmissions, respectively). Furthermore, we evaluated combinations of the PAREO operations: RE (no ARQ, no OH), RE + OH (no ARQ), RE + ARQ (no OH), and RE + OH + ARQ (i.e., PAREO), to evaluate the contribution of each operation individually. The MAC layer implementation used is TSCH with a single radio channel and the schedule used is pre-determined to allow one TX opportunity for each child-parent uplink when $RTX = 0$ and two when $RTX \geq 1$. By taking into account the trade-off between the slotframe size and the end-to-end la-

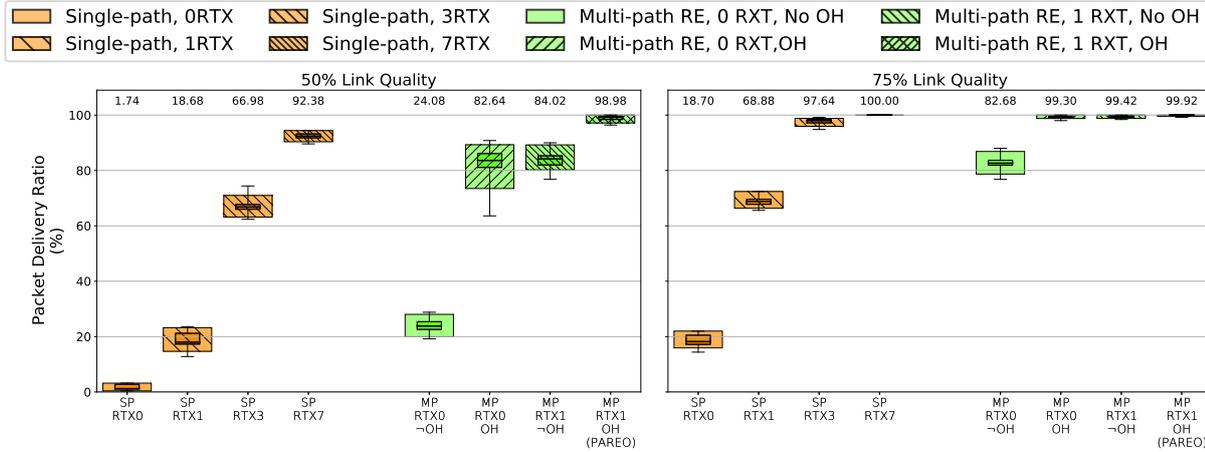


Figure 3.7 – PDR obtained by using the PAREO function.

tency in TSCH networks, we have therefore chosen to limit the number of transmission opportunities in a slotframe to 2 per child-parent link, i.e., a Slotframe contains 2 transmission Timeslots per node, which are usually used for one transmission and one possible RTX. Therefore, if an RTX is required after two losses, the node must wait for the next Slotframe.

For each routing algorithm choice (8 in total, 4 single-path plus 4 multi-path), we set different link qualities (50% and 75%) between all the nodes.

We have also executed 20 simulations with different random seed numbers per scenario, and in each simulation 250 packets were sent from the source node S to the destination node R. Therefore, for each of the 32 scenarios, 5000 packet transmissions are made for a total of 160000 for the whole work. We used Contiki OS, with modifications implementing the RE and OH operations¹, and the simulated network environment provided by COOJA.

PDR

Fig. 3.7 shows the average PDR obtained from all the simulated environments. Here, the x-axis represents the different environments in study, while the y-axis shows the statistical value of the number of packets received versus the total number of packets transmitted. Single-path requires the SP RTX 7 algorithm, with 7 re-transmissions and 8 total transmissions, in order to compete with PAREO. It can only compete when the link quality is higher (75%). With lower link qualities (i.e., 50%), PAREO is superior in all cases to SP RTX 7. Furthermore, in the case of 50% link quality, PAREO has an

1. <https://github.com/ariskou/contiki/tree/draft-ietf-roll-nsa-extension>

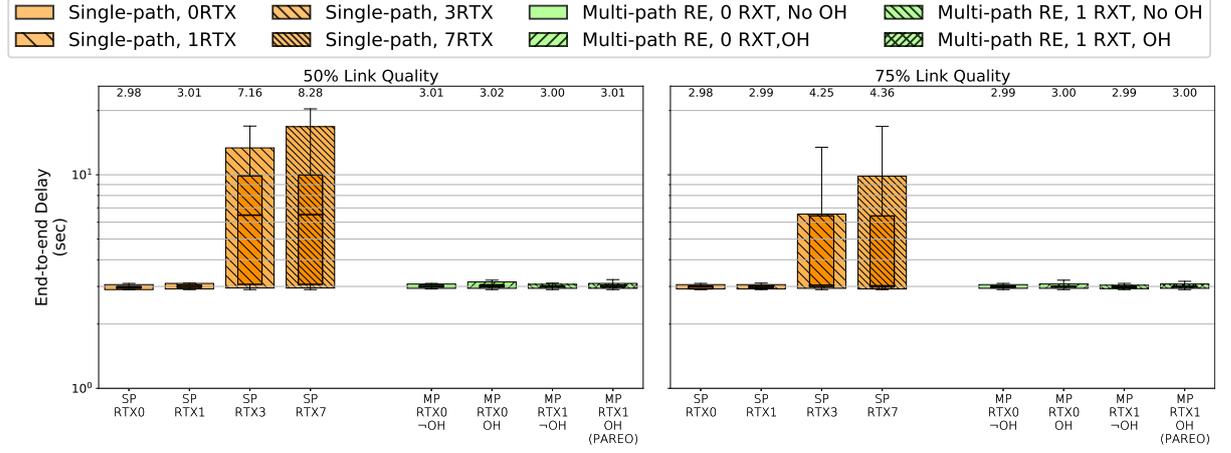


Figure 3.8 – Delay and Jitter (Percentiles of Delay) obtained by using the PAREO function.

approximately 7 times lower Packet Error Ratio (PER), where $PER = 100\% - PDR$:

$$\begin{aligned}
 PDR_{\text{PAREO}} &= 98.98\% \rightarrow PER_{\text{PAREO}} = 1.02\% \\
 PDR_{\text{SP RTX7}} &= 92.38\% \rightarrow PER_{\text{SP RTX7}} = 7.62\% \\
 \frac{PER_{\text{SP RTX7}}}{PER_{\text{PAREO}}} &\approx 7.47\times
 \end{aligned}$$

As it can be observed, for RE, the most significant advantage is provided by the addition of ARQ function, i.e., an extra MAC layer re-transmission, while OH helps but not to the same degree. More specifically, with 50% link quality, ARQ reduces Packet Error Ratio (PER) by 4.75 times while OH by 4.47 times. In overall, the results demonstrate that PAREO achieves very high PDR due to access to alternative routes.

Delay and Jitter

Regarding the latency, it is worth mentioning that the main advantage of using PAREO is that it can achieve bounded delay and jitter at the same time, as shown in Figure 3.8. All the combinations of the PAREO functions provide approximately the same performance given the same schedule, but for single-path, once the number of MAC layer re-transmissions attempted surpasses the number of transmission opportunities in the same slotframe (i.e., single-path with RTX 3, and 7) both delay and especially jitter are significantly impacted. While it is possible to only increase the number of transmission opportunities in the schedule to reduce jitter, this would nevertheless impact the latency.

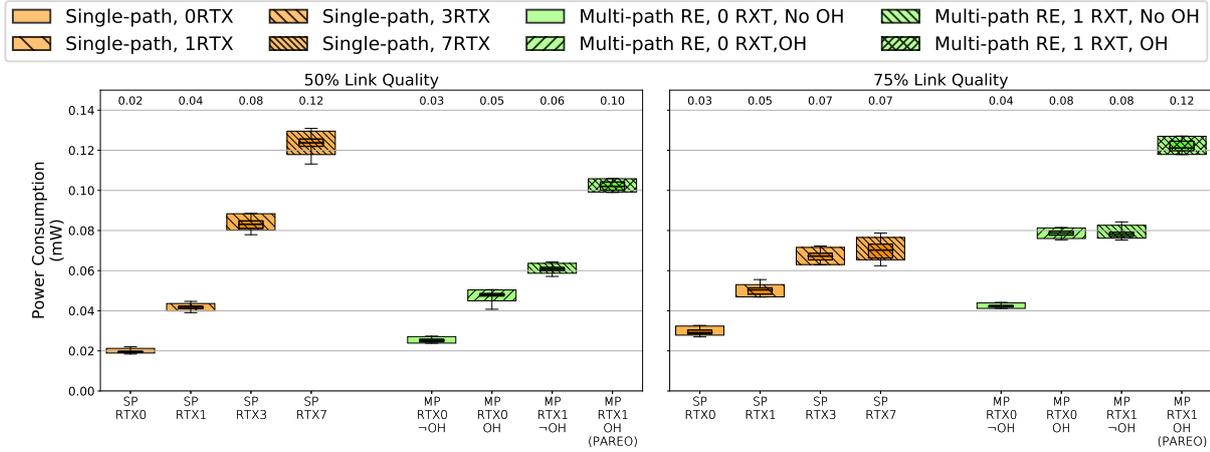


Figure 3.9 – PAREO power consumption per node per slotframe due to radio operation.

Power Consumption

Since COOJA only provides power-tracking in terms of time spent in the different radio states, we have extrapolated the radio power consumption of the whole network in mW by using the radio power consumption values of the Zolertia Z1 mote which uses the CC2420 radio transceiver module [10]. More specifically, we used the given specification sheet values ($P_{TX} = 52.2mW@3V$, $P_{RX} = 56.4mW@3V$, $P_{Idle} = 1.28mW@3V$) and that when being interfered the consumption is the same as when transmitting ($P_{TX} = P_{INT}$). Since the RTX 0 scenarios use a shorter slotframe than the rest of the RTX scenarios, we report the power consumption per slotframe, to allow meaningful comparisons. As shown in Fig. 3.9, the single-path scenarios generally have comparable power consumption to the RE/ARQ/OH combinations, with higher RTX values predictably leading to higher power consumption. It should be noted that the power consumption measured comprises the cost sending both data and control packets.

3.4 Conclusions

In this chapter, three types of redundant patterns were presented in addition to three functions that allow extending RPL with multi-path denominated PAREO functions. From the redundant patterns, we could conclude that their strategies are based on addressing transmission efforts towards a common objective. However, depending on how these selection strategies are made, the probability of success will vary.

We learned that the disjointed pattern consists of the construction of several disjointed paths that start from the source node and merge at the sink node. Because of this, the multi-path complexity of the algorithm resides in the source while the relay nodes focus only on maintaining the disjointed behavior. As a result, this pattern maintains a low action in the network compared to the other patterns, but in turn, this leads to lower transmission reliability than the other two strategies. The braided pattern, on the other hand, concentrate the greatest amount of links towards the next node each time a hop is required during transmission. As a result, it is the pattern with the best reliability performance. However, this can lead to a higher power consumption since a greater amount of links are used during transmission.

Given the above, we decided to follow the specifications that characterize the braided pattern given the high-reliability standards required by the industry.

Regarding the functions of PAREO, they were tested under the conditions of the braided pattern. In terms of reliability, it can be noted that ARQ performs better than OH, while when these two functions are activated in conjunction with RE, the PDR values can be close to 100%. On the other hand, the reliability of a single-path is directly subject to the number of RTX. Only when 7 RTX is employed, the performance might reach PAREO values. Regarding the delay and jitter, PAREO performs better in comparison with single-path. This is because fewer RTXs are required in PAREO since it provides more opportunities using RE and OH. Finally, in terms of power consumption, we could observe that single-path consumes more power as the links get worse since they will require a greater number of RTX per hop. With the PAREO functions, however, it is observed that as the quality of the links gets worse, the environments that used only OH consumed less energy than the environments that used only ARQ. This may be due to a decrease in APs in networks with low link quality.

In the next chapter, we will present and analyze the first multi-path algorithm called N-Disjoint. N-Disjoint establishes a multi-path routing considering the disjoint pattern and using the PAREO functions.

N-DISJOINT

4.1 Introduction

Considering the wireless network scenarios, the probability of a successful transmission will depend partially on the link quality where the transmission takes place. This is why one of the possible requirements that characterize multi-hop routing protocols is to determine the wireless links that satisfy forwarding requirements while maintaining good link quality. Unfortunately, given the irregular nature of wireless environments, a link, even if it has a good quality, cannot ensure transmission reliability. Furthermore, conditions will become worse as more hops are required for transmission.

This affects RPL negatively since, by default, it allows only single-path routing for transmission; however, several solutions have been proposed to mitigate this problem. One of the most popular mechanisms is ARQ, which, as discussed in the previous chapter, makes a controlled number of RTX each time the ACK transmission is not received. As a result, transmission reliability statistics improve, however for each RTX used, latency and jitter get worst.

On the other hand, RE makes multiple copies of a packet with an objective that at least one of these can reach the destination. Nevertheless, to achieve this, it requires a multi-path routing. Given the above, this function also improves transmission reliability; however, its complexity lies in the routing. Consequently, a poorly elaborated strategy can lead to network flooding, which leads to higher energy consumption.

Therefore, in this chapter, we propose the N-Disjoint algorithm, which aims to improve transmission reliability by using RE and ARQ functions over a modified disjoint pattern. Moreover, we present different scenarios to analyze the trade-offs of each function to achieve a balanced state between reliability, latency, and energy consumption. To perform these, two selection strategies were implemented on this algorithm.

4.2 RE and ARQ

RPL is a multi-hop routing protocol adapted to different LLN networks thanks to its repair methodologies and proactive routing methods based on metrics. Unfortunately, since its standard operation is limited to single-path route selection, this protocol does not reach the industrial environment's reliability levels. This is also due to further factors, such as the link's quality, or the multiple hops that the packets have to traverse.

A popular solution is to re-transmit a message a certain number of times whenever it has to pass a hop. However, for every RTX performed, delay and jitter are affected. For this reason, ARQ (Section 3.3.1) proposes a limited number of RTXs, which are only used each time an ACK transmission is not received. However, as the link qualities decline, ARQ will need to use more RTX to maintain high reliability, leading to worse latency.

On the other hand, RE (Section 3.3.2) allows multiple copies of a single packet to be transmitted over several different paths. However, to do that, RPL must be adapted to allow multi-path routing. This leads to the routing protocol's complexity since a selective strategy is required to forward the packets. Hence, we have chosen to use the Disjoint Pattern (Section 3.2.1), which is characterized by its policy of route isolation selection and the fact that multi-path propagation starts at the source. However, due to its selection policy, the different routes that constitute this pattern behave as a single path, which consequently leads the packet to depend on the conditions of these routes, i.e., the probability of a packet lost is given by the quality of the links that constitute the route.

Therefore, the N-Disjoint algorithm is defined by combining these two functions (ARQ and RE) in order to allow a balance between latency performance and power consumption while ensuring transmission reliability. This algorithm allows the source to select multiple nodes within PS to broadcast its message. Then, for each hop performed, the ARQ function is used to guarantee the packet's delivery.

4.3 N-Disjoint Strategies

Referring to Section 3.2.1, the disjoint pattern, unlike the triangular and braided patterns, is exposed to frequent losses since its stability depends on the distance between the source and the sink. This is because each relay node forwards the packet via a unique route. Therefore, their multiple links selected from the source are independent of each other, inheriting the same disadvantages of the single-path. However, even though this

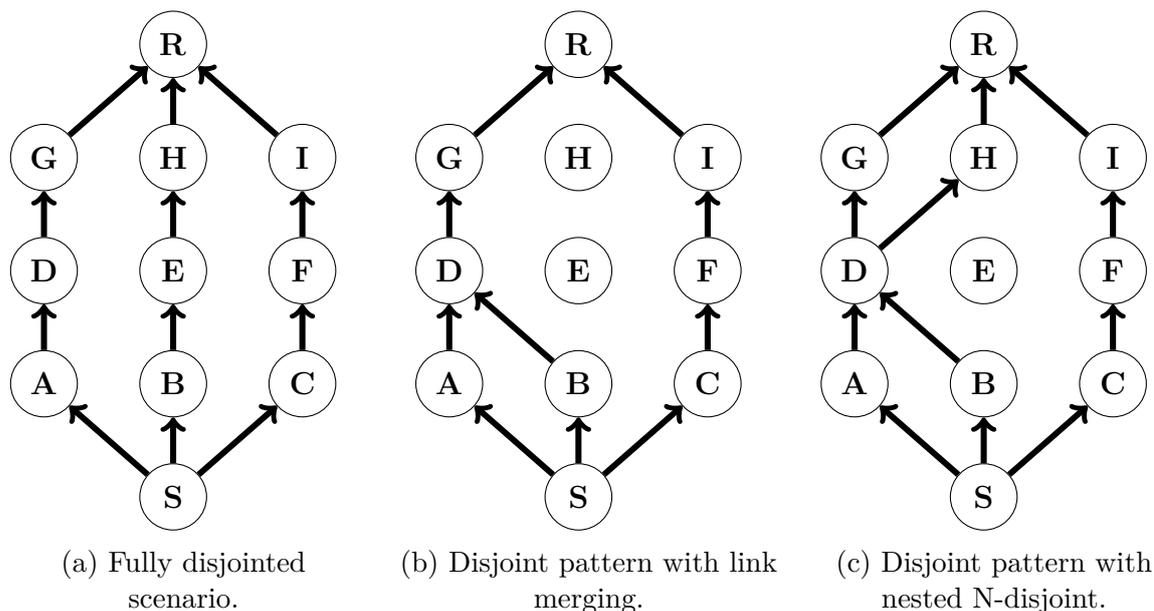


Figure 4.1 – Potential N-disjoint scenarios.

selection policy results in these disadvantages, it is attractive due to its simplicity, and because it maintains a low energy consumption profile.

Considering the limitations of this pattern, we have established the N-Disjoint algorithm, which aims to maintain a reliable network by taking into account some of the characteristics of the disjoint pattern. In addition, the operation of this algorithm depends on the PAREO ARQ and RE functions, which are used to reinforce the routing of the transmitted packets. Furthermore, the behavior of this algorithm depends on two types of selection strategies, which are detailed below.

4.3.1 Basic Strategy

This strategy consists of authorizing RPL to proceed with its routing without necessarily maintaining a disjoint pattern but keeping the multi-path performance only at the source node. Consequently, the routing of the relay nodes will be subject to the parent selection that RPL grants them, allowing nodes of the same hop distance to link towards the same parent, without this necessarily being the sink node. Figure 4.1 shows the behavior of this selection strategy where Figure 4.1a demonstrates a completely disjoint scenario, while Figure 4.1b demonstrates the existence of a merged road working in parallel with a disjoint road.

The operation of this selection strategy consists of giving only authorization to the source node to use the RE function with a limited number of replicas that do not exceed the size of the PS, while the ARQ is used in all transmission links with a bounded number of RTX. Given this, whenever a new packet needs to be transmitted from the source node, this node will select a PP plus multiple APs without exceeding the maximum AP boundary. The relay nodes, on the other hand, use the standard RPL routing, meaning that only a single-path is employed on them.

As a result, two or more links can merge before reaching the root node, which consequently reduces the number of disjointed links reaching it. However, although the number of disjointed links is reduced, those links formed by a merge will receive the attempts of the predecessor nodes increasing the possibilities to perform the next hop.

4.3.2 Advanced Strategy

This strategy consists of taking the basic strategy's selective policies but with the difference of authorizing some nodes to access multiple links, without these necessarily being the source node. In this way, transmission reliability is improved. Figure 4.1c illustrates how this strategy works, with node D being one of many possible nodes that can perform multi-path.

This strategy was developed taking into account that when several links are merged before reaching the final destination, the reliability of the transmission is restricted to the total number of disjointed paths that reach the destination node. Consequently, only ARQ is used, which makes the packet depending on the number of RTX. Besides, if the paths merge early in the network, the number of RTX sets may not be enough as more attempts are required to pass the additional hops.

This strategy mitigates these problems by establishing that nodes can send copies of a packet to the same number of parents as copies received from that packet. In this way, the total number of replicas will not be affected by link merging and will proceed normally.

It should be noted that, as well as the basic strategy, ARQ is used by each link.

PAREO		TSCH	
ARQ	True	Scheduling	Centralized
RE	True	Slotframe length	345 Timeslots
OH	False	Timeslot length	10 ms
		N ^o of channels	1
Simulation		Topology	
Duration	5000 pkts	Topology	Mesh topology
Data traffic	1 pkt/15 sec	N ^o of nodes	32
Routing protocol	RPL	N ^o of layers (L)	5
Parent set size (N)	6	N ^o of sources	1
PS_{MC} size (M)	6	Link Quality	50%, 75%

Table 4.1 – Simulation setup for N-disjoint.

4.4 Performance evaluation

4.4.1 Simulation Setup

We conducted a vast number of experiments to evaluate the trade-off between replications and RTXs based on *basic* and *advance* strategies. These experiments were conducted using the topology presented in Section 3.3.4 (see Figure 3.6) with a total of 32 nodes based on a source, a sink, and 30 relaying nodes.

The distance between the source and the sink is 6 hops ($L = 5$), and for each hop, there is a set of 6 nodes ($N = 6$) where each node can communicate with the other nodes of the set located in the adjacent vertical hops.

The MAC layer implementation used is TSCH (see in Section 2.1.2) with a single radio channel and the schedule used is pre-determined to allow one TX opportunity for each child-parent uplink when $RTX = 0$ and two when $RTX \geq 1$. By taking into account the trade-off between the slotframe size and the end-to-end latency in TSCH networks, we have therefore chosen to limit the number of transmission opportunities in a slotframe to 2 per child-parent link, i.e., a Slotframe contains 2 transmission Timeslots per node, which are usually used for one transmission and one possible RTX.

We evaluated a total of 80 scenarios composed of the combination of RTX 0, RTX 1, RTX 3, and RTX 7, with a total of 0 to 4 replications. In addition, for each scenario generated, each link is evaluated with a quality of 50% and 75%, respectively.

For the accuracy required for the evaluation of the algorithms, 20 different simulations

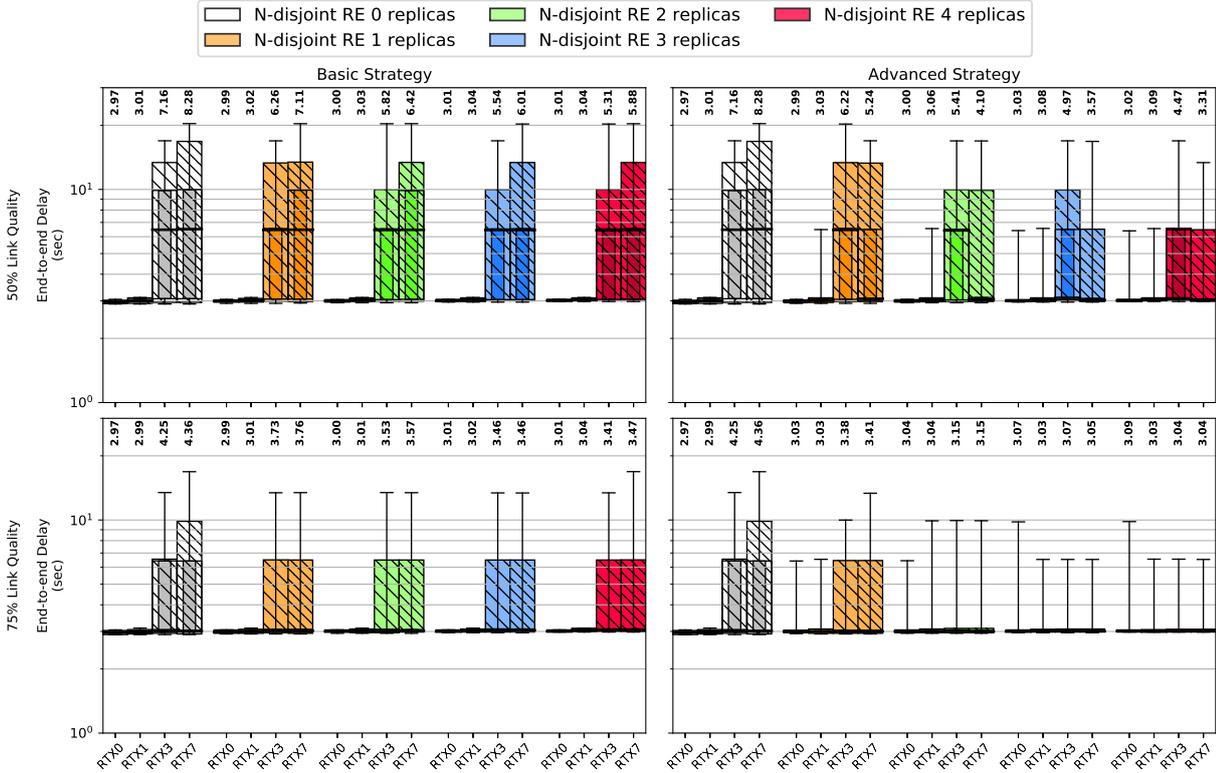


Figure 4.2 – N-disjoint latency using different replicas and RTX.

were executed for each generated scenario. These simulations consisted of a total of 250 packets per seed being transmitted every 15 seconds.

We used Contiki OS, with modifications implementing the RE and ARQ operations, and the simulated network environment provided by COOJA.

4.4.2 Simulation Results

Network Reliability (PDR)

Figure 4.3 shows the performance of the N-Disjoint algorithm in terms of network reliability. From the data obtained, it can be seen that reliability is mostly linked to the chosen RTX, reaching a PDR above 90% in ARQ with a maximum number of 7 RTX. On the other hand, RE is significantly affected by the link quality, showing poor behavior in environments with 50% link states.

It should be noted that when ARQ works with RTX 7, the PDR is maintained above 90%, regardless of whether RE is used.

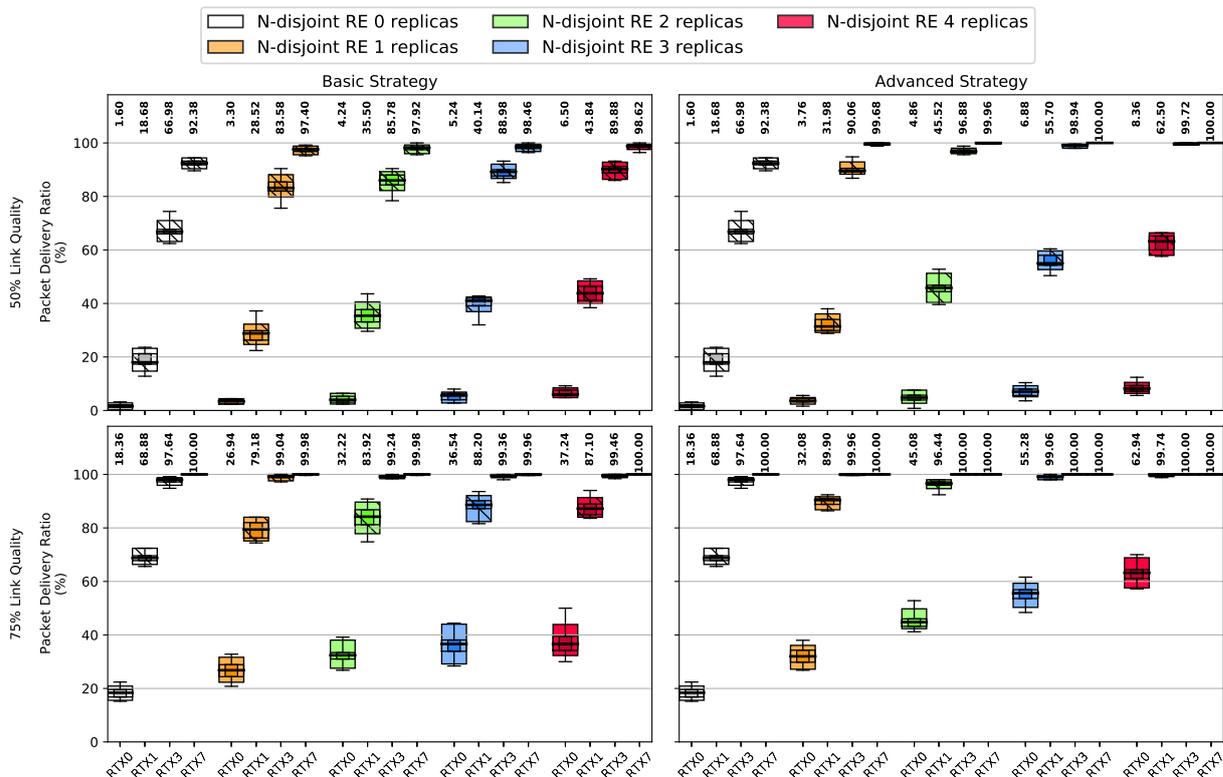


Figure 4.3 – N-disjoint PDR using different replicas and RTX.

In the advanced strategy, it can be observed that RE and ARQ have significantly better performance than in the environments with basic strategy.

Delay and Jitter

Figure 4.2 represents the latency of the different scenarios studied, where the median values represent the average delay between the time a packet was transmitted, and the time it was received by the target. The maximum values of each case provide us with the standard deviation of these delays (jitter). Given that the latency is strictly linked to the scheduler, the jitter will depend on the number of Slotframes used for transmission.

As defined in the simulation setup (see Section 4.4.1), a Slotframe contains 2 transmission Timeslots per node, which are usually used for one transmission and one possible RTX. Therefore, if an RTX is required after two losses, the node must wait for the next Slotframe.

Table 4.1 shows that a Slotframe consists of 345 TimeSlots, of which 33 are used for control transmissions and 312 for data transmission. Since a Timeslot has a duration of

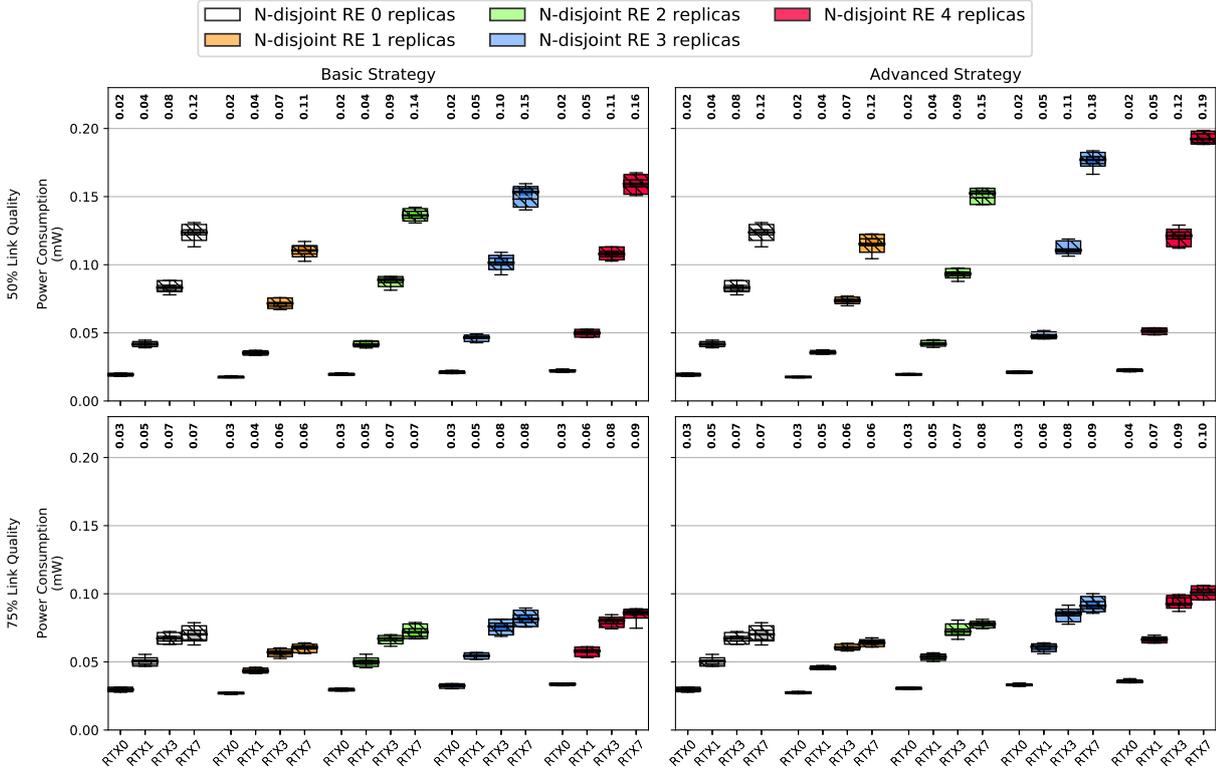


Figure 4.4 – N-disjoint energy performance.

10 ms, each transmission takes approximately 3120 ms from the source node to the sink.

Taking this into account, it can be seen that most packets use at least 2 Slotframes, excluding scenarios with 75% link quality and an advanced strategy.

Finally, if we focus on the environments with 50% link quality, we can observe that the basic strategy and the advanced strategy have similar performances in the scenarios with RTX 0, RTX 1, and RTX 3, however, in environments with RTX 7, the latency of the advanced strategy is better. In the case of environments with 75% link quality, the advanced strategy stands out in comparison to the basic strategy.

Power Consumption and number of replication per packet

In terms of power consumption, Figure 4.4 shows that as the link quality decreases, more energy is required. The reason for this is that each packet loss generates a new transmission, which consumes more energy. Especially if these attempts are used in multiple directions.

To detect the effects of RE in this context, we must consider the approximation of the

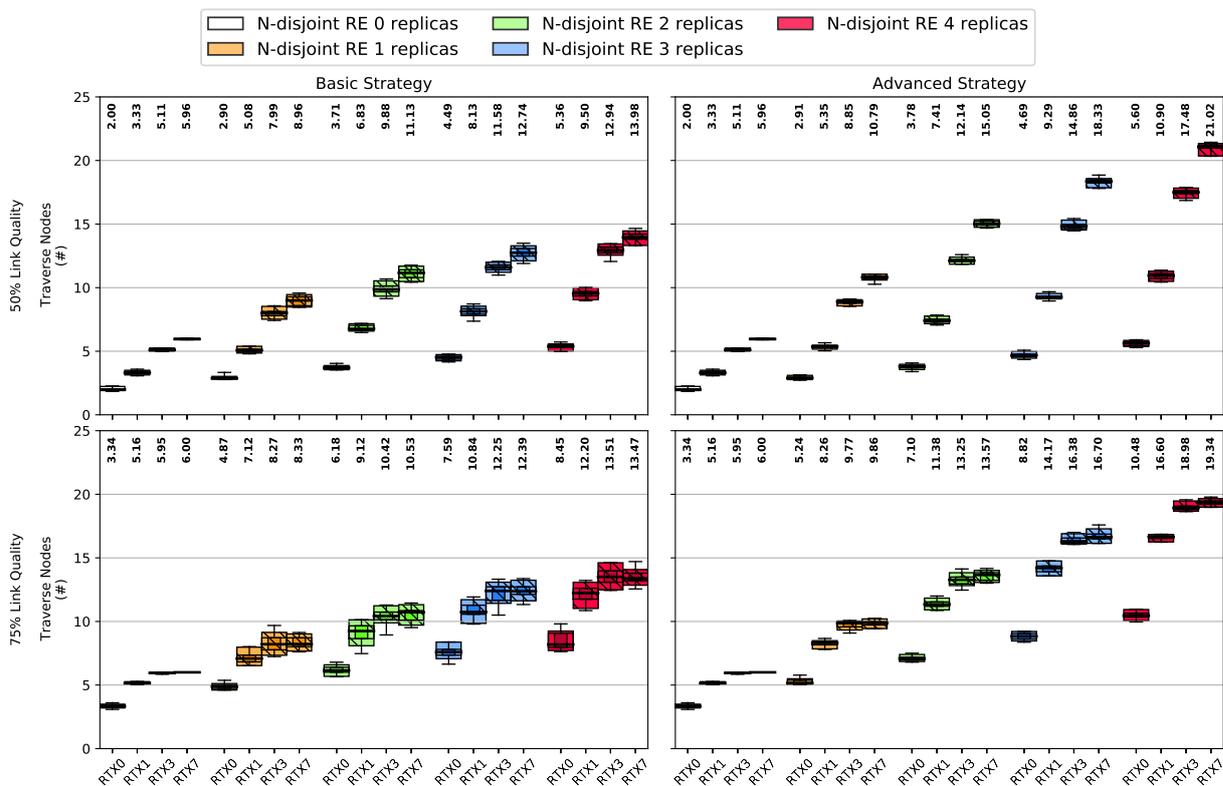


Figure 4.5 – Average relay nodes per transmission using N-disjoint.

total number of participating nodes per transmission. Figure 4.5 shows this, giving the average number of relay nodes for each transmission. To begin with, the number of relay nodes used by the advance strategy is higher than those used in the basic strategy. Besides, we can see that the number of nodes involved is not affected by the state variability of the link.

Based on the above, comparing Figure 4.5 with Figure 4.4 shows that the RTX provides most of the power consumption since this behavior is amplified in the scenarios with worse link quality.

4.5 Conclusions

In this chapter, the N-Disjoint routing algorithm was discussed and analyzed. As a selection method, this algorithm uses PAREO’s ARQ for RTX, and RE for replication by employing a modified disjoint pattern. Its goal is to achieve high transmission reliability through multi-path while balancing power and latency charge.

As a result, two selection strategies were defined, the basic strategy and the advanced strategy. The basic strategy methodology consists of allowing RPL to elaborate single-path along the network, excluding the source node which can replicate the packet to a limited number of parents using multi-path. On the other hand, the advanced strategy extends the basic strategy by allowing a relay node to replicate a packet to the same number of parents as the number of copies of that packet received by the node. In other words, the advanced strategy enables multi-path replication to all nodes, and disables the RE elimination function. It should be noted that these two methodologies are provided with 0, 1, 3, and 7 RTX.

We learned that the two N-disjoint strategies using RTX 7 can reach PDR over 90% in all environments studied. We also learned that the advanced strategy performs better than the basic strategy in PDR, latency, and power consumption.

From the results obtained, it was shown that in terms of PDR, ARQ has more influence than RE. However, by balancing these two functions, an acceptable PDR is achieved. This gain changes when we talk about delay and jitter since a higher number of RTX leads to an increase in these two variables.

Concerning power consumption, we can observe that ARQ plays a great part of this expense. Moreover, the power consumption gets higher as more replicas are performed.

In conclusion, this algorithm guarantees transmission reliability with its advanced strategy. However, given the high power consumption generated by each RTX, it is required to generate an adequate trade-off between replicas and RTXs based on the quality of the network.

In the following chapter, we will deal with CA selection to perform braided routing patterns using three selection algorithms.

COMMON ANCESTOR (CA) SELECTION

5.1 Introduction

As demonstrated in the previous chapter, the RE function combined with the ARQ function, improves the reliability of the network. However, maintaining a trade-off between the number of links and the amount of RTX is critical to keep a stable LLN.

This balance can be shaped in different combinations, and depending on the forwarding pattern employed, different strategies can be addressed. An example of this is given in the strategies required by a disjoint and a braided pattern. According to the previous chapter, an utterly disjointed pattern will depend on the RTX of its disjointed paths and on the number of links employed from the source node to ensure its reliability. This leads to a surplus in terms of latency and energy consumption. However, route merging, as seen in both basic and advanced strategies, reduces the load on RTX and the number of links to be used since transmission efforts converge towards a common point.

In braided patterns, each sibling node (nodes with the same hop distance) addresses at least one common parent. This implies that for each hop performed, it is required a balance between RE and ARQ.

In this chapter, we will focus on three algorithms that use the braided pattern to develop different strategies to find an AP. These algorithms use a CA selection policy, which consists in providing a common parent to multiple nodes except the source node each time a packet performs a hop. The purpose of these methodologies is to ensure high level of network reliability, as well as providing stable latency.

5.2 Selection models

A routing algorithm based on a braided pattern (see Section 3.2.3) is mainly characterized by having nodes directed at least to a common parent node, but with the condition that these nodes must be at the same distance from the sink node. This condition gives

advantages to the transmission reliability since it concentrates a greater number of transmission attempts at a specific point, increasing the probability of passing a hop. However, since it requires multi-path to be formed, a selection strategy must be formed to guarantee at least two parents per hop, while not over-selecting unnecessary parents.

To extend RPL with multi-path and direct it to perform a braided pattern, we applied CA pattern selection. This selection methodology consists in allowing a node to find an AP if this possible node candidate has within its PS a mutual node with the PS of the node's PP, i.e., if a node A has as PP a node B , and if a node C has inside its PS a node in common with the PS of B , then if the node C belongs to the PS A , this node can be selected as AP of A . In this way, a braided pattern can be formed, starting at the source and ending at the sink.

We have defined three algorithms for selecting an AP in CA selection. Each of these has different selection policies that directly affect both the probability of getting an AP and the number of nodes used to propagate a packet to the sink node. In addition, to demonstrate the behavior and compensations that these selections entail, we have tested them in different scenarios.

Below, we present the proposed algorithms. We use the topology presented in Section 3.3.4 to calculate the probability of finding an AP for node S and the rest of the nodes located in layers $2 \cdots L$, and where each layer is a set of N nodes. In addition, the common general equations used to derive the probability of having a CA and the existence of an AP are defined through the following steps:

$$\begin{aligned}
 P(\exists CA) &= 1 - P(\nexists CA) \\
 P(\nexists AP) &= (1 - P(\exists CA))^{N-1} \Rightarrow \\
 P(\exists AP) &= 1 - P(\nexists AP) = 1 - (1 - P(\exists CA))^{N-1}
 \end{aligned}
 \tag{5.1}$$

5.2.1 Strict CA

This algorithm allows a candidate node v be selected as an AP from a u node if the PP of the PP of u is the same as the PP of the AP candidate, i.e., if $PP(PP(u)) = PP(v)$. This is illustrated in Figure 5.1a.

Given this, this algorithm requires that a candidate node shares the same PP with the current PP. That is in terms of the number of relay nodes used for the transport of a packet. However, obtaining APs may be difficult since each node has different parent

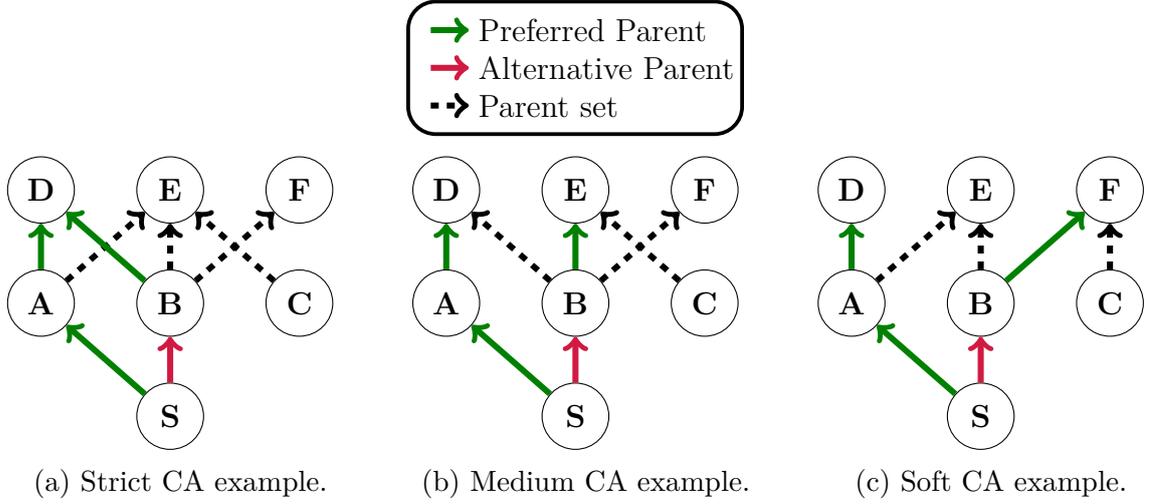


Figure 5.1 – Algorithm list for Common Ancestor (CA) selection.

sets. If we assume that the source and nodes of layer $2 \cdots L$ have the same number of neighbors N , the probability of finding an AP is represented by Equation 5.2.

$$P(\exists CA) = \frac{1}{N} \quad (5.2)$$

where $P(\exists CA)$ is the probability of having at least one CA and $P(\exists AP)$ is the probability of having at least one AP (the complement of probability of not being able to find any APs). Given these, the probability of finding an AP is calculated in Equation 5.3.

$$P(\exists AP) = 1 - \left(\frac{N-1}{N}\right)^{N-1} \quad (5.3)$$

5.2.2 Medium CA

The purpose of this algorithm is to guarantee to find an AP if this node has within its set of parents the PP of the node's PP. In other words, an AP candidate v can be chosen as the AP of a node u if the PP of the node's PP belongs to the PS_{MC} of the AP candidate, i.e., if $PP(PP(u)) \in PS_{MC}(v)$. To do so, we require the fixed number of addresses M contained into the PS_{MC} .

Contrary with Strict CA, this algorithm offers more nodes that cover the conditions to be AP. An example that presents this case is shown in Figure 5.1b and its probability

of having an AP is given by Equation 5.4.

$$P(\exists CA) = \frac{M}{N} \quad (5.4)$$

Using Equation 5.1, we obtain Equation 5.5.

$$P(\exists AP) = 1 - \left(\frac{N-M}{N}\right)^{N-1} \quad (5.5)$$

5.2.3 Soft CA

This algorithm aims to guarantee a node to find an AP by covering a wider range of allowed nodes. To achieve this, this algorithm allows nodes that have any common node from their parent set with those in the PP's parent set in order to be selected as AP. More specifically, an AP candidate v can be selected as an AP from a node u if the PS_{MC} of the PP of the node has some overlap with the PS_{MC} of the AP candidate, i.e., if $PS_{MC}(PP(u)) \cap PS_{MC}(v) \neq \emptyset$. Figure 5.1c shows an example of this case.

In Equation 5.6, the probability of obtaining a CA depends on the probability of having a common node in the two PS_{MC} parent sets across the PP and across the AP candidate.

$$P(\nexists CA) = \frac{\binom{N}{M} \cdot \binom{N-M}{M}}{\binom{N}{M} \cdot \binom{N}{M}} = \frac{\binom{N-M}{M}}{\binom{N}{M}} \Rightarrow$$

$$P(\exists CA) = 1 - \frac{\binom{N-M}{M}}{\binom{N}{M}} \quad (5.6)$$

Finally, using Equation 5.1, the probability of finding an AP is represented by the Equation 5.7.

$$P(\exists AP) = 1 - \left[\frac{\binom{N-M}{M}}{\binom{N}{M}} \right]^{N-1} \quad (5.7)$$

5.2.4 Discussion

Figure 5.2 shows the performance of Equations 5.3, (5.5), and (5.7). All these equations provide the probability of getting an AP. The X-axis of each figure covers different maximum sizes configured for the PS_{MC} set. On the other hand, the Z-axis includes a

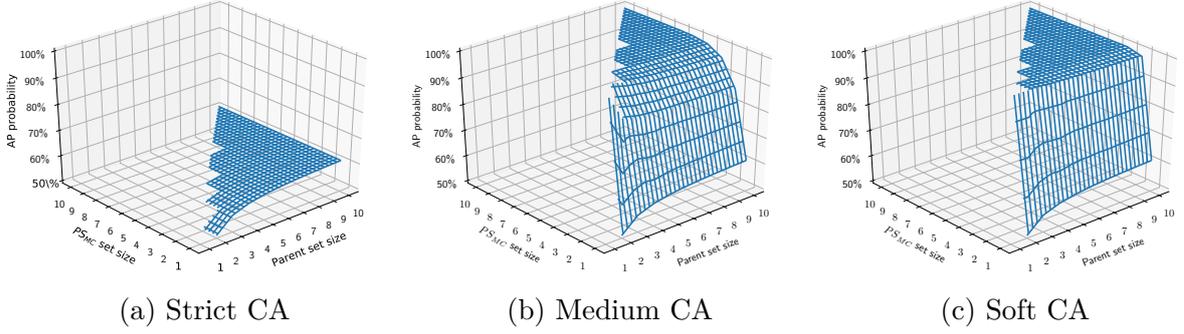


Figure 5.2 – Probability of obtaining an AP according to the algorithm.

different set of nodes that represent the size of the PS. Finally, the Y-axis presents the resulting probability of getting an AP by combining the X and Z axis parameters.

Analyzing the values of Figure 5.2a, it is observed that as the size of PS of a node increases, the probability of obtaining an AP increases, reaching its maximum with a $PS \geq 5$ and with values close to 60%. It should be noted that since this method does not use the information from the set of parents, the M size of the PS_{MC} does not affect the probability.

On the other hand, in the case of Figure 5.2b and Figure 5.2c, respectively, it can be seen that their behavior is similar. This is because both are directly influenced by the number M of addresses in the PS extension in the NSA MC. However, the Soft CA has a higher probability value than the Medium CA using the same parameters, since the Soft CA uses the intersection between the two sets of PS_{MC} , while the Medium CA uses only one parent (the PP) of one of the two sets of PS_{MC} that intersect with a set of PS_{MC} . Because of this, the Soft CA reaches values close to 100% with dimensions larger than one node in its PS_{MC} since there are N , M values with a guaranteed non-empty intersection.

Given all the above, the fact that a node does not have an AP directly affects the network reliability. As a result, in terms of network reliability, the performance of Strict CA will be lower than the rest of the algorithms, since it has a higher risk of performing a single-path transmission in some of its relay nodes.

5.2.5 Implementation Aspects

CA algorithms are implemented using a custom RPL OF, which extends MRHOF [GL12]. These algorithms use the same behavior that MRHOF uses for RANK calculation and PP selection but adding additional filtering to the candidate parents in the PS of an AP

PAREO		TSCH	
ARQ	True	Scheduling	Centralized
RE	True	Timeslot length	10 ms
OH	True	Slotframe length	357 Timeslots
		N ^o of channels	1
Simulation		Topology	
Duration	5000 pkts	Topology	Mesh topology
Data traffic	1 pkt/15 sec	N ^o of nodes	32
Routing protocol	RPL	N ^o of layers (L)	5
Parent set size (N)	6	N ^o of sources	1
PS_{MC} size (M)	6	Link Quality	50%, 75%

Table 5.1 – Simulation setup for CA algorithms.

50% and 75%, respectively.

To satisfy the precision required to perform the respective evaluations, 20 different simulations were executed for each scenario generated. These simulations consisted of a total of 250 packets per seed transmitted every 15 seconds.

Each scenario has a centralized TSCH program (referred to in Section 2.1.2) with an 297 Timeslots size of 10 ms. We use the Contiki operating system, with modifications in RPL and TSCH to enable RE, ARQ, and OH operations. Finally, the simulated network environment was provided by the COOJA simulator¹.

5.3.2 Simulation Results

PDR

In terms of PDR, the probability for a node to obtain an AP directly affects performance. This is because a larger number of relay nodes leads to a higher number of packet copies in the network, which in turn increases reliability. In Figure 5.4, we see the average PDR obtained from all the simulations performed, where the X-axis denotes the different environments under study, and the Y-axis the resulting PDR of each environment.

From this figure we can see that in environments where RPL was used with single-path routing, transmission reliability has improved as the RTX boundary increases, reaching values of 100% with no more than 7 RTX in environments with 75% link quality. However, with links with 50% quality the PDR of RPL single-path with 7 RTX reaches a maximum

1. <https://github.com/ariskou/contiki/tree/draft-ietf-roll-nsa-extension>

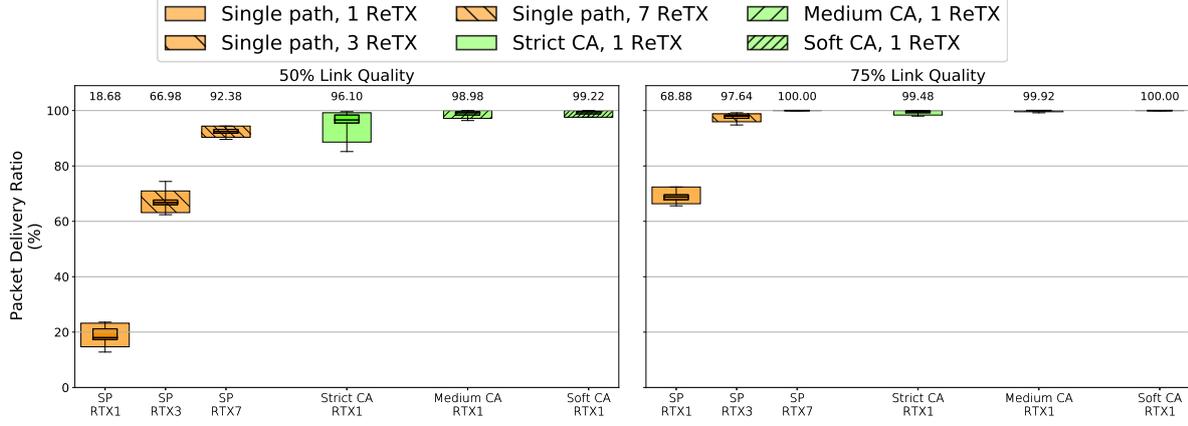


Figure 5.4 – PDR obtained by each CA algorithm.

of 92.38%.

Looking at the environment where RPL is configured with single-path, RPL= 1, and with 75% link quality, we can see that its PDR is 68.88%. This is because each transmission probability follows Equation 5.8.

$$P(q, r, d) = \left(\sum_{k=0}^{r-1} (1 - q)^k q \right)^d \quad (5.8)$$

Where q stands for link quality, r corresponds to the maximum number of transmissions and d to the total number of hops between the source node and the root node. In addition, taking into account the parameters of Table 5.1, we have that the required amount of hops d between the source node and the root node is $L + 1$. Therefore, the resulting theoretical PDR is given by the following equation.

$$\begin{aligned} d &= L + 1 \\ \text{PDR} &= P(q, r, d) \\ \text{PDR} &= P(0.75, 2, 6) \rightarrow 0,67893 \end{aligned} \quad (5.9)$$

Regarding the environments where the CA algorithms were used, it should be noted that all of them have values higher than 95% of PDR. Moreover, considering only the cases single-path and CA with RTX 1, we can see that the CA algorithms are considerably better than the single-path environments.

Finally, we can observe that in environments with 75% link quality, Soft CA achieved a PDR of 100% while Medium CA achieved 99.92% and Strict CA 99.48%.

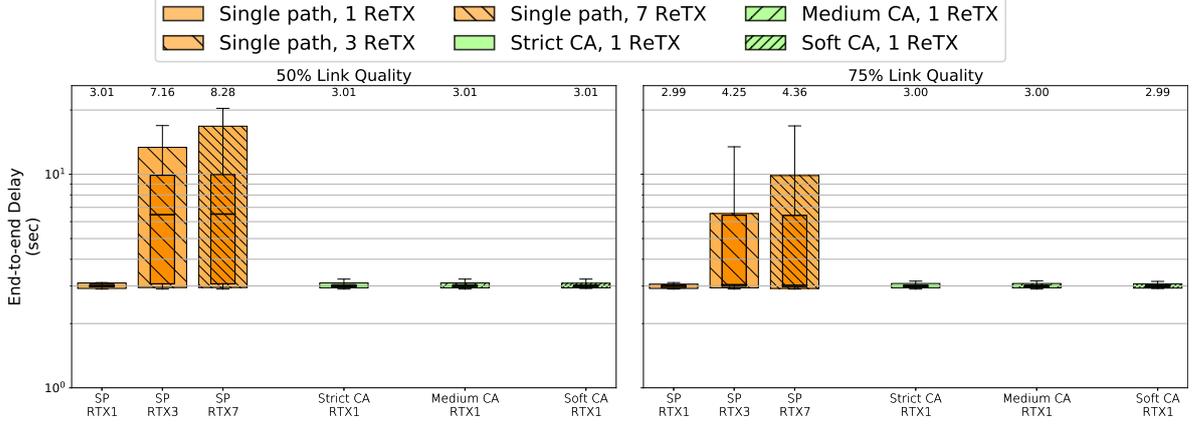


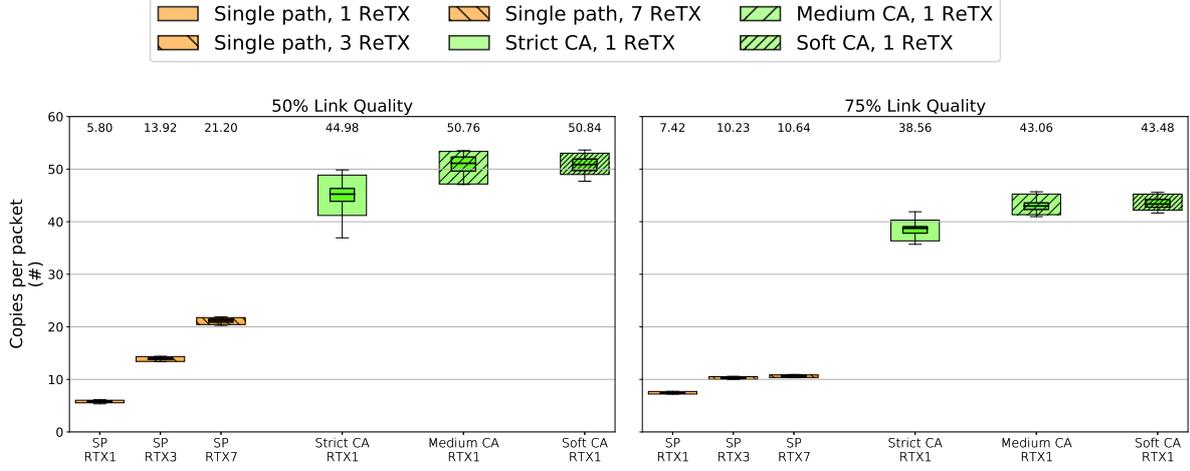
Figure 5.5 – Latency obtained by each CA algorithm.

Delay and Jitter

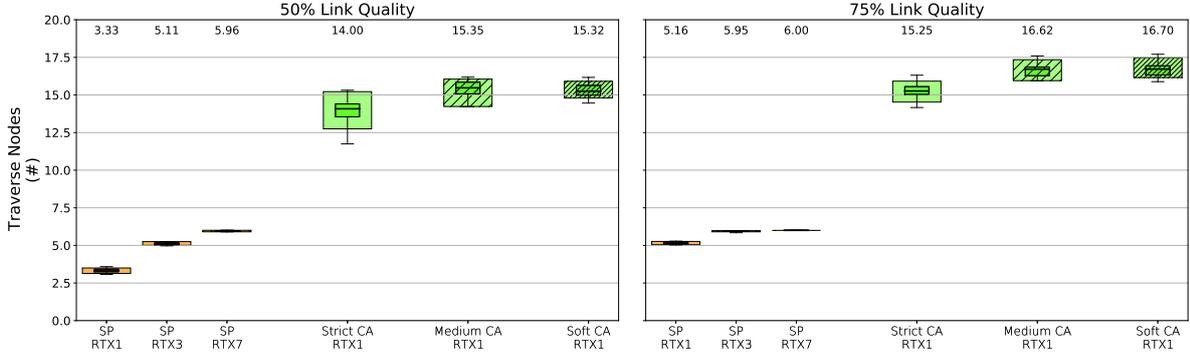
Figure 5.5 shows the average latency obtained from the different studied use-cases. Here the Y-axis delivers the average delay of a packet that started from the source node and ends at the root, while the jitter is represented as the standard deviation of each delay. This measurement is taken at the moment this packet is transported in its first Timeslot until one of its copies is received by the root node for the first time. It should be noted that each simulation is based on a centralized schedule, therefore the latency will depend mainly on the size of the Slotframe programmed.

$$\begin{aligned}
 T_{L+1} &= N \text{ (source layer)} \\
 T_{L\dots 2} &= N^2 \text{ (each intermediate layer)} \\
 T_1 &= N \text{ (layer of direct children of the root)} \\
 T &= L_{L+1} + (L - 1)L_{L\dots 2} + L_1 \\
 &= (L - 1)N^2 + 2N
 \end{aligned} \tag{5.10}$$

Taking into account the values defined in Table 5.1 and considering that the simulations were performed with the topology defined in Section 3.3.4, we obtain that the total distance between the source node and the sink node consists of $L + 1$ hops, i.e., 6 hops. Additionally, each node has a neighborhood with $N = 6$ possible parents per hop, excluding the children of the sink node and the sink itself. Considering that 32 nodes compose the topology, then the total number of up-links is 156. More specifically, if T is the total number of links, then its expression will be given by Equation (5.10) .



(a) Average number of copies made per transmitted packet.



(b) Average number of relay nodes involved in the transmission of a packet.

Figure 5.6 – average number of copies per packet (a), and average number of relay nodes (b) per simulated environment.

Finally, if we consider that there is a possible RTX per transmission (that is, two transmissions) and that each Timeslot lasts 10 ms, then the maximum delay per Slotframe is 3120 ms, i.e., taking in consideration Equation (5.10) and considering MD as the maximum delay per Slotframe, then:

$$\begin{aligned}
 MD &= T \times (1 \text{ TX} + 1 \text{ RTX}) \times 10 \text{ ms} \\
 MD &= 156 \times (2) \times 10 \text{ ms} \rightarrow 3120 \text{ ms}
 \end{aligned}
 \tag{5.11}$$

Figure 5.5 shows this analysis, which highlights the increase in delay and jitter in an environment with more than one RTX. Furthermore, the high jitter values of the single-path scenarios with RTX 3 and RTX 7 are mainly due to the fact that some of the

transmissions performed in these scenarios required more than one Slotframe.

Traffic overhead

In terms of traffic overload, Figure 5.6 gives us information on this behavior in different simulated environments. Here in Figure 5.6a, the y-axis gives the number of copies generated per transmitted packet, while Figure 5.6b gives the average of the number of nodes involved to make an upward transmission from the source node to the sink node.

Starting with single-single path RPL use-cases, the number of copies made per packet (Figure 5.6a) decreases as the link quality improves. In turn, the number of nodes involved increases (Figure 5.6b) as the link quality improves. This behavior is due to the number of attempts required to make a transmission will decrease. Consequently, the average number of nodes involved in the transmission will be higher since more packets will perform the total hop distance

This behavior is also reflected in the environments where the multi-path algorithms were used. However, the number of nodes involved will depend on the success of the propagation of the packet. Besides, the number of copies is higher than the single-path due to its propagation behavior. However, as with single-path, fewer copies are made as the link quality improves since less RTX is required.

Power Consumption

Figure 5.7 shows the results obtained in terms of power consumption by each of the simulated environments, which were calculated using the values of the Zolertia Z1 mote, which uses the CC2420 radio transceiver module (see Section 3.3.5). In this Figure, the Y-axis represents the total average energy used by each simulation in mW, taking as a starting point the establishment of DODAG and ending 15 seconds after the transmission of the last packet.

At first glance, it can be seen that the environments configured with single-path RPL are strongly affected as the quality of the links deteriorates. However, this is not reflected in the use-cases where CA selection algorithms were used, having a very small difference in energy expenditure between each link quality.

If we take into account the data in Figure 5.6a and combine it with the data in Figure 5.7, we can observe that the number of packets used and the amount of power spent are not related. This indicates that the amount of RTX performed affects, to a greater extent, the amount of power consumed. Consequently, as the quality of the links

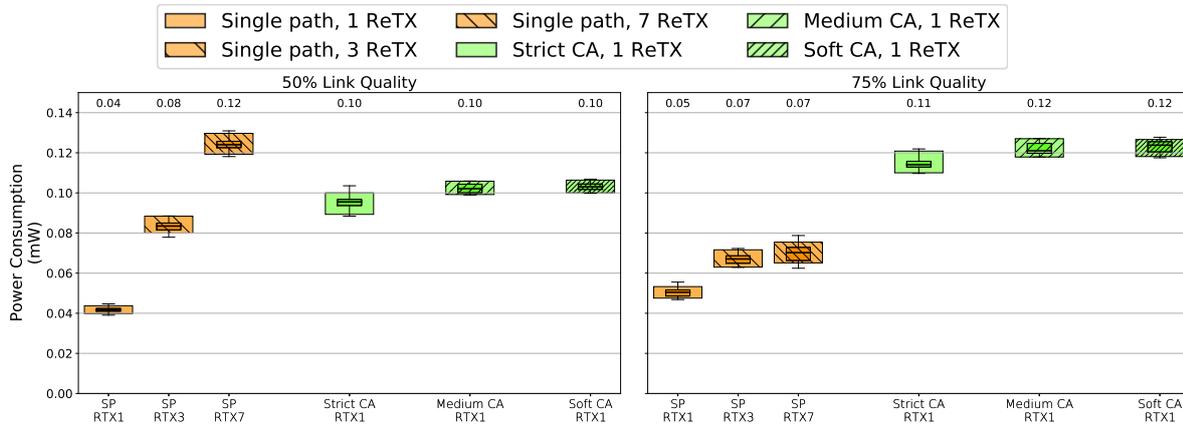


Figure 5.7 – Energy expenditure of each CA algorithm.

improves, the less RTX will be required, resulting in the single-path variations shown in Figure 5.7.

Thus, taking into account the environments where multi-path CA was used, the energy variation has a small difference since they can only perform one RTX at most. Therefore, the replication of multiple copies of a packet does not have a drastic impact on power consumption. However, due to their propagation policy, the average power consumption is about 0.11 mW.

Finally, analyzing these data we can affirm that although the overall power consumption of the network is higher for our algorithms compared to the default single-path algorithms, their trade-off is reasonable since they ensure transmission reliability and latency stability.

5.4 Conclusions

In this chapter, three algorithms were proposed using the CA selection method. Our objective is to ensure transmission reliability by following braided pattern features. However, due to the variable behavior of LLNs, finding and maintaining a stable AP for each node is complex, making it difficult to find one. Given this, each algorithm has different selection strategies.

The first algorithm, *Strict CA*, consists of allowing a node to obtain an AP if it has the same PP as the PP's PP. *Medium CA*, on the other hand, consists of allowing a node to select an AP if its PP belongs to the PS of the PP. Finally, *Soft CA* allows to select a

node as an AP if its PS contains a node in common with the PS of the PP.

As a result, each algorithm has different probabilities of finding an AP, which will depend on the number of valid parents that each algorithm allows. Algorithms that allow a greater range of valid parents will tend to disperse the transmission more, while those with a smaller number of valid parents will have less chance of finding an AP.

Experiments conducted in the Cooja simulator have shown that the proposed algorithms have a PDR over 95%, with those that have a greater range of possible AP (Medium CA, and Soft CA) close to 99% PDR. It was also observed that RPL with single-path depends directly on the RTXs to improve its transmission reliability.

In terms of latency, we have seen that CA algorithms have better delay and jitter than single-path use-cases with more than one RTX. This is because multi-path condenses a larger number of transmission attempts into shorter periods, requiring fewer RTX attempts.

What we have learned from this chapter is that even though CA algorithms have a higher traffic overhead, the main factor that triggers the power consumption is the amount of RTX. As a result, CA maintains a stable balance across different link qualities, standing out in links with very low qualities.

To conclude, these algorithms stand out for their reliability, delay and jitter, and transmission stability. However, they require a higher power consumption when they are compared to the single-path transmission operating in high link qualities.

In the next chapter, we will cover the ODeSe algorithm, based on CA selection policies, but with the distinction of keeping the routing propagation as compact as possible.

ON-DEMAND SELECTION (ODESE)

6.1 Introduction

Most industrial applications require strict guarantees in terms of network reliability and low latency performance. The delay for a data packet to traverse through the network must be bounded and constant. As we have seen throughout this work, there are multiple ways to meet these requirements. However, defining an algorithm that satisfactorily fits the requirements of the industry is a challenge since once a requirement is improved, a trade-off with other supplies is demanded. Moreover, an algorithm may tend to overspend resources, so optimization is crucial in its selection methods.

In the course of this research, we have shown that the use of redundant link patterns is a good strategy to provide higher network reliability in multi-path RPL routing. This could be seen in Chapter 4 with the N-disjoint algorithm which uses the disjoint patterns as the selection basis, and in Chapter 5 where the CA algorithms use the braided pattern to compose the routes.

We also note that some pattern structures may be insufficient as their behavior does not meet the required needs. This can be observed in multi-path patterns such as disjoint since each of its paths behaves with single-path performance. Therefore, it is important to use supplementary methods to overcome their weaknesses, such as those proposed in Section 4.3.1 and Section 4.3.2.

In addition, it must also be considered that the algorithms used for routing must be efficient with the resources available. Poor management of these algorithms can result in unnecessary use of network resources. However, generating efficient algorithms is also a challenge, since they require a predictive model that does not exist in wireless technologies.

In this chapter, we will introduce the ODeSe algorithm, which aims to ensure transmission reliability by using CA selection with minimum routing dispersion while keeping high AP matching rates. To do so, this algorithm centers its efforts on maintaining two relay nodes per-hop as long as these nodes both have the same PP and AP. In addi-

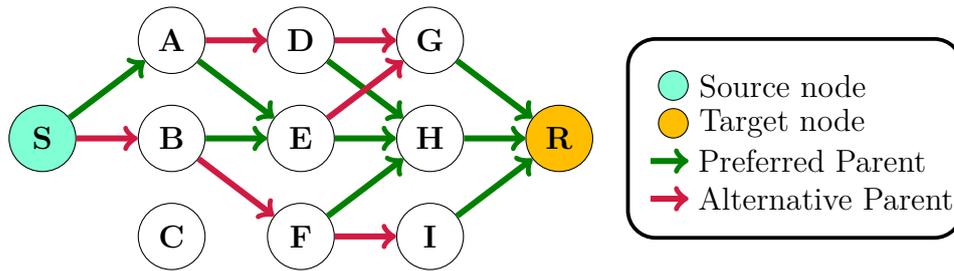


Figure 6.1 – Flooding using the strict CA algorithm.

tion, this chapter will provide a discussion of this algorithm against the CA algorithms to demonstrate ODeSe’s contribution.

6.2 CA Trade-offs

By analyzing the three CA algorithms, we can observe that the possible selection of an AP node is determined by the PS from a node’s PP. This is because each of these algorithms depends on the existence of a non-empty subset from the intersection between the PS of a candidate node and the PS of the node’s PP. However, as we have seen previously, the resulting subset of this intersection will correspond to the selection policies of each algorithm. Consequently, depending on which CA algorithm is being employed, the probability of finding possible candidates for AP will fluctuate, just as it has been shown in Chapter 5.

Checking at the Strict CA algorithm, we know that it is the CA algorithm that has the most difficulties to get an AP given its selection policy. As a result, the number of routing paths in the network is lower compared to its other peers, making it the CA algorithm with the lowest expenditure, but at the same time, with the lowest network reliability rate. However, if the ideal conditions are given, this algorithm can disperse its routing just as the Soft CA algorithm can.

An example of this is Figure 6.1, where the topology is defined by the notation presented from Section 3.3.4, being $L = 3$ and $N = 3$, i.e., 3 groups of 3 nodes between the source and the sink. Here we can observe that the routing includes almost all the nodes with exception of the node C. Besides, the routing is scattered between the transition from group {A, B, C} towards group {D, E, F}. This is mainly because even though each PP and AP has the same PP, this behavior does not involve their APs. More specifically, nodes A and B have node E as PP, however, nodes D and F are selected as AP since

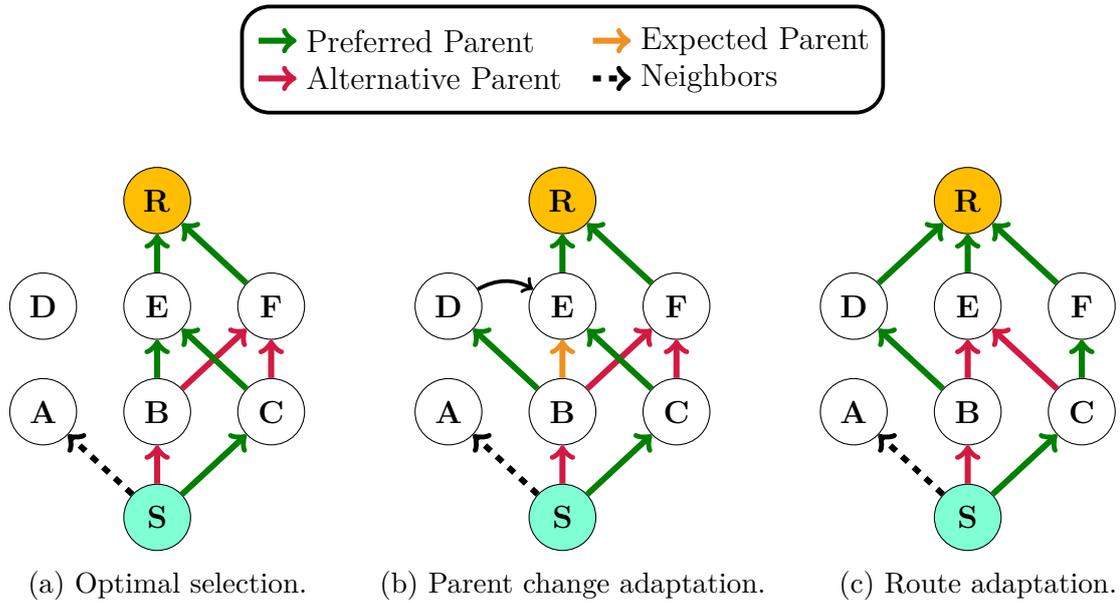


Figure 6.2 – On-Demand Selection (ODeSe) functions for parent selection

they have node H as PP, as well as node E. This shows that even though Strict CA is the algorithm with the lowest routing spread among the other CA algorithms, it may still generate the same traffic load as its peers. Given this, it is imperative to generate an algorithm that maintains a low routing dispersion while having a high AP finding rate.

6.3 ODeSe

While the Strict, Medium, and Soft CA algorithms perform well, the highest network reliability is achieved with Soft CA but at the cost of higher energy consumption due to higher flooding. In this section, we present ODeSe that aims to maintain the same level of network reliability while reducing the energy consumption. The main difference between the CA algorithms and the ODeSe is that in ODeSe each node decides not only its own forwarding parents (PP and AP) but also the PP and AP to be used in the next hop towards the destination. In other words, each node has a two-hop forwarding control. As it is shown in the performance evaluation, this results in tighter control of flooding that can be translated in lower energy consumption.

Algorithm 2 demonstrates the ODeSe selection methodology where for each hop, a node k selects its PP and AP using the Strict CA algorithm. In the case that no valid node is found to be AP, the node k will opt for a Medium CA strategy and consequently a Soft

Algorithm 1: Common Ancestor (CA)

Input: k : The current node,
 c : A candidate parent node
Output: True if c can be used as a PP, else False.
Function $IS_VALID_PP(k, c)$
| **return** $RANK(k, c) < RANK(k, PP(k))$
end

Input: k : The current node,
 c : A candidate parent node
Output: True if c can be used as an AP with Strict CA, else False.
Function $IS_VALID_AP_STRICT(k, c)$
| **return** $IS_VALID_PP(k, c) \text{ AND } PP(PP(k)) = PP(c)$
end

Input: k : The current node,
 c : A candidate parent node
Output: True if c can be used as an AP with Medium CA, else False.
Function $IS_VALID_AP_MEDIUM(k, c)$
| **return** $IS_VALID_PP(k, c) \text{ AND } PP(PP(k)) \in PS(c)$
end

Input: k : The current node,
 c : A candidate parent node
Output: True if c can be used as an AP with Soft CA, else False.
Function $IS_VALID_AP_SOFT(k, c)$
| **return** $IS_VALID_PP(k, c) \text{ AND } PS(PP(k)) \cap PS(c) \neq \emptyset$
end

Input: k : The current node,
 f_v : A candidate parent validity checking function (i.e., $IS_VALID_AP_STRICT$,
 $IS_VALID_AP_MEDIUM$, or $IS_VALID_AP_SOFT$).
Output: The AP node to use.
Function $CA(k, f_v)$
| **if** $AP(k) = NULL \text{ OR } NOT \text{ IS_VALID_PP}(k, AP(k))$ **then**
| | **forall** c **in** $PS(k)$ **do**
| | | **if** $f_v(k, c)$ **then** // Check if c is a valid AP
| | | | **return** c ; // Use c as AP
| | | **end**
| | **end**
| | **return** $NULL$; // No AP found
| **else**
| | **return** $AP(k)$; // Use existing AP
| **end**
end

where $PP(k)$, $AP(k)$, and $PS(k)$ correspondingly return the Preferred Parent (PP), Alternative Parent (AP), and Parent Set (PS) of node k . Also, $RANK(k, p)$ returns the RPL rank of node k through a parent node p .

Algorithm 2: On-Demand Selection (ODESe).

Input: HbH_{PP} : The stored Preferred Parent within the Hop-by-Hop IPv6 extension header,
 HbH_{AP} : The stored Alternative Parent within the Hop-by-Hop IPv6 extension header,
 k : The current node.

Output: Returns the parent set used for the next data packet forwarding (PP, AP).

```

Function  $ODESe(HbH_{PP}, HbH_{AP}, k)$ 
   $PP_{old} \leftarrow PP(k)$  ; // Save current PP
  if  $IS\_VALID\_PP(k, HbH_{PP})$  then // Replace current PP
     $PP(k) \leftarrow HbH_{PP}$  ;
  end
  if  $IS\_VALID\_AP\_STRICT(k, HbH_{AP})$  then // Use HbH_{AP} as AP
     $AP(k) \leftarrow HbH_{AP}$  ;
  else // Clear the AP before CA call // AP Strict
     $AP(k) \leftarrow NULL$  ;
     $AP(k) \leftarrow CA(k, IS\_VALID\_AP\_STRICT)$  ;
    if  $AP(k) = NULL$  then
       $AP(k) \leftarrow CA(k, IS\_VALID\_AP\_MEDIUM)$ 
      if  $AP(k) = NULL$  then
         $AP(k) \leftarrow CA(k, IS\_VALID\_AP\_SOFT)$ 
      end
    end
  end
   $PP_{aux} \leftarrow PP(k)$  ; // Save the PP to use for this packet
   $PP(k) \leftarrow PP_{old}$  ; // Restore old PP for other packets
  return  $(PP_{aux}, AP(k))$ 
end

```

where $PP(k)$ and $AP(k)$ correspondingly return the Preferred Parent (PP) and Alternative Parent (AP) of node k .

CA if it fails as well. Additionally, in the forwarded replicas, the node stores the addresses of the PP and the AP of its PP, i.e., $HbH_{PP} = PP(PP(k))$, $HbH_{AP} = AP(PP(k))$. More specifically, these two addresses are stored within the Hop-by-Hop option header of the data packet, to be delivered to the next-hop node, see Figure 6.3.

Once received at the next hop node(s) $k' \in \{PP(k), AP(k)\}$ (either the PP or the AP of a node k), the following checks are made. The next-hop node k' checks if the transported PP address HbH_{PP} is indeed a valid RPL PP for it, and if yes, it then uses HbH_{PP} as the new PP for this specific packet, see Figure 6.2b. The original PP is restored after finishing the forwarding operation for this packet. The validity check is required because information at node k might be “stale” due to not receiving up-to-date parent set information from k' via a DIO packet. If the HbH_{PP} is not a valid PP, then the default PP of node k' is used, see Figure 6.2c.

For the AP, node k' checks if the stored HbH_{AP} node within the Hop-by-Hop option

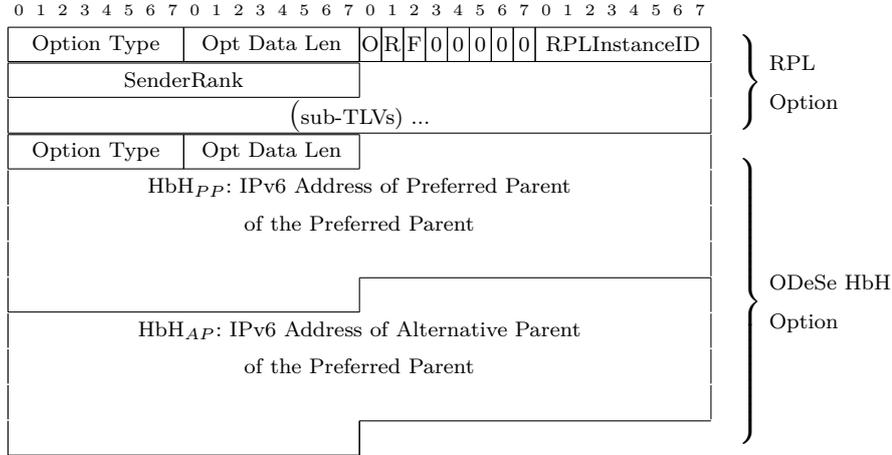


Figure 6.3 – Hop-by-Hop option fields (RPL option and ODeSe field).

field is also a valid AP by Strict CA policy. If it is, then k' uses HbH_{AP} as the AP, see Figure 6.2a. If it is not, then k' will try to obtain a new AP using first Strict CA, and if that fails, falls back to Medium CA and then Soft CA.

When k' has selected the AP and PP for the next hop, it does what node k at the previous hop did before it. It stores and replaces the addresses of the PP and the AP of its PP, i.e., $HbH_{PP} = PP(PP(k'))$, $HbH_{AP} = AP(PP(k'))$ in the forwarded replicas. Therefore, the entire process repeats until the data packet is delivered to the target node.

6.4 Performance evaluation

6.4.1 Number of relays

To evaluate ODeSe against the three CA selection algorithms, we used the Monte Carlo method using the topology studied in Section 3.3.4, which consists of a source node and a sink node, plus L layers or $L + 1$ hops between these and N nodes per L layer, excluding the layer where the sink node is located. The goal is to evaluate the average number of nodes used per layer between different distances and different sizes of parent sets, starting from the source and ending at the root.

The assessments described in Figure 6.4 consist of four scenarios, Strict CA, Medium CA, Soft CA, and ODeSe, respectively. The x-axis gives the distance $L + 1$ hops between the source and the root. The z-axis, on the other hand, gives the N number of existing nodes per hop. Finally, the y-axis gives the average number of relays nodes per segment $L \times N$.

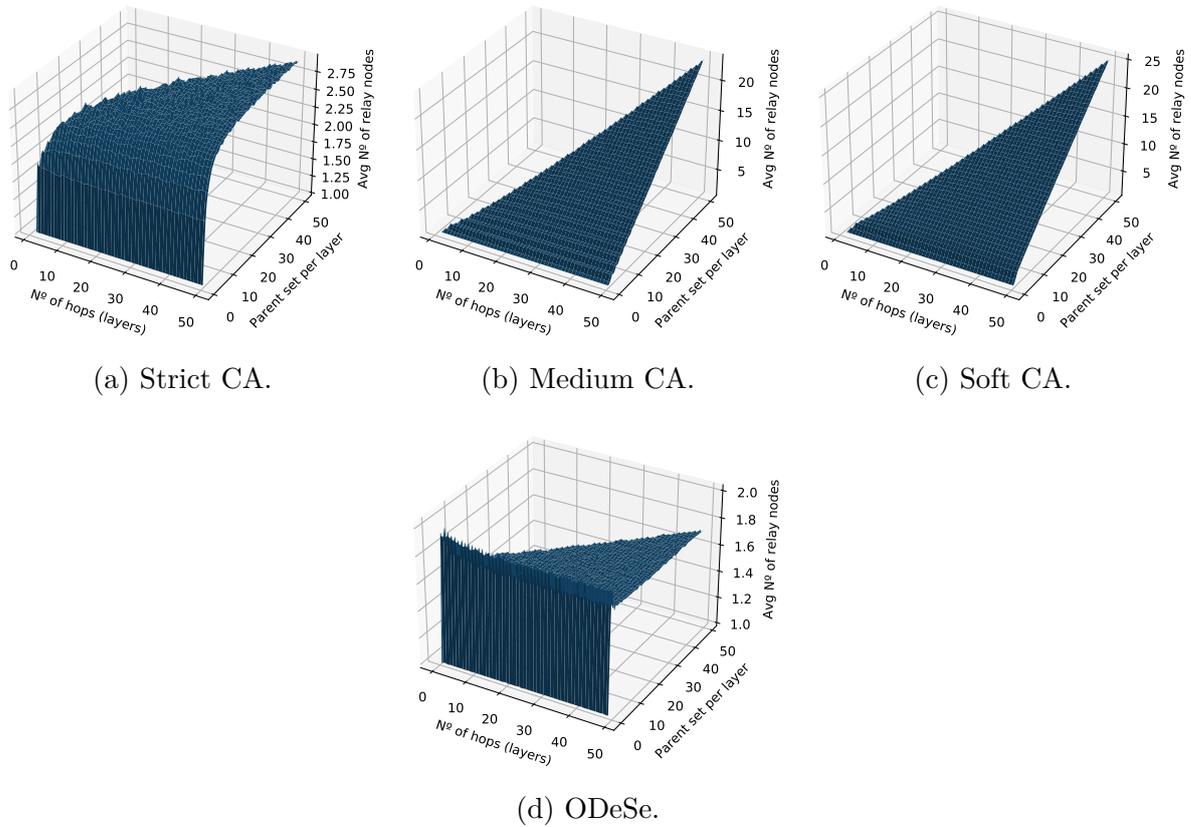


Figure 6.4 – Average number of relay nodes used per layer.

Figure 6.4a shows that the Strict CA algorithm increases the average number of relay nodes per layer until it settles in approximately three nodes. The reason why the number of relay nodes does not exceed the average of 3 nodes may be because Strict CA selection policies do not ensure high reliability in finding an AP. Therefore, routing propagation is restricted because some nodes perform single-paths.

Figure 6.4b and Figure 6.4c, on the other hand, have a linear upward increase. Due to the topology used, the differentiation between these two algorithms is relatively small, leaving Medium CA with 1 relay node lower than Soft CA in terms of the average of relay nodes.

Finally, ODeSe (Figure 6.4d), shows a settles average that does not exceed two nodes per layer yet with values close to those. This indicates that this algorithm does not exceed more than 2 relay nodes per hop. Additionally, given that its average value of 1.8 relay nodes per-hop and that it is estimated that only 2 relay nodes are obtained for each hop, this indicates that it has high reliability to achieve an AP.

PAREO		TSCH	
ARQ	True	Scheduling	Centralized
RE	True	Timeslot length	10 ms
OH	True	Slotframe length	357 Timeslots
		N ^o of channels	1
Simulation		Topology	
Duration	5000 pkts	Topology	Mesh topology
Data traffic	1 pkt/15 sec	N ^o of nodes	32
Routing protocol	RPL	N ^o of layers (L)	5
Parent set size (N)	6	N ^o of sources	1
PS_{MC} size (M)	6	Link Quality	50%, 75%

Table 6.1 – Simulation setup for ODeSe algorithm.

In the next section, we will study the algorithm by simulation to study what are the expected performance in terms of PDR and latency.

6.4.2 Simulation setup

To evaluate ODeSe, we performed several experiments to test it against CA algorithms. These experiments, similarly with the performance evaluation in the previous chapters, were carried out in a topology built with the rules of topology presented in subsection 3.3.4 (see Figure 3.6) with a total of 32 nodes of which one is the source, another the sink and the rest of the 30 nodes as relays.

The distance between the source and the sink is 6 hop ($L = 5$) and for each hop, there is a set of 6 nodes ($N = 6$) where each member can communicate with the other nodes from the adjacent L layers.

We evaluated a total of 2 ODeSe scenarios composed of two different link states of 50% and 75% respectively.

For each scenario generated, 20 different simulations were run. These simulations consisted of a total of 250 packets per seed transmitted every 15 seconds. All scenarios have a centralized TSCH program (referred to in subsection 2.1.2) with a 297 timeslots size of 10 ms. We use the Contiki operating system, with modifications in RPL and TSCH to allow RE, ARQ, and OH operations. Finally, the simulated network environment was provided by the COOJA simulator.

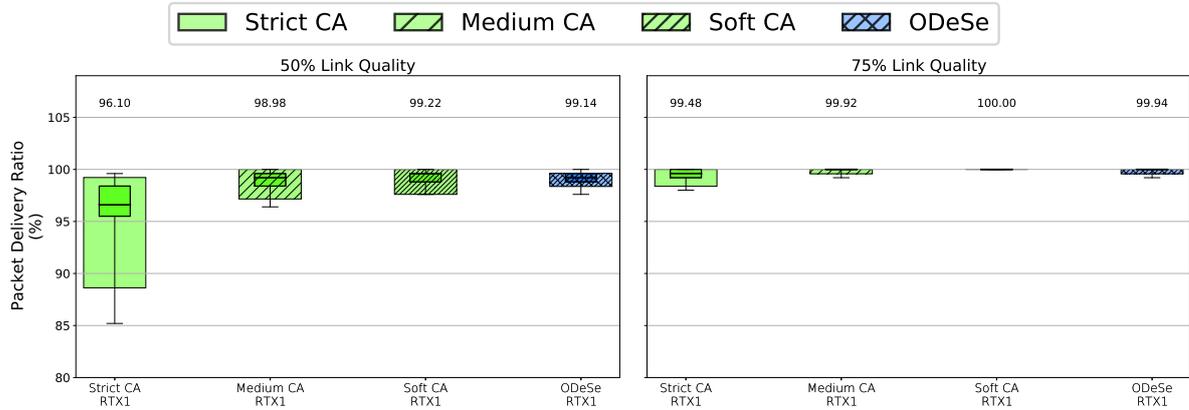


Figure 6.5 – CA algorithm PDR compared to ODeSe PDR.

6.4.3 Simulation results

PDR

Figure 6.5 represents the performance of the Strict CA, Medium CA, Soft CA, and ODeSe algorithms in terms of PDR. The y-axis shows the percentage of received packets out of the total 5000 packets sent (PDR).

As we can see, the performance of ODeSe is higher than Strict CA and Medium CA with a value of 99.14%. It also has similar values to Soft CA with a difference of 0.08%. The reason why ODeSe can reach these values is mainly that it distributes its links so that the upstream transmission involves 8 transmission attempts towards a single direction.

More specifically, since ODeSe selects the same PP and AP to the nodes of the same distance, the PAREO functions perform 1 transmission attempt + 1 RE attempt to one of its parents and duplication of this to the other parent given to the OH function. Knowing that these actions are performed by each parent, a single node delivers a total of 4 transmission attempts to the PP and AP. Finally, assuming there is another relay node at the same distance, these 4 attempts are duplicated.

Discussion

As we have learned throughout our conducted research that the use of multi-path with braided pattern (CA and ODeSe algorithms) guarantees reliability over 90% of transmission. Additionally, as we have learned in the previous chapter (Chapter 5), multi-path algorithms tend to consume a greater amount of energy because they use a greater num-

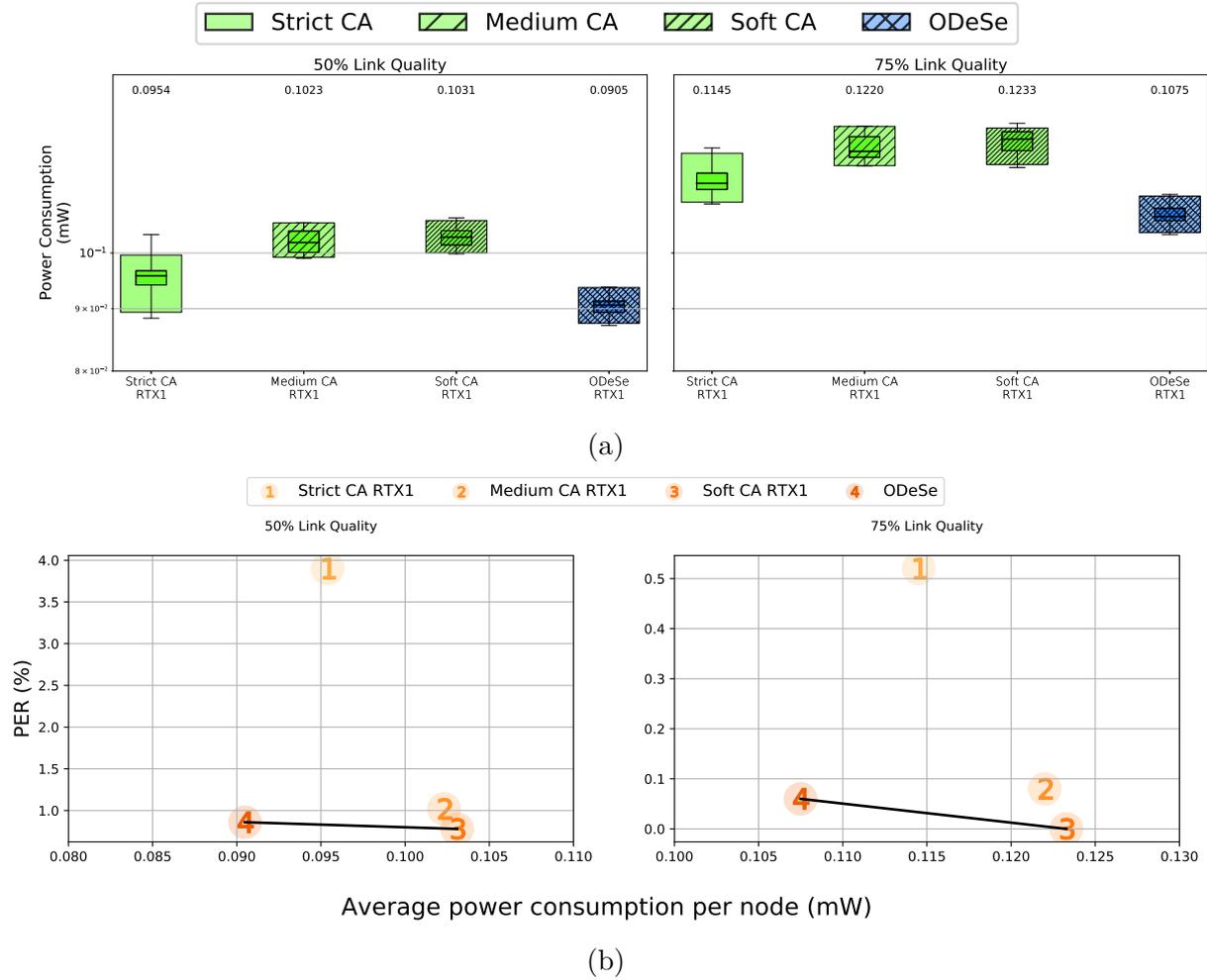


Figure 6.6 – (a) Average power consumption by CA and ODeSE algorithms and (b) Relationship between average power consumption and PER by algorithm. The line represents the Pareto’s frontier.

ber of relay nodes to transmit. Therefore it is important to optimize their consumption.

Given the above, let us focus on Figure 6.6. Figure 6.6a shows the average power consumption of the four algorithms, starting from DODAG’s completed fabrication until the last message was sent in the simulation. As it can be seen that ODeSe has a slight difference of 0.0049 mW with Strict CA in environments with 50% link quality and 0.0070 mW in environments with 75% link quality, proving to be the algorithm with the lowest power consumption.

Figure 6.6b, on the other hand, shows the relationship between the PER of each algorithm, and the average power consumption of them through a Pareto diagram. The black

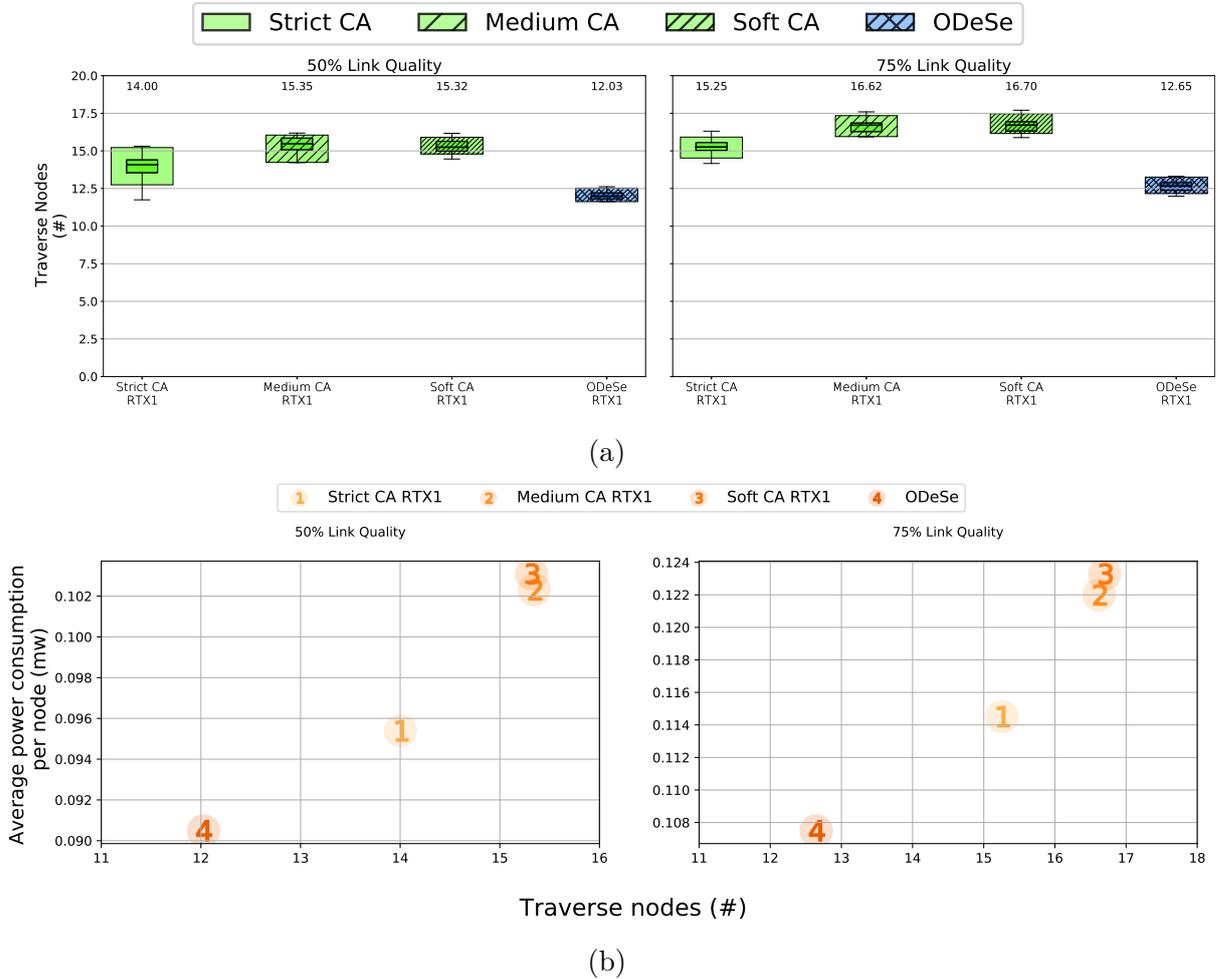


Figure 6.7 – (a) Average number of traverse nodes per transmission by CA and ODeSE algorithms and (b) Relationship between average power consumption and traverse nodes per transmission by algorithm. The line represents the Pareto’s frontier.

line represents the Pareto frontier which covers the algorithms with the best statistics in relation to the number of packets lost. From here it can be seen that ODeSe and Soft CA stand out with a PDR over 99%. Furthermore, it is shown that ODeSe has 12.2% less power consumption than Soft CA in environments with 50% link quality, and 12.8% in 75% link quality.

Finally, Figure 6.7 shows the relationship between power consumption and the number of average relay nodes per algorithm. As it can be observed in Figure 6.7a that ODeSe has an average of 12 relay nodes per transmission, which is the expected value for this algorithm given its selection policy (see Algorithm 2). Moreover in Figure 6.7b, we can

see that ODeSe has the lowest power consumption, being 5.1% lower than Strict CA in environments with 50% link quality, and 6.1% lower in environments with 75% link quality.

This demonstrates the advantages of ODeSe over CA algorithms.

6.5 Conclusions

In this chapter, the propagation problems of CA selection algorithms were addressed to introduce the ODeSe routing algorithm. More specifically, the propagation of CA algorithms consists in elaborating a redundant routing following the braided pattern principles. However, since these algorithms center their routing patterns on the PP, APs may tend to disperse. The goal of ODeSe is to set the same PP and AP to the nodes of the same distance to focus the transmission efforts towards 2 maximum nodes per hop.

To achieve this, ODeSe takes an on-demand policy every time a packet is transmitted. This consists of transporting the PP and AP addresses of a node's PP within the packet that this node is going to transmit to its PP and AP. In this way, the following nodes that will receive this message will select these transported PP and AP and then repeat the process.

As shown in the experiments, ODeSe has a high-reliability performance, reaching PDR values over 99% in environments with 50% and 75% link quality. In terms of delay and fluctuation, this algorithm has the same performance as the CA algorithms are given its braided propagation. Finally, in terms of power consumption, ODeSe has better performance than the CA algorithms since its selection policy concentrates all its transmission attempts towards the same direction, thus maximizing the resources provided by the PAREO functions.

Given the above, we believe that this algorithm meets the majority of the expectations established for migrating LLNs within the industrial networks.

CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this manuscript, different routing strategies were proposed to make LLN technologies meet the 4.0 industry's requirements. Since these technologies have low processing capability and high sensitivity to noise, the required routing methods are complex, as they must provide transmission reliability and, at the same time, ration their available resources. Because of this, the implementation of these routing strategies has been challenging since a balance must be maintained between their goals and the use of the resources available to accomplish them.

In Chapter 2, we introduced Ad-Hoc Networks and the requirements that they must meet to be used within the industrial networks. We also presented the proactive and the reactive routing protocols, demonstrating the adaptability in different applications and environments. Furthermore, RPL was also introduced in this chapter, proving that it is the de facto routing protocol for LLNs, due to its non-complicated manipulation of metrics within the OF and its convenient handling for new implementations. However, although RPL is a protocol focused on LLN technologies, its single-path transmission is not enough to cover the industry specifications. In view of this, we chose to extend RPL with multi-path in order to improve transmission reliability by replicating a packet by different paths.

In Chapter 3, we have presented the functions and requirements for extending RPL with multi-path. Here we learned of three types of redundant multipath patterns called disjoint, triangular and braided patterns. The main characteristic that encloses these patterns is the concentration of relay opportunities towards the root node thanks to the capability of providing merge paths along the route, and depending on the algorithm, these merges will act in different ways and appear in different quantities.

The second part consists of the PAREO functions, which provide the necessary features to extend the RPL to multi-path routing protocol. PAREO is made up of the RE, ARQ

and OH functions where the RE goal allow the transmission of several replicas of a packet in different paths; ARQ function delivers a controlled number of RTX for each failed transmission attempt; and OH function takes advantage of the wireless network conditions so that a node can annex a reception opportunity without the packet belonging to it. These functions were used throughout the entire research.

In Chapter 4, we introduced the first multi-path algorithm called N-disjoint, which is characterized by increasing the reliability of RPL by taking advantage of the disjoint pattern, in conjunction with the RE and ARQ functions. Moreover, given the conditions of the LLNs and the characteristics of the disjoint pattern, we established two types of selection strategies, the Basic and Advanced strategy, to highlight the different options of how to implement this algorithm.

The Basic strategy allows links to merge into nodes that are not necessarily the root node, keeping RE at the source. The Advanced strategy extends the methodology of the metric strategy allowing a node to perform the same number of RE replications as links that merge in it. Besides, each n-disjoint strategy has a certain number of ARQ RTXs. From the experiments we learn that ARQ consumes most of the power used and tends to extend the delay. Moreover, these behaviors intensify as the quality of the link deteriorates. However, this consumption can be balanced by reducing the number of RTX and complementing them with a certain amount of replication.

In Chapter 5, we established three algorithms based on the braided pattern. To do so, we established the CA selection, which consists of allowing a node to have an AP if this AP has from its PS a node in common with the PS of the PP. Under this condition, the Strict CA algorithm allows selecting an AP if the candidate node has the same PP as the PP of the node. Medium CA allows a node to have an AP if this AP has within PS the PP's PP. Finally Soft CA, selects an AP, if it has within its PS a node intersecting with the PS of PP.

The performance evaluation of the CA selection shows that all its algorithms provide transmission reliability since they reach PDR over 95% in all case studies. Moreover, it is observed that the algorithms with a higher probability of finding APs obtain better results in terms of PDR. However, the selection policies that favor the chances of finding a AP also contribute to the flooding of the network. Consequently, these algorithms tend to consume more energy.

This led us to our Chapter 6, where the deficiencies of the CA algorithms are presented and addressed with the ODeSe algorithm. According to the ODeSe algorithm, the CA

selection policy is maintained but with the differences that nodes of the same hop distance have to select the same PP, as well as the same AP. Consequently, a minimum node routing spread is maintained. The ODeSe results show network reliability close to 100%, as well as low jitter and delay, while the power consumption values remain close to the Strict CA.

7.2 Perspectives

Throughout this research, we have defined and developed different routing algorithms to further integrate LLNs in industrial networks. Hereafter, we present the potential future work that we have considered for multi-path routing protocols.

7.2.1 MAC layer and scheduling protocols

Throughout this research, we have employed a centralized TSCH schedule to conduct the transmissions. Thus, each node involved in the network have the same number of transmitting and receiving TSCH cells. The objective of this is to maintain a uniform condition to each node to provide a fair scenario to all studied algorithms. However, since RPL is a protocol that works independently of the other network layers, the resulting routing can take different directions, as well as provide different routes with variable hop distances.

Therefore, RPL requires a distributed schedule that can predict the behavior of this protocol. Moreover, this scheduling must be able to detect new members of a DODAG and be able to detect the action of potential mobile nodes. Therefore, a reactive TSCH schedule with cells that can quickly adapt to the network changes is required for multi-path routing protocols.

7.2.2 PAREO functions in reactive routing protocols

In the course of this research, we studied the performance of PAREO functions in multi-path algorithms using the RPL routing protocol. In addition, we observed that RPL routing can behave in a reactive manner thanks to the ODeSe algorithm.

Knowing the impact of PAREO functions in terms of reliability in multi-path routing, and given that it works in all the studied algorithms, the implementation of multi-path in reactive routing protocols such as LOADng is an interesting approach. Therefore, it would be interesting to extend this research to other routing protocols specialized in LLN.

7.2.3 Topology and experimenting environments

A pre-defined and a static topology was employed for the performance evaluation campaign to highlight the behavior of the different proposed routing algorithms. However, it would be interesting to see how these routing protocols can operate under different conditions, such as random topology or networks with mobile nodes.

Finally, although the RPL protocol was used in simulated environments, it is necessary to replicate the conditions studied in real-world conditions to corroborate the impact of multi-path algorithms in RPL.

7.2.4 Cooja simulator and multi-path RPL

Regarding the simulations, the Cooja simulator provided by the Contiki-os was employed. By default, this operating system lacks a default multi-path implementation for RPL.

In addition, it does not have a scalable implementation for modifications of a data packet header, more specifically the hop-by-hop packet. This is because this section is dedicated only to the RPL hop-by-hop option, not allowing the integration of other hop-by-hop options.

Furthermore, ETX computing performed by Contiki in simulated environments does not use probabilistic models at all.

Therefore, considering these previously presented issues, it is essential to develop a standard implementation in Contiki OS in order to meet these issues.

Appendices

RÉSUMÉ EN FRANÇAIS: VERS UNE LATENCE FIABLE ET LIMITÉE POUR L'INTERNET DES OBJETS

A.1 Contexte général

L'Internet des Objets (IoT) [Gub+13] est un ensemble de normes et de technologies développées afin de permettre à des objets intégrés et intelligents de se connecter à l'Internet, et qui se caractérisent par leur faible coût de production, leur grande adaptabilité à différents environnements et leur facilité de développement. Ces caractéristiques ont permis aux technologies IoT d'être massivement intégrées dans divers systèmes et applications, tels que le smart grid ou l'industrie. En réponse à cela, elles ont dû s'adapter à tous les environnements, tout en apportant des réponses rapides aux défaillances rencontrées. Ainsi, au cours de la dernière décennie, une migration constante vers les technologies sans fil a pu être constatée, celles-ci permettant une installation positionnelle flexible [Can+18].

L'industrie a récemment commencé à adopter les IoT dans ses chaînes de production, évoluant ainsi vers l'Industrie 4.0 [Lu17]. Un tel concept consiste à incorporer des appareils dans un réseau à la fois câblé et sans fil, afin de maintenir les différents systèmes de l'industrie de façon autonome grâce à la cohérence des communications qu'un appareil possède avec les autres nœuds de sa zone de travail. De plus, il permet une centralisation potentielle de la chaîne de production puisqu'un nœud IoT peut agir en tant qu'administrateur pour les autres appareils. Par conséquent, la production tend à s'automatiser davantage et, ainsi, à devenir moins coûteuse.

En adoptant l'approche de l'Industrie 4.0, nous couvrons différents concepts qui appartiennent à l'IoT, le WSN étant l'un des plus importants sur ce marché. Ces réseaux sont constitués de centaines de nœuds équipés de multiples capteurs permettant de surveiller les conditions physiques ou environnementales. Leur distribution se fait vers un nœud

centralisé, appelé la passerelle ou le puits. Ainsi, ces nœuds fonctionnent en coopération pour transférer leurs données vers le nœud centralisé. Cependant, le problème rencontré avec ces technologies est leur capacité limitée en termes de CPU et de mémoire, ainsi que leurs limites en termes de technologies sans fil.

Il est communément admis que les réseaux câblés maintiennent une communication stable entre les dispositifs de capteurs, tandis que les réseaux sans fil sont exposés à des conditions environnementales qui entraînent des interférences et des collisions de communication. Ces limitations sont actuellement atténuées par des systèmes prédictifs, en plus des processus intelligents qui assurent la robustesse de la communication. Malheureusement, pour qu'ils fonctionnent correctement, des systèmes ayant un niveau de traitement élevé sont nécessaires. De plus, en raison de la mobilité que permettent les technologies sans fil, de nombreux appareils sont alimentés par des piles, ce qui limite leur durée de vie.

Plusieurs approches ont été faites pour supporter ces technologies, l'une de ces approches étant la manipulation de l'acheminement. En effet, cette approche consiste à fournir un chemin en fonction des exigences de l'application, chemin qui sera ensuite utilisé pour acheminer la transmission d'un paquet. Néanmoins, pour rendre cela possible, différentes stratégies et décisions doivent être envisagées.

Comme mentionné dans [Pis+09], la topologie industrielle peut être constituée de 10 à 200 dispositifs interconnectés. De plus, dans les technologies de réseau intelligent, la topologie peut être constituée de milliers d'appareils [MZX18]. Cela peut avoir des répercussions sur les décisions des algorithmes de routage, puisqu'en effet plus le nombre de chemins vers la destination sera élevé, plus les informations à traiter seront nombreuses. En outre, étant donné la variabilité des réseaux sans fil, certains nœuds peuvent relayer des adresses non valides, ce qui rend nécessaire la détection des nœuds capables de fournir le routage correspondant. Pour cette raison, un système de routage doit garantir une communication rapide et sans perturbations entre les différents nœuds. De plus, certains systèmes nécessitant des échanges de communication rapides et précis, le protocole de routage doit garantir la fiabilité de la transmission car les pertes de paquets consécutives peuvent entraîner le dysfonctionnement de ces systèmes.

Ceci affecte considérablement les réseaux industriels, car ils exigent que les transmissions maintiennent un standard de qualité minimum. En effet, la QoS de l'industrie consiste à obtenir une grande fiabilité de transmission, une faible fluctuation de latence et une dépense énergétique optimisée. Les exigences mentionnées ci-dessus peuvent toute-

fois être satisfaites de différentes manières, comme le montre la publication [KK20]. Les auteurs expliquent ici qu’il existe plusieurs mécanismes stratégiques permettant de répondre à la QoS des cas d’utilisation industrielle. Cependant, chaque mécanisme ayant des comportements différents, leurs conditions sont également différentes, ce qui nécessite des compromis variés. Pour clarifier ce point, nous prendrons comme exemple les mécanismes sélectifs de différenciation et de regroupement des services.

La différenciation des services consiste à équilibrer efficacement un réseau limité de ressources sur la base d’un ou plusieurs critères. Cependant, cette différenciation des services nécessite un mécanisme qui détecte l’allocation dynamique des priorités ainsi que les conditions dynamiques du réseau.

Le regroupement, ou clustering, quant à lui, divise le réseau en petits clusters, chacun ayant une tête de cluster. Par conséquent, les nœuds concentreront leur transmission uniquement sur leurs têtes de cluster, tandis que les têtes de cluster concentreront leurs efforts de transmission sur le root. Cependant, étant donné la quantité de trafic devant passer par les têtes de cluster, les clusters épuisent leur énergie résiduelle plus rapidement.

Les deux exemples précités peuvent être utilisés pour améliorer la même exigence de l’industrie, mais leurs stratégies différentes indiquent que leurs trade-offs seront différents. Par conséquent, il est nécessaire de mettre en œuvre une stratégie qui s’adapte aux besoins d’un système sans causer un impact considérable sur les ressources du réseau. Dans ce manuscrit, nous avons l’intention d’aborder les exigences suivantes, celles-ci ayant pour finalité de garantir la qualité de service de l’industrie :

- Assurer un haut niveau de fiabilité du réseau;
- Maintenir une faible variation de délai (c’est-à-dire jitter);
- Optimiser la latence du réseau;
- Maintenir une consommation électrique faible et équilibrée.

A.2 Contexte de la thèse

Le LLN [Gha+18] désigne une catégorie de réseaux câblés et sans fil dont les nœuds sont très limités en termes de ressources et de technologies de communication. Leurs ressources limitées comprennent des réserves d’énergie, ainsi que des capacités de traitement et de stockage restreintes. Les LLN ont également un faible débit de données, des caractéristiques de liaison très asymétriques et des distances de communication courtes. Cependant, leur faible coût les rend attractifs pour le marché. L’un des composants qui

caractérisent le LLN est le saut, qui représente les différentes connexions d'une route entre les appareils d'un réseau. Il existe des LLN ayant des structures à plusieurs sauts comme les WSN, ou des LLN ayant des structures à un seul saut comme les LPWAN [Far18], ainsi que d'autres technologies IoT.

C'est pourquoi l'IETF a défini le protocole de routage RPL. Ce protocole de routage se caractérise par le fait qu'il ne dépend pas des autres couches du réseau qui lui sont rattachées, ainsi que par le fait qu'il fournit une méthode de routage adaptable à différents environnements réseau. En outre, il dispose d'un mécanisme de réparation des routes qui consiste à rediriger le trafic vers un autre nœud valide si un nœud de la route d'origine cesse de fonctionner ou parce que le nouveau nœud satisfait plus efficacement les besoins de routage établis. En conséquence, RPL est devenu l'un des protocoles de routage les plus populaires pour les LLN. Cependant, comme la mise en œuvre par défaut de ce protocole est le routage à une seule route, la fiabilité de bout en bout est généralement faible pour les réseaux industriels.

Au sein d'un LLN, la fiabilité de la transmission est généralement directement liée à la distance totale de saut qu'un paquet doit parcourir. En effet, la source étant située plus loin du root, le paquet est immergé dans un plus grand nombre de sauts, qui sont à leur tour soumis à des conditions sans fil. De plus, si l'un des nœuds de relais tombe soudainement en panne, la procédure de réparation RPL peut provoquer une augmentation significative du délai de transmission puisqu'il doit trouver une autre route.

Une solution standard à ce problème consiste à mettre en place plusieurs tentatives de transmission [FW02], ce qui donne plus de chances à un paquet d'effectuer une transmission réussie. Cependant, plus le nombre de tentatives de transmission est élevé, plus le retard et les fluctuations de performance sont importants.

Une autre solution possible est l'utilisation de plusieurs routes pour transmettre un seul paquet. Plus précisément, broadcast un paquet par plusieurs chemins au lieu d'un seul. Cependant, une diffusion non contrôlée peut entraîner une surcharge de trafic et une inondation. De plus, plus les appareils sont utilisés, plus leur espérance de vie sera courte. Par conséquent, il est nécessaire que la diffusion soit faite à des parents stratégiquement ciblés.

Dans ce manuscrit, nous nous concentrons sur différents outils et méthodologies pour étendre RPL avec multi-chemin pour assurer la fiabilité de la transmission, tout en maintenant une consommation électrique acceptable. De plus, nous fournissons une étude et une analyse de l'œuvre connexe dans l'état de l'art pour donner un aperçu du problème.

A.3 Contributions de thèse

Ce manuscrit vise à fournir des fonctions de routage redondantes et des algorithmes pour étendre le protocole de routage RPL multivoie. Notre objectif est de générer des algorithmes de routage pour les LLN qui garantissent la fiabilité de la transmission, une latence stable et une consommation électrique équilibrée pour une utilisation dans l'environnement de l'Industrie 4.0.

Nous commençons par les fonctions PAREO, dont le but est de fournir les outils nécessaires afin de prendre en charge le multitrajet en RPL. La fonction RE consiste à permettre à un nœud source de faire plusieurs copies d'un paquet ou des répliques vers les différents nœuds voisins que la source a sélectionnés lors de son routage pour effectuer une transmission. La fonction ARQ, quant à elle, fournit un nombre maximum de retransmissions à un nœud afin de les utiliser chaque fois qu'un paquet ne peut pas faire de saut. Enfin, la fonction OH permet à un nœud voisin de recevoir le même message d'un nœud récepteur si ce dernier est à portée du nœud émetteur.

Quant aux algorithmes multi-chemin, nous avons l'algorithme N-Disjoint qui consiste à exploiter les bases d'un motif disjoint en utilisant les fonctions de PAREO RE et ARQ. Son but est d'assurer une transmission fiable tout en maintenant un équilibre entre le nombre de retransmissions et le nombre de répliques.

L'algorithme CA implique l'élaboration de plusieurs chemins redondants, qui sont fusionnés de manière répétée avec le routage afin de concentrer un grand nombre de tentatives de retransmission vers un saut. Cependant, pour y parvenir, un nœud doit transmettre sa transmission à des nœuds stratégiques afin de maintenir un nombre élevé de fusions de chemins. Compte tenu de cela, il y a une probabilité de ne pas obtenir un parent approprié. Par conséquent, CA établit trois algorithmes : Strict CA, Medium CA, et Soft CA, chacun d'eux ayant une politique de sélection différente basée sur la probabilité de trouver un nœud approprié pour l'algorithme. L'objectif des algorithmes CA est d'assurer la fiabilité de la transmission en maintenant un nombre minimum de retransmissions.

Enfin, l'algorithme ODeSe adopte les mêmes principes que l'algorithme CA mais avec la différence qu'il les étend de manière on-demand à chaque nœud où un paquet transite. L'objectif de l'algorithme ODeSe est d'assurer la fiabilité des transmissions en maintenant un nombre minimum de retransmissions et en conservant une consommation d'énergie optimisée.

A.4 Modèles de sélection

Afin de comprendre le fonctionnement des différents algorithmes proposés, nous devons nous pencher sur les modèles de sélection d'itinéraire sur lesquels ces algorithmes sont basés. Tout d'abord, nous avons le modèle disjoint, qui se caractérise par deux ou plusieurs chemins disjoints partant du nœud source et fusionnant avec le nœud de destination. Sa complexité réside dans la source puisqu'elle est responsable de la sélection des multiples parents pour effectuer la transmission. En revanche, les nœuds de transmission ne peuvent sélectionner qu'un seul parent.

Ensuite, nous avons le modèle triangulaire qui, contrairement au modèle de disjoint, donne la priorité à la différenciation d'un PP d'un AP. La sélection consiste à permettre à un nœud de sélectionner un AP si ce nœud a le même parent que le PP, cependant, un AP ne peut pas sélectionner un autre AP. Par conséquent, un PP peut avoir plusieurs parents, et ces parents ont à leur tour un nœud parent commun.

Enfin, nous avons le modèle braided, qui utilise la même méthodologie de sélection de chemin que le modèle triangulaire, avec une extension selon laquelle les AP peuvent aussi avoir leurs propres AP. Les avantages de ce modèle résident dans une meilleure probabilité d'obtenir un AP, car une plus grande gamme de nœuds peut être utilisée pour ce modèle, ce qui permet de générer un plus grand nombre de chemins par rapport au modèle triangulaire.

A.5 Algorithme N-Disjoint

Le modèle disjoint, contrairement aux modèles triangulaires et braided, est exposé à des pertes fréquentes car sa stabilité dépend de la distance entre la source et le puits. En effet, chaque nœud de retransmission renvoie le paquet par un chemin unique. Par conséquent, ses multiples liens sélectionnés à partir de la source sont indépendants les uns des autres, héritant des mêmes inconvénients de la route unique. Cependant, bien que cette politique de sélection comporte des inconvénients, elle est attrayante en raison de sa simplicité, et parce qu'elle maintient un profil de faible consommation d'énergie.

En tenant compte des limites de ce schéma, nous avons établi l'algorithme N-Disjoint, qui vise à maintenir un réseau fiable en prenant en compte certaines des caractéristiques du modèle disjoint. De plus, cet algorithme se comporte selon deux types de stratégies de sélection, qui sont détaillées ci-dessous.

A.5.1 Stratégie de base

Cette stratégie consiste à autoriser RPL à poursuivre son routage sans nécessairement maintenir un motif disjoint mais en conservant les performances de la multitraiteance uniquement dans le nœud source. Par conséquent, le routage des nœuds de relais sera soumis à la sélection parentale accordée par RPL, permettant aux nœuds de même distance de saut d'être reliés au même parent, sans que celui-ci soit nécessairement le nœud de destination. Cette stratégie de sélection fonctionne en permettant au seul nœud source de choisir plusieurs liens pour transmettre plusieurs copies d'un paquet. D'autre part, les nœuds relais utilisent le routage RPL standard.

Par conséquent, deux ou plusieurs liens peuvent être fusionnés avant d'atteindre le nœud racine, ce qui réduit le nombre de liens qui atteignent le nœud racine. Cependant, alors que le nombre de liens est réduit, ces liens fusionnés recevront les tentatives des nœuds précédents, ce qui augmente les chances de faire le prochain saut.

A.5.2 Stratégie avancée

Cette stratégie consiste à prendre les politiques sélectives de la stratégie de base mais avec la différence d'autoriser certains nœuds à accéder à des liens multiples, sans que ceux-ci ne soient nécessairement le nœud source. De cette façon, la fiabilité de la transmission est améliorée.

Cette stratégie a été développée en tenant compte du fait que lorsque plusieurs liens sont fusionnés avant d'atteindre la destination finale, la fiabilité de la transmission est limitée au nombre total de chemins disjoints qui atteignent le nœud de destination.

Cette stratégie atténue ce problème en établissant que les nœuds peuvent envoyer des copies d'un paquet au même nombre de parents que les copies reçues de ce paquet. De cette façon, le nombre total de miroirs ne sera pas affecté par la fusion de liens et se déroulera normalement.

A.6 Les algorithmes d'ancêtres communs (CA)

Pour étendre RPL avec des trajets multiples et la diriger pour réaliser un modèle braided, nous avons appliqué la sélection de modèle CA. Cette méthode de sélection consiste à permettre à un nœud de trouver un AP si ce possible candidat a dans son PS un nœud mutuel avec le PS du PP du nœud, c'est-à-dire si un nœud A a comme PP un

nœud B, et si un nœud C a dans son PS un nœud en commun avec le PS de B, alors si le nœud C appartient au PS A, ce nœud peut être sélectionné comme AP de A. De cette façon, un motif tressé peut être formé, en commençant à la source et en finissant au puits.

Nous avons défini trois algorithmes pour sélectionner un AP dans la sélection CA. Chacun d'entre eux a des politiques de sélection différentes qui affectent directement à la fois la probabilité d'obtenir un AP et le nombre de nœuds utilisés pour propager un paquet vers le nœud récepteur.

A.6.1 Strict CA

Cet algorithme permet de sélectionner un nœud candidat v comme AP à partir d'un nœud u si le PP du PP de u est le même que le PP du candidat AP, c'est-à-dire si $PP(PP(u)) = PP(v)$. Compte tenu de cela, cet algorithme exige qu'un nœud candidat partage le même PP avec le PP actuel.

A.6.2 Medium CA

Cet algorithme permet de trouver un AP si un nœud candidat a dans son ensemble de parents le PP du PP du node. En d'autres termes, un candidat AP v peut être choisi comme AP d'un nœud u si le PP du PP du nœud appartient au PS_{MC} du candidat AP, c'est-à-dire si $PP(PP(u)) \in PS_{MC}(v)$. Pour ce faire, nous avons besoin du nombre fixe d'adresses M contenues dans le PS_{MC} .

A.6.3 Soft CA

Cet algorithme permet de sélectionner comme AP les nœuds qui ont un nœud commun avec ceux de l'ensemble parent du PP. Plus précisément, un candidat AP v peut être sélectionné comme AP à partir d'un nœud u si le PS_{MC} du PP du nœud présente un certain chevauchement avec le PS_{MC} du candidat AP, c'est-à-dire si $PS_{MC}(PP(u)) \cap PS_{MC}(v) \neq \emptyset$.

A.7 Algorithme de sélection on-demand (ODeSe)

Si les algorithmes Strict, Medium et Soft CA fonctionnent bien, la plus grande fiabilité du réseau est obtenue avec le Soft CA, mais au prix d'une consommation d'énergie plus

élevée en raison d'une plus forte inondation. ODeSe vise à maintenir un haut niveau de fiabilité de transmission tout en conservant une faible consommation d'énergie. La principale différence entre les algorithmes CA et ODeSe est que dans ODeSe, chaque nœud décide non seulement de ses propres parents d'expédition (PP et AP), mais aussi des PP et AP à utiliser lors du prochain saut vers la destination. En d'autres termes, chaque nœud dispose d'un contrôle d'expédition à deux sauts. Pour ce faire, l'ODeSe utilise l'en-tête "Hop by Hop" pour transporter la direction des nœuds qui correspondent le mieux à la sélection des parents pour le prochain saut.

A.8 Conclusions

Dans ce manuscrit, différentes stratégies de routage ont été proposées pour que les technologies LLN répondent aux exigences de l'Industrie 4.0. Comme ces technologies ont une faible capacité de traitement et une grande sensibilité au bruit, les méthodes de routage requises sont complexes, car elles doivent assurer la fiabilité de la transmission tout en rationnant les ressources disponibles. C'est pourquoi la mise en œuvre de ces stratégies de routage a été un défi, car il faut maintenir un équilibre entre leurs objectifs et l'utilisation des ressources disponibles pour les atteindre.

Au chapitre 2, nous avons présenté les réseaux ad hoc et les exigences qu'ils doivent réunir pour être utilisés au sein des réseaux industriels. Nous avons également présenté les protocoles de routage proactif et réactif, démontrant ainsi l'adaptabilité de différentes applications et environnements. En outre, RPL a également été introduite dans ce chapitre, prouvant qu'il s'agit du protocole de routage de facto pour les LLN, en raison de sa manipulation non compliquée des mesures au sein de l'OF et de sa manipulation pour les nouvelles mises en œuvre. Toutefois, bien que RPL soit un protocole axé sur les technologies LLN, sa transmission par voie unique n'est pas suffisante pour couvrir les spécifications de l'industrie. Dans cette optique, nous avons choisi d'étendre RPL avec multi-chemins afin d'améliorer la fiabilité de la transmission en répliquant un paquet.

Dans le chapitre 3, nous avons présenté les fonctions et les conditions requises pour étendre RPL à plusieurs voies. Nous y avons appris l'existence de trois types de modèles redondants à trajets multiples appelés modèles disjoints, triangulaires et tressés. La principale caractéristique qui englobe ces modèles est la concentration des possibilités de relais vers le nœud racine grâce à la capacité de fournir des chemins de fusion le long du parcours, et selon l'algorithme, ces fusions agiront de différentes manières et apparaîtront

en quantités différentes.

La deuxième partie est constituée des fonctions PAREO, qui fournissent les caractéristiques nécessaires pour étendre RPL au protocole de routage à trajets multiples. PAREO est composé des fonctions RE, ARQ et OH, où l'objectif RE permet la transmission de plusieurs répliques d'un paquet sur différents chemins; la fonction ARQ fournit un nombre contrôlé de RTX pour chaque tentative de transmission échouée; et la fonction OH tire parti des conditions du réseau sans fil pour qu'un nœud puisse annexer une possibilité de réception sans que le paquet lui appartienne. Ces fonctions ont été utilisées tout au long de la recherche.

Au chapitre 4, nous avons présenté le premier algorithme à trajets multiples appelé N-disjoint, qui se caractérise par l'augmentation de la fiabilité de RPL en tirant parti du modèle disjoint, en conjonction avec les fonctions RE et ARQ. De plus, étant donné les conditions des LLN et les caractéristiques du modèle disjoint, nous avons établi deux types de stratégies de sélection, la stratégie de base et la stratégie avancée, pour mettre en évidence les différentes options de mise en œuvre de cet algorithme.

La stratégie de base permet de fusionner les liens en des nœuds qui ne sont pas nécessairement le nœud racine, en gardant l'RE à la source. La stratégie avancée étend la méthodologie de la stratégie métrique en permettant à un nœud d'effectuer le même nombre de réplifications d'RE que les liens qui y fusionnent. En outre, chaque stratégie N-disjointe comporte un certain nombre de RTX ARQ. Les expériences nous apprennent que l'ARQ consomme la plus grande partie de la puissance utilisée et tend à prolonger le délai. De plus, ces comportements s'intensifient à mesure que la qualité du lien se détériore. Cependant, cette consommation peut être équilibrée en réduisant le nombre de RTX et en les complétant avec un certain degré de réplification.

Au chapitre 5, nous avons établi trois algorithmes basés sur le modèle tressé. Pour ce faire, nous avons établi la sélection CA, qui consiste à permettre à un nœud de sélectionner un AP si cet AP a, à partir de son PS, un nœud en commun avec le PS de la PP. Dans cette condition, l'algorithme de CA strict permet de sélectionner un AP si le nœud candidat a le même PP que le PP du nœud. Le CA moyen permet à un nœud d'avoir un AP si cet AP a à l'intérieur du PS le PP du PP. Enfin, le CA souple sélectionne un AP, s'il a dans son PS un nœud croisant le PS du PP.

L'évaluation des performances de la sélection de CA montre que tous ses algorithmes offrent une fiabilité de transmission puisqu'ils atteignent un taux de PDR supérieur à 95% dans toutes les études de cas. De plus, on observe que les algorithmes ayant une

plus grande probabilité de trouver les AP obtiennent de meilleurs résultats en termes de PDR. Cependant, les politiques de sélection qui favorisent les chances de trouver un AP contribuent également à l'inondation du réseau. Par conséquent, ces algorithmes ont tendance à consommer davantage d'énergie.

Cela nous a conduit à notre chapitre 6, où les lacunes des algorithmes de CA sont présentées et traitées avec l'algorithme ODeSe. Selon l'ODeSe la politique de sélection des CA est maintenue, mais avec les différences que les nœuds de la même distance de saut doivent sélectionner le même PP, ainsi que le même AP. Par conséquent, un étalement minimum du routage des nœuds est maintenu. Les résultats d'ODeSe montrent une fiabilité du réseau proche de 100%, ainsi qu'une faible jitter et un faible retard, alors que les valeurs de consommation d'énergie restent proches du CA strict.

BIBLIOGRAPHY

- [10] *Z1 Datasheet*, Z1, Zolertia, 2010, URL: http://zolertia.sourceforge.net/wiki/images/e/e8/Z1_RevC_Datasheet.pdf.
- [20] *IEEE Standard for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Std 802.15.4-2020, May 2020.
- [Arm+16] J. d. Armas, P. Tuset, T. Chang, F. Adelantado, T. Watteyne, and X. Vilajosana, « Determinism through Path Diversity: Why Packet Replication Makes Sense », *in: Proceedings of the International Conference on Intelligent Networking and Collaborative Systems (INCoS)*, Ostrawva, Czech Republic, Sept. 2016.
- [Can+18] J. C. Cano, V. Berrios, B. Garcia, and C. K. Toh, « Evolution of IoT: An Industry Perspective », *in: IEEE Internet of Things Magazine* 1 (2018), pp. 12–17.
- [Cen+19] N. Cen, J. Jagannath, S. Moretti, Z. Guan, and T. Melodia, « LANET: Visible-light ad hoc networks », *in: Ad Hoc Networks* 84 (2019), pp. 107–123.
- [CM99] S. Corson and J. Macker, *Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*, RFC 2501, IETF, Jan. 1999, pp. 1–12.
- [CYH17] T. Clausen, J. Yi, and U. Herberg, « Lightweight On-demand Ad hoc Distance-vector Routing - Next Generation (LOADng): Protocol, extension, and applicability », *in: Computer Networks* 126 (2017), pp. 125–140.
- [DB17] F. Drouhin and S. Bindel, « Routing and Data Diffusion in Vehicular Ad Hoc Networks », *in: Building Wireless Sensor Networks*, ed. by S. Femmam, Elsevier, 2017, chap. 3, pp. 67–96.

-
- [DLV13] S. Duquennoy, O. Landsiedel, and T. Voigt, « Let the Tree Bloom: Scalable Opportunistic Routing with ORPL », *in: Proceedings of the ACM Conference on Embedded Networked Sensor Systems (ACM Sensys)*, New York, NY, USA, Nov. 2013.
- [Far18] S. Farrell, *Low-Power Wide Area Network (LPWAN) Overview*, RFC 8376, IETF, May 2018, pp. 1–43.
- [Fot+07] M. Fotino, A. Gozzi, J.-C. Cano, C. Calafate, F. De Rango, P. Manzoni, and S. Marano, « Evaluating Energy Consumption of Proactive and Reactive Routing Protocols in a MANET », *in: Proceedings of the Wireless Sensor and Actor Networks (WSAN)*, Albacete, Spain, Sept. 2007.
- [FW02] G. Fairhurst and L. Wood, *Advice to link designers on link Automatic Repeat reQuest (ARQ)*, RFC 3366, IETF, Aug. 2002, pp. 1–27.
- [Gha+18] B. Ghaleb, A. Al-Dubai, E. Ekonomou, A. Alsarhan, Y. Nasser, L. Mackenzie, and A. Boukerche, « A Survey of Limitations and Enhancements of the IPv6 Routing Protocol for Low-power and Lossy Networks: A Focus on Core Operations », *in: IEEE Communications Surveys & Tutorials* 21 (2018), pp. 1607–1635.
- [GK12] O. Gaddour and A. Koubâa, « RPL in a nutshell: A survey », *in: Computer Networks* 56 (2012), pp. 3163–3178.
- [GL12] O. Gnawali and P. Levis, *The Minimum Rank with Hysteresis Objective Function*, RFC 6719, IETF, Sept. 2012, pp. 1–13.
- [GS19] M. A. Gawas and Govekar S. S., « A novel selective cross layer based routing scheme using ACO method for vehicular networks », *in: Journal of Network and Computer Applications* 143 (2019), pp. 34–46.
- [Gub+13] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, « Internet of Things (IoT): A vision, architectural elements, and future directions », *in: Future Generation Computer Systems* 29 (2013), pp. 1645–1660.
- [HF08] X. Huang and Y. Fang, « Multiconstrained QoS Multipath Routing in Wireless Sensor Networks », *in: Wireless Networks* 14 (2008), pp. 465–478.

-
- [Hwa+20] R.-H. Hwang, M.-C. Peng, C.-Y. Wu, and S. Abimannan, « A novel RPL-based multicast routing mechanism for wireless sensor networks », *in: International Journal of Ad Hoc and Ubiquitous Computing* 33 (2020), pp. 122–131.
- [ITN15] O. Iova, F. Theoleyre, and T. Noel, « Using multiparent routing in RPL to increase the stability and the lifetime of the network », *in: Ad Hoc Networks* 29 (2015), pp. 45–62.
- [JM19] J. Jagannath and T. Melodia, « VL-ROUTE: A Cross-Layer Routing Protocol for Visible Light Ad Hoc Network », *in: Proceedings of the IEEE International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Washington DC, USA, Apr. 2019.
- [KK20] T. Kaur and D. Kumar, « A survey on QoS mechanisms in WSN for computational intelligence based routing protocols », *in: Wireless Networks* 26 (2020), pp. 2465–2486.
- [Kou+18] R.-A. Koutsiamanis, G. Z. Papadopoulos, X. Fafoutis, J. M. D. Fiore, P. Thubert, and N. Montavont, « From Best Effort to Deterministic Packet Delivery for Wireless Industrial IoT Networks », *in: IEEE Transactions on Industrial Informatics* 14 (2018), pp. 4468–4480.
- [Kou+20] R.-A. Koutsiamanis, G. Z. Papadopoulos, N. Montavont, and P. Thubert, *Common Ancestor Objective Function and Parent Set DAG Metric Container Extension*, draft ietf-roll-nsa-extension-08, IETF, Mar. 2020, pp. 1–17.
- [Lag+18] T. Lagos Jenschke, R.-A. Koutsiamanis, G. Z. Papadopoulos, and N. Montavont, « Multi-path Selection in RPL Based on Replication and Elimination », *in: Proceedings of the International Conference on Ad-Hoc Networks and Wireless*, St. Malo, France, Aug. 2018.
- [Lev+11] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko, *The Trickle Algorithm*, RFC 6206, IETF, Mar. 2011, pp. 1–13.
- [LH09] G.-W. Lee and E.-N. Huh, « Reliable data transfer using overhearing for implicit ACK », *in: Proceedings of the ICROS-SICE International Joint Conference (ICCAS-SICE)*, Fukuoka, Japan, Aug. 2009.

-
- [LNM14] Q. Le, T. Ngo-Quynh, and T. Magedanz, « RPL-based multipath Routing Protocols for Internet of Things on Wireless Sensor Networks », *in: Proceedings of the International Conference on Advanced Technologies for Communications (ATC)*, Hanoi, Vietnam, Oct. 2014.
- [Lod+15] M. A. Lodhi, A. Rehman, M. M. Khan, and F. B. Hussain, « Multiple path RPL for low power lossy networks », *in: Proceedings of the IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, Bandung, Indonesia, Aug. 2015.
- [Lu17] Y. Lu, « Industry 4.0: A survey on technologies, applications and open research issues », *in: Journal of Industrial Information Integration* 6 (2017), pp. 1–10.
- [MKL17] P. Minet, I. Khoufi, and A. Laouiti, « Increasing reliability of a TSCH network for the industry 4.0 », *in: Proceedings of the IEEE International Symposium on Network Computing and Applications (NCA)*, Cambridge, MA, USA, Nov. 2017.
- [MNT19] E. Mozaffari Ahrar, M. Nassiri, and F. Theoleyre, « Multipath aware scheduling for high reliability and fault tolerance in low power industrial networks », *in: Journal of Network and Computer Applications* 142 (2019), pp. 25–36.
- [Moh+10] S. Mohseni, R. Hassan, A. Patel, and R. Razali, « Comparative review study of reactive and proactive routing protocols in MANETs », *in: Proceedings of the IEEE International Conference on Digital Ecosystems and Technologies (DEST)*, Dubai, United Arab Emirates, Apr. 2010.
- [MT14] M. N. Moghadam and H. Taheri, « High throughput load balanced multipath routing in homogeneous wireless sensor networks », *in: Proceedings of the Iranian Conference on Electrical Engineering (ICEE)*, Tehran, Iran, May 2014.
- [MZX18] S. Ma, H. Zhang, and X. Xing, « Scalability for Smart Infrastructure System in Smart Grid: A Survey », *in: Wireless Personal Communications* 99 (2018), pp. 161–184.
- [OG20] E. Oztemel and S. Gursev, « Literature review of Industry 4.0 and related technologies », *in: Journal of Intelligent Manufacturing* 31 (2020), pp. 127–182.

-
- [Pap+19] G. Z. Papadopoulos, R.-A. Koutsiamanis, N. Montavont, and P. Thubert, *Exploiting Packet Replication and Elimination in Complex Tracks in LLNs*, draft papadopoulos-raw-pareo-reqs-01, IETF, Dec. 2019, pp. 1–10.
- [Pat+14] D. N. Patel, S. B. Patel, H. R. Kothadiya, P. D. Jethwa, and R. H. Jhaveri, « A survey of reactive routing protocols in MANET », *in: Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES)*, Chennai, India, Feb. 2014.
- [PBD03] C. Perkins, E. Belding-Royer, and S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*, RFC 3561, IETF, July 2003, pp. 1–37.
- [Pis+09] K. Pister, P. Thubert, S. Dwars, and T. Phinney, *Industrial Routing Requirements in Low-Power and Lossy Networks Abstract*, RFC 5673, IETF, Oct. 2009, pp. 1–27.
- [SK14] R. Shenbagapriya and N. Kumar, « A survey on proactive routing protocols in MANETs », *in: Proceedings of the International Conference on Science Engineering and Management Research (ICSEMR)*, Chennai, India, Nov. 2014.
- [SVV15] A. Sgora, D. J. Vergados, and D. D. Vergados, « A Survey of TDMA Scheduling Schemes in Wireless Multihop Networks », *in: ACM Computing Survey* 47 (2015), p. 39.
- [Tan+16] 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 », *in: Journal of Sensors* 2016 (2016), pp. 1–11.
- [TDV14] J. Tripathi, J. C. De Oliveira, and J.P. Vasseur, « Proactive versus reactive routing in low power and lossy networks: Performance analysis and scalability improvements », *in: Ad Hoc Networks* 23 (2014), pp. 121–144.
- [Thu+17] P. Thubert, C. Bormann, L. Toutain, and R. Cragie, *IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Routing Header*, RFC 8138, IETF, Apr. 2017, pp. 1–37.
- [Thu12] P. Thubert, *Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)*, RFC 6552, IETF, Mar. 2012, pp. 1–14.
- [Usm18] M. Usman, « Analysis of the Impact of Different Parameter Settings on Wireless Sensor Network Lifetime », *in: International Journal of Advanced Computer Science and Applications* 9 (2018).

-
- [Vas+12] J. P. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, *Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks*, RFC 6551, IETF, Mar. 2012, pp. 1–30.
- [Vu +19] Q. Vu Khanh, N. Ban, V. Hoai Nam, D. Tuan, and N. Han, « Survey of Recent Routing Metrics and Protocols for Mobile Ad-Hoc Networks », *in: Journal of Communications* 14 (2019), pp. 110–120.
- [Win+12] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. P. Vasseur, and R. Alexander, *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, RFC 6550, IETF, Mar. 2012, pp. 1–157.
- [WPG15] 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*, RFC 7554, IETF, May 2015, pp. 1–23.
- [YWK05] Y. Yang, J. Wang, and R. Kravets, « Designing routing metrics for mesh networks », *in: Proceedings of the IEEE Workshop on Wireless Mesh Networks (WiMesh)*, Santa Clara, CA, United States, Sept. 2005.
- [ZWY17] L. Zhu, R. Wang, and H. Yang, « Multi-Path Data Distribution Mechanism Based on RPL for Energy Consumption and Time Delay », *in: Information* 8 (2017), p. 124.

Titre : Vers une latence fiable et limitée pour l'Internet des objets

Mot clés : RPL, Multitrajets, Ad-Hoc, LLNs, Déterminisme, PAREO

Résumé : Les Low Power and Lossy Network (LLN) sont des technologies Internet of Things (IoT) sans fil qui fonctionnent avec une puissance de traitement, une mémoire ou une puissance limitées. En outre, ils ont des liens caractérisés par des taux de perte élevés. Toutefois, en raison de leur faible coût et de leur facilité de manipulation, elles sont devenues populaires dans le industry 4.0. Par conséquent, pour que ces technologies puissent être intégrées dans l'industrie, elles doivent assurer une transmission fiable, rapide et stable. Le IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) est un protocole de routage vectoriel à distance spécialisé dans ces applications IoT. Son adaptabilité en a fait l'un des protocoles les plus po-

pulaires pour les LLNs. Cependant, sa transmission unidirectionnelle en amont n'est pas suffisante pour garantir la fiabilité de la transmission. Il en résulte un défi pour l'industrie et différentes fonctions et stratégies ont été proposées pour résoudre ce problème. Malheureusement, nombre de ces stratégies ne peuvent être reproduites pour différents environnements et nécessitent des compromis dans d'autres domaines. Dans cette thèse, notre objectif est de fournir une RPL personnalisée afin qu'elle puisse assurer la fiabilité de la transmission tout en maintenant un faible retard et une faible fluctuation. À cette fin, nous proposons différentes fonctions et algorithmes qui permettent l'extension des multitrajets en RPL.

Title: Toward Reliable and Bounded Latency for Internet of Things

Keywords: RPL, Multi-path, Ad-Hoc, LLNs, Determinism, PAREO

Abstract: The Low Power and Lossy Network (LLN) are wireless Internet of Things (IoT) technologies that operate with limited processing power, memory, or power. Furthermore, they have links characterized by high loss rates. However, due to their low cost and easy handling, they have become popular in the industry 4.0. Therefore, for these technologies to be incorporated into the industry, they need to ensure reliable, fast, and stable transmission. The IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) is a distance vector routing protocol specialized in these IoT applications. Its adaptability has made it one of the most popular protocols for LLNs. How-

ever, its one-way upstream transmission is not sufficient to guarantee transmission reliability. This has resulted in a challenge to the industry and different functions and strategies have been proposed to address this problem. Unfortunately, many of these strategies cannot be replicated for different environments and require trade-offs in other areas. In this thesis, our goal is to provide a customized RPL so that it can ensure transmission reliability while maintaining a low delay and fluctuation. For this purpose, we propose different functions and algorithms that allow the extension of multi-path in RPL.