DISSERTAÇÃO DE MESTRADO Nº 1055

INVESTIGATING MULTICRITERIA APPROACHES FOR THE ROUTING PROBLEM IN AD-HOC WIRELESS NETWORKS USING QOS-AWARE METRICS

Jean Nunes Ribeiro Araújo

DATA DA DEFESA: 21/05/2018

Universidade Federal de Minas Gerais

Escola de Engenharia

Programa de Pós-Graduação em Engenharia Elétrica

INVESTIGATING MULTICRITERIA APPROACHES FOR THE ROUTING PROBLEM IN AD-HOC WIRELESS NETWORKS USING QOS-AWARE METRICS

Jean Nunes Ribeiro Araújo

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do Título de Mestre em Engenharia Elétrica.

Orientador: Prof. Lucas de Souza Batista

Belo Horizonte - MG

Maio de 2018

Araújo, Jean Nunes Ribeiro.
A663i Investigating multicriteria approaches for the routing problem in ad-hoc wireless networks using QoS-aware metrics [manuscrito] / Jean Nunes Ribeiro Araújo. – 2018.
xii, 104 f., enc.: il.

Orientador: Lucas de Souza Batista.
Coorientador: Cláudio de Castro Monteiro.

Dissertação (mestrado) Universidade Federal de Minas Gerais, Escola de Engenharia.

Apêndices: f.85-96.

Bibliografia: f. 97-104.

1. Engenharia elétrica - Teses. 2. Otimização multicritério - Teses. 3. Roteamento (Administração de redes de computadores) - Teses. 4. Sistemas de comunicação sem fio - Teses. 5. Redes de sensores sem fio – Teses. I. Batista, Lucas de Souza. II. Monteiro, Cláudio de Castro. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título.
CDU: 621.3(043)

Abstract

Investigating Multicriteria Approaches for the Routing Problem in Ad-hoc Wireless Networks using QoS-aware Metrics

Ad-hoc Wireless Networks have aroused much interest of the scientific and business community in the last two decades. Sensor, vehicular, and mobile networks have evolved from this “ad-hoc” paradigm. Among the various emerging challenges of this research field, the provision of Quality of Service (QoS) is one of the most prominent, since these networks are prone to suffer from instabilities and interference in the wireless medium and frequent topology changes when mobility exists. Depending on application or scenario, the protocol needs to consider two or more QoS criteria when solving the routing problem. In this context, this work proposes to investigate if the use of multiple QoS-aware metrics can generate promising compromise solutions considering several network quality indicators in static and mobile Ad-hoc Wireless Networks. For that, a framework that supports several optimization objectives is developed to house the methods. Two new models are proposed - one based on weighted sum method with path size control and another based on compromise method (ϵ-constraint) with pruning mechanism and path size control - and compared with the standard weighted sum method. In order to map a single final solution, a utility function is proposed to choose the parameters (weights and constraints) of each method. In a network simulator, experiments are designed varying mobility, type of application, and packet generation rate. The following network quality indicators are measured: Packet Loss Ratio (PLR), Throughput, End-to-End Delay (E2ED), Network Lifetime (NLT), Normalized Routing Load (NRL), and Packet Error Rate (PER). The results show the proposed methods were more efficient in generating better trade-off solutions and promoting significant improvements in the quality indicators in most scenarios investigated, indicating that these approaches are promising and deserve to be further studied in future works.

Resumo

Investigando Abordagens Multicritério para o Problema de Roteamento em Redes Sem Fio Ad-hoc usando métricas conscientes de QoS

As redes sem fio ad-hoc têm despertado muito interesse nas comunidades científicas e empresariais nas últimas duas décadas. Redes de sensores, redes veiculares e redes móveis têm evoluído a partir deste paradigma "ad-hoc". Dentre os vários desafios deste campo de pesquisa, o fornecimento de Qualidade de Serviço (QoS) é um dos mais proeminentes, uma vez que essas redes são propensas a sofrer com instabilidades e interferências no meio sem fio e frequentes mudanças de topologia quando existe mobilidade. Dependendo da aplicação ou do cenário, o protocolo precisa considerar dois ou mais critérios de QoS ao resolver o problema de roteamento. Neste contexto, este trabalho propõe investigar se o uso de múltiplas métricas conscientes de QoS pode gerar soluções de compromisso promissoras considerando vários indicadores de qualidade em redes sem fio ad-hoc estáticas e móveis. Para isso, um framework que suporta vários objetivos de otimização é desenvolvido para abrigar os métodos propostos. Dois novos modelos são propostos - um baseado no método de soma ponderada com controle do tamanho do caminho e outro baseado no método de compromisso (ϵ-restrito) com mecanismo de poda e controle do tamanho do caminho - e comparados com o método de soma ponderada tradicional. Para mapear uma única solução final, uma função de utilidade é proposta para escolher os parâmetros (pesos e restrições) de cada método. Em um simulador de rede, projetou-se experimentos em cenários de rede que variam em termos de mobilidade, tipo de aplicação e taxa de geração de pacotes. Os seguintes indicadores de qualidade de rede são medidos: taxa de perda de pacotes, vazão, atraso fim a fim, tempo de vida da rede, carga de roteamento normalizado e taxa de erro de pacotes. Os resultados mostram que os métodos propostos foram mais eficientes em gerar melhores soluções de compromisso e promover melhorias significativas nos indicadores de qualidade na maioria dos cenários investigados, indicando que essas abordagens são promissoras e merecem ser mais estudadas em trabalhos futuros.

Acknowledgements

I praise God for the valuable opportunity to complete this significant stage in my life. Thank you, Lord, for not letting me be discouraged at any point in this trajectory.

I would like to thank my advisor, professor Lucas de Souza Batista, for all the support, availability, and serenity in the lead of this work. I would also like to say thank to my co-advisor, Claudio de Castro Monteiro, who also accompanied me as advisor and master, since the time of the technical course at the IFTO. I feel honored to have you as mentors and more excellent examples of conduct.

In particular, I want to thank my wife, Tatiane. When I came up with the idea of leaving Palmas and taking a master's degree in BH, instead of just saying, "This is crazy!!!", she said: "This is crazy!!! But, I'm with you...". Thank you for sharing this time with me.

In those two years, beyond the priceless learning, I had the company of wonderful people who shared with me a little of the joys and apprehensions of that journey. Thank you guys from the ORCS (Odd, Random, and no-Context Subjects) lab :). Special thanks to the Fillipe Goulart for helping me with English.

I thank my friends from the Presbyterian Church of Palmas who, even from a distance, have always been praying and cheering for us.

Finally, I thank my lovely parents, Jose Maria and Doracy, who have always encouraged me and have spared no effort to support my studies trajectory. This work is dedicated to you.

Contents

List of Figures

List of Tables

Abbreviations

E2ED	**E**nd-to-**E**nd **D**elay
ETX	**E**xpected **T**ransmission **C**ount
FANET	**F**lying **A**d Hoc **NET**work
LLT	**L**ink **L**ife**T**ime
MANET	**M**obile **A**d hoc **NET**work
MD	**M**inimum **D**elay
ML	**M**inimum **L**oss
MPR	**M**ulti**P**oint **R**elay
MST	**M**inimum-cost **S**panning **T**ree
ND-DEMO	**N**ode **D**eployment - **D**ifferential **E**volution for **M**ultiobjective **O**ptimization
NLT	**N**etwork **L**ife**T**ime
NRL	**N**ormalized **R**outing **L**oad
OLSR	**O**ptimized **L**ink **S**tate **R**outing
PDR	**P**acket **D**elivery **R**atio
PER	**P**acket **E**rror **R**ate
PLR	**P**acket **L**oss **R**atio
QoS	**Q**uality of **S**ervice
QM	**Q**uality **M**etric
REC	**R**esidual **E**nergy **C**ost
SPP	**S**hortest **P**ath **P**roblem
UWN	**U**nderwater **W**ireless **N**etwork
VANET	**V**ehicular **A**d Hoc **NET**work
WMN	**W**ireless **M**esh **N**etwork
WSN	**W**ireless **S**ensor **N**etwork

Chapter 1

Introduction

1.1 Presentation

Over the last few years, many systems have emerged intending to raise connectivity to a level higher than many people ever thought possible. Imagine the following events:

- Cars and transport infrastructure interconnected (Intelligent Transportation Systems and Vehicular Networks) [1, 2].
- Monitoring and communication happening in harsh and isolated scenarios (Wireless Sensor Networks) [3].
- Internet arriving at the rural zones (Wireless Mesh Network) [4, 5].
- Unmanned aerial vehicles capable of moving in convoy (Flying Networks) [6].
- Smart homes with sensors and actors interconnected performing distributed sensing and autonomous interventions (Wireless Sensor and Actor Networks) [7].
- Autonomous underwater vehicles monitoring the bottom of the deep ocean in search of different life forms (Underwater Sensor Networks) [8, 9].

Interconnected things are a reality today thanks to the evolution of data communication networks. However, looking at these promising applications, it is not always possible to have a network infrastructure available, as in Flying Networks [6, 10]. In other cases, the network may have a limited infrastructure, which does not meet all devices, as in Vehicular Networks [11].

Conceptually, an Ad-hoc Wireless Network is self-organized because it does not rely on any available fixed infrastructure [12]. So, the devices can move freely and, at the same time, organize themselves in a fashion such that the network stays functional [12–14].

In this sense, the devices are also called *networked* since they can “talk” to each other using routes earlier established by routing protocols. In that way, the nodes play the role of relays. Thus, a source node sends a message to a destination node utilizing a route that can have multiple hops [13].

The critical issue is: how to properly evaluate the routes to choose the best one when interconnecting two nodes? This issue is especially challenging as Ad-hoc Wireless Networks are built to work in specific scenarios. Because of that, the application requirements tend to be problem-dependent. For example, a monitoring network can require a reasonable level of Packet Delivery Reliability (PDR) [15, 16]. On the other hand, when the goal is to enable a VoIP (Voice over Internet Protocol) application, the network needs to keep under control the End-to-End Delay (E2ED) [15, 16]. In other words, distinct applications can demand different Quality of Service (QoS) guarantees. For that reason, QoS-aware routing protocols have emerged [14, 17–19].

These new protocols employ metrics that can assess the QoS during the route optimization process. However, besides instabilities of the wireless medium, the routing protocols should consider the fact that the different applications can have conflicting quality requirements. This reality may demand to optimize multiple objectives. As a result, future routing mechanisms must support two or more QoS requirements and take into account the trade-off between them, since state-of-the-art protocols do not have these features natively [16, 20].

Formally, an Ad-hoc Wireless Network is a directed graph $G(V, E)$, in which $V = \{1, 2, \ldots, n\}$ is a finite set of nodes, and $E = \{1, 2, \ldots, m\}$ is a finite set of links between these nodes. The protocol defines a neighborhood $N_{(v_i)} \subset V$ within the coverage area of the v_i node. To enable a message exchange between a source node $SN \in V$ and a destination node $DN \in V$, a path (route) that connects them must be available.

Together with the definition of these routes, it is necessary to ensure that all (or most of) packets are successfully transmitted as fast as possible to the destination. Thus, one or more quality measures must be defined to optimize the inclusion of links in the routes. Usually, QoS-aware metrics represent these measures. When more than one metric is relevant, the single-criterion problem turns into a multiple criteria problem [21].

In this context, this work proposes to investigate if the use of multiple QoS-aware metrics can generate better compromise solutions considering several network quality indicators in static and mobile Ad-hoc Wireless Networks. Because of the characteristics of the problem, the proposed methods are based on scalar methods (weighted sum and compromise). These approaches are presented in the course of the work.

1.2 Scope

When optimizing routes in Ad-hoc Wireless Networks, the following concerns arise:

1. Routing protocols need to transmit control packets frequently to handle topology changes and link breakages [22].
2. These exchanges of control packets cannot overload the network [22].
3. The objectives to be optimized and, consequently, the quality metrics utilized to reach these objectives must be well-defined [16].
4. Using advanced multi-objective optimization metaheuristics can be unfeasible mainly when devices are mobile and have low energy autonomy and computational resources [20].
5. Scalar methods are the most employed to deal with multiple criteria in this context. However, the higher difficulty lies in the fine-tuning of additional parameters [20, 23–25].
6. Most proposed routing schemes are specific to a given scenario or quality requirement, even when the solution uses multiple criteria [10, 23, 24, 26, 27].

Some protocols have been presented to cope with the topology definition in Ad-hoc Networks by focusing on how to solve the issues number 1 and 2 [22]. Proactive protocols fire periodic messages to maintain the topology updated and select routes *a priori*, while reactive protocols build the routes on-demand. Some protocols partition the network and employ both paradigms [22]. There are also protocols that generate the routes opportunistically as the data are transmitted [28]. These paradigms are described in the next chapter.

However, the remaining issues are paramount when talking about how to provide QoS in such networks, regardless of which routing paradigm a solution employs. The way how a link is evaluated impacts directly on the protocol performance and, hence, it affects users' Quality of Experience (QoE) [17, 29]. Therefore, in spite of proposing some changes in the routing protocol used, the scope of this work lies in how to cope with the issues number 3 to 6 adequately.

1.3 Goals of This Dissertation

This work aims to design a flexible multicriteria framework that supports several optimization objectives in order to verify if the use of multiple QoS-aware metrics can generate promising compromise solutions considering the main network quality indicators in static and mobile Ad-hoc Wireless Networks. The specific goals are presented next:

- Design of a flexible multicriteria framework which integrates multiple QoS-aware metrics for Ad-hoc Wireless Networks;
- Proposal of dedicated scalar-based methods inspired from weighted sum and compromise approaches;
- Proposal of a path size control mechanism so as to aid the definition of solutions more resistant to events such as collisions, noise, and interference;
- Design of a utility function in order to support the definition of parameters of the scalar-based approaches, such as weights and constraints;
- Proposal of test problems to validate and analyze the methods suggested.

1.4 Contributions

The contributions of this dissertation are summarized next:

- A flexible multicriteria framework is developed to house the implementation of the QoS-aware metrics and the adaptation of the MPR (MultiPoint Relay) and Shortest Path algorithms.
- Two new models are proposed to deal with multiple QoS-aware metrics, both based on scalar methods. In these models, a mechanism is introduced to control the growth of the number of hops.
- A multi-objective algorithm is proposed to define properly the position of the nodes in the plan when the network is static.
- A new method is proposed to define weights and constraints of the scalar methods. Unlike many works found in the literature that consider just the Packet Delivery Ratio (PDR) to determine these parameters, multiple network quality indicators are recognized when taking such a decision. This allows managing the network according to design objectives.

The list of publications arising from this work can be checked in Appendix B.

1.5 Structure of the Work

The subjects discussed in this work are organized as shown next.

Chapter 2 – Ad-hoc Routing Review: In this chapter, a review of ad-hoc routing protocols is made and QoS provisioning mechanisms are introduced.

Chapter 3 – QoS-aware Routing in Ad-hoc Wireless Networks: This chapter describes how optimization problems are related to QoS-aware routing in Ad hoc Wireless Networks. In addition, the QoS-aware metrics used in this work are detailed.

Chapter 4 – Multi-objective Optimization Methods: The characterization of multi-objective optimization problems is introduced in this chapter. Prominent methods applied to the Ad-hoc Wireless Networks are also introduced jointly with two models to solve the shortest path problem in these networks.

Chapter 5 – Framework and Algorithms: At the time of this chapter, the structure of the framework is demonstrated. Therein, all propositions and interventions described in earlier chapters are detailed.

Chapter 6 – Results and Discussion: This chapter describes the experimental design used to validate the proposed methods. The results are discussed considering the various scenarios studied.

Chapter 7 – Conclusions: At this point, general conclusions and discussions concerning the work are presented. In addition, this chapter delineates suggestions and insights for future works.

Chapter 2

Ad-hoc Routing Review

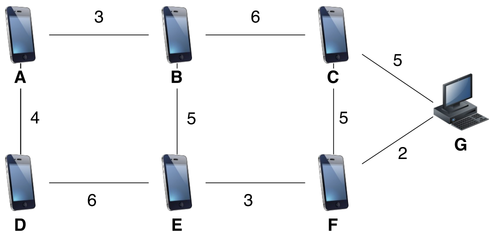
2.1 Ad-hoc Routing

Two approaches have been standard used to solve the routing problem in wired networks. The first one is Distance Vector Routing (DVR) [30]. In DVR, a node v_i regularly shares with its set of neighbors $N_{(v_i)}$ a vector that contains the distances between it and each node of this neighborhood. This node can determine the next hop from itself to all possible destination as these exchanges are performed. The distributed path computation is made using the Bellman-Ford algorithm. Apart from simplicity, an advantage of this approach is that only local information is required. However, DVR is more often used in small local area networks, since it may suffer from routing loops due to slow convergence when the network is denser [30].

The second one is Link State Routing (LSR). Unlike DVR, LSR seeks to build a global routing table. Path computation is performed using Dijkstra's algorithm. Although LSR claims higher storage space and computational complexity, it does not suffer from routing loops [30]. Since this work utilizes an LSR-based protocol, it is worth taking a closer look at how this approach works.

In LSR, a state determines the associated cost to a link. When this cost is infinite (∞), either that link does not exist, or it is broken. Considering a weighted graph, in which each link $k \in E$ has an associated cost w_k, the best route - starting from a source node $SN \in V$ to a destination node $DN \in V$ - is the one with the lowest cost. In this context, each node needs to find the shortest path between itself and all other nodes to be able to forward a packet to any node in the network [30].

In a network comprised of n nodes, there will be $n - 1$ shortest paths starting from each node to any other node. It means that $n \times (n - 1)$ shortest paths are needed to interconnect all nodes. In a scenario with seven nodes as that presented in Figure 2.1a, the routing protocol must provide 42 shortest paths.



(A) Weighted graph

	A	B	C	D	E	F	G
A	0	3	∞	4	∞	∞	∞
B	2	0	6	∞	5	∞	∞
C	∞	6	0	∞	∞	5	5
D	4	∞	∞	0	6	∞	∞
E	∞	5	∞	6	0	3	∞
F	∞	∞	5	∞	3	0	2
G	∞	∞	5	∞	∞	2	0

(B) Link State Database

FIGURE 2.1: Network with Link State DateBase (LSDB) example

A better way to see these paths is by combining them into a Minimum-cost Spanning Tree (MST), which is a tree that has SN as root and that extends throughout the graph so that all other nodes are visited [30]. Each path between SN and any other node is the shortest. Figure 2.2 shows an example considering the network of Figure 2.1a.

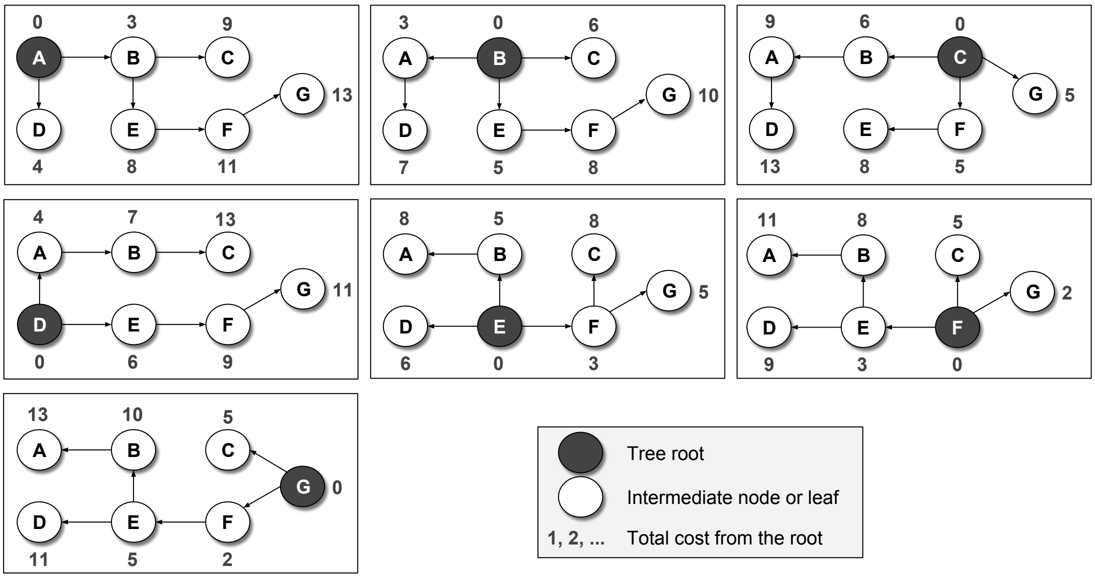


FIGURE 2.2: MSTs for the nodes of the network of the Figure 2.1a.

The nodes need to know the state of each link to build the MST. From these pieces of information, the protocol generates a Link State Database (LSDB), and draw a complete network map. Thus, there will be only one LSDB for the entire network, and each node must have a copy of it to be able to create its own MST [30]. The LSDB is a two-dimensional vector (as it is shown in Figure 2.1b), in which each cell contains a value that defines the cost of the corresponding link.

Since there are links with infinite cost (∞) in Figure 2.1b, it would not be possible to create the MST of Figure 2.2. Therefore, the protocol needs to create an LSDB that contains information about the entire network. This task is made possible through a flooding process that works as follows [30]:

- Each node sends some probe messages to all its direct neighbors (the ones to which it is directly connected) to collect two information of them: i) the identity and ii) the link cost (state).
- From this information, the protocol generates and sends a packet called Link-State Packet (LSP) to the neighborhood, as it is shown in Figure 2.3.
- When an LSP arrives, the protocol compares this newly arrived version with the current LSP (if the node has already received any). If the new LSP is older than the current one (the protocol determines it by checking the sequence number), it is discarded. Otherwise, it replaces the current one and, after that, a copy is sent to all neighbors (except for the one that sent the LSP).

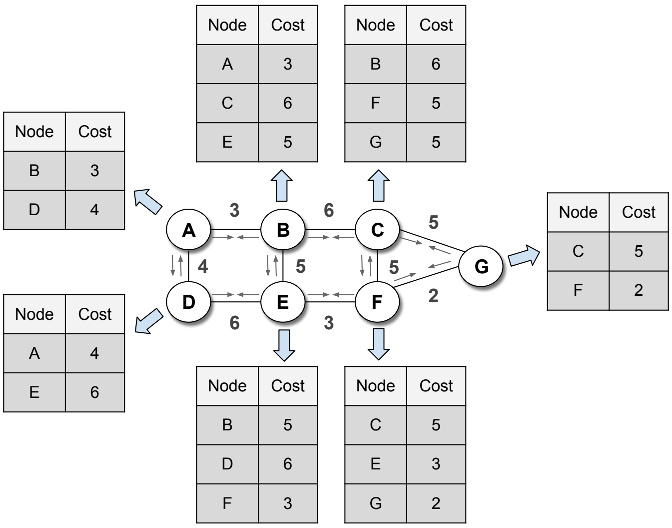


FIGURE 2.3: LSPs created and sent by each node for construction of the LSDB.

Such a strategy ensures that, upon receiving all new LSPs, each node will be able to create a complete and identical LSDB and having global knowledge of the network. Now, the nodes can build their MSTs by executing the well-known Dijkstra's algorithm.

Figure 2.4 shows an MST structure for the graph of Figure 2.1a considering the node A as root. It is necessary to go through initialization and six iterations to compute the MST. Later, in Chapter 5, it is presented a modified version of this algorithm to cover the proposals described in this work.

Initially, the DVR-based and LSR-based protocols emerged to address the demands of wired networks [28]. However, Ad-hoc Wireless Networks have been gaining increasingly space, thanks to factors such as flexibility, the variety of applications, and support

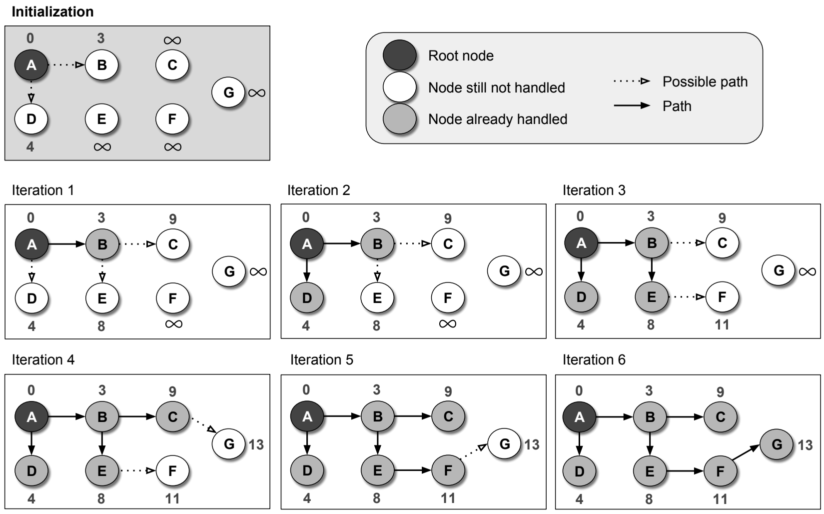


FIGURE 2.4: Composition of the MST using the Dijkstra's algorithm.

to mobility. For that reason, the research community and the industry have engaged in developing routing protocols that tailor to the unstable nature of these networks [22, 28, 31, 32].

2.2 Ad-hoc routing paradigms

The author in [28] classifies the Ad-hoc Wireless Routing protocols into four paradigms, namely: reactive, proactive, hybrid, and opportunistic.

2.2.1 Reactive routing

Reactive protocols compute the paths on-demand. The idea is to determine a route just when a message needs to be transmitted. The goal is to reduce routing, processing, and storage overhead. At this paradigm, Ad-hoc On-demand Distance Vector (AODV) is the most historically studied protocol [13]. Based on it, new protocols have emerged, such as DYMO [33] and BATMAN [34].

In AODV, when a source node SN needs to deliver a message to a given destination node DN, the AODV initiates a process called route discovery. SN sends a route request (RREQ) message to its neighbors. This message is flooded through the network until it reaches DN or until it reaches an intermediate node that knows how to arrive at DN. DN (or an intermediate node) then sends back to SN a route reply (RREP) unicast message to acknowledge the route establishment. In this process, the routing tables of the nodes that compose the route are updated [35]. A setback of this protocol is that the route discovery process may take too much time [31, 32]. This characteristic can be harmful if real-time multimedia applications are used [32].

2.2.2 Proactive routing

Instead of reacting to the demands as reactive protocols usually do, proactive protocols implement mechanisms to compute and maintain the routing tables updated, that is, each node sends its table to its neighbors periodically.

Proactive protocols share the same difficulty. As they need to ensure the routing tables remain consistent to find optimal paths, control packets must constantly be transmitted. This factor, on its own, can create an excessive overhead [22, 32]. To lighten the effects of this problem, the Optimized Link State Routing (OLSR) was designed [36–38].

OLSR is an LSR-based protocol whose principal function is to narrow the number of nodes that forward Link State Packets (LSPs) and, hence, reduce the impact of the flooding. For this purpose, it introduces a technique called MPR (MultiPoint Relay). The idea is to define MPR nodes that must be in charge of exchanging the routing information periodically. So, when a node needs to advertise a control packet on the network, it will send to all its neighbors, but only an MPR node will relay the information forward. In this way, each node will only receive the information once [13, 22].

In this paradigm, OLSR is the most representative protocol, and it is utilized in this work. It is more detailed in Chapter 5, where it is presented the proposed framework.

2.2.3 Hybrid routing

Some papers have dedicated to proposing hybrid protocols that combine reactive and proactive paradigms. For instance, Zone Routing Protocol (ZRP) [39] proactively maintains routes within a zone and reactively builds routes between these zones [22].

2.2.4 Opportunistic routing

The protocols mentioned so far are adaptations of protocols that are designed to work on wired networks [28]. In scenarios characterized by intermittent connectivity and limited infrastructure, they may not be so adequate. Under this premise, the authors in [40] introduce the concept of opportunistic routing.

The basic operation is summarized as follows: i) the source node propagates the data packet; ii) after receiving this packet, a relay node is selected from among the candidate nodes; iii) the selected node proceeds with the transmission of the packet opportunistically selecting the next relay. This process continues until the packet reaches the destination [28]. Note that, unlike the traditional approaches, the selection of the relay nodes occurs in real time (on-the-fly). Many protocols have been presented following this reasoning, as for example: ExOR [40], LCAR [41], PLASMA [42], and COR [19].

2.3 Ad-hoc routing evaluation

As important as determining the best routing protocol or paradigm to be used in a particular scenario is to define how to evaluate the cost of a link or node. In short, these values represent some quality metric. In the literature, it is possible to realize that such a "quality metric" can be many things [16, 29].

The most basic metric assigns the cost equal to 1 to the links, meaning a hop between two nodes. So, the best route is the one that has the minimum number of hops. However, this metric does not fit to environments where the link states vary a lot, and the connectivity is intermittent [28, 43]. Because of that, many QoS-aware metrics have emerged. Information like distance, signal power, packet delivery rate, delay, residual energy, and link stability, can serve as a basis to derive these new quality metrics [14].

Taking into account the multitude of possibilities for QoS-aware metrics, it is natural to ask "which one should be adopted regarding a particular scenario or application?" The answer to this question is straightforward: it depends directly on the objective to be optimized. Some objectives have been stressed in literature, such as: minimizing packet loss ratio, minimizing end-to-end delay, maximizing network lifetime, maximizing route stability, minimizing energy consumption, and minimizing the number of hops [16, 17].

A quality metric is designed to answer a specific objective. However, researchers have observed the need to address more than one objective to provide a satisfactory overall network performance [19, 24–27, 44–49]. At this point, multicriteria optimization strategies are promising because they can be used to design protocols that can meet heterogeneous scenarios and applications [20].

The next chapter elaborates better the idea of applying multicriteria optimization in the routing problem in Ad-hoc Wireless Networks.

Chapter 3

QoS-aware Routing in Ad-hoc Wireless Networks

3.1 Presentation

Guaranteeing QoS in Ad-hoc Wireless Networks is a challenging task. Many factors can impact this process, such as network characteristics, users' experience demands and their applications, mobility patterns, and the number of devices. Considering these factors, the network designer needs to define one or more design objectives that the solution should address. In short, the routing protocol must be flexible enough to meet the quality requirements related to design objectives defined [15].

This chapter presents some general issues about the foremost design objectives observed when planning ad-hoc networks. Also, the QoS metrics used in this work and their position in literature are detailed. As a starting point, it is introduced the characterization of a single-objective optimization problem related to a simple routing problem.

3.2 Single-objective Optimization Problem

Optimization is the process of minimizing or maximizing an objective function that is subject to a set of equality or inequality restrictions. Considering minimization, solving a single-objective optimization problem can be understood as determining a decision vector $\mathbf{x}^*$ such that $f(\mathbf{x}^*)$ attains its smallest value possible. $\mathbf{x}^*$ is called a global minimum under these circumstances.

As $\mathbf{x}$ is an arbitrary combination of input parameters, the formulation of a single-objective optimization problem can be written as [21]:

$$\begin{aligned} &\text{minimize} \quad && f(\mathbf{x}) \\ &\text{subject to} \quad && \mathbf{x} \in S \subseteq \mathbb{X}, \end{aligned} \tag{3.1}$$

wherein $\mathbb{X}$ is the generic set of decision vectors and S is the set of all vectors $\mathbf{x}$ satisfying

$$\begin{aligned} \mathbf{g}(\mathbf{x}) &\leq \mathbf{0}, \\ \mathbf{h}(\mathbf{x}) &= \mathbf{0}, \\ \mathbf{x}_{lower} \leq \mathbf{x} &\leq \mathbf{x}_{upper}, \end{aligned} \tag{3.2}$$

in which $\mathbf{g} : \mathbb{X} \mapsto \mathbb{R}^l$ and $\mathbf{h} : \mathbb{X} \mapsto \mathbb{R}^m$ represent inequality and equality constraints, respectively. When the model does not claim that the solution must satisfy any constraint, the problem is called unconstrained[1]. In these cases, S is identically equal to $\mathbb{X}$. Otherwise, when $\mathbf{x}$ is subject to a set of constraints, such as equality and inequality functions, S is a subset of $\mathbb{X}$ [20].

To see how the single-objective optimization can be a tool to solve a simple routing problem, consider the following example.

Example 3.1. *Professor John wants to invite his class from the ornithology course (a branch of zoology that concerns the study of birds) of the Biological Sciences Faculty at the UFMG for a practical activity that he calls:* ***Field Day****. The goal is to find out which bird species live in the Serra do Cipó, Minas Gerais. The professor decides to divide the class into groups. As the region is large, he organizes the teams as follows: i) each group must be responsible for a given observation area; ii) a student from each group must stay in a region with Internet access (base station) to receive photos, videos, and information from the gathering, as well as to clarify doubts of who is in the field.*

There is no network infrastructure connecting the observation field to the base station. Thus, the communication should be done by multiple hops. The professor then consulted the Computer Network Team (CNT) of the campus to know whether such an action would be possible. Figure 3.1 illustrates the topology presented by him. He stressed the students should be able to make voice calls on the network.

Consider that a voice call needs to be made from GROUP 1 (A) to base station (BS). As it is a real-time transmission, CNT has judged minimizing the end-to-end delay (E2ED) would be more appropriate. In many cases, this is achieved by minimizing the number of hops[2]*. Table 3.1 shows the available routes and the number of hops of each one.*

---

[1] When a problem has only box constraints, it is also named unconstrained [50].

[2] Minimizing the number of hops is not always equivalent to minimize end-to-end delay. When optimizing the number of hops, there is a considerable risk that, by disregarding the link stability for example, many packet losses occur, which generates retransmissions and consequently more delay [51]. In this case, due to the simplicity of the scenario, it can be assumed that minimizing number of hops is enough to minimize E2ED as well.

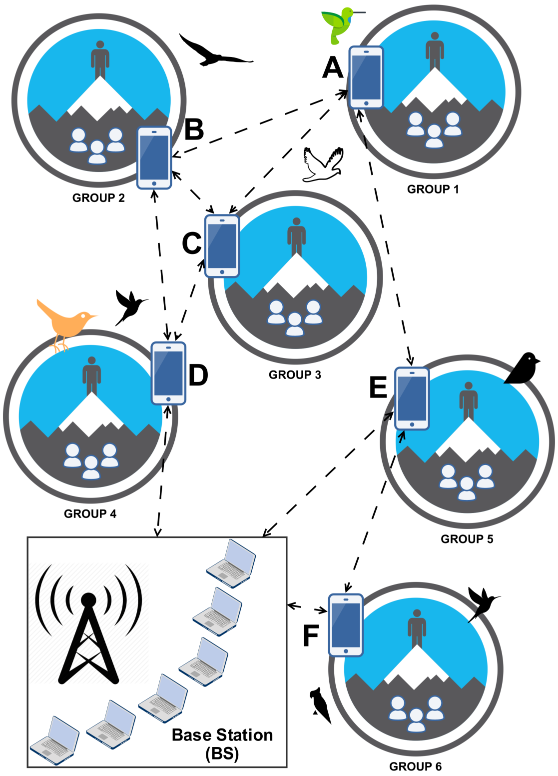


FIGURE 3.1: Example of an Ad-hoc Wireless Network topology. Dashed lines are links connecting the nodes. For sake of simplicity, the links are symmetrical.

TABLE 3.1: Available routes with the number of hops.

Solution	Route	Number of Hops
1	A → B → C → D → BS	4
2	A → B → D → BS	3
3	A → C → D → BS	3
4	A → E → F → BS	3
5	A → E → BS	2

The goal is to find the route that minimizes the number of hops, that is,

$$\begin{aligned} &\textit{minimize} \quad f(\boldsymbol{x}) = [\textit{number of hops}] \\ &\textit{subject to} \quad \boldsymbol{x} \in S, \end{aligned} \tag{3.3}$$

wherein S is the feasible routes set. Herein, a route is feasible when it does not have cycles. According to the objective, the routing protocol should choose the 5th solution.

Although the number of hops is the metric adopted by state-of-art routing protocols, other metrics are more efficient regarding QoS [43, 52–54]. Choosing the best metric is often tied to the objective to be optimized. In this context, some of the main objectives discussed in the literature are presented below.

3.3 Design Objectives

An Ad-hoc Wireless Network is called a self-organizing network when the nodes themselves discover and maintain the routes in a distributed way [14, 28, 55]. Thus, nodes must be available and prepared to play the role of relays. Therefore, whenever nodes want to "talk" to each other, routes must be established. Routing protocols are in charge of defining rules and procedures to provide such routes [13]. The link or node quality is the feedstock utilized by these protocols to make the best routing decision regardless of the algorithms or techniques specifically employed to find the route. This quality is usually measured by metrics that evaluate the node or link status in real time or even make estimates or forecasts based on network parameters [5, 16, 24].

This section states which are the fundamental design objectives that must be taken into account while designing a metric to be used by the routing algorithm. Subsequently, some well-known metrics are outlined.

3.3.1 Minimizing the number of hops

Minimizing the number of hops is an objective frequently associated with simplicity and low computational overhead [16]. For example, OLSR uses *hop-count* to compute the shortest path, while AODV verifies *hop-count* to update a table entry [35, 36]. Apart from simplicity, optimizing this objective can reduce the end-to-end delay [23, 32]. Moreover, often a shorter path decreases the energy consumption and increases network lifetime [56].

However, minimizing only the number of hops is an overly simple objective when applied to Ad-hoc Wireless Networks. In these networks, most of the times, *hop-count* information is not enough for selecting a route with adequate performance [28, 43]. In fact, paths with fewer hops usually result in longer distance hops, which implies in less reliable links [13]. Also, in dense networks, routes between source and destination with the same number of hops will inevitably exist. An arbitrary tie-breaking decision may not select the best available route. Hence, additional clues are often necessary to yield paths with reasonable QoS, at least.

3.3.2 Minimizing end-to-end delay

Interactive real-time multimedia applications, like voice and video over IP, are highly sensitive to delay. In [57], the authors argue that the human auditory system does not perceive delays shorter than 150 milliseconds in voice communication. Delays between

150 and 400 milliseconds may cause some discomfort, but are acceptable. More than 400 milliseconds may mean uncomfortable and poorly understood conversations. For addressing this demand, a routing solution can occasionally tolerate packet drops provided that the application smooth out slight damage. Today, modern multimedia applications are very efficient in providing this service [58]. Considering the spread of this type of application, minimizing end-to-end delay is a critical design objective.

3.3.3 Minimizing packet loss ratio

The messages exchanged between users are the most valuable assets on a network. So, minimizing packet loss is very significant. The reliability is closely associated with the packet delivery rate that the routing protocol can guarantee. In interactive real-time multimedia applications, drops higher than 20% may result in unclear chats [57].

Concerning to elastic data, such as FTP, email, and file download, the sensitivity to packet drops is even more significant. For instance, a file transfer process demands full integrity of the packets. If this does not occur for some reason, retransmissions are required [30]. In this type of application, longer delays can be tolerated provided that there is a reasonable assurance of delivery reliability [57].

3.3.4 Minimizing energy consumption

The energy depletion control is a considerable issue in Ad-hoc Wireless Networks as it impacts the network lifetime. When this requirement is not observed correctly, the utility and efficiency of the network could be reduced mainly in scenarios where the nodes do not have a regular source of energy such as in wireless sensor and mobile networks [46, 56].

For meeting this requirement, some works use a direct approach. Each node needs to broadcast its residual energy periodically to the neighborhood, and this information is announced for the entire network [45, 59]. Prediction-based approaches have emerged to minimize the amount of energy spent to do this mapping [24, 60].

3.3.5 Maximizing route stability

The mobility and the instability of the mobile networks contribute towards the frequent topological changes. In this sense, a link can drop and get up numerous times during the network lifetime. Therefore, maximizing route stability is a valuable design objective. Usually, the packet drops caused by disconnections reduce when the protocol considers the link stability in the route discovery process. In general, the link stability is measured through its age. So, the older links are considered more stable [29].

3.3.6 Considering trade-offs and combining complementary objectives

Many works have identified trade-offs between these objectives. In general, a route achieves higher throughput and packet delivery ratio values at the cost of increasing the end-to-end delay, mostly in static networks [16, 23, 24]. The possible compromise between delay and delivery reliability symbolizes the challenge of designing solutions capable of supporting applications with different quality requirements. The energy issue is another factor that has drawn the attention. In most cases, a compromise between energy consumption and delivery reliability also arises [20, 60].

Some design objectives are correlated in some way. For instance, when maximizing route stability, the packet loss ratio tends to decrease. Thus, it makes sense to consider both objectives (maximizing route stability and minimizing packet loss ratio) when maximizing the delivery reliability [18].

Multicriteria problems have been studied mainly in fields as decision theory and operations research [21]. Also, multicriteria techniques have been drawing attention as a tool to design methods and protocols that deal with two or more QoS requirements [23, 56, 60]. The idea is to develop robust and adaptive solutions that fit in distinct scenarios [14].

3.4 QoS-aware Routing Metrics

The optimization or not of the objectives mentioned above depends on the quality measures used to evaluate the links or nodes. This section discusses some of the principal metrics used in the literature.

3.4.1 Expected Transmission Count (ETX)

Expected Transmission Count (ETX) was proposed in [53] to tackle the limitations of the *hop-count* metric. The primary objective is to maximize the network throughput. ETX is the estimated number of transmissions so that a packet, leaving the source, reaches its destination. It is computed as follows [53]

$$ETX_{(v_i,v_j)} = \frac{1}{P_{(v_i,v_j)}} \tag{3.4}$$

$P_{(v_i,v_j)}$ is the expected probability that a packet is successfully received and acknowledged. It is given by

$$P_{(v_i,v_j)} = d_f \times d_r \tag{3.5}$$

in which d_f is the probability of success in delivering a packet of v_i to v_j and d_r is the probability of success in delivering a packet to v_i confirming that v_j has received the packet sent. The ETX of a route is the sum of the ETX values along the route as described below

$$ETX_{route} = \sum_{(v_i,v_j)\in E} ETX_{(v_i,v_j)} \tag{3.6}$$

Example 3.2. *In example 3.1, CNT team proposes to employ the number of hops as the quality metric in the Professor John's network. However, one of the team members discovered the possibility of using QoS-aware metrics. ETX was then tested. Table 3.2 shows the estimated probability of packet delivery and ETX values for all links in the topology illustrated in Figure 3.1.*

Table 3.2: Estimated probability of packet delivery and ETX values for all links.

Link	Probability of Packet Delivery	ETX metric
1	$P_{A,B} = 0.94$	$ETX_{A,B} = 1/0.94 = 1.064$
2	$P_{A,C} = 0.95$	$ETX_{A,C} = 1/0.95 = 1.053$
3	$P_{A,E} = 0.86$	$ETX_{A,E} = 1/0.86 = 1.163$
4	$P_{B,C} = 1$	$ETX_{B,C} = 1/1 = 1$
5	$P_{B,D} = 0.93$	$ETX_{B,D} = 1/0.93 = 1.075$
6	$P_{C,D} = 0.99$	$ETX_{C,D} = 1/0.99 = 1.010$
7	$P_{D,BS} = 0.94$	$ETX_{D,BS} = 1/0.94 = 1.064$
8	$P_{E,F} = 0.91$	$ETX_{E,F} = 1/0.91 = 1.099$
9	$P_{E,BS} = 0.50$	$ETX_{E,BS} = 1/0.50 = 2.041$
10	$P_{F,BS} = 1$	$ETX_{F,BS} = 1/1 = 1$

ETX of the routes can be computed according to Equation (3.6). Table 3.3 contains these values related to $A \rightarrow BS$ routes.

Table 3.3: Available routes with the number of hops and ETX values.

Solution	Route	Hops	ETX of the route
1	A → B → C → D → BS	4	$1.064 + 1 + 1.010 + 1.064 = 4.138$
2	A → B → D → BS	3	$1.064 + 1.075 + 1.064 = 3.203$
3	A → C → D → BS	3	$1.053 + 1.010 + 1.064 = \mathbf{3.127}$
4	A → E → F → BS	3	$1.163 + 1.099 + 1 = 3.262$
5	A → E → BS	**2**	$1.163 + 2 = 3.163$

As it can be seen, the 3rd solution got the lowest ETX value. Hence, it would be the one chosen. Concerning packet delivery ratio, ETX tends to select a route that overcomes that one taken by hop-count. However, if any of the links of this path degrades minimally (for example, reducing the probability of packet delivery to 90%), the 5th solution would be the best one. Note that, the 5th solution would also be the one chosen by the hop-count metric. Observe that one of the links of this solution has a packet loss ratio of 50%

($E \rightarrow BS$), despite the minimum number of hops. It means that ETX can often select routes with high packet loss ratio (5th solution) even when there are alternative paths with low loss (2nd, 3rd, and 4th solutions). Besides that, if there are small up-and-down variations, there will also be a frequent flipping between the 3rd solution and the 5th one, thus generating an instability.

The probability of packet delivery is measured using the Hello protocol [61]. This protocol and its variants are adopted by much routing protocols to establish [36] and maintain [35] bi-directional neighbor relationships. Generally, HELLO packets are sent every t seconds. Each node counts the number of HELLOs received within in a given time window w and divides it by the expected number of HELLOs in the same period. This operation is described as follows

$$d_f = \frac{hl}{{}^{w}\!/_{t}} \tag{3.7}$$

in which hl is the number of HELLOs successfully received during w and ${}^{w}\!/_{t}$ is the number of HELLOs sent during w. For example, if the protocol sends HELLO packets every 2 seconds (t) and the time window is 20 seconds (w), then the number of HELLO packets that should be received every 20 seconds is equal to 10. If a node received only 8 packets (hl), the estimated d_f of the link is 0.8 (80%). For each HELLO packet received, the node sends an acknowledgment so that the sender can calculate the reverse delivery ratio d_r.

The worse the probability of packet delivery, the higher the ETX value. The reader can consult details and some variations of this metric in [55].

3.4.2 Minimum Loss (ML)

Minimum Loss (ML) metric was proposed in [62] with the objective of minimizing the packet loss. Considering that the probability of a packet is successfully received and acknowledged is given by Equation (3.5), the ML value of the route is given by the product of the probabilities of packet delivery along the route, as described below

$$ML_{route} = 1 - \left(\prod_{(v_i, v_j) \in E} P_{(v_i, v_j)} \right) \tag{3.8}$$

As ETX, chosen paths based on minimum loss rates lead to high throughput. However, ML has additional advantages. First, the routes tend to be more stable. Thus, packet loss rates are lower. Second, while ETX needs to deal with inversion and sum of n products, ML has to deal with the multiplication of n products only. Thereby, it generates less computational overhead than ETX [17].

Example 3.3. *Table 3.4 shows the ML computation in Professor John's network. Wishing to figure out which metric is one that best fit this network, all values of metrics studied in this section are listed in Table 3.11.*

Table 3.4: Available routes with ML values.

Sol.	Route	ML
1	A → B → C → D → BS	1 - $(0.94 \times 1 \times 0.99 \times 0.94) = 0.125$
2	A → B → D → BS	1 - $(0.94 \times 0.93 \times 0.94) = 0.178$
3	A → C → D → BS	1 - $(0.95 \times 0.99 \times 0.94) = \mathbf{0.116}$
4	A → E → F → BS	1 - $(0.86 \times 0.91 \times 1) = 0.217$
5	A → E → BS	1 - $(0.86 \times 0.50) = 0.570$

Reflecting specifically on ETX and ML, both would get the same solution (3rd). However, there is a meaningful difference. Since ML seeks to minimize the probability of packet loss, the 5th solution becomes the last option because of the bottleneck E → BS that drops 50% of packets. The 5th solution is the second best option in ETX, and it would likely be used at some time. In ML, the 1st solution is the second best one, even though it has the highest number of hops. In both ETX and hop-count metrics, this option would be the last one, and it would hardly be used. In summary, ML orders the routes according to the reliability of packet delivery, regardless of the number of hops.

ETX and ML are not able to assess the quality accurately when a link is asymmetric. Moreover, they do not take into account requirements such as link delay, link stability, node residual energy, and even the number of hops on the route.

3.4.3 Minimum Delay (MD)

In [54], the authors proposed to use the per-hop packet pair delay metric in wireless networks. At that point, this metric was widely used in the context of wired networks. The technique consists in transmitting, at fixed intervals, two probe packets back-to-back to each neighbor to measure the packet dispersion. In short, the receiver v_j measures the delay by computing the difference between the reception times of the packet pair, and then it sends back this value to the sender v_i. Finally, the sender sets up this value as the link quality measurement. In [63], the Ad-Hoc Probe algorithm was designed based on this strategy. In [52], the authors integrated this metric with the OLSR protocol and named it Minimum Delay (MD). The formula used to calculate the $MD_{(v_i,v_j)}$ from a given packet pair sample k is given by

$$MD_{(v_i,v_j)} = (RT_{2,k} - ST_k - \delta) - (RT_{1,k} - ST_k - \delta) \tag{3.9}$$

where δ is the clock offset among nodes and ST_k is the sent time of the packet k stamped at the sender node v_i. $RT_{1,k}$ and $RT_{2,k}$ are the receiving time of the packet k measured at the receiver node v_j. The MD value of the route is given by the sum of the one-way delays along the route, as described below

$$MD_{route} = \sum_{(v_i,v_j)\in E} MD_{(v_i,v_j)} \tag{3.10}$$

The authors also proposed a way to measure the delay in the reverse direction. The idea is to allow that v_j announces the measured value for each of its neighbors using a modified HELLO packet. When v_i sees itself listed in the HELLO message, it can compute the packet dispersion in the direction $v_j \rightarrow v_i$.

Example 3.4. *In the Professor John's network, students need to transmit real-time audio. Because of that, CNT decides to use a delay-driven metric. Looking over at Figure 3.1, links formed by nodes that are closer tend to have lower transmission delay. On the other hand, routes with a lower number of hops may accumulate less delay. Table 3.5 contains an example of estimated MD values for each link, while Table 3.6 summarizes the computed MD values for all routes.*

Table 3.5: Estimated delay values for all links.

Link	Estimated delay of the link(s)
1	$MD_{A,B} = 0.025$
2	$MD_{A,C} = 0.032$
3	$MD_{A,E} = 0.045$
4	$MD_{B,C} = 0.012$
5	$MD_{B,D} = 0.027$
6	$MD_{C,D} = 0.024$
7	$MD_{D,BS} = 0.022$
8	$MD_{E,F} = 0.031$
9	$MD_{E,BS} = 0.026$
10	$MD_{F,BS} = 0.007$

When using this metric, the routing algorithm tends to select routes with fewer hops (Table 3.11). It occurs because the small number of nodes collaborates to make the end-to-end delay smaller. MD is more flexible than hop-count since it can detect the link instabilities and breaks. It is important to say the use of a delay-driven metric does not guarantee routes that minimize the end-to-end delay. In an unstable medium, shorter routes can cause high packet loss ratio and generate retransmissions. That causes more delay. Similar to hop-count, MD is hardly effective when applied alone.

Table 3.6: Available routes with MD values.

Sol.	Route	MD(s)
1	A → B → C → D → BS	0.025 + 0.012 + 0.017 + 0.022 = 0.076
2	A → B → D → BS	0.025 + 0.027 + 0.022 = 0.074
3	A → C → D → BS	0.032 + 0.024 + 0.022 = 0.078
4	A → E → F → BS	0.045 + 0.031 + 0.007 = 0.083
5	A → E → BS	0.045 + 0.026 = **0.071**

Minimizing end-to-end delay is undoubtedly one of the principal objectives to be satisfied when designing a routing protocol. However, in proportion to its importance, it is challenging to obtain a metric capable of accurately measuring or predicting the delay in transmission. For instance, though MD can guarantee some precision, it does not assess the queuing delays of the outbound packet. Furthermore, the dual packet technique provokes a significant increase in routing traffic [17]. For details on the performance of ETX, ML, and MD in a static Ad-hoc Network, [16] and [17] can be queried.

3.4.4 Link Lifetime (LLT)

Link breakages occur all the time in Ad-hoc Wireless Networks. Add mobility to this recipe and such breaks become even more frequent. Therefore, in addition to performance, a protocol should consider the estimated lifespan of links.

The author in [64] put forth the natural idea that the older the link the more stable it is. In the proposal, the link is stable when its lifetime surpasses a given time threshold L_{st} plus a small offset. The offset considers the possibility the link breaks immediately after the L_{st} time. The idea is to draw a line separating stable and unstable links. Such a division is based on two observations. First, most of the links fail after L_{st} threshold. Second, if a link is alive for longer than L_{st}, it is very likely that either the nodes are stopped, or they are moving at similar speeds in the same direction. Both situations suggest this link will be around for quite some time.

The authors in [65] show how link lifetime is affected by parameters such as speed and mobility pattern. In [66], the authors demonstrate the link lifetime distributions are well-suited to a normal distribution, regardless of mobility model. This analysis is remarkable because it allows a node to predict the link lifetime based entirely on age.

Inspired in [64] and [66], the authors in [24] derived a metric that adds the following premise: when a new link is detected, probably it was active short time ago. In a sense, the proposal seeks to encourage the use of short-lived links, unlike metrics strictly based on age. Equation (3.11) shows a nested if-else statement in which the estimation of stability can be relatively obtained, as presented in [24].

$$LLT_{(v_i,v_j)} = \begin{cases} \log_{10}(L_{st} - 1), & \text{if link ended up to be created} \\ \log_{10}(L_{st} - age_{(v_i,v_j)}), & \text{if } (age_{(v_i,v_j)} < 0.98 \times L_{st}) \\ \log_{10}(0.02 \times L_{st}), & \text{if } (0.98 \times L_{st} \leq age_{(v_i,v_j)} \leq 1.02 \times L_{st}) \\ \log_{10}(age_{(v_i,v_j)}), & \text{if } (1.02 \times L_{st} < age_{(v_i,v_j)} \leq 1.05 \times L_{st}) \\ \log_{10}(1.5 \times age_{(v_i,v_j)}), & \text{if } (age_{(v_i,v_j)} \geq 1.05 \times L_{st}) \end{cases} \tag{3.11}$$

$LLT_{(v_i,v_j)}$ is the estimated stability (or residual lifetime) of the link (v_i, v_j) at a given age. When the link is new, that can mean that it was recently broken and it gets the age of one second. For example, if L_{st} is equal to 20 seconds, then the estimated relative residual lifetime (LLT) of the link is $\log_{10}(20-1) = 1.2788$ seconds. Inasmuch as the age increases towards the L_{st} threshold, $LLT_{(v_i,v_j)}$ decreases. When the age achieves the range $[0.98 \times L_{st}, 1.02 \times L_{st}]$, $LLT_{(v_i,v_j)}$ reaches its minimum value. Passing through this bound, the link starts a transition towards the stable state. After that, the link is considered reliable and a factor of 1.5 is applied in order to increase faster the $LLT_{(v_i,v_j)}$ value. The normalized distribution is illustrated in Figure 3.2.

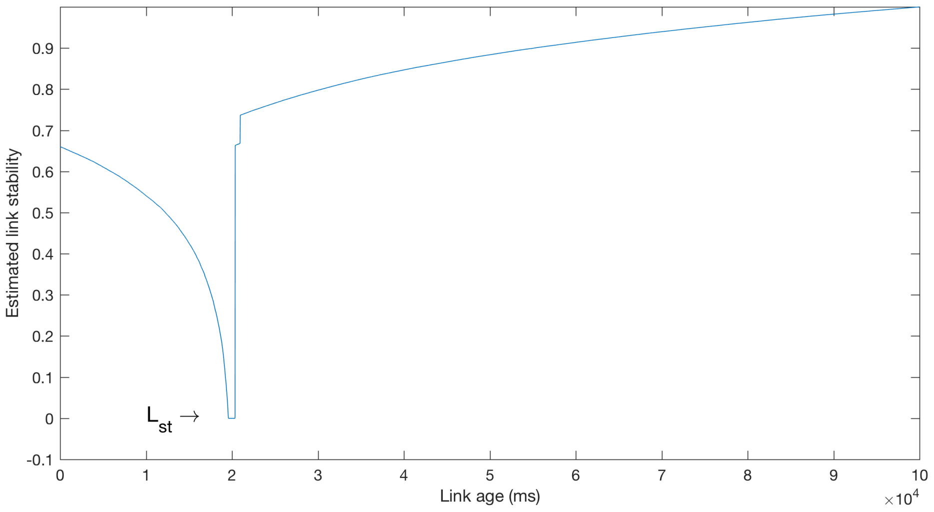


FIGURE 3.2: Normalized distribution of estimated stability of a link at a given age.

LLT value of the route is given by the product of the LLTs of links along the route, as described below

$$LLT_{route} = \prod_{(v_i,v_j)\in E} LLT_{(v_i,v_j)} \tag{3.12}$$

Finally, L_{st} is calculated as follows:

$$L_{st} = \frac{min(TR_{v_i}, TR_{v_j}) \times 2}{MS_{v_i} + MS_{v_j}} \tag{3.13}$$

in which TR_{v_i} and TR_{v_j} are transmission ranges of nodes v_i and v_j, respectively. If two nodes have different transmission ranges, only the shorter transmission range is used to calculate L_{st}. MS_{v_i} and MS_{v_j} are moving speeds of nodes.

Example 3.5. *Now, CNT also desires to estimate the link stability in the Professor John's network. For the sake of simplicity, the transmission range and the moving speed of nodes are standardized in 250m and 2 m/s respectively. Thereby, using the Equation (3.13), the measured* L_{st} *is 125 seconds. Table 3.7 contains the estimated values of LLT for each link and Table 3.8 shows the route cost using Equation (3.12). A simple* *normalization is applied to fit the LLT values within a range between 0 and 1 (Table 3.7). Note the LLT is a metric that works in maximization context.*

TABLE 3.7: Estimated link lifetime values for all links.

Link	Age (s)	LLT (s)
{A,B}	120	0.699
{A,C}	90	1.544
{A,E}	10	2.061
{B,C}	300	2.653
{B,D}	126	0.398
{C,D}	420	2.799
{D,BS}	81	1.644
{E,F}	20	2.021
{E,BS}	250	2.574
{F,BS}	150	2.352

TABLE 3.8: Available routes with LLT metric.

Sol.	Route	LLT
1	A → B → C → D → BS	$0.108 \times 0.806 \times 0.858 \times 0.445 = 0.033$
2	A → B → D → BS	$0.108 \times 0 \times 0.445 = 0$
3	A → C → D → BS	$0.409 \times 0.858 \times 0.445 = 0.156$
4	A → E → F → BS	$0.594 \times 0.580 \times 0.698 = 0.240$
5	A → E → BS	$0.594 \times 0.777 =$ **0.462**

The chosen solution has the newest link $(A \rightarrow E)$ *and one link already stable* $(E \rightarrow BS)$*. Note something interesting in 2nd solution. According to the estimation, there is a link that is close to dying* $(A \rightarrow B)$ *and another that can break at any moment* $(B \rightarrow D)$*. The latter makes the route unfeasible. This solution becomes feasible again when the links in question turn stable.*

More details about other stability-aware methods can be found in [29].

3.4.5 Residual Energy Cost (REC)

An Ad-hoc Wireless Networks can be composed of nodes with limited energy. Consequently, the network lifetime depends on the residual energy of nodes. This factor is especially relevant in Wireless Sensor Networks (WSNs) when there is no possibility of recharging. For example, in rescue scenarios, if a node discharges its battery, vital information will likely be missed. In this context, a well-known and straightforward measurement is the node residual energy [10, 19]. Herein, this measurement is arranged so that it becomes a cost metric, as follows

$$REC_{v_i}^{t} = \frac{EC_{v_i} - RE_{v_i}^{t}}{EC_{v_i}} \tag{3.14}$$

in which $RE^t_{v_i}$ is the estimated residual energy of node v_i at the end of time interval t and EC_{v_i} is the energy capacity of node v_i. The greater the node energy level, the lowest the cost of choosing it as a relay. In [24], the authors argue nodes can have different power consumption for packet transmission. So, Equation (3.14) becomes

$$REC2^t_{v_i} = TE_{v_i} + (\mu \times \gamma \times REC^t_{v_i}) \tag{3.15}$$

in which TE_{v_i} is the energy consumed by the node when transmitting a unit length of packet, μ is a scaling factor for normalizing the range of $REC^t_{v_i}$ relative to TE_{v_i}, and γ is a term to control the relative significance of $REC^t_{v_i}$ related to TE_{v_i}. However, if the nodes have the same power consumption, Equation (3.14) is satisfactory. REC value of the route is given by the sum of REC values along the route, as it is described below

$$REC_{route} = \sum_{(v_i)\in E} REC^t_{v_i} \tag{3.16}$$

Example 3.6. *The Professor John's network is an example of a harsh scenario, wherein the network needs to work as long as possible, but there is no place to recharge the devices. CNT knows that is crucial to take the energy factor into account in routing decisions. For the sake of simplicity, the power consumption is the same for all nodes. Let's consider the per-node energy capacity (EC) equal to 400 Joules. Taking fictitious examples of residual energy values after some time of network operation, Table 3.9 contains the estimated RECs for each node and Table 3.10 shows the RECs for each route.*

TABLE 3.9: Estimated REC values for all nodes.

Node	Residual Energy (RE)	Residual Energy Cost (REC)
A	383	$REC_A = {}^{(400-383)}/_{400} = 0.0425$
B	350	$REC_B = {}^{(400-350)}/_{400} = 0.1250$
C	318	$REC_C = {}^{(400-318)}/_{400} = 0.2050$
D	322	$REC_D = {}^{(400-322)}/_{400} = 0.1950$
E	369	$REC_E = {}^{(400-369)}/_{400} = 0.0775$
F	380	$REC_F = {}^{(400-380)}/_{400} = 0.0500$

TABLE 3.10: Available routes with REC values.

Sol.	Route	REC
1	A → B → C → D → BS	$0.0425 + 0.125 + 0.205 + 0.195 = 0.5675$
2	A → B → D → BS	$0.0425 + 0.125 + 0.195 = 0.3625$
3	A → C → D → BS	$0.0425 + 0.205 + 0.195 = 0.4425$
4	A → E → F → BS	$0.0425 + 0.077 + 0.050 = 0.1700$
5	A → E → BS	$0.0425 + 0.077 = \mathbf{0.1200}$

Table 3.11 lists all metrics presented up to now. When trying to minimize energy consumption, the effect is the generation of routes with few hops. Also, nodes with higher energy supply are more often chosen because they have the lower cost.

This metric does not equate to minimize the number of hops or even end-to-end delay. For example, let's suppose that the GROUP E of students, unfortunately, forgot to recharge the device. With the battery in half (200 *Joules), the cost of the routes 4 and 5 are equal to* 0.5925 $(0.0425 + 0.50 + 0.050)$ *and* 0.5425 $(0.0425 + 0.50)$*, respectively. Hence, the lower energy autonomy of the device of the GROUP E weakened the 4th and 5th solutions. After that, the 2nd solution becomes the best one, even though it has neither the minimum number of hops nor the minimum end-to-end delay.*

Table 3.11: Available routes with ETX, ML, MD, LLT, and REC values.

Sol.	Route	Hops	ETX	ML	MD(s)	LLT	REC
1	A → B → C → D → BS	4	4.138	0.125	0.076	0.033	0.567
2	A → B → D → BS	3	3.203	0.178	0.074	0.000	0.363
3	A → C → D → BS	3	**3.127**	**0.116**	0.078	0.156	0.443
4	A → E → F → BS	3	3.262	0.217	0.083	0.240	0.170
5	A → E → BS	**2**	3.163	0.570	**0.071**	**0.462**	**0.120**

3.5 Objectives *vs.* Metrics

QoS-aware metrics are much more sensitive to the variations, disconnections, and changes of topology. These metrics tend to have better performance than *hop-count* metric [43, 52–54]. Table 3.12 summarizes the association "objective *vs.* metric" considering the earlier mentioned objectives and metrics. About the objectives, the table lists: i) design requirements for the routing algorithm (what?), ii) reasons for meeting these requirements (why?), and iii) possible alternatives to meet these requirements (how?). Furthermore, the metrics capable of helping to reach these objectives are marked according to some works found in the literature.

Some metrics address objectives that tend to be conflicting. For example, when a routing protocol uses MD, the propensity is to reduce the path length. In general, this leads to an augment in the packet loss ratio. When a routing protocol uses ML, routes with closer relays are more likely to be generated because of the signal quality. So, the number of hops tend to increase, which can lead to more significant latency.

Some studies have shown that reliability-oriented and link stability-aware metrics are usually complementary [19, 24, 45, 59]. For example, ML aims to minimize the packet loss ratio, while LLT strives to maximize the route lifetime. Usually, links with higher packet delivery rates are more stable. More stable links tend to lose fewer packets [55]. However, optimizing ML does not mean to optimize LLT and vice versa since they are different metrics. Moreover, in wireless networks, congestion and interference influence more on the ML. On the other hand, mobility and sudden link breakages affect more the LLT [13].

In this context, utilizing two or more metrics can be necessary. Precisely for this reason, all metrics are marked in the last two lines in Table 3.12. So, any combination between them can mean some trade-off or a complement between objectives.

In this dissertation, it is proposed a framework that arranges and incorporates these metrics into the routing protocol. The goal is to support several optimization objectives and demonstrate that the network quality indicators can be improved, in terms of trade-off, when multicriteria approaches are utilized.

Table 3.12: Objectives and metrics.

Objectives			**Metrics**					
What?	**Why?**	**How?**	*Hop-count*	MD	ML	ETX	REC	LLT
Minimizing number of hops [16, 52]	Shorter routes may decrease the routing end-to-end delay and the contention need.	By reducing the number of hops of the routes.	✓	✓				
Minimizing end-to-end delay [17, 24, 52, 63, 67]	Low latency is an essential QoS indicator for real-time applications.	By reducing the number of hops of the routes or estimating the links delay.		✓				
Minimizing packet loss ratio and maximizing throughput [16, 52, 62]	Taking routes with a sound assurance of packet delivery helps to decrease drops and retransmissions.	By considering the packet delivery rates (or packet delivery probabilities) of links			✓	✓		
Minimizing energy consumption [10, 24, 45, 48, 60]	Selecting routes with energy autonomy avoid packet drops and retransmissions caused by energy full depletion and, ultimately, it results in a greater the network lifetime.	By avoiding certain nodes from being overloaded as relays and selecting stable routes in order to decrease retransmissions.					✓	✓
Maximizing route lifetime [19, 24, 29, 45, 60]	Stable routes decrease the packet drop rates mostly in mobile networks or when the data flow is high.	By capturing the network load or even estimating the links lifetime.						✓
Considering performance trade-offs [19, 23, 24, 27, 60, 67]	The targeted objectives can be conflicting, which can require to find a proper trade-off solution.	By utilizing some multicriteria technique in order to reach a compromise between the targeted objectives.	✓	✓	✓	✓	✓	✓
Combining complementary metrics [19, 24]	The targeted objectives can be correlative, thus, the quality indicators can improve if the metrics are merged.	By utilizing some multicriteria technique to combine complementary metrics.	✓	✓	✓	✓	✓	✓

Chapter 4

Multi-objective Approaches

4.1 Presentation

This chapter introduces the main strategies to deal with multi-objective problems in Ad-hoc Wireless Networks. Besides that, the QoS-aware metrics earlier mentioned are arranged using scalar approaches. Initially, the first section seeks to analyze the characterization of these problems. The graph structure described in Chapter 1 is adopted.

4.2 Multi-objective Optimization Problem

Many real-world problems are grounded in the idea of optimizing multiple objectives. For instance, two of the primary objectives of a company or business, namely to minimize costs and maximize performance (production), tend to conflict in many scenarios [68].

In a multi-objective approach, the fundamental purpose is to try to optimize two or more objectives at the same time without violating the constraints imposed in the problem formulation. This problem can be seen *a priori* as an expansion of the single-objective problem:

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & [f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_k(\mathbf{x})] \\ \text{subject to} \quad & \mathbf{x} \in S \subseteq \mathbb{X}, \end{aligned} \tag{4.1}$$

in which k is the number of objective functions, S is the feasible set, $\mathbf{x}$ is the decision vector, and $f_i(\mathbf{x})$ describes the scalar value of the i-th objective [21].

To see how this affects the decision process, consider the **Field Day** example presented in the previous chapter.

Example 4.1. *In the first part, CNT was working to enable the use of interactive real-time communication on the network. This application demands a short end-to-end delay. So, minimizing the number of hops was the criterion employed to get the best route. However, students also need to exchange information about the data gathering, such as customs, lifestyle, and organization of the birds to classify them into species, gender, and families.*

Thus, it is necessary to guarantee reliability and integrity of the data. In time, "minimization of packet loss" is included as another objective. The larger the distance between the nodes, the more the probability of packet loss, since the link tends to fade as the nodes are getting away. According to this objective, nodes closer are preferable. Table 4.1 complements Table 3.1 with a new estimation of packet loss ratio (ML[1]).

Table 4.1: Available routes with the number of hops and packet loss probability.

Solution	Route	Number of Hops	Packet Loss Ratio (ML)
1	A → B → C → D → BS	4	0.12 (12%)
2	A → B → D → BS	3	0.19 (19%)
3	A → C → D → BS	3	0.22 (22%)
4	A → E → F → BS	3	0.29 (29%)
5	A → E → BS	2	0.25 (25%)

Figure 4.1 shows the conflict in a graphical form. For each route, the horizontal axis gives its number of hops, and the vertical axis gives its probability of packet loss.

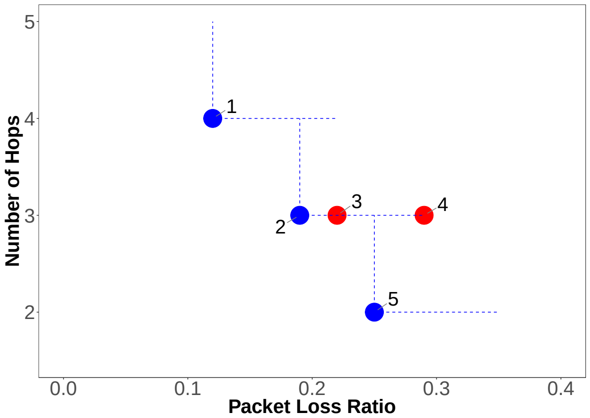


Figure 4.1: **Field Day example**. For each route (according to Table 4.1), the figure shows the packet loss ratio (horizontal axis) and the number of hops (vertical axis).

The 3rd and 4th solutions are not good options because the 2nd option provides a smaller packet loss ratio with the same number of hops. The 5th option also overcomes the 4th one because it has a shorter path and inferior packet loss ratio. However, among the 1st, 2nd, and 5th solutions, which one is the best? These three solutions are called incomparable (or non-dominated) and compose the approximated Pareto front [69].

---

[1] This metric is detailed in section 3.4.2.

4.3 Multi-objective Optimization Methods

When designing routing algorithms that consider multiple quality requirements, many studies have given preference to scalar approaches. Since mobile devices have limited processing and power resources, the algorithm needs to run as fast as possible. These methods turn the multi-objective problem into a single-objective problem. Thus, the vector-driven approach gives way to the usual scalar approach [21]. This section highlights more deeply two scalarizing methods, namely, weighted sum and compromise.

4.3.1 Weighted Sum method

In this method, the vector function of Equation (4.1) turns into a scalar function by aggregating its objectives through a weighted sum [21]. The problem then becomes

$$\begin{aligned} &\text{minimize} \quad && f_w(\mathbf{x}) = \sum_{i=1}^{k} w_i \times f_i(\mathbf{x}) \\ &\text{subject to} \quad && \mathbf{x} \in S, \end{aligned} \tag{4.2}$$

wherein the weights must respect the following relation: $w_i \geq 0$, $\forall i \in \{1, \ldots, k\}$ and

$$\sum_{i=1}^{k} w_i = 1 \tag{4.3}$$

in which w_i is a non-negative weight attached to the “i-th” objective. These weights play the role of identifying the relative importance of the quality metrics. In other words, the vector of weights “speaks about” the decision maker’s preferences. When there is no a clear indicator about these preferences, it is common to generate several vectors of weights, which can uniformly, to produce a set of representative solutions.

Example 4.2. *In* ***Field Day*** *Example (3.1), if Professor John thinks that delay is more important than packet loss ratio, he can assign* $w_1 = 0.8$ *for the* MD *metric and* $w_2 = 0.2$ *for the* ML *metric. It means roughly that “reducing latency deserves 80% of his attention”. On the other hand, if he wants delivery reliability, he can set* $w_1 = 0.2$ *for the* MD *and* $w_2 = 0.8$ *for the* ML*.*

The main drawbacks of this method are [21]:

- Fine-tuning of the weights is a complex task, and it can be by itself an optimization problem.

- Weights uniformly distributed does not guarantee a uniform distribution of the points in the front.
- The method is not able to reach points in non-convex regions of the front.

Many new and reformed protocols have emerged based on this approach. The main reasons are i) convenience for combining multiple metrics; ii) the preferences can be assigned relatively through the weights; iii) ease of incorporating into the routing protocols.

In a Wireless Mesh Network (WMN), the authors in [23] developed a *Dijkstra*-based weighted sum minimization algorithm to meet two objectives: minimizing end-to-end delay and maximizing throughput. In [24], the authors designed a utility function that employs normalized weights for two metrics: delay and energy. The proposal was tested in a Mobile Ad-hoc Network (MANET). Both studies utilize OLSR as routing protocol.

Applied to a MANET, the authors in [27] derived two metrics, namely, Secrecy Outage Probability (SOP) and Connection Outage Probability (COP). The former is concerned to meet security requirements, and the latter seeks to address different QoS requirements. The model uses weights to assign preference to the metrics. In [25], a weighted sum metric was modeled considering ETX and residual energy as quality measures.

Focusing in WSNs, the authors in [70] derived a metric that combines queuing delay, residual energy, and distance from the source node to the neighbors and the sink node. The proposal employs the outranking method TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) to make the routing decision. The weights were chosen giving preference to the residual energy metric. In [49], the authors seek to improve the reliability of the OLSR protocol using multiple metrics. The weights to ML and ETX metrics are defined accordingly to three different type of applications.

In an opportunistic route discovery process over a WSN, the authors in [19] propose to evaluate the quality taking into account some context criteria, such as signal quality, geographic progress, and node residual energy. A weight is assigned to each criterion, composing the metric called Dynamic Forwarding Delay (DFD). The authors in [67] investigate the trade-off between energy consumption and E2ED when selecting cluster-heads in WSNs. A weighted metric is introduced to choose the cluster-head. To compute the routes, a depth-first search (DFS) algorithm was implemented.

The weighted sum method is widely used for dealing with more than one QoS-aware metric in ad-hoc networks, regardless of routing paradigm or route optimization algorithm. In fact, it is a valuable strategy that can be explored in a vast number of applications.

4.3.2 Compromise method

This method is also called "ϵ-constraint method" [21]. Herein, instead of merging the objective functions, the idea is to transform a multi-objective problem into a single-objective problem using additional constraints. The steps to perform this transformation are summarized next [21]:

1. the objective function with "higher priority" is chosen to be optimized. The objective function to be optimized has the index 1.
2. a vector of constraints that plays the role of upper bound is defined as ϵ_i, $i \in \{2, \ldots, k\}$, $\epsilon_i \geq 0$.
3. the problem is remodeled by establishing the objective function to be optimized and transforming all other objective functions into inequality constraints as follows:

$$\begin{aligned} \text{minimize} \quad & f_1(\mathbf{x}) \\ \text{subject to} \quad & f_2(\mathbf{x}) \leq \epsilon_2 \\ & \vdots \\ & f_k(\mathbf{x}) \leq \epsilon_k \\ & \mathbf{x} \in S, \end{aligned} \tag{4.4}$$

In this method, the approximated front is obtained by altering the upper bounds ϵ_i. For example, ϵ_2 would have values between $\epsilon_{2_{min}}$ and $\epsilon_{2_{max}}$. Since one varies ϵ_2, a new point is generated when optimizing f_1.

Example 4.3. *Looking back over the discussion "delay* vs. *delivery reliability" of* ***Field Day*** *Example (4.2), the following model can be defined:*

$$\begin{aligned} \textit{minimize} \quad & MD_{route} \\ \textit{subject to} \quad & ML_{route} \leq \epsilon \\ & \boldsymbol{x} \in S, \end{aligned} \tag{4.5}$$

This model can be used to design quality policies. In this example, the desired delivery reliability may be modeled as: 1) high: $\epsilon = 0.05$*; 2) average:* $\epsilon = 0.1$*; 3) fair:* $\epsilon = 0.2$*.*

What if there is no any route that meets this constraint ϵ? In Ad-hoc Wireless Network, such a condition means that at that time, a feasible route will not be found. In short, when the designer has a reasonable clarity about the network characteristics and the bounds to be reached regarding a given objective, this technique can be more effective since it can achieve non-convex regions of the *Pareto front* [21]. When the bounds and the network conditions are uncertain, finding the proper ϵ for one metric can be as tricky as finding well-suited weights in weighted sum method.

Multi-Constrained Shortest Path (MCSP) problem [71] is NP-complete [72]. Because of that, the constrained model presented in Equation (4.5) can be relaxed. The restriction $ML_{route} \leq \epsilon$ becomes $ML_{(v_i,v_j)} \leq \epsilon$. This strategy is known as a pruning procedure [23, 49]. Pruning is a process of eliminating links that have quality values smaller than a given threshold [49]. The idea is to optimize one metric (e.g., MD) and then to prune the tree locally by discarding links that violate the bound assigned to another metric (e.g., ML). The shortest path algorithm with pruning is fast and has relatively small computational complexity [73].

If $ML_{(v_i,v_j)}$ is over-estimated, or the ϵ is underestimated, the route with maximum reliability may be discarded before the destination node is handled. Hence, the proper estimation of the quality metric and the ϵ value are critical for the pruning algorithms [74]. In next chapter, a new procedure is proposed to choose a suitable ϵ value.

4.3.3 Hybrid method

It is possible to combine the weighted sum and the compromise methods into a new one called "hybrid method" [21]. Now, Equation (4.1) is modified the following way:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{k} w_i \times f_i(\mathbf{x}) \\ \text{subject to} \quad & f_j(\mathbf{x}) \leq \epsilon_j, j = 1, \ldots, k \\ & \mathbf{x} \in S, \end{aligned} \tag{4.6}$$

This method allows holding the advantages of each method. Besides that, it can be used to solve both convex and non-convex problems. However, getting well-tuned parameters is even tougher [21].

4.4 Proposed Multi-objective Models

Two multi-objective models are proposed in this dissertation considering the mentioned scalarizing methods and the shortest path problem. In formulations, $QM^{(1)}$ and $QM^{(2)}$ represent some of those metrics presented in section 3.4, and $QM^{(3)}$ stands for the *hop-count* metric. The models are written as:

1) **Pruning Model** (ρ)

$$\begin{array}{rlll} \min & QM^{(1)}_{route} & & \\ \text{s.t.} & QM^{(2)}_{(i,k)} & \leq & \epsilon, \;\; k \in N(v_i) \\ & QM^{(3)}_{route} & < & \theta + [\theta \times \sigma] \end{array} \tag{4.7}$$

2) **Hybrid Model** (h)

$$\begin{array}{rlll} \min & w_1 \times QM^{(1)} + w_2 \times QM^{(2)} & & \\ \text{s.t.} & QM^{(3)}_{route} & < & \theta + [\theta \times \sigma] \end{array} \tag{4.8}$$

The first model is based on compromise method, but with a pruning mechanism. With this treatment, it is hoped to make sure that $QM^{(1)}$ is optimized without the existence of any link or node of the route that violates the bound imposed to the $QM^{(2)}$. Thus, $QM^{(1)}_{(route)}$ is the primary quality metric and $QM^{(2)}_{(i,k)}$ is the secondary quality metric. The scalar term ϵ represents the constraint that $QM^{(2)}_{(i,k)}$ must satisfy to try to be part of the route. Therefore, it defines which links are pruned of the neighborhood $N(v_i)$.

The other constraint is proposed to avoid overgrowth of the path length. The variable θ is the number of hops of the current route, and $\sigma \in \{0, 1\}$ is the *hop control factor*. This treatment allows the route to be updated only if the number of hops of the new route ($QM^{(3)}$) is lower than $\theta + [\theta \times \sigma]$ hops of the current one. This particular control seeks to i) avoid excessive contention need in the channel, ii) reduce interference, iii) save energy, and iv) decrease the link breakages.

The second model is based on weighted sum method, in which the $QM^{(1)}$ and $QM^{(2)}$ quality metrics compound the objective function to be optimized. However, this model is considered hybrid because, just like pruning model, it includes the constraint to avoid overgrowth of the path length.

Finally, to assure the connectivity, the remaining constraints are needed [26].

$$\sum_{(v_i,v_j)\in E} x_{i,j} - \sum_{(v_j,v_i)\in E} x_{j,i} = 0, \;\; \forall i \in V \backslash \{SN, DN\} \tag{4.9}$$

$$\begin{array}{l} \displaystyle\sum_{(SN,v_j)} x_{SN,j} = 1, \\ \displaystyle\sum_{(v_j,DN)} x_{j,DN} = 1, \end{array} \tag{4.10}$$

$$x_{i,j} \in \{0, 1\}, \;\; \forall (v_i, v_j) \in E \tag{4.11}$$

The constraint (4.9) imposes the flow conservation and the set of constraints (4.10) ensures that exists at least one path between the source node and the destination node. Lastly, the variable $x_{i,j}$ is equal to 1 if the link (i, j) integrates the route and 0, otherwise.

The QoS-aware metrics are arranged according to these two formulations. The compositions obtained are analyzed and compared with the traditionally weighted sum and single-objective methods. The methods are listed in Tables 4.2 and 4.3 according to two batches of comparisons. The first batch has methods that do not apply MD metric. This partition allows a fairer comparison since MD adds extra routing packets in the network. Besides, each batch has different objectives that are better detailed in next chapters. The prefixes ρ, h and ω are used to label the arrange when using **pruning**, **hybrid**, or **standard weighted sum** models, respectively.

Table 4.2: Batch of comparisons between methods that do not use MD metric.

Batch 1	
Method	Design objective
$\rho REC\text{-}LLT$	Minimize the energy consumption and ensure the links of the routes do not violate the stability constraint
$\rho REC\text{-}ML$	Minimize the energy consumption and ensure the links of the routes do not violate the packet loss constraint
$hETX\text{-}REC$	Maximize the throughput and minimize the energy consumption **(with hop-count control)**
$hML\text{-}REC$	Minimize the packet loss and minimize the energy consumption **(with hop-count control)**
$\omega ML\text{-}REC$	Minimize the packet loss and minimize the energy consumption **(without hop-count control)**
ML	Minimize the packet loss

Table 4.3: Batch of comparisons between methods that use MD metric.

Batch 2	
Method	Design objectives
$\rho MD\text{-}LLT$	Minimize the end-to-end delay and ensure the links of the routes do not violate the stability constraint
$\rho MD\text{-}ML$	Minimize the end-to-end delay and ensure the links of the routes do not violate the packet loss constraint
$hMD\text{-}ETX$	Minimize the end-to-end delay and maximize the throughput **(with hop-count control)**
$hMD\text{-}ML$	Minimize the end-to-end delay and minimize the packet loss **(with hop-count control)**
$\omega MD\text{-}ML$	Minimize the end-to-end delay and minimize the packet loss **(without hop-count control)**
MD	Minimize the end-to-end delay

Chapter 5

Framework and Algorithms

5.1 Presentation

The idea of using multiple objectives applied to the routing problem over Ad-hoc Wireless Networks is not a novelty [44, 75]. However, it has been drawing attention over the last few years due to the increasing number of applications and scenarios that can be met thanks to the ad-hoc paradigm. Some recent works have explored these possibilities [24, 45, 46, 49, 60]. Nevertheless, some issues have been identified in these studies:

1. The works concentrate on proposing new quality metrics or new routing protocols to meet either a particular network scenario or an earlier defined quality requirement.
2. To deal with multiple design objectives, the routing problem is modeled, in most cases, using the standard weighted sum method.
3. Many works give preference for one of the QoS-aware metrics when defining the weights. For instance, in [10, 19, 24, 76], the weights that guarantee higher Packet Delivery Ratio (PDR) in preliminary experiments are chosen. In [45] and [70], the weights vector is defined giving preference to the residual energy metric. In [49], the weights are modified accordingly to the type of application. In some works, as in [25] and [27], this issue is not even discussed.
4. The works that deal with the routing problem in Static Ad-hoc Wireless Networks (SAWN) usually do not face the deployment problem [23, 25, 45, 67, 70]. However, optimal deployment of the nodes can contribute to guarantee a reasonable level of QoS and help to build more efficient routes [26, 46].

One of the goals of this work is to design a framework in which it is possible to incorporate any QoS-aware metric. Still, this proposal enables these metrics to be combined into multi-objective models, thus supporting multiple design objectives. Well-known quality metrics were implemented and included in the module to serve the routing protocol (section 3.4). Since these metrics cover the foremost design objectives (section 3.3), the module can be adapted to fit any scenario either turning on or turning off the metrics. Looking at issue number 1, this strategy can favor the definition of QoS policies that meet the requirements of different applications.

In relation to the issue number 2, two new models were proposed (section 4.4). The idea is to compare the performance of both models with the single-objective method and standard weighted sum method. An original multicriteria utility function is proposed to choose the weights and constraints in order to address the issue number 3. About the issue number 4, routing and deployment problems are considered in this work. For that, a multi-objective evolutionary algorithm was designed to solve the deployment problem *a priori*. In this chapter, all these propositions and interventions are detailed.

5.2 Structure of the Framework

Figure 5.1 presents the structure of the framework divided into numbered modules. This section details the implementation of each module.

5.2.1 Routing Messages (RM) module

Description: *OLSR uses these messages to build and maintain the topology.*
Proposal: *This module must modify the messages to broadcast the QoS-aware metrics.*

The framework uses OLSR as routing protocol. OLSR is a proactive protocol that, in a distributed way, transmits and collects information to achieve full awareness of the network. In short, every node regularly broadcasts HELLO messages to its neighbors. A HELLO message carries the address of the neighbor interfaces and the state of links (asymmetric or symmetric). The nodes obtain information about their 2-hop neighbors as these packets are exchanged.

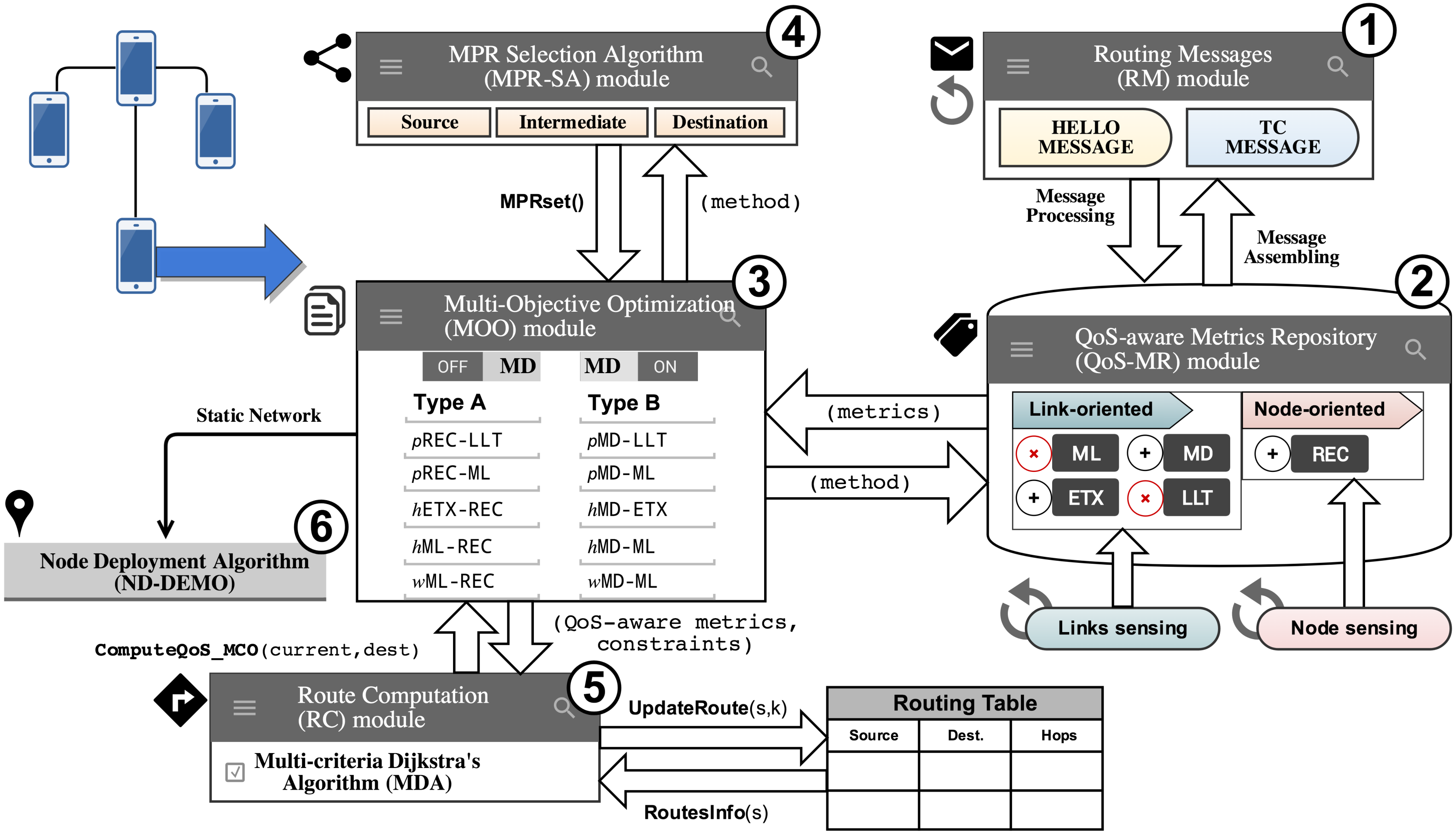


FIGURE 5.1: Structure of the framework.

The nodes use the Topology Control (TC) messages to broadcast their Multipoint Relays (MPRs) set in the network. Based on the information of the TC messages, the shortest path algorithm builds the routes. These routing packets were modified to share those QoS-aware metrics detailed in section 3.2.

5.2.1.1 HELLO messages

The HELLO messages are fired on a regular interval to detect neighbors and the state of the links [36]. Figure 5.2 illustrates the adjustment of the packet to convey the QoS-aware metric and the method information.

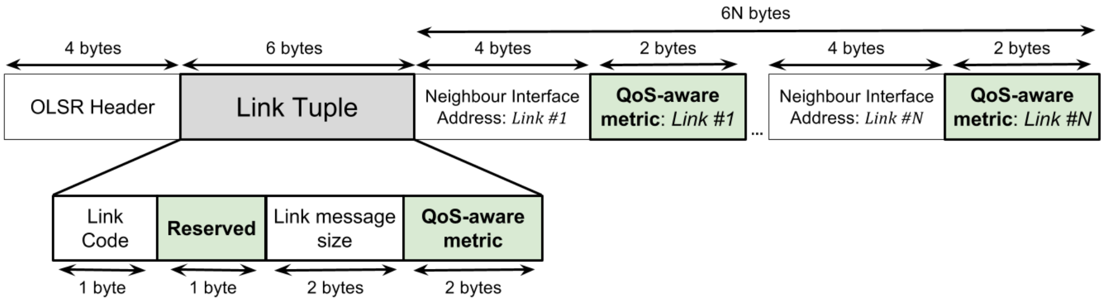


FIGURE 5.2: Adapted HELLO packet.

In Figure 5.1, *MOO* module "tells" to the *QoS-MR* module which multicriteria method is being used among those presented in Tables 4.2 and 4.3. Therefore, the protocol must advertise both *metric value* and *method information*. The field **QoS-aware metric** was created in the data structure to transmit the *metric value* accordingly to the method. As a floating-point value, it can be represented using 2 bytes. The *method information* is indicated by another field that contains the method and the pruning constraint[1]. Instead of creating another field, the one-byte **Reserved** field in the original HELLO message was occupied to reduce extra messaging overhead. The first four bits must inform the method, while the remaining bits must notify about the pruning constraint.

5.2.1.2 TC messages

TC messages are in charge of announcing the current link states considering all MPR selectors of the local node. Figure 5.3 illustrates the adjustment of the packet to convey the *metric value*.

If the Ad-hoc Wireless Network is heterogeneous regarding node speed or transmission range, the fields **Moving speed** and **Trans. range** are used to announce the moving

---

[1] Necessary information when the method uses the pruning mechanism

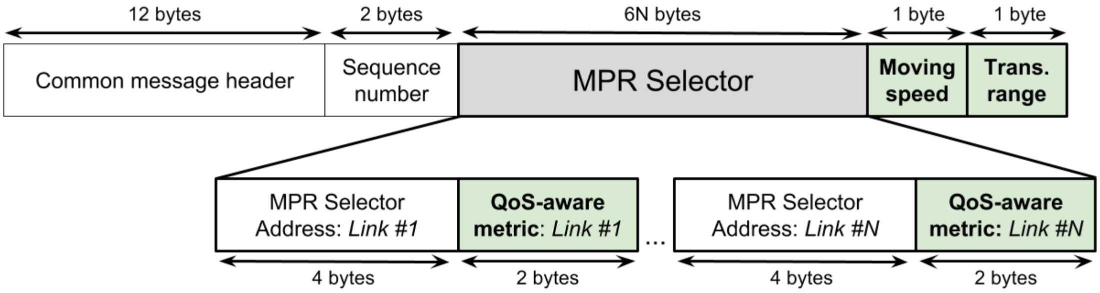


FIGURE 5.3: Adapted TC packet.

speed and the transmission range[2]. However, no extra cost is added since these data occupy the two bytes reserved in the original TC message. Finally, TC message is extended to carry the estimated *metric value* of its selectors through of the field **QoS-aware metric**.

About the routing overhead, the fixed header length includes 4 bytes of the OLSR packet header, 8 bytes of the UDP header, 20 bytes of the IP header, and 34 bytes of the MAC 802.11 header, thus totaling 66 bytes. Considering that an OLSR control packet encapsulates m HELLO and TC messages, its total length is then $66 + \sum_{k=1}^{m}(6 + 6N_k)$ and $66 + \sum_{k=1}^{m}(16 + 6N_k)$, respectively.

5.2.2 QoS-aware Metrics Repository (*QoS-MR*) module

Description: *Module in which the quality metrics are calculated and provided to the MOO module.*

Proposal: *This module must implement three types of QoS-aware metrics: i) delay-driven (MD), ii) reliability-oriented (ETX and ML), iii) link stability/lifetime-aware (REC and LLT). These metrics must be turned on or off according to the method selected in the MOO module.*

Observe there are twofold types of metrics: i) link-oriented and; ii) node-oriented. The former depends on the constant exchange of routing packets to assess the performance of the link. The latter, in turn, only needs local information (namely node residual energy). The computation and updating of these metrics are periodically realized through the functions *Links sensing* and *Node sensing*. The module performs this sensing when a new HELLO or TC message arrives (when the trigger *Message Processing* is fired). Also, before a new routing message is sent, the metrics are incorporated into the packet during the *Message Assembling* process and broadcasted together.

---

[2] These fields are used if the LLT metric is turned on. This information are necessary for the calculation of the L_{st} parameter (section 3.4.4).

Only the metrics utilized by the multicriteria method of the *MOO* module are computed and transmitted, thus reducing extra processing overheads. The *MOO* and *QoS-MR* modules use the parameters *method* and *metrics* to allow this communication. For example, the ρ*REC-LLT* method calculates and transmits only *REC* and *LLT* metrics.

5.2.3 Multi-Objective Optimization (*MOO*) module

Description: *Module in which the types of traffic are defined and, from this information, the quality metrics are arranged.*
Proposal: *This module must define which QoS-aware metrics and which multicriteria method will be utilized considering the type of traffic. Furthermore, it must execute the deployment algorithm to better place the nodes when the network is static.*

If the application can figure out what it needs, and reveal those requirements to underlying layers, some strategy can be put into practice so that these requirements are reached. Herein, two types of application are considered. **Type A** has two design objectives: minimizing packet loss rate (high reliability[3]) and maximizing throughput. **Type B** requires to minimize packet loss rate (high reliability) and minimize end-to-end delay. Looking at *MOO* module, the set of methods chosen to deal with **Type A** application has *MD* metric turned off and *REC* metric turned on. In another way, when dealing with **Type B**, MD metric is turned on, and the *REC* metric is turned off. Both metrics are also used in the expectation of increasing the network lifetime.

Type A represents multimedia applications with some tolerance to delay. Live Audio/Video Streaming applications[4] are good examples since they can reach this resilience through buffers [57]. **Type B** describes interactive real-time multimedia applications[5]. For instance, a video or an audio conference with an uncontrolled delay is annoying [57].

Based on the method, the *QoS-MR* module computes the required metrics periodically via HELLO or TC messages and provides the values to the *MOO* module. *MOO* receives and manages these values. Besides that, both *MPR-SA* and *RC* modules use the method information to compute the MPR set and the routes, respectively.

---

[3] For any data transmission network, some reliability is desired regardless of the application.

[4] A sender transmits audio/video to a group of receivers (e.g., Facebook Live).

[5] Two or more people interact in a real-time conversation (e.g., Google Hangouts).

5.2.4 MPR Selection Algorithm (*MPR-SA*) module

Description: *Multipoint Relays (MPR) set is a node-set that covers all nodes that are two hops away from the current node.*
Proposal: *When more than one 1-hop neighbors are covering the same number of uncovered 2-hops neighbors, the method defined in the MOO module must be used to choose which one will be selected.*

Proactive protocols have a drawback. They throw into the network a constant stream of packets to maintain topology information updated. In many cases, this ceaseless activity causes degradation in performance. OLSR protocol has been proposed to preserve the proactive feature while endeavoring to minimize the flooding of control messages.

The scheme is as follows: each node must elect a neighbor node set (MPR set). This set will be in charge of relaying its routing data to its 2-hop neighbors. So, instead of flooding the network with routing packets so that the message reaches the entire network, just MPR set must be used for this [36]. The challenging is to find the MPR set with the minimum number of nodes that covers all 2-hops neighbors of the current node. Figure 5.4 illustrates both mechanisms.

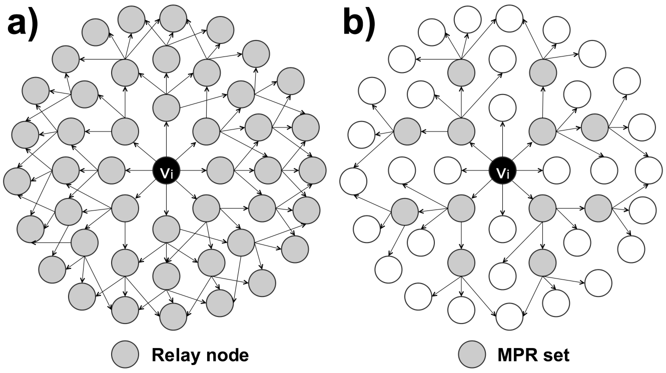


FIGURE 5.4: Flooding mechanisms for diffusion of a broadcast message up 3-hops: a) all nodes retransmitting and b) MPR node-set retransmitting.

With MPR mechanism, the number of retransmissions is remarkably reduced. However, the authors in [77] showed that finding the MPR set with minimal size is an NP-hard problem. Because of this, the standard OLSR builds this set through a heuristic that uses two criteria: i) only symmetric links are considered and ii) the 1-hop neighbor that covers the higher number of uncovered 2-hops neighbors is selected to be part of the set. This latter criterion is called the degree of reachability [36].

The algorithm below describes the heuristic. $N(v_i)$ is the set of 1-hop neighbors of the node v_i, $N^2(v_i)$ is the set of 2-hops neighbors of the node v_i, and $MPRset(v_i)$ is the set of multipoint relays of the node v_i. $R(v_j)$ is the degree of reachability of the node v_j, i.e., the number of nodes in $N^2(v_i)$ that are not yet covered by at least one node in the $MPRset(v_i)$, and that are reachable through v_j. $D(v_j)$ is the number of symmetric neighbors of the node $v_j \in N(v_i)$, excluding all the members of $N(v_i)$ and the v_i itself.

Algorithm 1 MPR set selection

```
 1: $MPRset(v_i) \leftarrow 0$
 2: Calculate $D(v_j)$, $\forall v_j \in N(v_i)$
 3: Add to $MPRset(v_i)$ those 1-hop neighbors in $N(v_i)$ that are the only covering some
    2-hops neighbor
 4: Remove from $N^2(v_i)$ these now covered 2-hops neighbors
 5: while $N^2(v_i)$ is not empty do
 6:     Calculate $R(v_j)$, $\forall v_j \in N(v_i)$
 7:     Add in $MPRset(v_i)$ the node $v_j \in N(v_i)$ with the greatest $R(v_j)$
 8:     if there is more than one node $v_j$ with the greatest $R(v_j)$ then
 9:         Add in $MPRset(v_i)$ the node $v_j \in N(v_i)$ with the maximum degree $D(v_j)$
10:     end if
11:     Remove from the $N^2(v_i)$ all nodes covered by this node $v_j$
12: end while
```

It was proved in [77] that the above heuristic is within a *log n* complexity factor. Since the MPR set is defined, the topology discovery is only made through these nodes. Thus, each MPR node regularly broadcasts TC messages to announce its list to the network. These TC messages help to build and maintain the routing tables. However, the MPR set calculation is based primarily on the degree of reachability.

The proposal is to take the QoS-aware metric as tiebreaker criteria (line 8). Suppose the protocol uses the *ρREC-LLT* method. Figure 5.5 shows that more than one 1-hop neighbors are covering the same number of uncovered 2-hops neighbors (M_1, M_2, M_3). Consider the required minimum stability ϵ[6] is equal to 0.60. Note the node M_2 has the lowest *REC* (0.20), but the *LLT* (0.55) violates the ϵ. Therefore, M_3 is selected because it does not violate ϵ and its *REC* is smaller than the *REC* of the M_1 node.

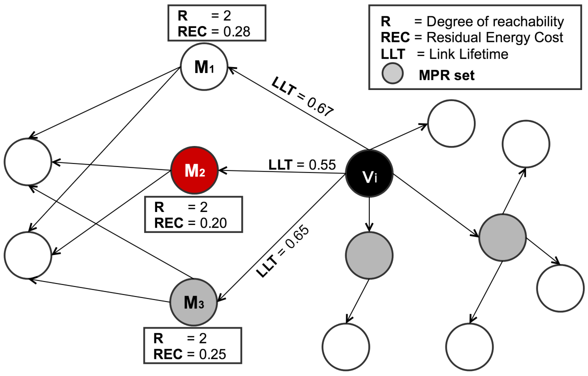


FIGURE 5.5: MPRs selection with *ρREC-LLT* method.

---

[6] As it was defined in the previous section, ϵ establishes a minimum quality value that needs to be guaranteed for a given solution to be considered feasible.

5.2.5 Route Computation (*RC*) module

Description: *Module that implements the shortest path algorithm.*

Proposal: *This module must provide the means to compute the routes taking into account multiple objectives.*

The default OLSR protocol uses the shortest path algorithm ($Dijkstra's$ algorithm) to find paths, having the number of hops as link state metric [36]. The Algorithm 2 is a modified version of this algorithm based on methods presented in Tables 4.2 and 4.3.

Algorithm 2 Route Computation (RC) algorithm

```
 1: function ROUTECOMPUTATION($V, s$)
 2:     $c[s] \leftarrow 0$                          ▷ Set the cost from $s$ to zero
 3:     for $i \leftarrow 1 : V$ do                  ▷ Update costs from 1-hop neighbors of $s$
 4:         if $i \in N(s)$ then                     ▷ $N(s)$ contains the 1-hop neighbors of $s$
 5:             $c[i] \leftarrow AnalyzeQoS(s, i)$   ▷ Analyze QoS of the 1-hop neighbors of $s$
 6:         else
 7:             $c[i] \leftarrow \infty$
 8:         end if
 9:         $ant[i] \leftarrow \varnothing$   ▷ $ant[i]$ is the previous node on the shortest path from $s$ to $i$
10:     end for
11:     $SP \leftarrow RoutesInfo(s)$   ▷ $SP$ contains nodes whose paths are already known
12:     $Q \leftarrow V$
13:     while $Q \neq \varnothing$ do                ▷ While there are nodes still not handled
14:         $j \leftarrow ExtractBestQoS(Q)$         ▷ Extract best node among those not handled
15:         for $k \leftarrow 1 : N(j)$ do           ▷ $N(j)$ is the neighbors set of the node $j$
16:             if $k \notin SP$ then   ▷ There is still no path that connects $s$ to $k$ ($c[k]$ is $\infty$)
17:                 $c[k] \leftarrow AnalyzeQoS(j, k)$   ▷ Update cost from $s$ to $k$
18:                 $ant[k] \leftarrow j$
19:                 $UpdateRoute(s, k)$                  ▷ Update Routing Table
20:                 $SP \leftarrow SP \cup \{k\}$
21:             else                            ▷ There is already a path that connects $s$ to $k$
22:                 $cost \leftarrow AnalyzeQoS(j, k)$
23:                 if $cost \neq \infty$ then               ▷ New route is better
24:                     $c[k] \leftarrow cost$               ▷ Update cost from $s$ to $k$
25:                     $ant[k] \leftarrow j$
26:                     $UpdateRoute(s, k)$                  ▷ Update Routing Table
27:                 end if
28:             end if
29:         end for
30:         $Q \leftarrow Q - \{j\}$
31:     end while
32: end function
```

Having s as source node, the *RC* algorithm executes twofold preliminary procedures: i) it analyzes at the start the QoS cost of 1-hop neighbors (line 5) and, ii) it takes information from routes that already exist (line 11). This latter procedure consults existing routes to compare with the new ones and then try to avoid unnecessary route updates. Now, when the algorithm finds a new route between s and k, basically it verifies if there is already a current route. If the route does not exist, the new route is included in the routing table (line 19). Otherwise, the algorithm analyzes if the new route is better than the current one (line 22). If the function returns a quality value not equal to ∞ (line 23), the new one is better and the routing table is updated (line 26).

Algorithm 3 details the function *AnalyzeQoS* and how the comparison between routes is made. Initially, some information are obtained from *MOO* module (line 6), namely,

Algorithm 3 New and current route cost comparison in terms of QoS

```
 1: function ANALYZEQOS($current, dest$)
 2:                                  ▷ $c[new]$ is the main QoS-aware metric of the new route
 3:                                  ▷ $\sigma$ is the hop control factor
 4:                                  ▷ $p[new]$ is the secondary QoS-aware metric used as pruning factor
 5:                                  ▷ $\epsilon$ is the minimum required for $p[new]$
 6:     $\{c[new], \sigma, p[new], \epsilon\} \leftarrow ComputeQoS\_MOO(current, dest)$
 7:
 8:                                  ▷ If there is already a route, check if new route meets the constraint(s)
 9:                                  ▷ If there is still not a route, just return the cost of the current route.
10:     if $c[dest] \neq \infty$ then                    ▷ There is already a path
11:                                  ▷ Check if the new route has not grown much and if it is worth upgrading it
12:         $\theta \leftarrow hops[dest]$               ▷ $\theta$ is the number of hops of the current route
13:         if $hops[new] \leq abs(\theta \times \sigma)$ then
14:                                  ▷ Check if the new route is better the current route in terms of QoS
15:             if $c[dest] > c[new]$ then
16:                 if $\epsilon \neq \varnothing$ then      ▷ $\rho$ methods
17:                     if $p[new] \geq \epsilon$ then       ▷ pruning
18:                         return $c[new]$      ▷ Route met $\epsilon$ constraint of the $\rho$ method.
19:                     else
20:                         $c[new] \leftarrow \infty$     ▷ Route violates $\epsilon$ constraint of the $\rho$ method.
21:                     end if
22:                 else                                 ▷ $h$ methods
23:                     return $c[new]$
24:                 end if
25:             else
26:                 $c[new] \leftarrow \infty$ ▷ Current route is better than new route in terms of QoS
27:             end if
28:         else
29:             $c[new] \leftarrow \infty$               ▷ Route has grown more than allowed
30:         end if
31:     end if
32:     return $c[new]$
33: end function
```

the new route cost $c[new]$ and the *hop control factor* σ. When an ρ method is used, the pruning factor $p[new]$ (local quality metric) and the ϵ constraint are retrieved.

RC algorithm uses the σ to prevent the sudden growth of the number of hops (line 13). The idea is to allow the path length grows in a controlled way. This treatment tries to: i) avoid excessive contention need in the channel, ii) reduce interference, iii) save energy, and iv) decrease the link breakages, which reduce the triggering of TC messages, consequently. The variable θ is the number of hops of the current route (line 12). Algorithm 4 covers the *ComputeQoS_MOO* function into *MOO* module.

Algorithm 4 Compute QoS according to the method

```
function COMPUTEQOS_MOO($current, dest$)
    $pruningMethods \leftarrow [\rho REC\text{-}LLT, \rho REC\text{-}ML, \rho MD\text{-}LLT, \rho MD\text{-}ML]$
    $nodeMetric \leftarrow [REC]$
    $multiplicativeMetric \leftarrow [ML, LLT]$
    $QMs \leftarrow GetInfoMetrics(method)$                ▷ Interface with the QoS-MR module
    for each $i \in QMs$ do                                ▷ For each Quality Metric
        if $QM^{(i)} \in nodeMetric$ then
            $QM^{(i)}_{(d)} \leftarrow QM^{(i)}_{(dest)}$                    ▷ Node-oriented metric
        else
            $QM^{(i)}_{(d)} \leftarrow QM^{(i)}_{(current,dest)}$            ▷ Link-oriented metric
        end if
    end for
    if $QM^{(1)} \in multiplicativeMetric$ then
        $QM^{(1)}_{route} \leftarrow QM^{(1)}_{(current)} \times QM^{(1)}_{(d)}$
    else
        $QM^{(1)}_{route} \leftarrow QM^{(1)}_{(current)} + QM^{(1)}_{(d)}$
    end if
                                                 ▷ Compute the cost of the new route
    if $method \notin pruningMethods$ then                 ▷ $h$ methods
        if $QM^{(2)} \in multiplicativeMetric$ then
            $QM^{(2)}_{route} \leftarrow QM^{(2)}_{(current)} \times QM^{(2)}_{(d)}$
        else
            $QM^{(2)}_{route} \leftarrow QM^{(2)}_{(current)} + QM^{(2)}_{(d)}$
        end if
        $c[new] \leftarrow \omega_1 \times QM^{(1)}_{route} + \omega_2 \times QM^{(2)}_{route}$
        $p[new] \leftarrow \varnothing$
        $\epsilon \leftarrow \varnothing$
    else                                                   ▷ $\rho$ methods
        $c[new] \leftarrow QM^{(1)}_{route}$
        $p[new] \leftarrow QM^{(2)}_{(d)}$                              ▷ pruning factor
        $\epsilon \leftarrow GetEpsilon()$
    end if
                ▷ $\epsilon$ is the pruning constraint and $\sigma$ is the hop control factor
    $\sigma \leftarrow GetSigma()$
    return $\{c[new], \sigma, p[new], \epsilon\}$
end function
```

The values of the QoS-aware metrics (QMs) are retrieved from the *QoS-MR* module using the *method* as a parameter in the $GetInfoMetrics$ function (line 5). The metric can be *node* or *link-oriented*. It can also be an addictive or multiplicative metric.

The hybrid methods (h) compute the end-to-end quality for both quality metrics (lines 13-17; lines 20-24). On the other hand, pruning methods (ρ) compute the end-to-end quality just for $QM^{(1)}$ since that $QM^{(2)}$ is treated as a pruning factor[7] (line 30). Ultimately, ω_i defines the weights of each metric, ϵ is the minimum quality when testing a link or node, and σ is the *hop control factor*. In the next chapter, the definition of these parameters is discussed.

5.2.6 Node Deployment Algorithm (ND-DEMO)

Description: *The multi-objective evolutionary algorithm that targets to deploy the nodes in the area by considering two objectives: i) maximizing the covered area and; ii) maximizing the packet reception ratio.*

Proposal: *When the network is static, this module must execute the algorithm with the aim of positioning the nodes properly.*

A telecom company wants to increase the extension of its access network. A mining company wants to design a network that enables communication in harsh and isolated regions as in underground tunneling. A given security department desires to deploy a sensor network to monitor its frontier. In such scenarios, the position of nodes matters. Thus, before the physical deployment, it is important to draw a proper layout so that the network can cover as maximum as possible of a given area.

In this context, an initial optimization process seeks to figure out the coordinates of the nodes in a two-dimensional plane. So, the decision variables are arranged in a vector $P = \{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n\}$ that contains the positions of the nodes. The layout is defined in such a way that satisfies the following objectives: i) maximizing the Packet Reception Ratio (PRR) and ii) maximizing the network coverage [26]. PRR is estimated based on the signal quality between the nodes as follows [19]

$$PRR_{(v_i,v_j)} = \left(1 - \frac{1}{2}\exp\left(-\frac{SNR_{(v_i,v_j)}}{2}\frac{B_N}{R}\right)\right)^{8L} \tag{5.1}$$

in which B_N is the noise bandwidth, R is the data rate in bits, and L is the packet size. These parameters are set to default values. $SNR_{(v_i,v_j)}$ is given by:

$$SNR_{(v_i,v_j)} = RSS_{(v_i,v_j)} - Pn \tag{5.2}$$

---

[7] $QM^{(2)}$ is a local quality metric

$RSS_{(v_i,v_j)}$ is the received signal strength in dB without noise as a function of the distance between v_i and v_j. It is computed as: $P_t - PathLoss_{(v_i,v_j)}$, in which P_t is the transmission power in dB and $PathLoss_{(v_i,v_j)}$ is the path loss in dB as a function of distance, which corresponds to the log-normal shadowing path loss model [78]. Pn is the sampled noise floor in dB. Thus, the first objective function is the average of PRRs considering all links, and it is computed as

$$f_1 = \frac{1}{n} \times \sum_{(v_i,v_j)\in E} PRR_{(i,j)} \tag{5.3}$$

in which n is the number of nodes.

To model network coverage, the whole area is fragmented in the form of s small squares. The center of each square is considered as a demand point, generating a set $D = \{1, 2, \ldots, s\}$ of demand points [26]. Each node has a coverage ratio defined. The demand point within of this radio is considered to be covered by such a node. The function below is derived to maximize the network coverage.

$$f_2 = \frac{\sum\limits_{i\in D} h_i}{|D|} \tag{5.4}$$

in which h_i assumes the value 1 (one) if the demand point i is covered. Otherwise, h_i assumes the value 0 (zero). At last, $|D|$ describes the cardinality of the set D.

Algorithm 5 is a modified implementation of the Differential Evolution for Multiobjective Optimization (DEMO) algorithm that was used to solve the previous problem [79].

Differential Evolution (DE) is a well-known population-based metaheuristic in which each candidate solution of the population (parent) is perturbed by adding a weighted moving vector and modifying the value of some randomly selected coordinates [80]. DEMO is a multi-objective version that applies some strategies inherited from specialized algorithms in solving multi-objective optimization problems like NSGA [79]. In the function *CreateCandidate* (line 6), a mutated individual is generated from perturbation applied to some individual of the population, as follows

$$\mathbf{y}_i = \mathbf{x}_{n_1} + \lambda \times (\mathbf{x}_{non\_dominated} - \mathbf{x}_{n_1}) + \beta \times (\mathbf{x}_{n_2} - \mathbf{x}_{n_3}) \tag{5.5}$$

in which $\lambda \in [0, 1]$.

In the differential mutation of the classical algorithm, the base vector $\mathbf{x}_{n_1}$ is extracted from the population at random way. A strategy that can be employed to promote convergence is to extract this vector from the non-dominated set at random way. However, in spite of suggesting faster convergence, this strategy might rapidly reduce diversity.

Algorithm 5 *DEMO*: Differential Evolution for Multiobjective Optimization

```
1: $pop \leftarrow Evaluate(pop)$        ▷ Evaluate population of individuals randomly generated
2: while stop condition is not met do
3:     $j \leftarrow 1$
4:     for $i \leftarrow 1 : N$ do
5:                                   ▷ Create candidate solution (offspring) from parent $pop_i$
6:         $new \leftarrow CreateCandidate(pop_i)$
7:         $new \leftarrow Evaluate(new)$
8:         if $new$ Pareto-dominates $pop_i$ then
9:             $pop_i \leftarrow new$                 ▷ offspring takes the place of the parent
10:        else if $pop_i$ Pareto-dominates $new$ then
11:            $new \leftarrow \varnothing$                              ▷ discard solution
12:        else
13:            $pop_{(N+j)} \leftarrow new$                 ▷ adds candidate in population
14:            $j \leftarrow j + 1$
15:        end if
16:    end for
17:    $pop \leftarrow Truncation(pop, N)$
18:    $FP \leftarrow GetParetoFront()$
19: end while
```

This behavior can provoke premature convergence, mainly when optimizing multimodal functions. To reduce this selective pressure, the above equation was proposed. Now, $\mathbf{x}_{n_1} + \lambda \times (\mathbf{x}_{non\_dominated} - \mathbf{x}_{n_1})$ represents the base vector which is generated from non-dominated solution. So, from now on, a point on the segment that connects $\mathbf{x}_{n_1}$ to $\mathbf{x}_{non\_dominated}$ is randomly constructed.

Another consideration is, in the classical *DE*, a constant value is assigned to the scale factor β. The author in [80] argues that varying β within a continuous interval can generate more diverse difference vectors, which possibly contributes to avoiding the algorithm stagnation. So, the β value is defined in each mutation following the Equation (5.6):

$$\beta_i = \beta_0 + \mathcal{U}_{[-\alpha,\alpha]}, \ i \in \{1, \ldots, n\} \tag{5.6}$$

in which α is a value very close to 0 that delimits the uniform interval that controls the variation. Note that β will vary around β_0. These two modifications allow the algorithm to be more effective in the convergence process without neglecting the diversity provision.

Concerned in dealing with the multi-objective aspect of the problem, the selection criterion of the *DEMO* is based on the Pareto dominance concept. Observe that the parent (pop_i) will only be replaced if the candidate solution (new) dominates it (line 8). When they are incomparable solutions, both stay alive (line 13).

The algorithm uses two criteria to make the cut: the non-dominated sorting and the crowding distance metric [79]. The first one consists of clustering the points using ranks of dominance. The non-dominated elements are assigned to the *rank* 1 *set*. After removing the points that belong to the *rank* 1 *set*, new non-dominated points are found and assigned to the *rank* 2 *set*. This process is repeated until all elements are ranked. After that, the individuals of the same front are evaluated according to the crowding distance metric [81]. For the i-th point in the objective space, closest neighbors are discovered. The crowding distance value is the semiperimeter of the hyper-rectangle formed taking these neighbors as vertices [81]. The truncation procedure (line 17) selects only the best N individuals regarding these two metrics and keeps them into the population for the next step of the algorithm. In [79], the authors argue that these steps help to preserve elitism and contribute to the uniform spread of solutions along the front.

Finally, from these best individuals, the estimated Pareto front is formed by the first front (line 18).

Chapter 6

Results and Discussion

After planning, developing and deploying the framework, this work proposes to investigate the performance of the proposed multicriteria approaches. Before discussing the results in section 6.4, section 6.1 introduces the network quality indicators that were used to evaluate the performance of the methods. The experimental design is detailed in section 6.2. Finally, the section 6.3 presents the utility function proposed to assist in the process of setting weights and constraints.

6.1 Network Quality Indicators

Through the network quality indicators, one seeks to answer how good a routing protocol or method is. In this study, the following indicators are utilized: Packet Loss Ratio (PLR), Throughput, End-to-End Delay (E2ED), Network Lifetime (NLT), Normalized Routing Load (NRL) and Packet Error Rate (PER). Their definitions are given next.

6.1.1 Packet Loss Ratio (PLR)

Packet Delivery Ratio (PDR) refers to the ratio between the data packets received by the destination and the total number of packets sent out by the source. In short, this indicator pictures the performance of the protocol regarding delivering data packets to the destination. This measure is outlined in (6.1) [47]:

$$PDR = \frac{1}{F} \sum_{i=1}^{F} \frac{r_i}{t_i} \tag{6.1}$$

in which F is the total number of connection flows, i is the unique flow identifier, r_i is the number of unique packets received from flow i, and t_i is the number of packets transmitted to flow i. The Packet Loss Ratio (PLR) is given by $1 - PDR$.

For any network, PLR is a paramount quality indicator. From a user's standpoint, it represents the delivery reliability. Likewise, from a network design standpoint, it can identify issues that might lead to a reduced throughput or even a high delay. Naturally, the performance is better when PLR is low.

6.1.2 Throughput

Throughput is the total packets successfully delivered to individual destinations over time. It is computed as follows [15]

$$Throughput = \frac{1}{F}\sum_{i=1}^{F}\frac{r_i}{simtime} \tag{6.2}$$

in which F is the total number of connection flows, i is the unique flow identifier, r_i is the number of unique packets received from flow i, and $simtime$ is the simulation time in seconds. The performance is better when throughput is high.

6.1.3 End-to-End Delay (E2ED)

E2ED is the average time delay that data packets take to leave the source and reach the destination. It includes processing, queuing, and propagation delay. This measure is obtained from (6.3) [15, 47]:

$$E2ED = \frac{1}{D}\sum_{i=1}^{D}(tr_i - ts_i) \tag{6.3}$$

in which D is the total number of data packets received, ts_i is the time at which data packet i is sent, and tr_i is the time at which data packet i is received. The performance is better when the E2ED is low, mainly when real-time applications are utilized.

6.1.4 Network Lifetime (NLT)

NLT is the network "remaining" lifetime after a period t of normal operating. Initially, the residual lifetime of all nodes is computed as follows [26]

$$lt_i = \frac{E_i}{e_i} \times simtime \tag{6.4}$$

For each node i, E_i is the initial energy capacity, e_i is the residual energy, and *simtime* is the simulation time in seconds. Such an operation generates a vector **lt** that contains the estimated remaining lifetimes to each node in seconds.

Concerning to the network lifetime, one can consider that a network stops to work in its maximum capacity when the first node breaks down due to full energy depletion. So, the network lifetime is given by [26]

$$NLT = \min(\mathbf{lt}) \tag{6.5}$$

This indicator has particular relevance when it comes to sensor networks or when devices can not have access to a recharge source. The method tends to get better results when it provides routes with few hops or considers the residual energy of nodes.

6.1.5 Normalized Routing Load (NRL)

NRL is defined as the number of routing packets transmitted per data packet delivered at the destination, as follows [82]

$$NRL = \frac{\sum P_r}{\sum P_d} \tag{6.6}$$

wherein P_r is the total routing control packets sent and P_d is the total data packets delivered. This indicator provides an idea of how much bandwidth is consumed transmitting routing packets. The method tends to get lower values of NRL when it does not require numerous exchanges of routing packets to compute the QoS-aware metrics. Moreover, the NRL can reduce when the method selects routes composed of more stable links since the number of routing updates will decrease.

6.1.6 Packet Error Rate (PER)

PER is the number of incorrectly received data packets divided by the total number of received packets. A packet is declared incorrect if at least one bit is corrupted [83]. The expectation value of the PER is denoted Packet Error Probability P_p, which for a data packet length of N bits can be expressed as [83]

$$P_p = 1 - (1 - P_b)^N = 1 - e^{N \log(1 - P_b)} \tag{6.7}$$

in which P_b is the expectation value of the bit error ratio. OMNET simulator provides some models to compute bit errors. Here, the Nist error rate model is utilized as recommended in [84]. PER is an important indicator since bit errors usually occur due to noise, collisions, and interference.

6.2 Experimental Design

In Chapter 3, the QoS-aware metrics (ETX, ML, MD, LLT, and REC) were introduced. In Chapter 4, two multicriteria methods (pruning and hybrid) were proposed. In Chapter 5, the framework was detailed and incorporated to the OLSR routing protocol. Through the framework, it was possible to obtain twofold benefits i) combining the metrics according to the proposed methods such that the proposal can support several optimization objectives; ii) automating the experiments.

In this chapter, the proposed methods are compared with the standard weighted sum method and with a single-objective method. The performance of each one is analyzed considering the earlier mentioned network quality indicators.

For purposes of statistical inference, the tests were performed with the following objectives: i) to determine, under a significance level α, if the method A is significantly superior (or inferior, or even not significantly different) to B considering the quality indicator I; ii) to assess the size of these differences (if they exist).

The tests are executed independently for three mobility scenarios and considering three packet transmission rates. For each of the ten independent runs of each method, the quality indicators are evaluated. The statistical analysis is then conducted using such values. The following statement is tested:

> *"The proposed method* ***m*** *has better performance than the single-objective method* ***s*** *considering the quality indicator* ***i****".*

The null hypothesis assumes the equality of the median values, against the alternative bilateral hypothesis. For example, the hypothesis for the comparison between the $\rho REC\text{-}ML$ and ML methods using the PLR quality indicator in the non-mobility (static) scenario is defined as follows:

$$\begin{aligned} H_0^{static;PLR;(\rho REC\text{-}ML,ML)} &: \tilde{\mu}^{static}_{\rho REC\text{-}ML,PLR} = \tilde{\mu}^{static}_{ML,PLR} \\ H_1^{static;PLR;(\rho REC\text{-}ML,ML)} &: \tilde{\mu}^{static}_{\rho REC\text{-}ML,PLR} \neq \tilde{\mu}^{static}_{ML,PLR} \end{aligned} \tag{6.8}$$

wherein *static* indicates the mobility scenario, $\tilde{\mu}^{static}_{\rho REC\text{-}ML,PLR}$ and $\tilde{\mu}^{static}_{ML,PLR}$ represent the median values of the PLR quality indicator considering that the protocol worked with the ρREC-ML and ML methods, respectively. Multicriterio methods are compared with the best single-objective method in the *all* vs. *one* schema.

A hypothesis is tested according to one of the after principles: i) if the assumptions of normality of residuals and homogeneity of variances are not violated, the one-way ANOVA is used to identify differences between populations and the pairwise *t-tests* are employed to size such differences; ii) otherwise, the Kruskal-Wallis non-parametric test is applied to identify differences between populations and the *Wilcoxon Rank-Sum* test is used to scale out these differences [85]. The significance level (α) is equal to 0.05.

6.3 Defining Constraints and Weights

In section 4.4, the proposed multicriteria methods are presented. Therein, it is possible to note that some parameters need to be defined. For the pruning method, it is necessary to specify the value of ϵ. Likewise, for the hybrid method, the values of w_1 and w_2 must be provided. In hindsight, ϵ serves as pruning factor of the neighboring nodes that do not meet the constraint related to a given metric. Also, w_1 and w_2 are parameters that represent the importance of each metric that composes the method.

Several works define these parameters, giving preference to one of the criteria. For example, in [19] and [24], the authors determined the weights according to the highest packet delivery rate. In [45] and [70], the weights vector is defined giving preference to the residual energy indicator.

In this study, a more general schema of choosing those parameters is proposed. The idea is to create a utility function that considers a set of quality indicators I. So, the set that maximizes this function for each method $m \in M$ is defined as the best one, as it is described in 6.9

$$
\begin{array}{rrcl}
\arg\max & U^m = \sum_{i=1}^{I} \tilde{\mu}_i^m \times e_i^m \times \eta_i & & \\
\text{subject to} & e_i^m & \geq & 0 \\
 & \sum_{i=1}^{I} \eta_i & = & 1 \\
 & \eta_i & \geq & 0 \\
 & \forall i & \in & I \\
 & \forall m & \in & M
\end{array}
\tag{6.9}
$$

The term e_i^m undergoes special treatment. For example, imagine there is a significant difference between m and *hop-count* for an indicator i that one wants to minimize (e.g., PLR). Since the effect size e_i^m describes the magnitude of the difference, if $\tilde{\mu}_i^m < \tilde{\mu}_i^{hop\text{-}count}$, e_i^m will be a negative value. It means that m has a positive effect on reducing the indicator i. Therefore, in order to guarantee $e_i^m \geq 0$, all indicators are handled within maximization context. Now, $e_i^m < 0$ means that m was not able to overcome the reference method in the indicator i.

Upon a running sequence of a given method m, $\tilde{\mu}_i^m$ is the median value of the i-th network quality indicator. When comparing m with a reference method, e_i^m represents the effect size that m causes considering the quality indicator i. In this dissertation, *hop-count* is utilized as the reference method.

At last, η_i is the weight that shows the importance of the i-th quality indicator, considered non-negative, and whose sum over all indicator is assumed to be 1. These weights represent the "relative importance" of the indicators. For instance, if one thinks that reducing the packet losses is more important than decreasing the $E2ED$, it can be defined $\eta_{PLR} = 0.8$ and $\eta_{E2ED} = 0.2$. Naturally, the others η's receive zero.

This proposal gives flexibility for the network designer since the protocol can be changed adjusting these preferences whenever necessary. It is also possible to enable or disable any indicator. For example, if the devices have a permanent energy supply, maximizing the network lifetime will not be a concern. Then, such a quality indicator can be turned off by assigning 0 (zero) to η_{NLT}.

For purposes of defining these parameters, 10 preliminary simulations were performed for each method considering the two proposed models (pruning and hybrid). It was used moderately severe network conditions with average packet generation rate equals to **30kbps** and node speed ranging between 0.5 and 4 m/s. Table 6.1 summarizes the other simulation parameters.

Table 6.2 lists the ϵ, w_1, and w_2 values defined according to the proposed utility function (U). Tables 6.3 and 6.4 show the effect sizes for cases in which significant differences were found. The last column presents the values of the utility function U. All indicators have the same preference ($\eta_i = 0.2$), except for the Throughput ($\eta = 0.0$) that, in the simulations, is strongly correlated with the PLR indicator.

In a macro-view, the methods that work with MD metric (delay-driven) do not bear any practical effect on the NRL indicator when compared to the *hop-count* method. On the contrary, the effect is negative [-]. That is an expected outcome, since MD calculation requires constant transmission of control packet pairs, as it is detailed in section 3.4.3. For this reason, the routing overhead tends to increase.

Table 6.1: Parameters of the preliminary simulations

Parameter	*Configuration*
Simulation area	500 m x 500 m
Simulation duration	200 seconds
Traffic flow	Constant bit rate UDP packets
Number of flows	20 IP unidirectional
Packet size	64 bytes
Number of nodes	50
Mobility pattern	Random Waypoint Mobility
Pause time between node moves	10 seconds
Per-node initial energy	400 J
Packet transmission power consumption	600 mW
Packet reception power consumption	395 mW
Idle power consumption	300 mW
HELLO message interval	2 seconds
TC message interval	5 seconds

Another pertinent observation is that the $hML\text{-}REC$ method is not significantly different from the *hop-count* method for the NLT indicator regardless of the weights set (Table 6.4). As all devices start their operations with the same amount of energy, the impact of the REC metric is slight.

Except for these two cases, multicriteria methods outperformed the *hop-count* in all other indicators. The weights of the standard weighted sum methods ($\omega ML\text{-}REC$ and $\omega MD\text{-}ML$) were defined based on the criterion applied in [19] and [24], namely, the smallest PLR value[1].

Table 6.2: Constraints and weights defined to each multicriteria method.

Type A application		**Type B application**	
Method (m)	Parameters (p)	Method (m)	Parameters (p)
$\rho REC\text{-}LLT$	$\epsilon = 0.8$	$\rho MD\text{-}LLT$	$\epsilon = 0.1$
$\rho REC\text{-}ML$	$\epsilon = 0.6$	$\rho MD\text{-}ML$	$\epsilon = 0.6$
$hETX\text{-}REC$	$w_{ETX} = 0.5, w_{REC} = 0.5$	$hMD\text{-}ETX$	$w_{MD} = 0.9, w_{ETX} = 0.1$
$hML\text{-}REC$	$w_{ML} = 0.9, w_{REC} = 0.1$	$hMD\text{-}ML$	$w_{MD} = 0.2, w_{ML} = 0.8$
$\omega ML\text{-}REC$	$w_{ML} = 0.7, w_{REC} = 0.3$	$\omega MD\text{-}ML$	$w_{MD} = 0.1, w_{ML} = 0.9$

Still looking at section 4.4, in both models, the parameter σ also need to be specified. In this study, σ is in the model to control the fast growth of the path length. It was defined that $\sigma = 0.51$ is a suitable value to do this control properly. In practice, this means that if a new route has a number of hops 50% (plus a little) greater than the number of hops of the current one, the routing table will not be updated.

---

[1] The greatest Packet Delivery Ratio (PDR).

Table 6.3: Comparison between *hop-count* and pruning (ρ) methods. Effect size: not-significant effect (×) and negative effect (−).

	p=ϵ	PLR	E2ED	PER	NRL	NLT	U
$\rho REC\text{-}LLT$	0.1	0.392	0.384	0.239	0.282	0.043	0.179
	0.4	**0.418**	0.395	0.234	0.302	**0.123**	0.200
	0.5	0.404	0.381	0.265	0.309	×	0.186
	0.6	0.367	0.392	0.283	0.309	×	0.184
	0.7	0.350	0.359	0.297	0.308	×	0.174
	0.8	0.393	**0.401**	**0.321**	**0.361**	×	**0.208**
$\rho REC\text{-}ML$	0.1	0.373	0.376	0.234	0.276	0.044	0.168
	0.2	0.373	0.376	0.260	0.304	0.019	0.173
	0.3	0.351	0.372	0.231	0.296	0.061	0.167
	0.4	0.334	0.344	0.222	0.291	×	0.150
	0.5	0.338	0.353	0.184	0.310	0.060	0.157
	0.6	**0.409**	**0.403**	0.207	**0.359**	0.074	**0.199**
	0.7	0.338	0.368	**0.260**	0.296	**0.083**	0.172
$\rho MD\text{-}LLT$	**0.1**	**0.148**	0.238	0.226	−	0.083	**0.076**
	0.4	0.050	0.166	0.177	−	**0.107**	0.049
	0.5	0.082	0.181	0.208	−	0.011	0.049
	0.6	0.031	0.165	0.208	−	×	0.042
	0.7	0.037	0.175	0.226	−	×	0.046
	0.8	0.092	**0.254**	**0.289**	−	×	0.075
$\rho MD\text{-}ML$	0.1	×	0.071	0.178	−	×	0.022
	0.2	0.099	0.223	0.247	−	0.060	0.069
	0.3	0.117	0.237	**0.311**	−	×	0.076
	0.4	0.142	0.264	0.250	−	0.010	0.077
	0.5	0.112	0.259	0.250	−	0.012	0.072
	0.6	**0.148**	0.270	0.278	−	**0.081**	**0.089**
	0.7	0.118	**0.283**	0.225	−	0.024	0.076

TABLE 6.4: Comparison between *hop-count* and hybrid (h) methods. Effect size: not-significant effect (×) and negative effect (−).

	p=(w_1, w_2)	PLR	E2ED	PER	NRL	NLT	U
$hETX$-REC	(0.1, 0.9)	0.436	0.382	0.284	0.335	×	0.201
	(0.2, 0.8)	0.415	0.406	0.240	0.318	×	0.192
	(0.3, 0.7)	0.372	0.339	0.207	0.273	×	0.155
	(0.4, 0.6)	0.388	0.376	0.237	0.302	×	0.175
	(0.5,0.5)	0.437	0.413	0.281	**0.347**	**0.062**	**0.216**
	(0.6, 0.4)	0.393	0.355	0.230	0.295	0.054	0.174
	(0.7, 0.3)	**0.439**	**0.417**	**0.319**	0.343	×	0.215
	(0.8, 0.2)	0.428	0.376	0.257	0.341	×	0.195
	(0.9, 0.1)	0.423	0.395	0.261	0.340	×	0.198
hML-REC	(0.1, 0.9)	**0.448**	**0.408**	0.254	0.351	×	0.208
	(0.2, 0.8)	0.417	0.391	0.231	0.313	×	0.186
	(0.3, 0.7)	0.418	0.394	0.233	0.321	×	0.189
	(0.4, 0.6)	0.413	0.393	0.237	0.311	×	0.185
	(0.5, 0.5)	0.389	0.364	0.244	0.293	×	0.171
	(0.6, 0.4)	0.384	0.356	0.233	0.293	×	0.166
	(0.7, 0.3)	0.407	0.378	0.241	0.299	×	0.179
	(0.8, 0.2)	0.341	0.315	0.173	0.255	×	0.135
	(0.9,0.1)	0.447	0.405	**0.265**	**0.361**	×	**0.211**
hMD-ETX	(0.1, 0.9)	0.154	0.250	0.210	−	0.066	0.073
	(0.2, 0.8)	0.130	0.241	0.194	−	0.022	0.064
	(0.3, 0.7)	0.171	0.222	0.158	−	0.088	0.066
	(0.4, 0.6)	0.144	0.239	0.195	−	0.126	0.075
	(0.5, 0.5)	0.104	0.223	0.173	−	0.152	0.069
	(0.6, 0.4)	0.174	0.234	**0.218**	−	0.033	0.070
	(0.7, 0.3)	0.149	0.245	0.198	−	0.028	0.067
	(0.8, 0.2)	0.146	0.234	0.168	−	0.079	0.066
	(0.9,0.1)	**0.186**	**0.272**	0.196	−	**0.147**	**0.090**
hMD-ML	(0.1, 0.9)	**0.225**	**0.285**	0.223	−	×	0.087
	(0.2,0.8)	0.215	0.259	**0.265**	−	0.084	**0.093**
	(0.3, 0.7)	0.194	0.243	0.203	−	×	0.071
	(0.4, 0.6)	0.192	0.270	0.248	−	0.025	0.085
	(0.5, 0.5)	0.174	0.274	0.206	−	0.058	0.083
	(0.6, 0.4)	0.113	0.230	0.196	−	0.029	0.061
	(0.7, 0.3)	0.135	0.218	0.164	−	**0.113**	0.066
	(0.8, 0.2)	0.141	0.234	0.218	−	×	0.068
	(0.9, 0.1)	0.111	0.194	0.230	−	0.059	0.061

6.4 Experimental Results

All methods were executed 10 times in each mobility scenario for each average packet generation rate and, after each execution, the six network quality indicators were computed. A different generating seed was used for each of the 10 replications.

In summary, the characteristics of the experimental setup are

- Independent variables, or factors:
    - Methods for **Type A** application: ML, $\rho REC\text{-}LLT$, $\rho REC\text{-}ML$, $hETX\text{-}REC$, $hML\text{-}REC$, $\omega ML\text{-}REC$, and $\rho REC\text{-}ML^{rand}$ (static scenario).
    - Methods for **Type B** application: MD, $\rho MD\text{-}LLT$, $\rho MD\text{-}ML$, $hMD\text{-}ETX$, $hMD\text{-}ML$, $\omega MD\text{-}ML$, and $\rho MD\text{-}ML^{rand}$ (static scenario).
    - Mobility scenarios: no-mobility (static), 0.5-4 m/s and 4-9 m/s.
    - Average packet generation rates: **10kbps** (≈ 20 packets per second), **20kbps** (≈ 40 packets per second), and **30kbps** (≈ 60 packets per second).
    - Seed used to generate the scenarios: from 1 to 10.
- Dependent variables, or response factors:
    - PLR, Throughput, E2ED, NLT, NRL, and PER.

The methods were implemented on the software OMNET++ [86], which is an extensible, modular, component-based C++ simulation library primarily used as a framework to build network simulations. The confidence intervals were computed in the software R [87]. Once the indicators were normalized, the effect sizes can be seen as a percentage of losses or gains.

6.4.1 No-mobility scenario

6.4.1.1 Network Layout

As it was seen in the earlier chapter, the proposed framework provides the algorithm $ND\text{-}DEMO$. The idea behind the algorithm is to suggest the best coordinates (x, y) for nodes deployment on a flat space. It aims to optimize two conflicting objectives: covered area and packet reception rate.

In this experiment, a square area of $500m$ length on each side was considered where supposedly a sensor network should be deployed for monitoring purpose. The area is divided into squares ($10m \times 10m$), wherein the center of each square represents a demand point. So, the area was discretized considering $50 \times 50 = 2500$ demand points. The radius of coverage is the same for all sensors ($50m$). Finally, the following parameters of the algorithm are adopted as in the original paper [79]:

- Population size: 100;
- Stopping condition: 100 iterations;
- Crossover factor of the recombination (CR): 0.3;
- Scale factor of the base vector (λ): 0.5;

The parameters below were empirically chosen for this work.

- Central value for variation of scale factor (β_0): 0.1
- Allowable variation of β starting from β_0 (α): 0.099 (e.g. $\beta = \beta_0 \pm \alpha$)

ND-$DEMO$ algorithm returns a set of solutions that compose a front in which it is possible to see the associated trade-off. Note in Figure 6.1 that, as the average packet reception rate decreases, the covered area increases. This fact occurs because, in order to enlarge the sensed area, it is necessary to spread the nodes, which naturally generates packet losses as a result of the signal deterioration caused by the greater distance between the nodes. Likewise, the covered area decreases as the nodes are more grouped.

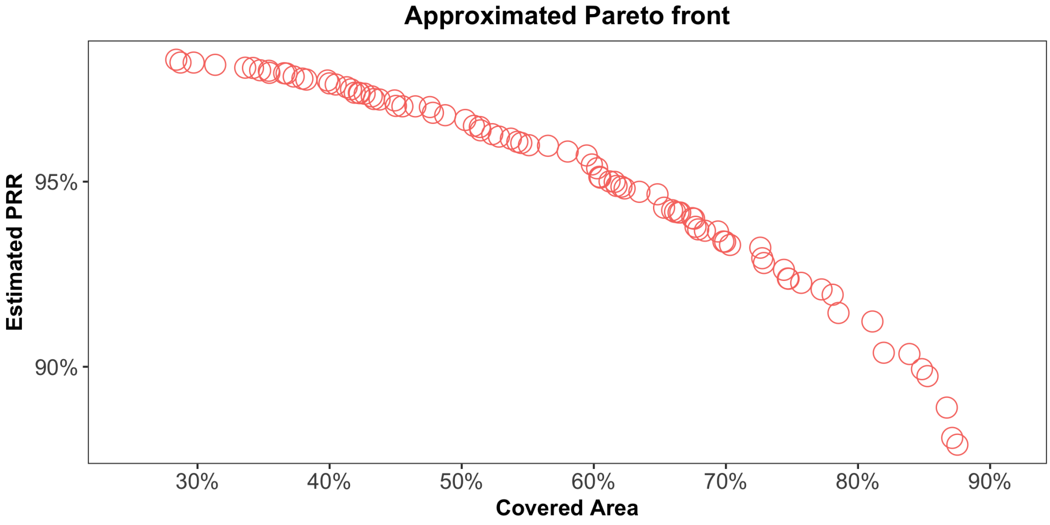


FIGURE 6.1: Example of the approximated Pareto front with solutions found by the ND-DEMO algorithm.

From the solutions obtained by the algorithm, it is necessary to apply some policy that helps in the decision making. An alternative is to use Compromise Programming (CP) [21], as the formulation below

$$\begin{aligned} &\underset{\mathbf{x}}{min} \left[\sum_{i=1}^{n_f} w_i |f_i(\mathbf{x}) - f_i^*|^r \right]^{1/r} \\ &\text{s.t.} \quad \mathbf{x} \in S, \end{aligned} \tag{6.10}$$

with $r = 2$ the Euclidean distance is calculated in relation to the reference point (an ideal solution) of each $f_i(\mathbf{x})$. In this way, a final practical solution can be chosen [21]. The weight vector $\mathbf{w}$ is composed of non-negative values assigned to each objective function $f_i(\mathbf{x})$ and represents the preferences of the decision maker.

In our approach, the weights are equal for the two objectives ($\mathbf{w} = [0.5, 0.5]$) and the reference used was the point $\mathbf{f}^* = [0.95, 1]$. What is desired is to choose a solution that guarantees the closest values of 95% packet reception rate and 100% covered area.

6.4.1.2 Initial remarks

Before proceeding with more detailed analysis, it is helpful to take a look at the data to get a prior insight into the results. Figures 6.2 and 6.3 show the six quality indicators for each QoS-aware method when executing in three different average packet generation rates and considering **Type A** and **Type B** applications. The graphs reveal the median of 10 replications. If only the point is visible, it means that the variability of the observations is small. When the quality indicator is preceded by [+], the higher its value, the better. When the quality indicator is preceded by [-], the smaller its value, the better.

Figure 6.2 shows the ML was the worst method in all quality indicators. On the other hand, proposed multicriteria methods (ρ[2] and h[3]) were the best ones in PLR, Throughput, E2ED, NRL, and PER indicators, mainly when the average packet generation rate is greater than **10kbps**.

Only to reinforce, *ωML-REC* represents the standard weighted sum method, such as presented in [19] and [23], for example. Also, $\rho REC\text{-}ML^{rand}$ implements the models proposed in this work, but without executing the *ND-DEMO* algorithm, that is, nodes are positioned randomly. To be more assertive in this investigation, confidence intervals of such differences were computed for each pair of methods, as it is described hereafter.

---

[2] *ρREC-LLT* and *ρREC-ML*

[3] *hETX-REC* and *hML-REC*

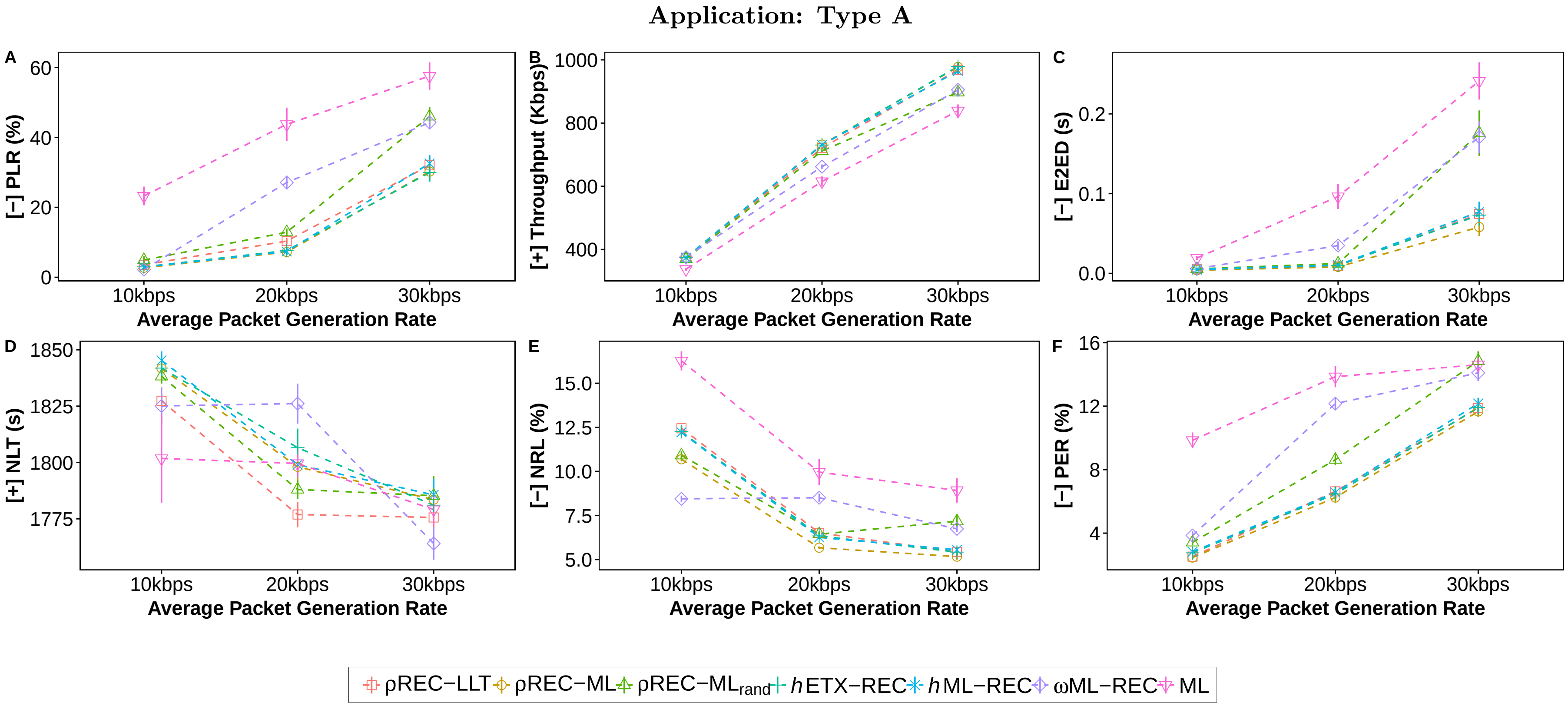


FIGURE 6.2: **Application: Type A**. Results for each method on each quality indicator, with 10kbps, 20kbps, and 30kbps of average packet generation rate.

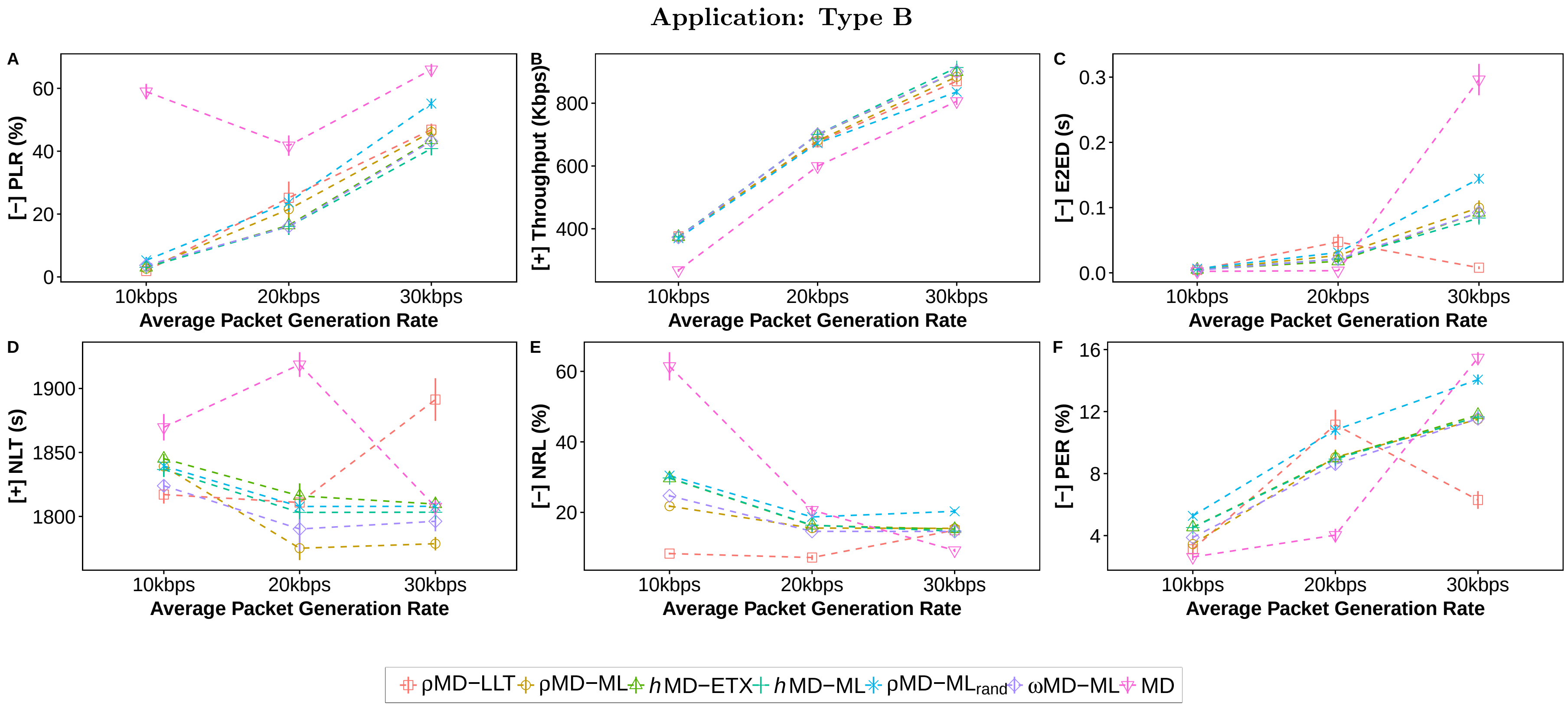


FIGURE 6.3: **Application: Type B**. Results for each method on each quality indicator, with 10kbps, 20kbps and 30kbps of average packet generation rate.

6.4.1.3 Analyzing the magnitude of the differences

Differences between the single and the multicriteria methods were analyzed. *All* vs. *ML* and *all* vs. *MD* comparisons were performed for **Type A** and **Type B** applications, respectively. The confidence intervals were computed in the software *R*. To get the overall confidence of 95%, a *familywise error rate* of 0.05 was defined.

To understand the intervals, check, for instance, the Figure 6.4. The axis signaling $[-]$ and $[+]$ labels the quality indicators either as "the lower, the better" or as "the higher, the better". When the observation is completely on the left of the dashed line in 0, it can be said the difference between these methods is negative. For example, the pair "$\omega ML\text{-}REC - ML$" in PLR indicator has this behavior. Since the PLR is a "the lower, the better" indicator, $\omega ML\text{-}REC$ is significantly better than ML. The same reasoning applies to the intervals completely on the right of line 0. Intervals crossing the line indicate the two methods are not significantly different at the chosen confidence level.

Application: Type A

- **PLR and Throughput:** Figure 6.4 shows the confidence intervals for the indicators in the three average packet generation rates considered in this work.
    - **10kbps:** For practical purposes, all multicriteria methods presented similarly good performances.

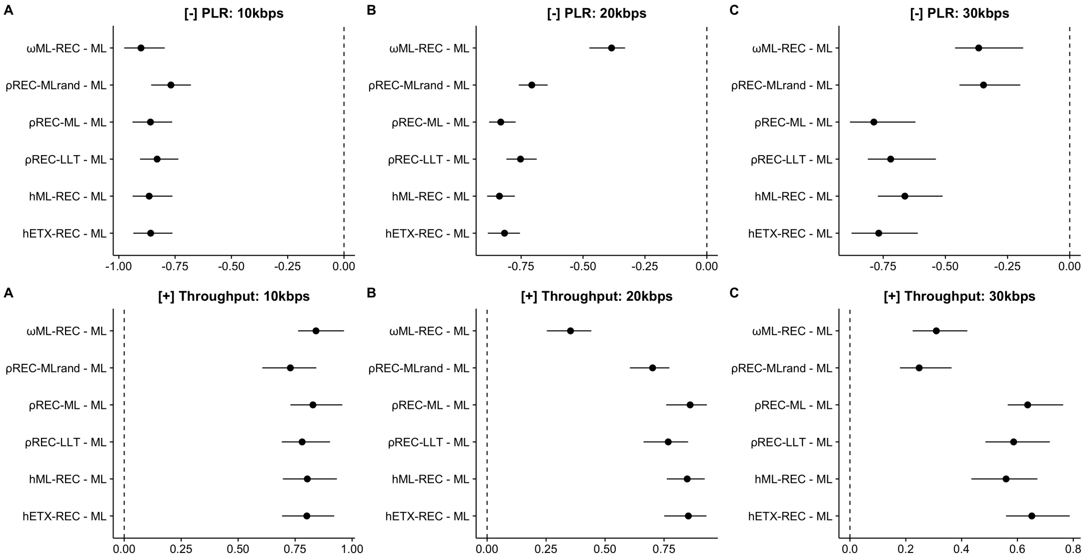


FIGURE 6.4: **Application: Type A**. 95% confidence intervals for differences in PLR and Throughput indicators. From left to right, the panels show the intervals for 10kbps, 20kbps, and 30kbps of average packet generation rate.

- **20kbps:** When traffic doubles, differences begin to accentuate. This rate generates approximately 40 packets per second, which enables to simulate a reality closer to a typical VoIP transmission [58].

  The proposed methods (ρ and h) reached remarkably best results when compared to the weighted sum (ω). Two strategies implemented in this work can justify this difference: i) weights and constraints are selected based on a multicriteria utility function and ii) the path length is considered into the model including a constraint to control its growth. The former strategy chooses parameters that tend to guarantee better compromise solutions. The latter strategy helps to reduce losses caused by collisions and interference since, by applying it, the average number of hops is likely to decrease.

  Summarily, in this scenario, working with multiple reliability metrics tends to promote performance gains in both PLR and Throughput. For practical purposes, ρ and h methods presented similar performances.

- **30kbps:** In this rate, considerations mentioned above are valid here. However, some specific observations can be made. Initially, the results suggest that the *a priori* network layout definition becomes essential to provide better performance when there is intense traffic. That is, having nodes well-positioned helps to create shorter paths and consequently reduces packet losses caused by interference and collisions. Table 6.5 reinforces this argument when it shows that the average number of hops of the proposed methods (excluding $\rho REC\text{-}ML^{rand}$) are inferior to the remaining ones.

  TABLE 6.5: **Application: Type A**. Average number of hops.

Method	10kbps	20kbps	30kbps
$\rho REC\text{-}LLT$	**1.93**	1.99	2.23
$\rho REC\text{-}ML$	**1.93**	**1.97**	**2.21**
$\rho REC\text{-}ML^{rand}$	2.23	2.33	2.70
$hETX\text{-}REC$	**1.93**	1.98	2.24
$hML\text{-}REC$	**1.93**	1.99	2.29
$\omega ML\text{-}REC$	2.29	2.85	3.13
ML	2.69	2.70	2.65

  All proposed methods had similar efficiency. These results show the weights arrangement, the *hop control factor*, and the node deployment algorithm help to guarantee promising values of delivery reliability.

- **E2ED:** When Figure 6.5 is analyzed, it can be perceived results have essentially the same fashion as PLR does. Basically, as the average packet generation rate increases, ρ and h methods are gaining prominence. Looking at **10kbps** rate, multicriteria methods were statistically equivalent, and they beat the single-objective

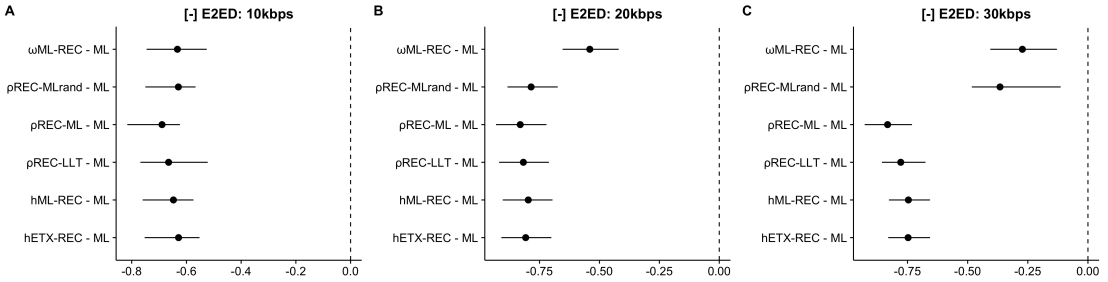


FIGURE 6.5: **Application: Type A**. 95% confidence intervals for differences in E2ED.

method. As to **20kbps** rate, ρ and h methods become significantly better than ω*ML-REC*. Finally, in the scenario with heavy traffic of **30kbps**, the ρ and h methods with *ND-DEMO* algorithm reached the best values.

The central explication to these results is that the proposed methods generate shorter routes due to the *hop control factor*, as it is shown in Table 6.5. This fact collaborates to create paths with lower delay. However, the most promising is that these shorter routes also perform better to delivery reliability (PLR and Throughput) in this type of application.

It is important to highlight that low delay conveys the feeling the network is working fast, especially when real-time or small-buffered multimedia applications are being used.

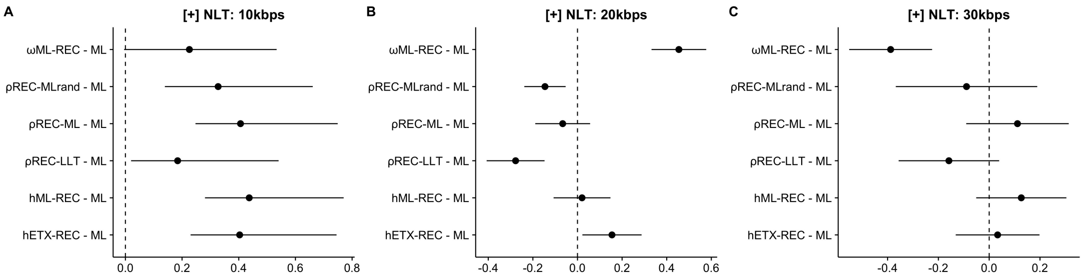


FIGURE 6.6: **Application: Type A**. 95% confidence intervals for differences in NLT.

- **NLT:** Watching through the confidence intervals in Figure 6.6, it is harder to extract patterns from NLT quality indicator like it was made previously when analyzing PLR, Throughput, and E2ED. Considering the next statements, some light can be put in this analysis.

  1. Methods with greater PLR could spend less energy on average because some relay nodes did not spend energy retransmitting early lost packets. This case may have occurred for the ω*ML-REC* method at the **20kbps** rate. That is, the losses may have collaborated for an NLT on average 40% higher when

compared to ML. Possibly, this also explains why ML reached competitive results just for NLT indicator, especially at the **20kbps** and **30kbps** rates.

2. For pruning methods (ρ), the value of ϵ should not be so restrictive as to make it difficult to obtain feasible routes. When comparing all proposed methods, it can be seen that *ρREC-LLT* got the worst results mainly in **20kbps**, exactly because it was not always able to find links with stability greater than or equal to 0.8 ($\epsilon = 0.8$). Not finding feasible routes can impact the NLT because it is needed to perform new searches, which yield more energy depletion.

Simply put, a trade-off can exist between reliability and node or network lifetime [49]. In this context, proposed methods reached better compromise solutions.

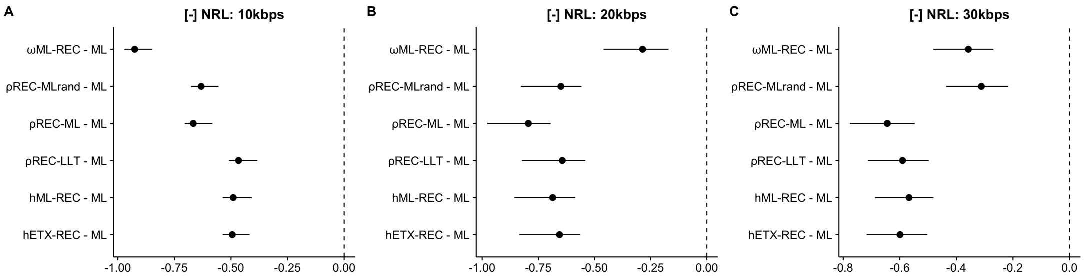


FIGURE 6.7: **Application: Type A**. 95% confidence intervals for differences in NRL.

- **NRL:** Figure 6.7 shows the confidence intervals for this indicator. In OLSR protocol, routing packets are broadcasted in two moments: i) periodically: both HELLO and TC messages are fired from time to time (few seconds), and ii) occasionally: additional TC messages may be transmitted to increase the reactiveness to link failures, so a node generates a TC message immediately when a change in the MPR set is detected and this change can be attributed to a link failure [88]. Those methods that are able to reduce the link breakages will decrease the routing overhead caused by occasional TC messages transmissions.

  - **10kbps:** In a lighter scenario in terms of data traffic, the ωML-*REC*'s result is the best, with NRL ≈ 80% lower than ML's. ωML-*REC* does not use the *hop control factor*. That is, for such a specific scenario and looking at this indicator alone, having any control over the path growth is not necessary or even inefficient. Therefore, when allowing the routes are updated without controlling the path size, more reliable links were found.
  - **20kbps:** In this rate, nodes receive, transmit, and retransmit theoretically twice as many packets as the **10kbps** rate. Herein, not controlling the paths

size can generate more contention need, in addition to demanding more storage space. In this scenario, links are more required and can fail or break more times. Unlike **10kbps**, ρ and h methods beat ω*ML-REC*.

- **30kbps:** Proposed methods that applied *ND-DEMO* algorithm had an advantage and were equally good. Such results suggest the node deployment algorithm is more effective when the average packet generation rate is high, which is especially interesting if multimedia applications are being used.

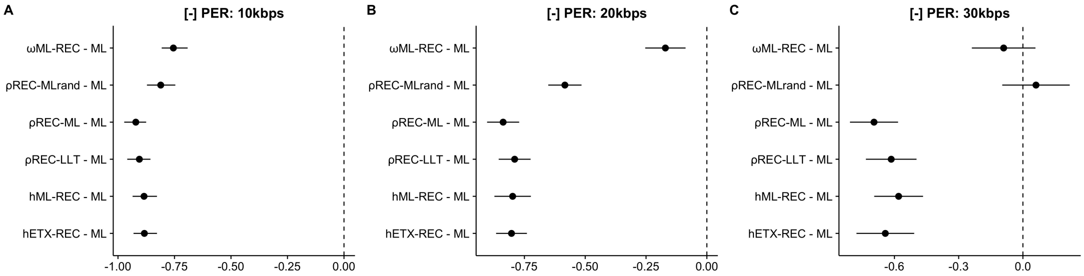


FIGURE 6.8: **Application: Type A**. 95% confidence intervals for differences in PER.

- **PER:** This quality indicator is important because it measures the percentage of packet loss due to noise, collisions, and interference. Results clearly showed the proposed methods (ρ and h) were significantly better than others. It is particularly interesting to see the effect of the use of *ND-DEMO* algorithm. The idea of the algorithm is to distribute homogeneously the nodes on the plane. So, it is reasonable to think the main effect is an automatic reduction of node agglomerations in some regions of the area, which reduces the PER. That was exactly the effect observed when studying the results in Figure 6.8.

This work aims to verify if the proposed methods (ρ and h) can generate promising compromise solutions considering the six network quality indicators analyzed. Besides, one of these methods would need to be selected and applied in an ad-hoc network. For both tasks, the utility function (U) presented in the optimization problem (6.9) was solved considering the results aforementioned.

The key idea of the function is that one can handle η_i according to the preferences associated with the application and network requirements. As **Type A** application has two design objectives, namely, high reliability and high throughput, the preferences of indicators were defined such as: $\eta_{PLR} = 0.25$, $\eta_{Thr.} = 0.25$, $\eta_{E2ED} = 0.15$, $\eta_{NLT} = 0.15$, $\eta_{NRL} = 0.10$, and $\eta_{PER} = 0.10$. The ML method was taken as reference.

Table 6.6 contains the utility function values. The ρ*REC-ML* method reached better results for all transmission rates. Taking **20kbps** and **30kbps** rates, it should also be noted the other three best ones belong to the set of proposed methods (ρ and h). These values indicate their potential for finding promising compromise solutions.

TABLE 6.6: **Application: Type A**. Multi-criteria utility function (U).

Method	10kbps	20kbps	30kbps
$\rho REC\text{-}LLT$	0.57	0.53	0.42
$\rho REC\text{-}ML$	**0.67**	**0.64**	**0.50**
$\rho REC\text{-}ML^{rand}$	0.54	0.46	0.10
$hETX\text{-}REC$	0.61	0.61	0.46
$hML\text{-}REC$	0.63	0.60	0.41
$\omega ML\text{-}REC$	0.64	0.21	0.11

Application: Type B

Type B application is different from **Type A**. It has a very small buffer to simulate a multimedia application that demands both reliability and low end-to-end delay. Herein, REC metric is replaced by MD metric. The idea is to combine a reliability-oriented/stability-driven metric and a delay-aware metric to try to meet these objectives.

In this section, $\omega MD\text{-}ML$ represents the standard weighted sum method, such as presented in [19] and [23], for example. Also, $\rho MD\text{-}ML^{rand}$ implements the models proposed in this work, but without executing the $ND\text{-}DEMO$ algorithm. In Figure 6.3, in spite of the superiority of the proposed methods (ρ and h) have not been seen in all quality indicators, they seem to get better results than MD in practically all scenarios. Confidence intervals of such differences were computed for each pair of methods.

As in the previous section, a comprehensive analysis was performed. However, the principal goal at this point is to recognize possible conflicts regarding the quality indicators objectively. From this, it is possible to find which approaches offer the best compromise solutions. The reader can access the more detailed analysis in Appendix A.

Figure 6.9 shows that all multicriteria methods were significantly superior when compared to MD in PLR indicator. In this indicator, MD presents a performance that varies non-uniformly in the three rates (see Table 6.3) since it does not take into account the delivery reliability. In E2ED, MD outperformed the others when the network works with **10kbps** and **20kbps** rates. This result occurs because MD generates short paths (check Appendix A). As to NLT, the methods that spent more energy are those that got better performance in PLR.

Therefore, in this type of application, it is possible to affirm the existence of two trade-offs: i) delivery reliability *vs.* delay, ii) delivery reliability *vs.* network lifetime.

Repeating the same idea of the **Type A** application analysis, the multi-criteria utility function (U) proposed in Equation (6.9) was used to evaluate the methods considering

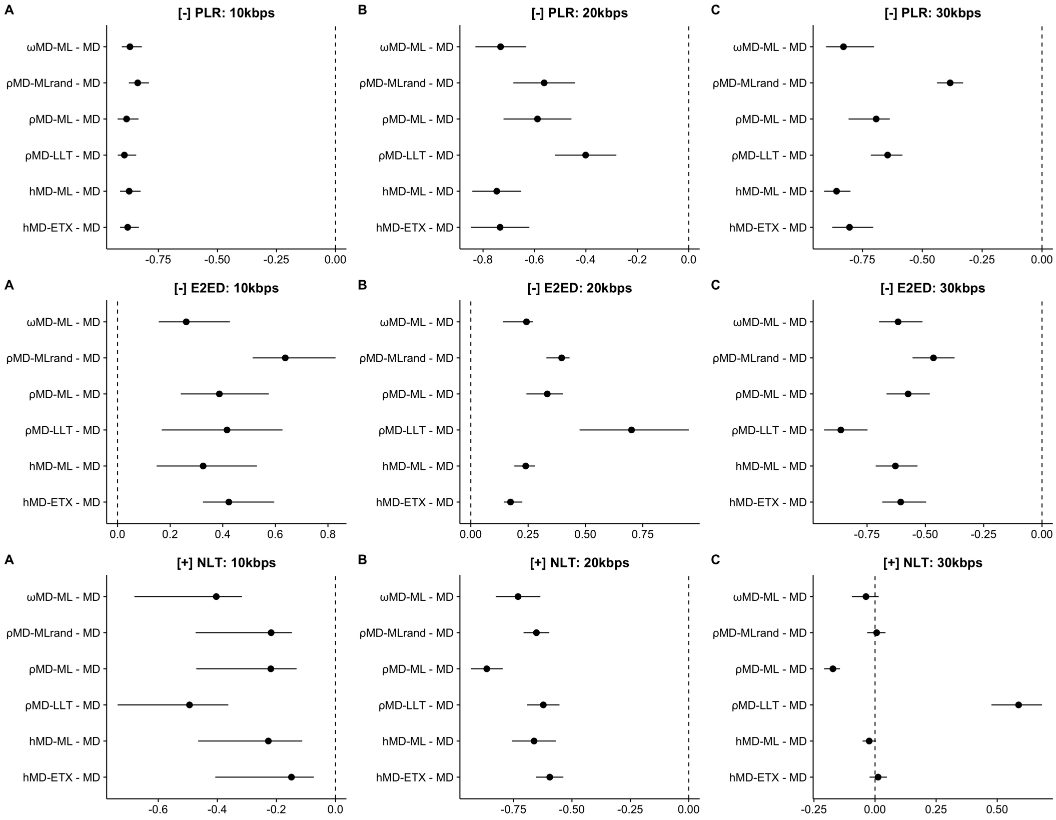


FIGURE 6.9: **Application: Type B**. 95% confidence intervals for differences in PLR, E2ED, and NLT indicators. From left to right, the panels show the intervals for 10kbps, 20kbps, and 30kbps of average packet generation rate.

the six network quality indicators analyzed. The goal is to verify if the proposed methods (ρ and h) generate promising compromise solutions.

The objectives are to guarantee: i) high reliability and ii) low end-to-end delay. So, the preferences of indicators were defined such as: $\eta_{PLR} = 0.25$, $\eta_{Thr.} = 0$, $\eta_{E2ED} = 0.25$, $\eta_{NLT} = 0.20$, $\eta_{NRL} = 0.15$, and $\eta_{PER} = 0.15$. The MD was taken as reference.

TABLE 6.7: **Application: Type B**. Multi-criteria utility function (U).

Method	10kbps	20kbps	30kbps
$\rho MD\text{-}LLT$	**0.28**	**0.08**	**0.49**
$\rho MD\text{-}ML$	0.21	0.01	0.20
$\rho MD\text{-}ML^{rand}$	0.18	-0.04	0.10
$hMD\text{-}ETX$	0.18	0.05	0.23
$hMD\text{-}ML$	0.17	0.04	0.27
$\omega MD\text{-}ML$	0.19	0.07	0.23

Table 6.7 contains the utility function values. The $\rho MD\text{-}LLT$ reached the best results, mainly for **10kbps** and **30kbps** rates. According to values, ρ and h methods are capable of guaranteeing a quality of service that considers the several network indicators. That is, they were the ones that best dealt with the trade-offs presented in this scenario.

6.4.2 Average speed scenario

After investigating the results in a static scenario, it is the moment to analyze the methods in an average speed scenario. In this environment, mobility contributes to even more frequent changes in topology.

Figure 6.10 presents a preliminary insight into the results of the **Type A** application. The ML single-objective method was the worst in most quality indicators. On the other hand, proposed methods (ρ and h) were the best ones in PLR, Throughput, E2ED, NRL, and PER when the average packet generation rate was equal to **30kbps**.

6.4.2.1 Analyzing the magnitude of the differences

In the same way of the static scenario, a careful and detailed analysis was performed in these mobile scenarios. To make the discussion more straightforward, the following sections show the results giving particular emphasis on the main differences.

Application: Type A

In Figure 6.10, it can be seen the ML is significantly worse than most of the others, which suggests the multicriteria methods are superior in basically all indicators. Initially, Table 6.8 shows a brief overview of the results considering each indicator. In Appendix A, more punctual discussions are held.

Table 6.8: **Application: Type A.** Overview of results.

Indicator	Overview
PLR	**10kbps** and **20kbps**: multicriteria methods presented similarly good performances for practical purposes. **30kbps:** the proposed methods (ρ and h) were the only ones able to maintain the PLR around 20%.
Throughput	**10kbps** and **20kbps**: multicriteria methods were equally efficient. **30kbps:** proposed methods overcome standard weighted sum (ωML-*REC*) and single-objective (ML) methods.
E2ED	**10kbps** and **20kbps**: multicriteria methods presented similarly good performances for practical purposes. **30kbps**: Proposed methods are significantly better than ML and ωML-*REC*.
NLT	**10kbps**: multicriteria methods overcome the ML. **20kbps** and **30kbps:** ωML-*REC* reached the best results.
NRL	All proposed methods were equally good in the three rates.
PER	**10kbps** and **20kbps**: multicriteria methods were significantly better than ML. **30kbps:** proposed methods surpassed standard weighted sum ωML-*REC* and ML methods.

Application: Type A

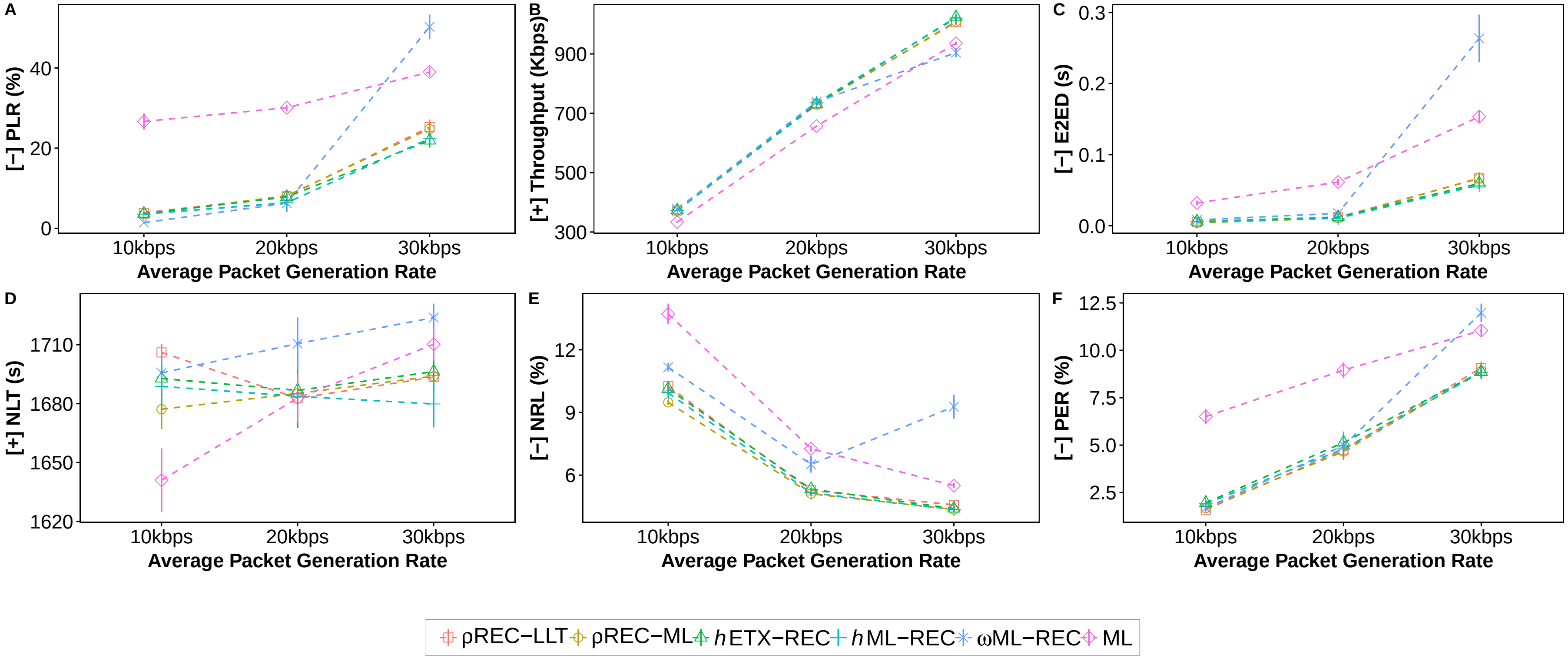


FIGURE 6.10: **Application: Type A**. Results for each method on each quality indicator, with 10kbps, 20kbps and 30kbps of average packet generation rate.

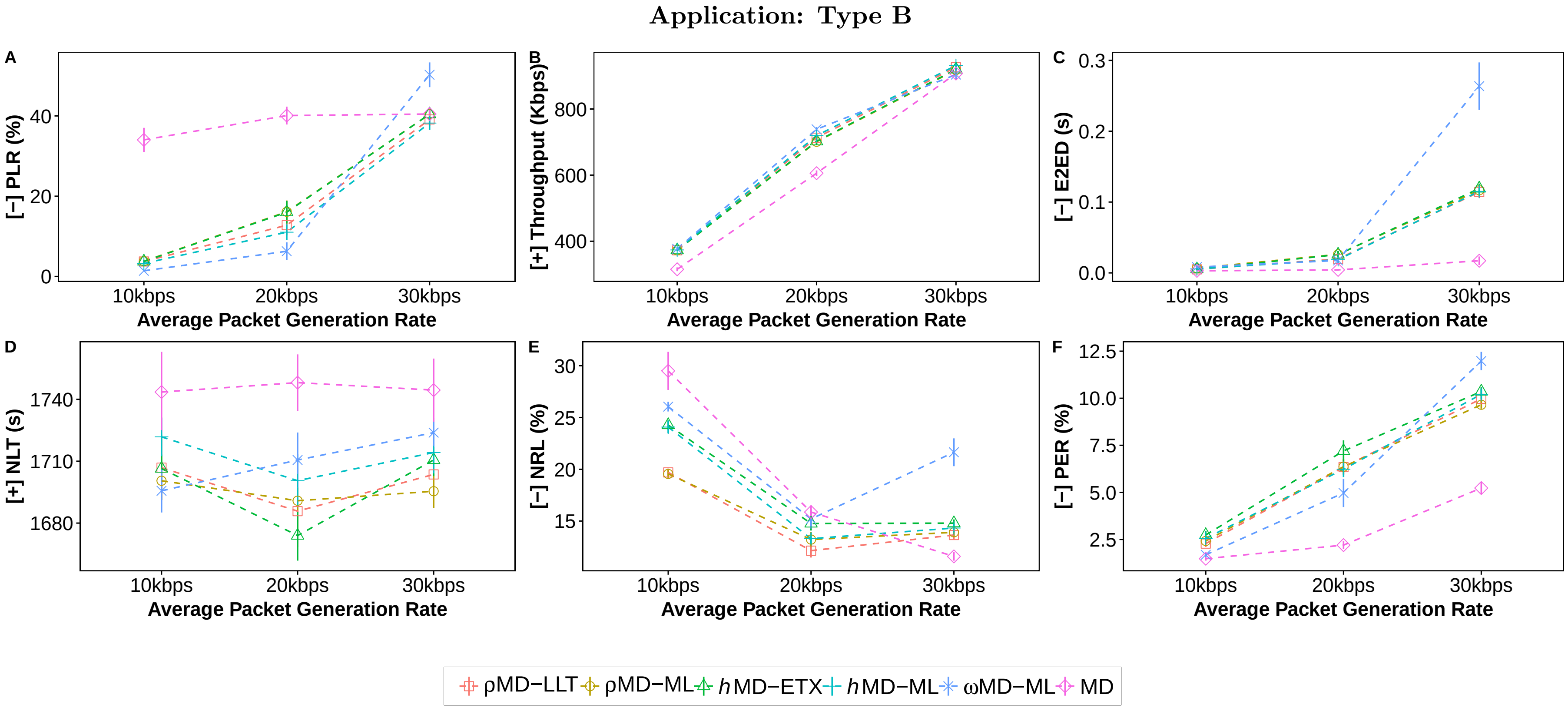


FIGURE 6.11: **Application: Type B**. Results for each method on each quality indicator, with 10kbps, 20kbps and 30kbps of average packet generation rate.

As in static scenario, the proposed multicriteria utility function (Equation 6.9) is used to evaluate which method is the best one in terms of compromise solutions considering the preferences that were defined in the earlier section. Table 6.9 contains the utility function values. For the **10kbps**, ρ*REC-LLT* and ω*ML-REC* reached better results. In **20kbps**, *hML-REC* would be the chosen one. In **30kbps**, ρ and h methods had an advantage. These results show once again the potential of the methods proposed mainly in ad-hoc networks with higher data traffic.

TABLE 6.9: **Application: Type A.** Multi-criteria utility function (U).

Method	10kbps	20kbps	30kbps
ρ*REC-LLT*	**0.68**	0.56	0.23
ρ*REC-ML*	0.65	0.57	0.25
hETX-REC	0.64	0.55	**0.30**
hML-REC	0.65	**0.62**	0.29
ω*ML-REC*	**0.68**	0.57	-0.03

Application: Type B

Type B application demands reliability and low end-to-end delay. From Figure 6.11, it can be seen multicriteria methods were superior to the single-objective method (*MD*) only for PLR and Throughput indicators and when the rate was equal to **10kbps** or **20kbps**. Table 6.10 shows a brief overview of the results.

TABLE 6.10: **Application: Type B.** Overview of results.

Indicator	Overview
PLR and Throughput	**10kbps**: multicriteria methods presented similarly good performances for practical purposes. **20kbps**: standard weighted sum (ω*ML-REC*) had the best result. **30kbps:** no method was able to guarantee the PLR lower than 20%. In Throughput, there was no difference between the methods.
E2ED, NLT, and PER	*MD* had better performance in all scenarios.
NRL	**10kbps** and **20kbps**: pruning methods (ρ) reached the best results. **30kbps**: *MD* had better performance.

From the idea of evaluating the compromise solutions and selecting a method for each packet generation rate, it is applied the multi-criteria utility function (Equation 6.9), as it was done in the static scenario. Preferences of each indicator were also defined in the previous section in order to generate solutions that try to guarantee high reliability and low delay, without neglecting other quality measures.

According to Table 6.11, the ρ*MD-LLT* method reached better result for **10kbps** and **20kbps** rates. In **30kbps**, the single-objective method *MD* guarantees best trade-off solutions. However, in this latter scenario, none of the methods were able to provide PLR smaller than 20%. This outcome suggests that such an application may be particularly hard to support under such network conditions.

Table 6.11: **Application: Type B.** Multi-criteria utility function (U).

Method	10kbps	20kbps	30kbps
ρMD-LLT	**0.18**	**0.16**	-0.11
ρMD-ML	0.17	0.10	-0.13
hMD-ETX	0.12	0.07	-0.14
hMD-ML	0.12	0.11	-0.10
ωMD-ML	0.16	0.12	-0.10

In fact, OLSR protocol is not well-fitted to deal with mobility and high data traffic, mainly in scenarios in which real-time applications are used. Since it follows the proactive paradigm, the route updates may not be fast enough to catch the topology variations produced by node moves and frequent link breaks. Reactive and opportunistic protocols must be tested to try to circumvent this problem. Although the proposed framework can be adapted to any routing paradigm, this issue will be postponed to future work.

6.4.3 High-speed scenario

A preliminary insight is obtained observing the Figures 6.12 and 6.13. As the results of this scenario are very similar to the results of the average speed scenario, the main differences are highlighted in Appendix A.

Table 6.12 contains the utility function values considering the preferences that are set in previous scenarios. In **Type A** application, for **10kbps** and **20kbps** rates, ωML-REC reached the best results mainly because of its simplicity when compared to the remaining multicriteria methods. When traffic gets heavier (**30kbps**), proposed methods (ρ and h) were better than ωML-REC.

For **Type B** application, ρMD-LLT, ρMD-ML, and ωMD-ML were the best ones in **10kbps** rate. In **20kbps** rate, ρMD-LLT and ωMD-ML reached better results. In **30kbps**, hMD-ML was the only method that overcame the single-objective method.

Table 6.12: Multi-criteria utility function (U).

Type A Application				**Type B Application**			
Method	10kbps	20kbps	30kbps	Method	10kbps	20kbps	30kbps
ρREC-LLT	0.48	0.48	0.26	ρMD-LLT	0.19	**0.12**	-0.04
ρREC-ML	0.53	0.42	**0.50**	ρMD-ML	**0.20**	0.07	-0.02
$hETX$-REC	0.48	0.42	0.42	hMD-ETX	0.14	0.06	-0.03
hML-REC	0.49	0.50	0.22	hMD-ML	0.14	0.09	**0.09**
ωML-REC	**0.68**	**0.60**	-0.02	ωMD-ML	**0.20**	**0.12**	-0.02

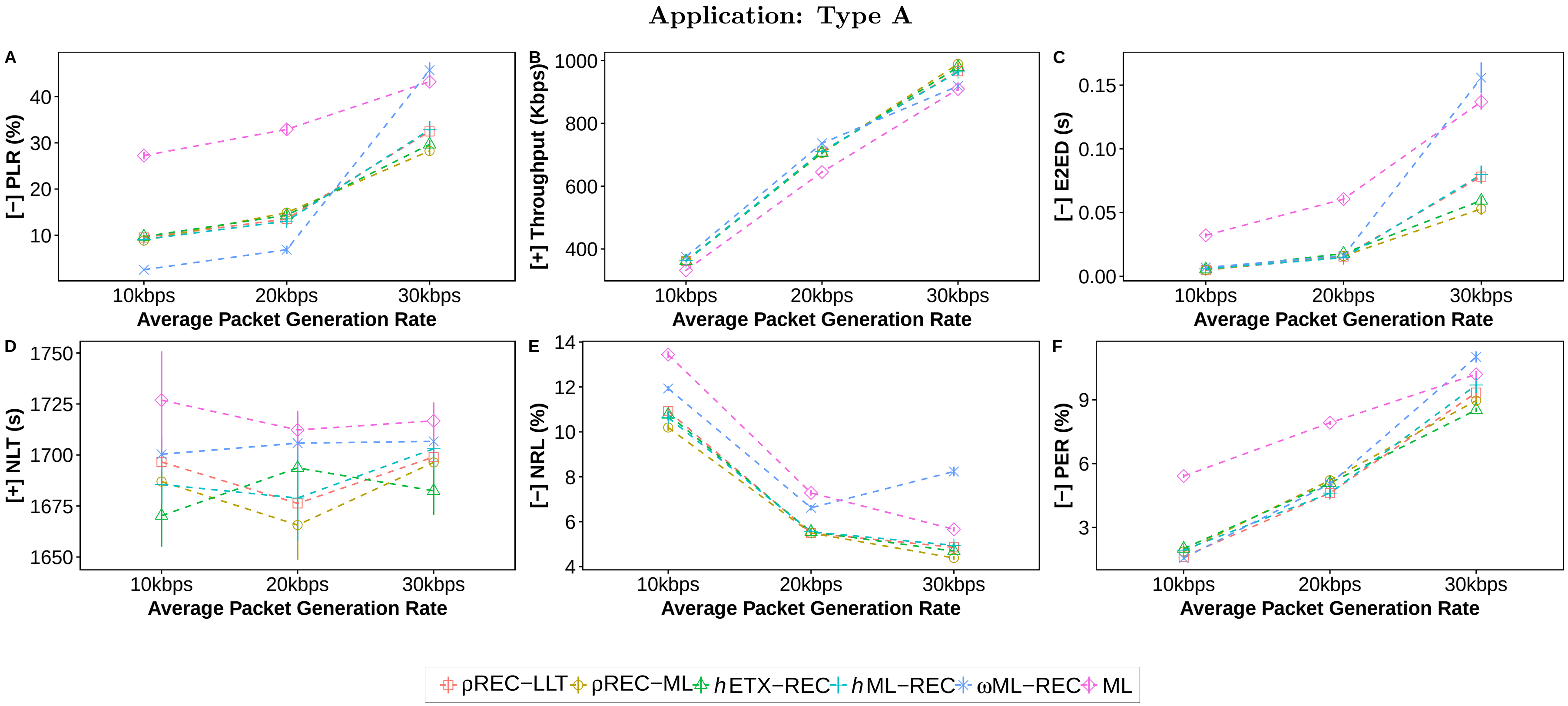


FIGURE 6.12: **Application: Type A**. Results for each method on each quality indicator, with 10kbps, 20kbps and 30kbps of average packet generation rate.

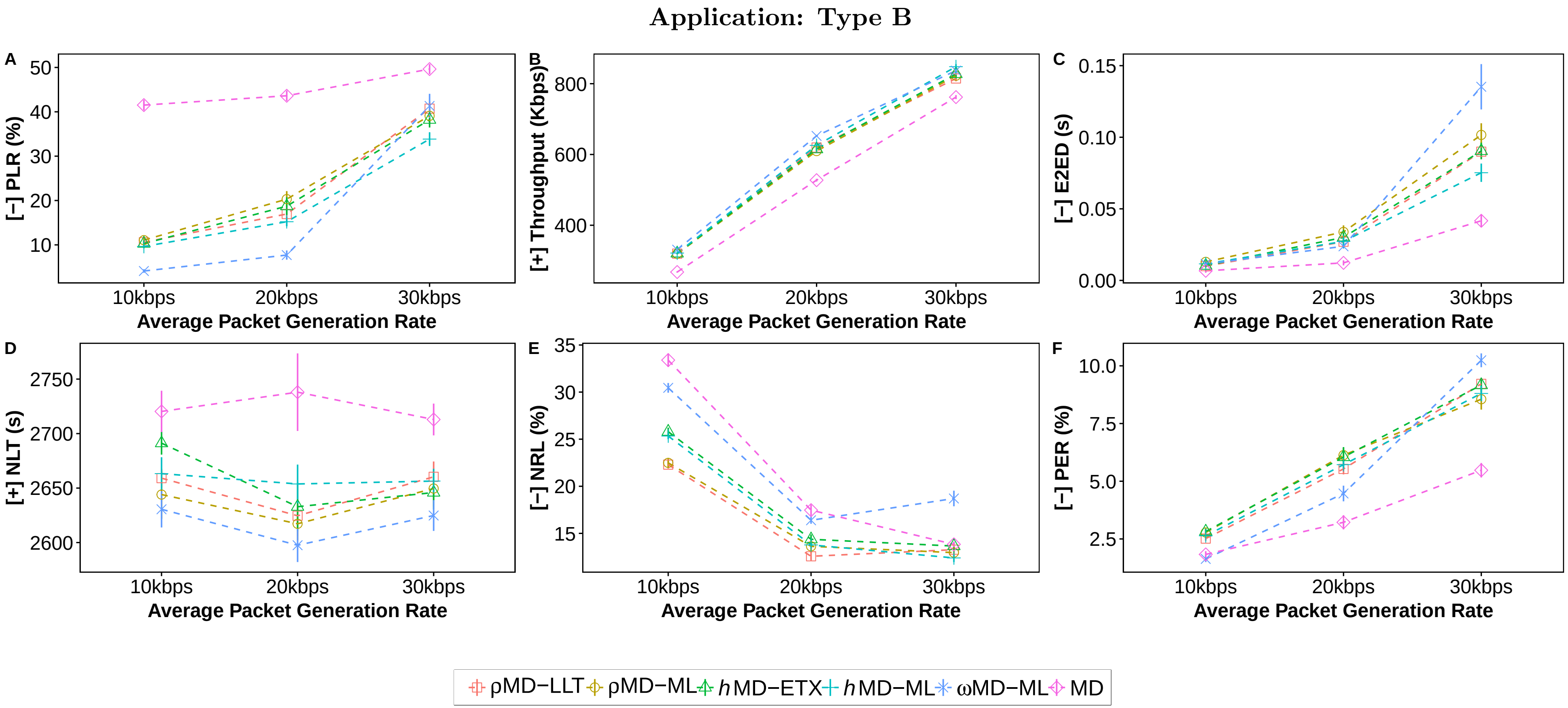


FIGURE 6.13: **Application: Type B**. Results for each method on each quality indicator, with 10kbps, 20kbps and 30kbps of average packet generation rate.

6.4.4 Discussion

This work aims to verify if the use of multiple QoS-aware metrics can generate promising compromise solutions considering several network quality indicators in static and mobile Ad-hoc Wireless Networks. For that, it is proposed a framework that implements some multicriteria methods, which is compared with standard weighted sum and single-objective methods. This section summarizes the results mentioned above.

6.4.4.1 Static scenario

According to the results in the static scenario, proposed methods were significantly better than the standard weighted sum and single-objective ones in practically all quality indicators. In PLR and Throughput indicators, their values were encouraging for both **Type A** and **Type B** applications regardless of the average packet generation rate. In addition, the use of $ND\text{-}DEMO$ algorithm was very effective.

In **Type A** application, $\rho REC\text{-}ML$ was the method that, overall, best fit the scenarios and got the best results. It occurred because of the pruning constraint ($\epsilon = 0.6$) was not overly restrictive. At the same time, it allowed that more reliable links were selected. In conclusion, all proposed methods (except for $\rho REC\text{-}ML^{rand}$) presented a satisfactory performance for all packet generation rates.

Regarding **Type B** application, $\rho MD\text{-}LLT$ obtained a great prominence in the **30kbps** rate, mainly in E2ED, NLR, and PER indicators. The fact the pruning constraint ($\epsilon = 0.1$) allows to select links that are "headed" for instability (but they are not stable, yet) explains its better performance in such a heavy scenario.

6.4.4.2 Mobile scenarios

As it reported by the results in average and high-speed scenarios, proposed methods were very competitive, although on some occasions the single-objective ones achieved better results, mostly in E2ED and NLT indicators for the **Type B** application.

In some situations, standard weighted sum methods (ω) got better results than proposed methods (ρ and h), mainly when the rate was equal to **10kbps** or **20kbps**. In fact, ω methods are straightforward because they have no constraints, while hybrid ones (h) have the hop control constraint, and pruning ones (ρ) have the hop control and the pruning constraints. Despite this, these proposed methods obtained promising compromise solutions practically in all scenarios.

6.4.5 Final remarks

The central proposal of this work was to design a framework to test multicriteria methods in the resolution of the routing problem in ad-hoc networks. Since the structure supports various optimization objectives, it is possible to verify if the use of multiple QoS-aware metrics can lead to promising compromise solutions, considering critical network quality indicators.

In a broader perspective, the multicriteria approaches proposed and incorporated into the framework showed to be very promising. Such an assertion is supported by the significant improvements obtained in terms of trade-off solutions considering the key quality indicators.

It was tried to generalize the test and simulation scenarios as much as possible. For example, the mobility scenarios cover a good set of class of networks. Another example comprises the two type of applications considered, which encompass multimedia applications with different characteristics. Also, three traffic models were employed to simulate distinct network conditions. The results suggest the proposed methods may find their place when applied in static or with reduced mobility networks.

The main inconvenience of the multicriteria approaches lies in how to define parameters like weights and constraint accurately. A multi-criteria utility function was designed to deal with this problem. Unlike most studies, which consider only the Packet Delivery Ratio (PDR), our proposal allows to include multiple quality indicators in this decision-making process.

A promising alternative would be to define these parameters according to the network particular conditions. Since ad-hoc networks tend to vary constantly in their state (regarding traffic and mobility), a strategy could be designed to accomplish this dynamic redefinition. However, this leads to an impasse: how to do this task without provoking extra processing and routing overhead? More than that, how to make possible to test weight and constraint arrangements while data are being transmitted and, at the same time, to define which QoS-aware metrics best fit in the current context? These are tough issues that, unfortunately, will be postponed to future works.

Finally, it is important to point out that the results are adherent to the properties and features of the routing protocol used, namely, the OLSR protocol. It is intended to adopt the proposed framework to other paradigms hereafter.

Chapter 7

Conclusion

The process of providing Quality of Service (QoS) in Ad-hoc Wireless Networks involves solving the routing problem. It means that routes must be calculated using the nodes as relays, thus allowing decentralized communication. Over the last years, several types of networks have emerged thanks to this ad-hoc paradigm, such as Wireless Sensor Networks, Mobile Networks, Vehicular Networks, and Flying Networks. Consequently, new and reformed routing protocols (e.g., OLSR, AODV, and DYMO) have arisen specifically to try to provide QoS for these new scenarios and applications.

These protocols use, in their standardized versions, the *hop-count* metric to evaluate and manage the routes. However, many studies have shown the employment of QoS-aware measures can lead to significant improvements in the performance of these protocols. Several of these metrics already exist with the aim of addressing different quality requirements such as minimizing end-to-end delay, minimizing packet loss ratio, minimizing energy consumption, maximizing route stability, and so on.

Some works have observed there is a wide range of possibilities to design solutions and protocols that consider multiple design objectives and trade-offs between these objectives. In this context, more than one QoS-aware measure can be arranged and combined with the route discovery process.

The weighted sum is a scalar method widely used in literature to deal with multiple criteria in the routing problem over Ad-hoc Wireless Networks. In addition to it, this work proposed two new models. The first one uses the compromise scalar method as its basis. But, instead of imposing a global constraint, a local constraint that operates as a pruning mechanism was defined. Under this circumstance, links become available if they meet the earlier established quality constraint. This strategy makes the method lighter and more adaptable to the ad-hoc networks.

Another proposal is to maintain, in some way, the *hop-count* metric in the model. So, a mechanism to control the length of the paths was introduced. This strategy aims to avoid excessive growth in route sizes and, consequently, the increase of contention and interference. The second model is equal to the weighted sum but with the addition of the *hop control factor* mentioned above.

The Optimized Link State Routing (OLSR) routing protocol was utilized. OLSR is very popular thanks to its ability to build a MultiPoint Relay (MPR) set that is in charge of propagating the routing messages for the entire network. This approach helps to avoid the flooding of redundant control packets. Five well-stated QoS-aware metrics were implemented. MPR and Shortest Path algorithms were adapted to work with the multicriteria models. All these proposals formed a modular framework developed in the OMNET++ network simulator.

The proposed methods were compared with single-objective and standard weighted sum ones considering an experimental set composed of scenarios that differ regarding mobility, type of application, and the packet generating rate. The goal was to assess which methods were able to produce better and more promising trade-off solutions according to the principal network quality indicators.

In Type A application, the results showed the proposed methods were significantly better than the other ones in the vast majority of the analyzed scenarios. As to Type B, the protocol had to deal with the trade-offs reliability *vs.* delay and reliability *vs.* network lifetime. In this context, the proposed methods were efficient in generating satisfactory compromise solutions, surpassing the simple weighted sum method in most scenarios.

The weights and constraints of proposed methods were chosen utilizing a multicriteria utility function designed to allow the decision maker to set the preferences according to the application or network requirements. This resource contributed to these methods overcome the standard weighted sum method in most of the scenarios studied.

In future studies, two possibilities will be investigated: i) weights and constraints can be defined dynamically, thus accompanying the constant variations of the network. Possible storage and processing overhead must be faced here and, ii) a more straightforward alternative would be to use some multi-path strategy. In this stand, more than one route can be obtained by varying the weights and constraints. Then one of the paths must be chosen based on some decision-making method. Again, issues about extra processing overhead must be analyzed.

Even glimpsing these promising possibilities, the starting point for future investigations is to adjust the present framework to be integrated to other routing protocols such as AODV, DYMO, BATMAN, and ExOR. In this way, a more extensive overview of the performance of routing protocols when applying the proposed framework can be reached. In a bolder view, a unique protocol that integrates the advantages of all routing paradigms can be envisioned from the proposed generic structure.

Finally, according to presented results, the addition of multiple QoS-aware metrics is a promising approach to handle the routing problem in Ad-hoc Wireless Networks. It is hoped that this work is an attractive source to promote further research in this area.

Appendix A

Complementary analysis

A.1 No-mobility scenario

Application: Type B

Figure A.1 shows the confidence intervals of differences computed for each pair of methods.

- **PLR and Throughput:** In **10kbps:** rate, the move of using a multicriteria method that combines a reliability-oriented metric and the delay-aware metric was enough to reduce packet losses (about 80% to 90%) and to increase the throughput (about 85% to 95%) significantly when compared to MD. Besides, all multicriteria methods were equally good in both quality indicators.

  In **20kbps** and **30kbps**, note that ωMD-LLT and h methods outperformed the others. They performed better because pruning methods (ρ) may not find feasible routes in the search process at some point.

- **E2ED and NLT:** Figure A.1 shows the MD outperformed the others when the network works with **10kbps** and **20kbps** rates. In these scenarios, due to the weak performance of MD in PLR, the trade-off between delivery reliability and delay is clear. The main advantage of the MD lies in its ability to generate short paths, as it is shown in Table A.1. In NLT, the methods that got better performance in PLR naturally spent more energy because of their higher packet delivery rate.

  However, when shifting to the **30kbps** rate, MD is not so efficient in E2ED. In such a heavy traffic scenario, the risk of building routes formed by overloaded links is high, which causes more delay due to queues congestion. Multicriteria methods

Table A.1: **Application: Type B**. Average number of hops.

Method	10kbps	20kbps	30kbps
ρMD-LLT	1.75	2.06	2.38
ρMD-ML	1.93	2.15	2.47
ρMD-ML^{rand}	1.86	2.53	2.87
hMD-ETX	1.90	2.07	2.37
hMD-ML	**1.67**	2.07	2.33
ωMD-ML	2.25	2.58	3.04
MD	1.70	**1.79**	**2.09**

were significantly better because they include a reliability/stability metric (ML, ETX, or LLT) that attenuates this risk.

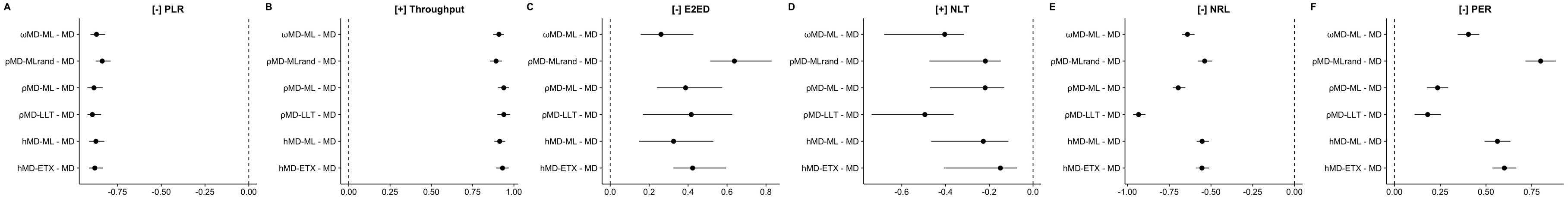
Still in **30kbps**, it is particularly interesting to check how the ρMD-LLT extends the network lifetime (from 50% to 60%) when compared to the MD. The choice of the ϵ value contributed to the good performance. With the restriction that the link stability should be greater than or equal to 0.1 ($\epsilon = 0.1$), only those links that are estimated to be close to dying are pruned (Figure 3.2). Accordingly, the routes take longer to break down as they are formed by more stable or newly links.

- **NRL:** Getting a good performance in PLR usually means that some reliability or stability metric was used. These metrics also help to reduce link breakages and the need for TC message transmissions. For that reason, multicriteria methods should have the best results in this indicator. This occurs in **10kbps** and **20kbps** rates, as it is shown in Figure A.1. In **30kbps**, due to the heavy traffic, the average number of hops had apparently a strong impact on the indicator. In fact, the smaller the number of hops, the smaller is the flood of routing packets when some link breaks down (Table A.1).

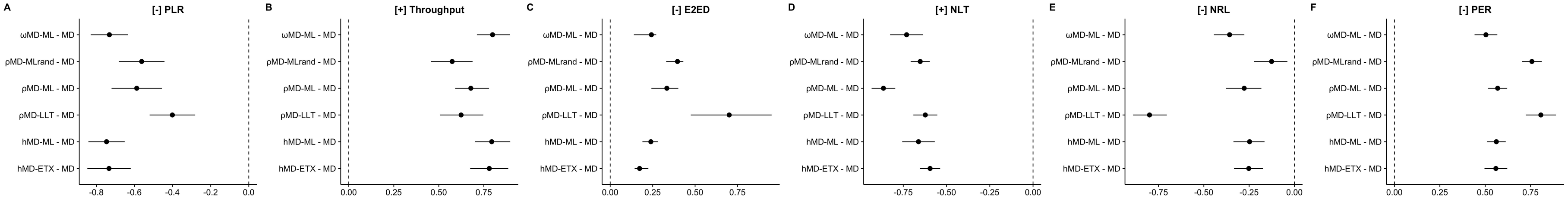
- **PER:** In digital transmission, PER describes the percentage of received bits of a data stream that have been altered due to noise, interference, distortion or bit synchronization errors. In this particular scenario, the difference between multicriteria methods and MD is significantly big regarding PLR. Therefore, it is natural that PER is higher for these methods, as it occurs in **10kbps** and **20kbps** rates (Figure A.1). As to **30kbps**, multicriteria methods beat MD with emphasis on the ρMD-LLT. This result suggests these methods can reduce the PER when the network has high data traffic.

Average packet generation rate for Type B application

10kbps



20kbps



30kbps

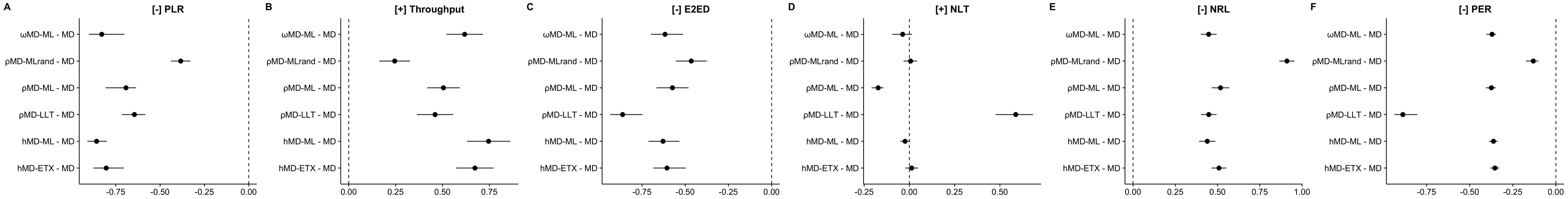


Figure A.1: **Application: Type B**. 95% confidence intervals for differences for the QoS indicators, with 10kbps, 20kbps and 30kbps of average packet generation rate.

A.2 Average speed scenario

Application: Type A

Looking at Figure A.2, the weighted sum method (ωML-*REC*) draws attention in PLR, Throughput, E2ED, and NLT indicators. In **10kbps** and **20kbps**, it got results equally good when compared with ρ and h methods. As to **30kbps**, it lost 33% more packets and got 20% smaller throughput than ML. The path size helps to explain this disparity. Table A.2 shows that ωML-*REC* obtained an average number of hops similar to the ML and higher than ρ and h methods. Therefore, in such a mobility and high traffic scenario, the *hop control factor* helps to guarantee the best values of proposed methods.

Still, ωML-*REC* had the highest E2ED and PER because of the greater average number of hops.

Table A.2: **Application: Type A.** Average number of hops.

Method	10kbps	20kbps	30kbps
ρ*REC-LLT*	**1.75**	**1.81**	1.99
ρ*REC-*ML	1.77	1.82	**1.98**
$hETX$-*REC*	1.78	1.85	**1.98**
hML-*REC*	1.79	1.84	**1.98**
ωML-*REC*	1.79	1.87	2.39
ML	2.46	2.38	2.41

The routing solutions supposedly spend less energy if PLR is high and the application works with UDP (a transport protocol that does not retransmit lost packets). In this context, higher PLR promotes energy savings as some intermediate nodes are not triggered to relay these lost packets. In short, dropping packets may mean longer network lifetime. Because of that, ωML-*REC* got the highest NLT in the **30kbps** rate, since it is the worst method in PLR indicator. Again, there is a trade-off between reliability and network lifetime in this type of application.

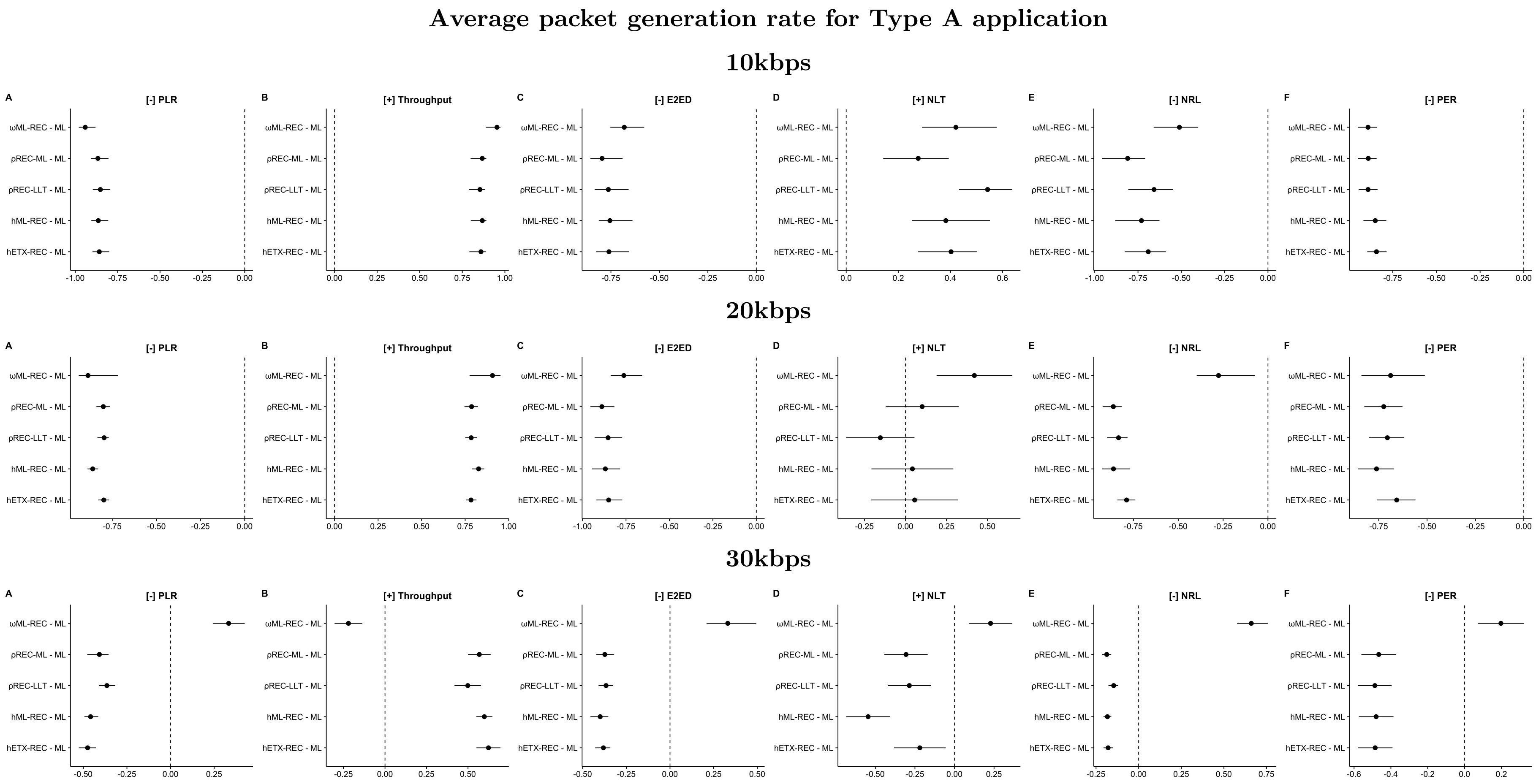


Figure A.2: **Application: Type A**. 95% confidence intervals for differences for the QoS indicators, with 10kbps, 20kbps and 30kbps of average packet generation rate.

Application: Type B

Through confidence intervals shown in Figure A.3, it is possible to highlight the main differences between the methods and draw some conclusions. Looking at PLR and Throughput in **10kbps** and **20kbps** rates, the results show considerable gains when combining a reliability-oriented metric and a delay-aware metric. These outcomes strengthen the proposal to use multicriteria methods also in mobility scenarios and with real-time multimedia applications.

In high traffic scenario (**30kbps**), the proposed methods (ρ and h) were not significantly better than MD in PLR indicator. The smaller number of hops of the MD (Table A.3) may have helped to equate its performance. However, this considerable PLR tends to promote discomforts that may drastically impair the QoS provided to the application.

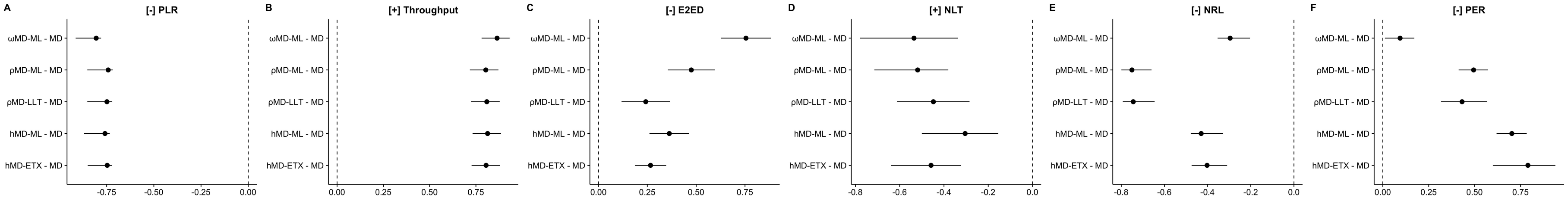
Table A.3: **Application: Type B.** Average number of hops.

Method	10kbps	20kbps	30kbps
ρMD-LLT	1.75	**1.87**	2.19
ρMD-ML	1.75	1.89	2.21
hMD-ETX	**1.65**	**1.87**	2.20
hMD-ML	1.67	1.89	2.17
ωMD-ML	1.87	1.91	2.34
MD	1.93	2.42	**2.02**

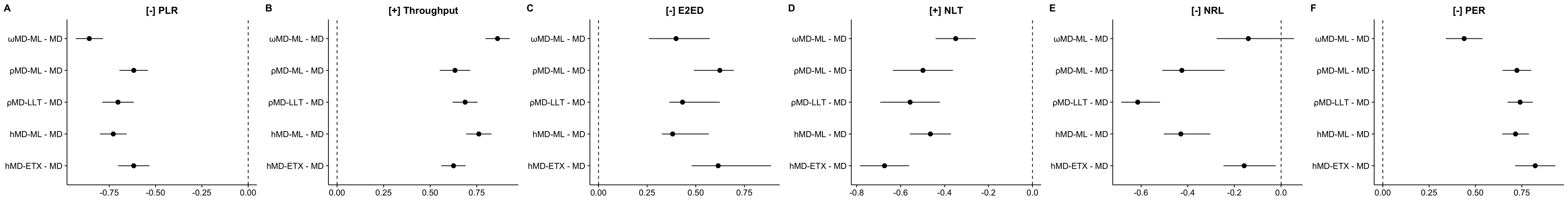
In E2ED, NLT, NLR, and PER, MD was more effective. These results again show the trade-offs: delivery reliability *vs.* delay and delivery reliability *vs.* network lifetime. In E2ED, the inferior performance of ρ and h methods is not critical. Figure 6.11 shows that values are always below 0.15 seconds (150 milliseconds), which is an acceptable limit for real-time multimedia applications [51, 57].

Average packet generation rate for Type B application

10kbps



20kbps



30kbps

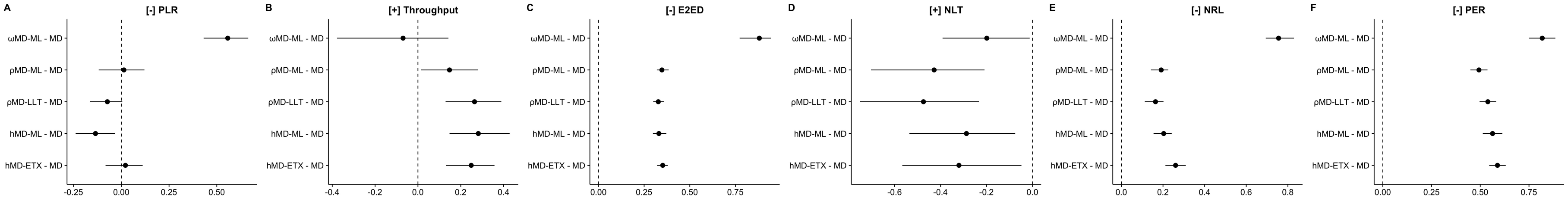


Figure A.3: **Application: Type B**. 95% confidence intervals for differences for the QoS indicators, with 10kbps, 20kbps and 30kbps of average packet generation rate.

A.3 High-speed scenario

The results of this scenario are very similar to the average speed scenario. The main difference is that in this one the variability increased mainly because of higher mobility. So, for the sake of objectivity, only the relevant differences are commented at this spot.

Application: Type A

Table A.4 shows a brief overview of the results considering each of the quality indicators.

Table A.4: **Application: Type A.** Overview of results.

Indicator	Overview
PLR	**10kbps** and **20kbps**: ωML-REC was more efficient. **30kbps:** neither method was able to keep the rate of loss less than 20%.
Throughput	**10kbps** and **20kbps**: ωML-REC was more efficient. **30kbps:** proposed methods overcome standard weighted sum (ωML-REC) and single-objective (ML) methods.
E2ED	**10kbps** and **20kbps**: multicriteria methods presented similarly good performances for practical purposes. **30kbps**: Proposed methods are significantly better than ML and ωML-REC.
NLT	there were no significant differences between the methods.
NRL	all proposed methods (ρ and h) were equally good in the three rates.
PER	**10kbps** and **20kbps**: multicriteria methods were significantly better than ML. **30kbps:** it was not possible to observe any significant difference between the proposed methods.

From Figure A.4, it can be seen the ML method is significantly worse than most of the other ones in practically all quality indicators. In this context, ωML-REC was more efficient when analyzing PLR and Throughput indicators in **10kbps** and **20kbps** rates.

In the heavy traffic scenario (**30kbps**), proposed methods (ρ and h) reversed the score. Unlike the other rates, *hop control factor* had a positive impact here (Table A.5). Since ωML-REC does not use it, its average number of hops grows (2.34) and, hence, its PLR grows together. The same effect can be observed to Throughput.

Table A.5: **Application: Type A.** Average number of hops.

Method	10kbps	20kbps	30kbps
ρREC-LLT	1.77	**1.80**	2.02
ρREC-ML	1.77	1.84	1.99
$hETX$-REC	1.78	1.81	**1.98**
hML-REC	**1.76**	1.82	2.08
ωML-REC	**1.76**	1.85	2.34
ML	2.44	2.39	2.40

In NLT, in spite of using an energy-aware metric, multicriteria methods were not able to excel the ML at any of the rates, mainly because of the high mobility and their better performance in PLR indicator.

Application: Type B

Table A.6 shows a brief overview of the results considering each of the quality indicators.

Table A.6: **Application: Type B.** Overview of results.

Indicator	Overview
PLR and Throughput	**10kbps** and **20kbps**: ωML-REC had the best results. **30kbps:** no method was able to guarantee the PLR lower than 20%. But, multicriteria methods were the best in this heavier scenario.
E2ED, NLT, and PER	MD performed better overall.
NRL	**10kbps** and **20kbps**: proposed methods performed better overall. **30kbps**: MD and proposed methods had similar performance.

Multicriteria methods were more efficient than the other ones in getting shorter routes (Table A.7), which reduces the PLR generated by congestion in the relay nodes queues. Glancing at the confidence intervals in Figure A.5, ωMD-ML got better results in **10kbps** and **20kbps** in PLR and Throughput.

Table A.7: **Application: Type B.** Average number of hops.

Method	10kbps	20kbps	30kbps
ρMD-LLT	1.76	1.81	2.04
ρMD-ML	1.75	1.85	2.02
hMD-ETX	**1.73**	1.82	2.00
hMD-ML	**1.73**	**1.79**	**1.96**
ωMD-ML	1.80	1.86	2.21
MD	2.44	2.60	2.84

Lastly, proposed methods (ρ and h) had good results in NRL, mainly because they seek to build more stable routes. So, TC messages will be possibly less frequent.

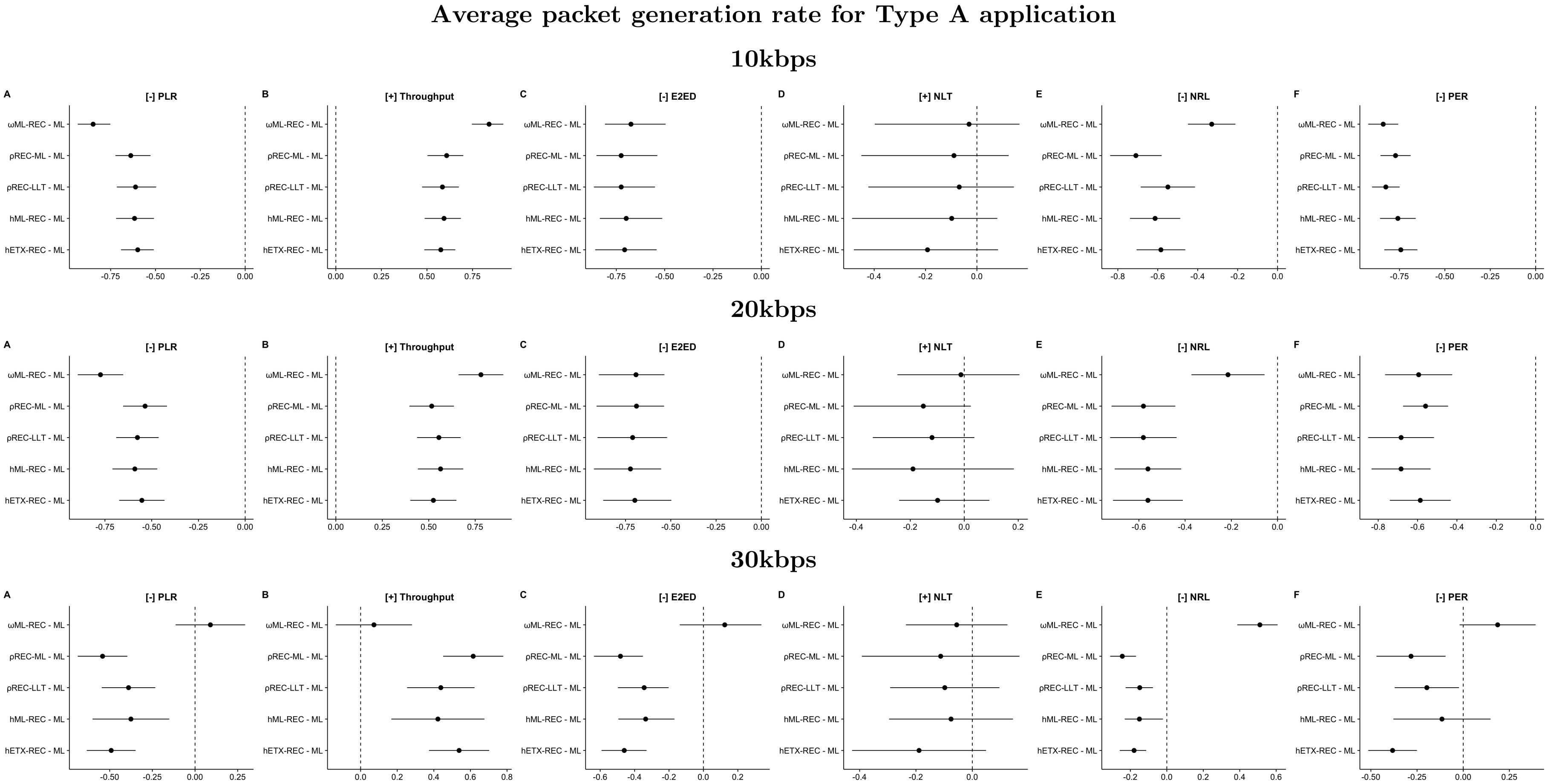
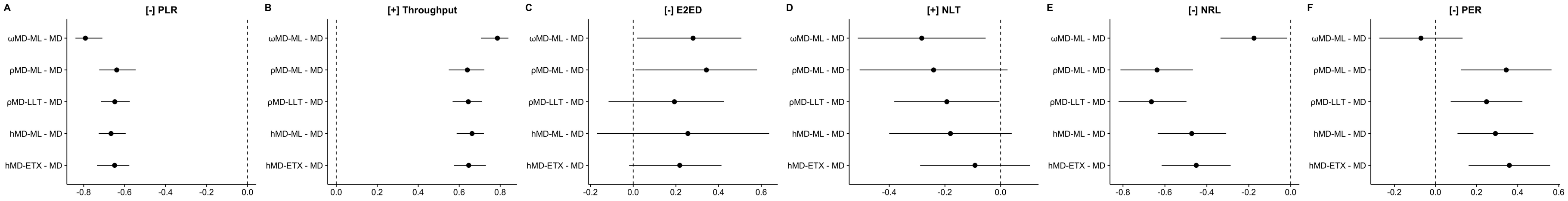


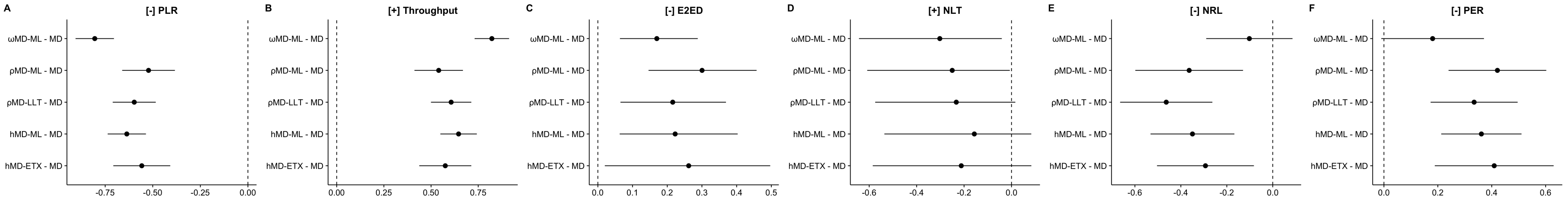
Figure A.4: **Application: Type A**. 95% confidence intervals for differences for the QoS indicators, with 10kbps, 20kbps and 30kbps of average packet generation rate.

Average packet generation rate for Type B application

10kbps



20kbps



30kbps

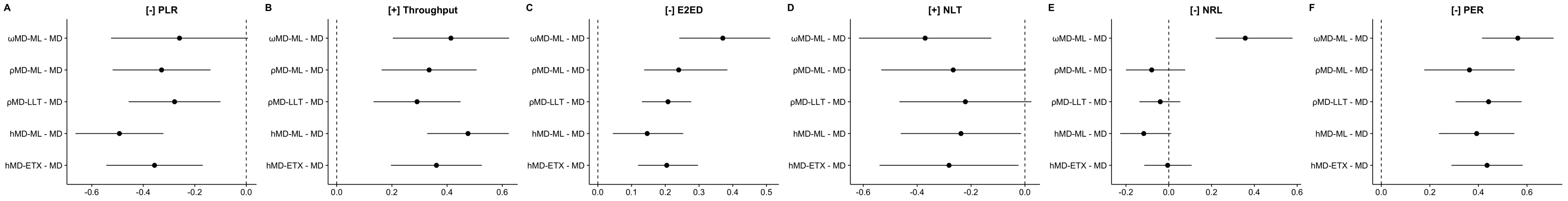


FIGURE A.5: **Application: Type B**. 95% confidence intervals for differences for the QoS indicators, with 10kbps, 20kbps and 30kbps of average packet generation rate.

Appendix B

Publications arising from this work

ARAÚJO, J. N. R.; MONTEIRO, C. C. ; BATISTA, L. S. . Multicriteria QoS-aware Solution in Wireless Multi-hop Networks. In: The Thirteenth International Conference on Wireless and Mobile Communications (ICWMC 2017), 2017, Nice. Proceedings of the 13th ICWMC, 2017. v. 7. p. 17-23.

ARAÚJO, J. N. R.; MONTEIRO, C. C. ; BATISTA, L. S. . Roteamento em redes sem fio ad-hoc: uma abordagem multiobjetivo. In: Simpósio Brasileiro de Automação Inteligente (SBAI), 2017, Porto Alegre. Anais do SBAI, 2017. p. 1-7.

ARAÚJO, J. N. R.; PEREIRA, L. M. ; BATISTA, L. S. . Definição do Layout de uma rede de sensores via algoritmo de evolução diferencial multiobjetivo.. In: Brazilian Congress on Computational Intelligence (CBIC), 2017, Rio de Janeiro. Proceedings of the XIII Brazilian Congress on Computational Intelligence, 2017. p. 1-12.

Bibliography

[1] Muhammad Alam, Joaquim Ferreira, and José Fonseca. Introduction to intelligent transportation systems. In *Intelligent Transportation Systems*, pages 1–17. Springer, 2016.

[2] Saleh Yousefi, Mahmood Fathy, and Saeed Bastani. Vehicular ad hoc networks: Current issues and future challenges. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 329–378, 2016.

[3] Shu Yinbiao, Kang Lee, Peter Lanctot, Fan Jianbin, Hu Hao, Bruce Chow, and Jean-Pierre Desbenoit. Internet of things: wireless sensor networks. *White Paper, International Electrotechnical Commission, http://www.iec.ch*, 2014.

[4] Marco Centenaro, Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios. *IEEE Wireless Communications*, 23(5):60–67, 2016.

[5] Christos Zachos, Jonathan Loo, and Shafiullah Khan. Wireless mesh network: Architecture and protocols. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 425–448, 2016.

[6] Ozgur Koray Sahingoz. Networking models in flying ad-hoc networks (FANETs): Concepts and challenges. *Journal of Intelligent & Robotic Systems*, 74(1-2):513–527, 2014.

[7] Daniele Miorandi, Sabrina Sicari, Francesco De Pellegrini, and Imrich Chlamtac. Internet of things: Vision, applications and research challenges. *Ad Hoc Networks*, 10(7):1497–1516, 2012.

[8] Luiz Filipe M Vieira. Underwater sensor networks. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 413–424, 2016.

[9] Miguel Garcia, Sandra Sendra, Marcelo Atenas, and Jaime Lloret. Underwater wireless ad hoc networks: A survey. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 379–411, 2016.

[10] Denis Rosário, Zhongliang Zhao, Torsten Braun, Eduardo Cerqueira, Aldri Santos, and Islam Alyafawi. Opportunistic routing for multi-flow video dissemination over flying ad-hoc networks. In *15th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–6. IEEE, 2014.

[11] Uichin Lee, Ryan Cheung, and Mario Gerla. Emerging vehicular applications. *Vehicular Networks: From Theory to Practice. Chapman and Hall/CRC*, 2009.

[12] Xiuzhen Cheng, Xiao Huang, and Ding-Zhu Du. *Ad hoc wireless networking*, volume 14. Springer Science & Business Media, 2013.

[13] Jonathan Loo, Jaime Lloret Mauri, and Jesús Hamilton Ortiz. *Mobile ad hoc networks: current status and future trends.* CRC Press, 2016.

[14] Lyes Khoukhi, Ali El Masri, and Dominique Gaiti. Quality-of-service state information-based solutions in wireless mobile ad hoc networks: A survey and a proposal. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 279–311, 2016.

[15] Waheb Jabbar, Mahamod Ismail, and Rosdiadee Nordin. On the performance of the current MANET routing protocols for VoIP, HTTP, and FTP applications. *Journal of Computer Networks and Communications*, 2014.

[16] Nadeem Javaid, Muti Ullah, and Karim Djouani. Identifying design requirements for wireless routing link metrics. In *Global Telecommunications Conference (GLOBECOM)*, pages 1–5. IEEE, 2011.

[17] Nadeem Javaid, Ayesha Bibi, Akmal Javaid, ZA Khan, K Latif, and M Ishfaq. Investigating quality routing link metrics in wireless multi-hop networks. *Annals of Telecommunications*, 69(3-4):209–217, 2014.

[18] Ali Moussaoui, Fouzi Semchedine, and Abdallah Boukerram. A link-state QoS routing protocol based on link stability for mobile ad hoc networks. *Journal of Network and Computer Applications*, 39:117–125, 2014.

[19] Zhongliang Zhao, Denis Rosário, Torsten Braun, and Eduardo Cerqueira. Context-aware opportunistic routing in mobile ad-hoc networks incorporating node mobility. In *Wireless Communications and Networking Conference (WCNC)*, pages 2138–2143, 2014.

[20] Bego Blanco Jauregui and Fidel Liberal Malaina. New approaches to mobile ad hoc network routing: application of intelligent optimization techniques to multicriteria routing. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 171–200, 2016.

[21] Yann Collette and Patrick Siarry. *Multiobjective optimization: principles and case studies*. Springer Science & Business Media, 2013.

[22] Jonathan Loo, Shafiullah Khan, and Ali Nasser Al-Khwildi. *Mobile Ad Hoc Rougint Protocols*. CRC Press, 2016.

[23] Nuha Alwan. Performance analysis of Dijkstra-based weighted sum minimization routing algorithm for wireless mesh networks. *Modelling and Simulation in Engineering*, page 32, 2014.

[24] Zhihao Guo, Shahdi Malakooti, Shaya Sheikh, Camelia Al-Najjar, and Behnam Malakooti. Multi-objective OLSR for proactive routing in MANET with delay, energy, and link lifetime predictions. *Applied Mathematical Modelling*, 35(3):1413–1426, 2011.

[25] Run Ye, Azzedine Boukerche, Houjun Wang, Xiaojia Zhou, and Bin Yan. E3TX: an energy-efficient expected transmission count routing decision strategy for wireless sensor networks. *Wireless Networks*, pages 1–14, 2017.

[26] Soumyadip Sengupta, Swagatam Das, MD Nasir, and Bijaya K Panigrahi. Multi-objective node deployment in WSNs: In search of an optimal trade-off among coverage, lifetime, energy consumption, and connectivity. *Engineering Applications of Artificial Intelligence*, 26(1):405–416, 2013.

[27] Yang Xu, Jia Liu, Yulong Shen, Xiaohong Jiang, and Tarik Taleb. Security/QoS-aware route selection in multi-hop wireless ad hoc networks. In *International Conference on Communications (ICC)*, pages 1–6. IEEE, 2016.

[28] Nessrine Chakchouk. A survey on opportunistic routing in wireless communication networks. *IEEE Communications Surveys & Tutorials*, 17(4):2214–2241, 2015.

[29] Ali Moussaoui and Abdallah Boukeream. A survey of routing protocols based on link-stability in mobile ad hoc networks. *Journal of Network and Computer Applications*, 47:1–10, 2015.

[30] Behrouz A Forouzan and Firouz Mosharraf. *Redes de computadores: uma abordagem top-down*. AMGH Editora, 2013.

[31] Raquel Lacuesta, Miguel García, Jaime Lloret, and Guillermo Palacios. Study and performance of mobile ad hoc routing protocols. *Mobile Ad Hoc Networks: Current Status and Future Trends*, page 71, 2016.

[32] Jose Costa-Requena. Ad hoc routing modeling and mathematical analysis. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 102–148, 2016.

[33] Ian Chakeres. Dynamic MANET on-demand (DYMO) routing. *Internet Draft (draft-ietf-manet-dymo-14)*, 2008.

[34] Axel Neumann, Corinna Aichele, Marek Lindner, and Simon Wunderlich. Better approach to mobile ad-hoc networking (BATMAN). *IETF draft*, pages 1–24, 2008.

[35] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (AODV) routing. Technical report, IETF, 2003.

[36] Thomas Clausen and Philippe Jacquet. Optimized link state routing protocol (OLSR). Technical report, IETF, 2003.

[37] Ying Ge, Thomas Kunz, and Louise Lamont. Quality of service routing in ad-hoc networks using OLSR. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, pages 9–pp. IEEE, 2003.

[38] Hakim Badis and Khaldoun Al Agha. QOLSR, QoS routing for ad hoc wireless networks using OLSR. *Transactions on Emerging Telecommunications Technologies*, 16(5):427–442, 2005.

[39] Nicklas Beijar. Zone routing protocol (ZRP). *Networking Laboratory, Helsinki University of Technology, Finland*, 9:1–12, 2002.

[40] Sanjit Biswas and Robert Morris. Opportunistic routing in multi-hop wireless networks. *SIGCOMM Computer Communication Review*, 34(1):69–74, 2004.

[41] Henri Dubois-Ferrière, Matthias Grossglauser, and Martin Vetterli. Valuable detours: Least-cost anypath routing. *IEEE/ACM Transactions on Networking (TON)*, 19(2):333–346, 2011.

[42] Rafael Laufer, Pedro B Velloso, Luiz Filipe M Vieira, and Leonard Kleinrock. Plasma: A new routing paradigm for wireless multihop networks. In *Proceedings of the INFOCOM*, pages 2706–2710. IEEE, 2012.

[43] Nan Liu and Winston KG Seah. Performance evaluation of routing metrics for community wireless mesh networks. In *7th International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pages 556–561. IEEE, 2011.

[44] Floriano De Rango, Francesca Guerriero, Salvatore Marano, and Emilia Bruno. A multiobjective approach for energy consumption and link stability issues in ad hoc networks. *IEEE Communications Letters*, 10(1):28–30, 2006.

[45] Waheb A Jabbar, Mahamod Ismail, and Rosdiadee Nordin. Multi-criteria based multipath OLSR for battery and queue-aware routing in multi-hop ad hoc wireless networks. *Wireless Networks*, 21(4):1309–1326, 2015.

[46] Muhammad Iqbal, Muhammad Naeem, Alagan Anpalagan, Ashfaq Ahmed, and Muhammad Azam. Wireless sensor network optimization: multi-objective paradigm. *Sensors*, 15(7):17572–17620, 2015.

[47] Mitra Ahmadi, Mohammad Shojafar, Ahmad Khademzadeh, Kambiz Badie, and Reza Tavoli. A hybrid algorithm for preserving energy and delay routing in mobile ad-hoc networks. *Wireless Personal Communications*, 85(4):2485–2505, 2015.

[48] Nicola Costagliola, Pedro Garçia López, Francesco Oliviero, and Simon Pietro Romano. Energy and delay-efficient routing in mobile ad hoc networks. *Mobile Networks and Applications*, 17(2):281–297, 2012.

[49] Yakubu Tsado, K Gamage, Bamidele Adebisi, David Lund, Khaled M Rabie, and Augustine Ikpehai. Improving the reliability of OLSR routing in smart grid NAN based wireless mesh network using multiple metrics. *Energies*, 10(3), 2017.

[50] Michael Emmerich and André Deutz. Multicriteria optimization and decision making. *LIACS. Leiden university, NL*, 2006.

[51] Larry L Peterson and Bruce S Davie. *Computer Networks: A Systems Approach*. Morgan Kaufmann, 2011.

[52] Weverton Cordeiro, Elisangela Aguiar, Waldir Moreira Junior, Antonio Abelem, and Michael Stanton. Providing quality of service for mesh networks using link delay measurements. In *Proceedings of 16th International Conference on Computer Communications and Networks (ICCCN)*, pages 991–996. IEEE, 2007.

[53] Douglas SJ De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing. *Wireless Networks*, 11(4): 419–434, 2005.

[54] Richard Draves, Jitendra Padhye, and Brian Zill. Comparison of routing metrics for static multi-hop wireless networks. In *ACM SIGCOMM Computer Communication Review*, volume 34, pages 133–144. ACM, 2004.

[55] Fei Shi, Dongxu Jin, and Jooseok Song. A survey of traffic-based routing metrics in family of expected transmission count for self-organizing networks. *Computers & Electrical Engineering*, 40(6):1801–1812, 2014.

[56] Christos Papageorgiou, Panagiotis Kokkinos, and Varvarigos Emmanouel. Energy-efficient unicast and multicast communication for wireless ad hoc networks using multiple criteria. *Mobile Ad Hoc Networks: Current Status and Future Trends*, pages 201–230, 2016.

[57] James F Kurose and Keith W Ross. *Computer networking: a top-down approach*, volume 6. Pearson, 2013.

[58] David Salomon. *A guide to data compression methods*. Springer Science & Business Media, 2013.

[59] Waheb A Jabbar, Mahamod Ismail, and Rosdiadee Nordin. MBA-OLSR: a multipath battery aware routing protocol for MANETs. In *5th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, pages 630–635. IEEE, 2014.

[60] Floriano De Rango, Francesca Guerriero, and Peppino Fazio. Link-stability and energy aware routing protocol in distributed wireless networks. *IEEE Transactions on Parallel and Distributed systems*, 23(4):713–726, 2012.

[61] Venkata C Giruka and Mukesh Singhal. Hello protocols for ad-hoc networks: overhead and accuracy tradeoffs. In *World of Wireless Mobile and Multimedia Networks (WoWMoM)*, pages 354–361. IEEE, 2005.

[62] Diego Passos, Douglas Vidal Teixeira, Débora C Muchaluat-Saade, Luiz C Schara Magalhães, and Célio Albuquerque. Mesh network performance measurements. In *International Information and Telecommunicatios Technologies Symposium (I2TS)*, pages 48–55, 2006.

[63] L-J Chen, Tony Sun, Guang Yang, Medy Young Sanadidi, and Mario Gerla. Ad hoc probe: path capacity probing in wireless ad hoc networks. In *First International Conference on Wireless Internet*, pages 156–163. IEEE, 2005.

[64] Chai-Keong Toh. Associativity-based routing for ad hoc mobile networks. *Wireless personal communications*, 4(2):103–139, 1997.

[65] KN Sridhar and Mun Choon Chan. Stability and hop-count based approach for route computation in MANET. In *Proceedings of the 14th International Conference on Computer Communications and Networks (ICCCN)*, pages 25–31. IEEE, 2005.

[66] Ingo Gruber and Hui Li. Link expiration times in mobile ad hoc networks. In *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks (LCN)*, pages 743–750. IEEE, 2002.

[67] Trong-Thua Huynh, Anh-Vu Dinh-Duc, and Cong-Hung Tran. Delay-constrained energy-efficient cluster-based multi-hop routing in wireless sensor networks. *Journal of Communications and Networks*, 18(4):580–588, 2016.

[68] Vira Chankong and Yacov Y Haimes. *Multiobjective decision making: theory and methodology*. Courier Dover Publications, 2008.

[69] Fillipe Goulart. *Preference-guided Evolutionary Algorithms for Optimization with Many Objectives*. PhD thesis, Universidade Federal de Minas Gerais, 2014.

[70] Liangrui Tang, Sen Feng, Jianhong Hao, and Xiongwen Zhao. Energy-efficient routing algorithm based on multiple criteria decision making for wireless sensor networks. *Wireless Personal Communications*, 80(1):97–115, 2015.

[71] Fernando Kuipers, Piet Van Mieghem, Turgay Korkmaz, and Marwan Krunz. An overview of constraint-based path selection algorithms for QoS routing. *IEEE Communications Magazine*, 40(12):50–55, 2002.

[72] Michael R Garey and David S Johnson. *Computers and Intractability: A Guide to the Theory of NP Completeness*, volume 29. W.H. Freeman & Co., 2002.

[73] Gang Liu and KG Ramakrishnan. A* prune: an algorithm for finding K shortest paths subject to multiple constraints. In *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, volume 2, pages 743–749. IEEE, 2001.

[74] Liping Fu, D Sun, and Laurence R Rilett. Heuristic shortest path algorithms for transportation applications: state of the art. *Computers & Operations Research*, 33(11):3324–3343, 2006.

[75] Natarajan Meghanathan. Stability-energy consumption tradeoff among mobile ad hoc network routing protocols. In *Wireless and Mobile Communications, 2007. ICWMC'07. Third International Conference on*, pages 9–9. IEEE, 2007.

[76] Denis Rosário, Zhongliang Zhao, Torsten Braun, Eduardo Cerqueira, Aldri Santos, and Zan Li. Assessment of a robust opportunistic routing for video transmission in dynamic topologies. In *Wireless Days (WD), IFIP*, pages 1–6. IEEE, 2013.

[77] Amir Qayyum, Laurent Viennot, and Anis Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pages 3866–3875. IEEE, 2002.

[78] Nouha Baccour, Anis Koubâa, Maissa Ben Jamâa, Denis Do Rosario, Habib Youssef, Mário Alves, and Leandro B Becker. Radiale: A framework for designing and assessing link quality estimators in wireless sensor networks. *Ad Hoc Networks*, 9(7):1165–1185, 2011.

[79] Tea Robič and Bogdan Filipič. Differential evolution for multiobjective optimization. In *Evolutionary multi-criterion optimization*, pages 520–533. Springer, 2005.

[80] Frederico G Guimarães. *Evolução diferencial*, chapter 7, pages 141–161. Imprensa da Universidade de Coimbra, 2012.

[81] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[82] Nadeem Javaid, Mariam Akbar, Zahoor Ali Khan, Turki Ali Alghamdi, Malik Najmus Saqib, and Majid Iqbal Khan. Modeling enhancements in routing protocols under mobility and scalability constraints in VANETs. *International Journal of Distributed Sensor Networks*, 10(7):261823, 2014.

[83] Ramin Khalili and Kavé Salamatian. Evaluation of packet error rate in wireless networks. In *Proceeding of MSWIM*, volume 4, 2004.

[84] Guangyu Pei and Thomas R Henderson. Validation of OFDM error rate model in ns-3. *Boeing Research Technology*, pages 1–15, 2010.

[85] Douglas C Montgomery and George C Runger. *Applied statistics and probability for engineers*. John Wiley & Sons, 2010.

[86] OMNeT++ Core Team. OMNeT++ discrete event simulator. `https://omnetpp.org/`, 2017. Accessed: 2017-10-05.

[87] R Core Team. R: a language and environment for statistical computing. `https://www.r-project.org`, 2017. Accessed: 2017-10-05.

[88] OLSR implementation. `http://www.olsr.org/`, 2017. Accessed: 2017-05-24.