Technische Universität München
Lehrstuhl für Kommunikationsnetze
Prof. Dr.-Ing. Wolfgang Kellerer

Master‘s Thesis

Joint Functional Split Adaptation and Path Selection of Automated Vehicles in Industrial Radio Access Networks

Author: Torres Suárez, Jose
Address: Arcisstraße 21
80333 München
Germany
Matriculation Number: 03770196
Supervisor: Kröger, Nicolai
Martínez Alba, Alberto
Begin: 01. April 2023
End: 26. September 2023

With my signature below, I assert that the work in this thesis has been composed by myself independently and no source materials or aids other than those mentioned in the thesis have been used.

München, 26.09.2023

Place, Date

Signature

This work is licensed under the Creative Commons Attribution 3.0 Germany License. To view a copy of the license, visit http://creativecommons.org/licenses/by/3.0/de

Or

Send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

München, 26.09.2023

Place, Date

Signature

To my parents for the opportunity.

To Alberto and Nicolai for the confidence.

To all those who watch over us from heaven.

Thank you.

Kurzfassung

Seit der industriellen Revolution gibt es einen stetigen Drang zur industriellen Automatisierung mit dem Ziel, menschliche Arbeit durch maschinenbetriebene Aufgaben zu ersetzen. Jüngste Fortschritte in mobilen Funkzugangsnetzen, Positionierungssystemen und künstlicher Intelligenz haben die Integration von autonom gesteuerten Fahrzeugen (AGVs) in industriellen Umgebungen vorangetrieben. Diese AGVs erfüllen Aufgaben wie den Transport von Materialien zwischen festen Stationen und die Überwachung von Fabrikzuständen, alles ohne menschliches Eingreifen. Für eine optimale Funktion sind AGVs auf eine drahtlose Verbindung zum größeren industriellen Netzwerk angewiesen. Um eine robuste Konnektivität über ausgedehnte industrielle Flächen zu gewährleisten, wird die Installation mehrerer Basisstationen erforderlich. Aufgrund des begrenzten drahtlosen Spektrums teilen diese Stationen jedoch oft Funkressourcen, was zu möglichen Zell-zwischen-Interferenzen führen kann. Solche Störungen können den Betrieb von AGVs beeinträchtigen und zu Konnektivitätsproblemen, betrieblichen Ungenauigkeiten, temporären Stopps oder sogar kompletten Serviceausfällen führen.

Ein mögliches Mittel gegen Zell-zwischen-Interferenzen ist der Einsatz von Störungskoordinations- oder -annullierungstechniken zwischen Basisstationen. Diese Lösungen erfordern in der Regel die Zentralisierung der Funktionen der koordinierten Basisstationen. Während eine vollständige Zentralisierung herausfordernd ist, kann ein partieller und adaptiver funktionaler Split implementiert werden, um die Interferenz in einer ausgewählten Gruppe von Basisstationen zu steuern. Es ist entscheidend zu erkennen, dass jeder funktionale Split, wenn er nicht auf die dynamischen Positionen der AGVs zugeschnitten ist, dennoch zu Konnektivitätsproblemen führen könnte, selbst wenn er für durchschnittliche Szenarien optimiert ist. Darüber hinaus könnten AGVs, die sich ohne Berücksichtigung des aktuellen funktionalen Splits bewegen, unbeabsichtigt durch hoch gestörte Zonen fahren, anstatt durch Bereiche, in denen die Interferenz gesteuert wird.

In dieser Arbeit beabsichtigen wir, Strategien zu untersuchen, die eine gleichzeitige Anpassung des funktionalen Splits des Funkzugangsnetzes und der Pfadauswahl von AGVs in industriellen Kontexten ermöglichen. Diese Untersuchung konzentriert sich auf die Gewährleistung der Konnektivität, die Optimierung der Leistung und die Minimierung der Betriebskosten.

Abstract

Since the industrial revolution, there has been a consistent push towards industrial automation with the goal of replacing human labor with machine-driven tasks. Recent advancements in mobile radio access networks, positioning systems, and artificial intelligence have spurred the integration of autonomously guided vehicles (AGVs) in industrial settings. These AGVs serve purposes such as transporting materials between fixed stations and monitoring factory statuses, all without human intervention. For optimal functioning, AGVs rely on a wireless connection to the broader industrial network. To ensure robust connectivity across expansive industrial spaces, it becomes necessary to install multiple base stations. However, due to the limited wireless spectrum, these stations often share radio resources, leading to potential intercell interference. Such interference can impair AGV operations, causing connectivity issues, operational inaccuracies, temporary halts, or even complete service disruptions.

One potential remedy for intercell interference is the employment of interference coordination or cancellation techniques among base stations. These solutions typically necessitate centralizing the functions of the coordinated base stations. While achieving complete centralization is challenging, a partial and adaptive functional split can be implemented to manage the interference across a select group of base stations. It's crucial to recognize that any functional split, if not tailored to the dynamic positions of AGVs, might still result in connectivity issues, even if optimized for average scenarios. Moreover, AGVs that move without considering the present functional split might inadvertently traverse high-interference zones, rather than areas where interference is managed.

In this thesis, we aim to explore strategies that allow for the concurrent adaptation of the radio access network's functional split and AGVs' path selection in industrial contexts. This exploration will focus on ensuring connectivity, optimizing performance, and minimizing operational costs.

Contents

Contents 6

1 Introduction 8

2 Background 10
2.1 Mobile Radio Access Network 10
2.2 Interference and Functional Split 12
2.2.1 Functional Split Layers 13
2.3 Industrial Mobile Network 15
2.4 Path Optimization Models 17
2.5 Mixed-integer Linear/Quadratic Problem 19
2.5.1 Gurobi 19
2.5.2 Glover's Constraints: Zero-One Polynomial Problem 20

3 Theory 21
3.1 Problem Statement 21
3.2 Initial Problem Simplified 27
3.3 Optimization algorithm formulation 30

4 Implementation and Evaluation 34
4.1 Implementation 34
4.1.1 Simplified Problem 36
4.1.2 Path and Functional Split Optimization Algorithm 38
4.1.3 Many AGVs are Present in the Given Scenario. 41
4.2 Results 43
4.2.1 Comparison Between Two Presented Methods 43

5 Conclusions and Outlook 53

List of Figures 55

List of Tables 57

A Notation und Abkürzungen **58**

Bibliography **59**

Chapter 1

Introduction

Industrial automation has been sought since the days of the industrial revolution, with the objective of substituting human labor with self-governing machinery. The recent progress in radio access networks, positioning systems and artificial intelligence has accelerated the employment of Autonomous Guided Vehicles (AGVs) in industrial surroundings. AGVs are used to ferry supplies and equipment between fixed points, oversee the state of the factory, etc. without human intervention.

AGVs must be wirelessly connected to the underlying industrial network through a radio access network that enables communication with other factory components. To ensure reliable connectivity across a vast industrial area, multiple base stations are required with partially overlapping coverage areas. Due to the limited and expensive radio spectrum, base stations are typically designed to share radio resources, potentially causing interference between cells. Connectivity loss can pose significant issues for AGVs, causing inaccurate operations, temporary stalling, or the complete cessation of service.

One potential resolution to inter-cell interference involves the adoption of coordination or cancellation procedures between base stations. Nonetheless, the application of these methods necessitates centralizing the functions of coordinating base stations to enable swift and efficient connectivity. Consequently, centralized functions must communicate with the remote locations where the radio equipment is located. This often necessitates prohibitively high link capacities, rendering full centralization inviable. An alternative is partial, adaptive function centralization, also known as adaptive Functional Split (FS). This method coordinates interference from a subset of base stations that can be dynamically selected.

However, a Functional Split that fails to consider the different positions of AGVs will result in a reduction in connectivity, even if they are optimized for the average situation. Moreover, AGVs that disregard the instantaneous Functional Split in their movement are likely to traverse areas with high interference instead of those where coordination of interference is occurring. At present, no solution has been proposed to tackle this issue. We outline a potential methodology for collaboratively adjusting the functional separation and choosing the course of AGVs in order to fulfill connectivity limitations, boost efficiency, and/or decrease operational expenses.

The structure of this Master Thesis is organized as follows:

- **Chapter 2**: We provide the background necessary to understand the progression of this thesis.
- **Chapter 3**: We delve into the detailed theory developed throughout this work. Besides presenting the Functional Split selection problems, we introduce the formulation and variables employed to address these challenges. Additionally, we propose multiple adaptation strategies for guiding the dynamic selection of the Functional Split.
- **Chapter 4**: The experimental implementation is showcased, along with results that evaluate both the model and the proposed adaptation strategies.
- **Chapter 5**: We present the conclusions and open questions arising from this study, suggesting potential directions for future work.

Chapter 2

Background

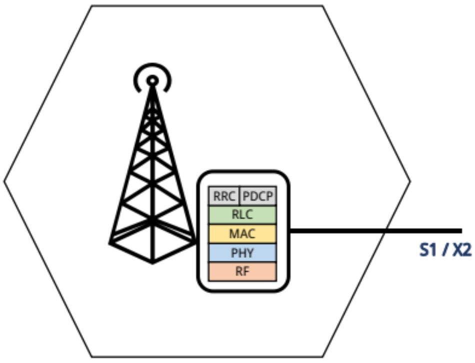
The rapid evolution of the fifth generation (5G) networks, marked by a surge in bandwidth requirements and the promise of ultra-reliable low-latency communication (URLLC), has spurred innovative research areas within the mobile communication domain. A pivotal facet within this paradigm is the intersection of Industrial Radio Access Networks (iRANs) and the consequent challenges and opportunities it presents. This chapter delineates the key foundational concepts and theories that set the stage for this thesis.

2.1 Mobile Radio Access Network

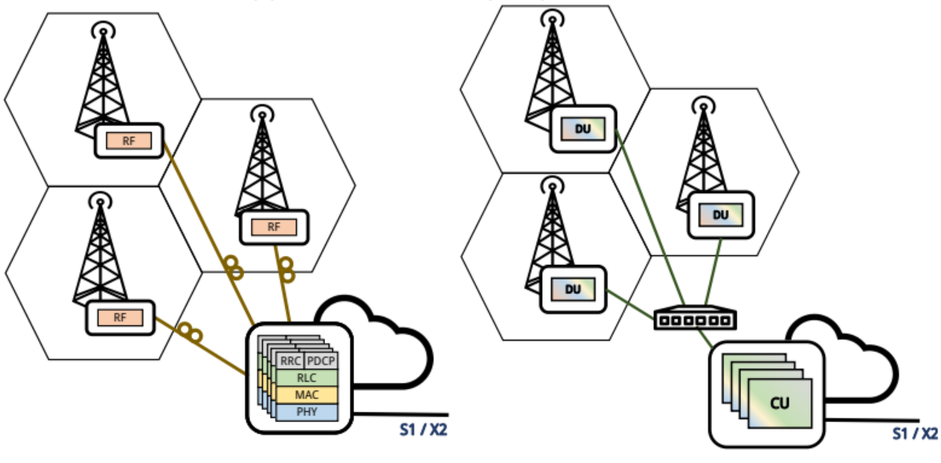
Radio Access Networks (RAN) constitute the foundation of mobile communication, acting as a bridge between user equipment (UE) and the core network. Essentially, RAN is responsible for managing radio resources to ensure user mobility and provide essential connectivity. The advent of 5G has resulted in revolutionary changes in the architecture and complexities of RAN, necessitating the need for more adaptable and flexible infrastructures to cater to data-intensive applications.

The history of the RAN is one of continuous technological development, adapting and evolving with each generation of mobile technology. During the 2G era, RAN moved from analog to digital systems, with GSM quickly becoming the leading standard and serving as the foundation for more advanced mobile communications. The introduction of faster data services and integrated internet-centric services during the pivotal 3G phase acted as a bridge between traditional mobile telephony and the data-driven world we recognize today. The subsequent 4G or LTE phase marked a paradigm shift, emphasizing exponentially higher data rates and introducing an IP-based architecture that fundamentally transformed user experience by accommodating bandwidth, intensive applications such as HD streaming and online gaming. Cloud-RAN concepts are emerging, demonstrating a shift from inflexible, hardware-based solutions to adaptable, software-defined architectures. Each developmental stage of RAN highlights the industry's response to changing customer needs, technological advancements, and the continuous drive towards a more connected global landscape [MA22].

The emergence of 5G in the Mobile RAN sector heralds an array of advancements and challenges, setting it apart from its forerunners. A prominent change involves exploring new frequency opportunities. Unlike prior mobile generations, 5G examines higher frequency bands, such as millimeter waves (mmWaves) ranging from 30-300 GHz. Whilst these frequencies hold the potential for vast data transfer capabilities, their limited range and proneness to interference from physical objects pose difficulties. In contrast, the Sub-6 GHz frequencies serve to strike a harmonious balance, providing a wide coverage range alongside commendable data capacity [MA22].



(a) Distributed RAN (LTE) Architecture



(b) Traditional C-RAN Architecture (c) Functional Splits Architecture

Figure 2.1: 5G RAN Architecture Evolution [GV18].

The technological backbone of the 5G RAN is supported by sophisticated antenna technologies. Massive MIMO is an exemplary technique that equips base stations with an array of antennas, enabling them to simultaneously service multiple users and amplify efficiency and capacity. Beamforming, a complementary adept technique, narrows the focus of signals toward specific users, optimizing signal quality, and curbing interference.

In response to the intrinsic limitations of *mmWaves*, 5G is experiencing a trend towards network densification. There is a noticeable trend towards the implementation of smaller cells, commonly referred to as small cells, which supplement the traditional macro cells, resulting in an intricate network. This sophisticated strategy amplifies the overall network's capacity and availability. A further enhancement of this arrangement is the innovative flexible network infrastructure. Network slicing, a key feature of 5G, enables operators to create several virtual networks from one physical RAN that cater to specific applications such as IoT, AR/VR and autonomous vehicles. Edge computing enhances this feature by enabling data processing to take place closer to its source, which helps to reduce latency.

The 5G RAN is characterized by a well-considered allocation of functionalities between the CU and the DU (Figure 2.1). The CU is responsible for non-real-time functions, while the DU is oriented towards real-time tasks, creating a scalable and adaptable RAN design. This concept complements the goals of RAN virtualization, which involves a shift from specialized hardware to adaptable software solutions that operate on standard servers. This transition reduces costs while enhancing flexibility.

Additionally, there is an urge for interoperability and open architectures within the 5G RAN sector. Projects promoting Open RAN are expanding the limits to guarantee that RAN components from different vendors can work together. This approach democratizes not only to hedge against vendor lock-in but also to lower costs and inspire innovation.

However, the journey towards 5G RAN is not without obstacles. The increased data rates and capacities impose strict requirements on backhaul, necessitating connections that match fiber-level speeds between base stations and the core network. Additionally, given a dense constellation of base stations, the need for energy efficiency becomes even more critical. The intricate management of sophisticated technologies such as beamforming and Massive MIMO necessitates advanced supervisory tools. 5G represents more than an incremental advancement in the RAN's development - it is a revolutionary breakthrough. With commitments to super-fast data transmission, unmatched data rates, and an interconnected planet, the multifaceted interactions between technologies in 5G RAN shape the contours of the upcoming mobile communication era [MA22].

2.2 Interference and Functional Split

One of the most significant challenges facing modern RAN is inter-cellular interference, which results from the overlap of signals from neighbouring cells, causing disruption. As wireless networks become denser to meet growing data demands, this interference worsens. There are several techniques to mitigate inter-cell interference, from static interference mitigation to dynamic interference mitigation to interference cancellation [MA22].

However, the introduction of the Functional Split concept has tackled this issue by distributing functions between the centralized and distributed units, resulting in a RAN architecture that is more adaptable and flexible. By coordinating these functions appropriately, interference can be mitigated with greater efficacy.

Addressing this is the concept of the Functional Split in 5G RAN. The Functional Split pertains to how the traditional base station functions are divided between the Centralized Unit and the Distributed Unit. By judiciously deciding which functions to centralize and which to distribute, operators can strike a balance between fronthaul efficiency and the flexibility to manage interference [AK19]. For instance, certain split options allow for more centralized coordination, which can be instrumental in implementing techniques like Coordinated Multi-Point (CoMP) operation, effectively mitigating inter-cell interference.

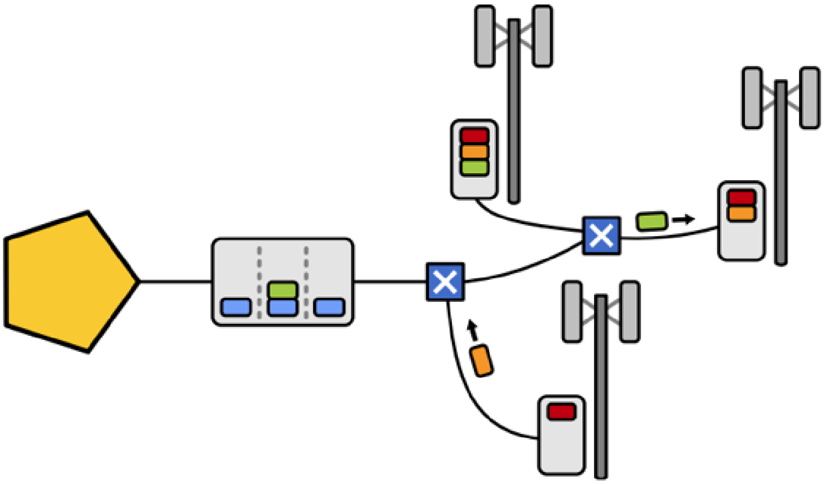


Figure 2.2: Dynamically centralized radio access networks [MA22].

Furthermore, the choice of Functional Split has ramifications beyond just interference management. It impacts latency, scalability, and overall network deployment costs. In the 5G era, the notion of a one-size-fits-all RAN architecture has given way to more adaptive, scalable solutions tailored to specific use-cases and deployment scenarios. Here, the Functional Split plays a pivotal role, offering network designers the flexibility to mold the RAN architecture based on specific needs.

2.2.1 Functional Split Layers

The level of centralization for a base station is established by its Functional Split, which divides the processing functions between those running at the DU and those running at the CU (Figure 2.2).

This split impacts the application of interference mitigation techniques between neighbouring cells. High levels of centralization, such as C-RAN (displayed as option 8 in Figure 2.3) or Intra-PHY (options 7.x), allow for more advanced techniques to cancel interference, such as joint transmission and reception. Options with lower levels of centralization, such as MAC-PHY (option 6), still permit less sophisticated techniques, such as coordinated scheduling and/or beamforming. However, options with low centralization, like the PDCP-RLC split (option 2), only allow for slow and basic interference mitigation [MAK22].

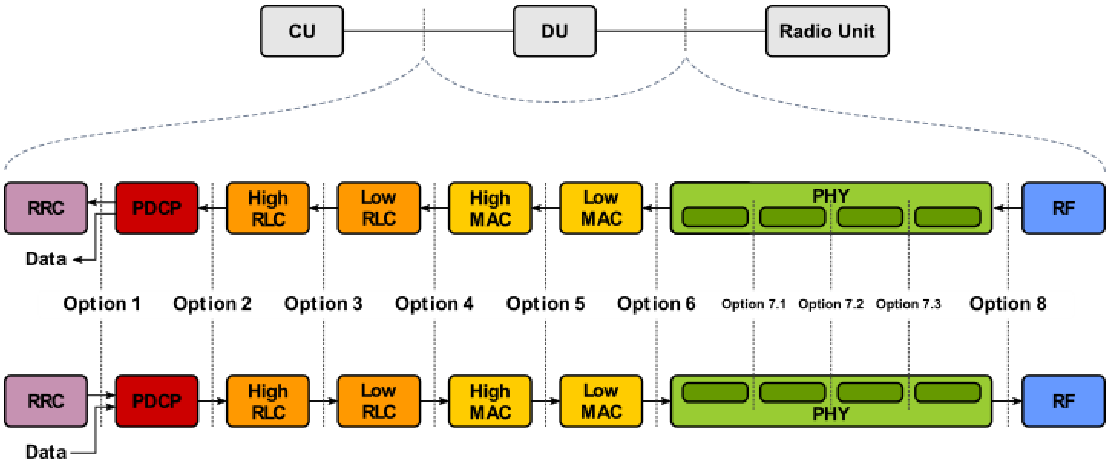


Figure 2.3: Possible Functional Split options along the protocol stack in a 5G RAN [MA22].

RRC-PDCP

The RRC-PDCP delineation facilitates a distinct bifurcation between the control plane and the data plane. Notably, user traffic circumvents the CU, thereby channeling packets directly from the SGW to the DU. In contrast, control traffic, predominantly characterized by NAS messages, is routed from multiple DUs to the central unit, enabling potential centralization of NAS components such as the HSS and MME.

PDCP-RLC

By centralizing the PDCP and RRC layers, a substantial enhancement in user mobility is achieved. The handover protocols are executed exclusively within the CU, except when the UE traverses between two DUs governed by disparate CUs. Such an arrangement expedites the execution of mobility management algorithms. Additionally, this configuration fosters the integration of multiple heterogeneous networks, given the superior compatibility of the PDCP layer with divergent technologies as compared to other radio layers [MZB+18].

It's also noteworthy that tasks like payload ciphering and robust header compression, undertaken by the PDCP layer, are more efficiently executed in a cloud environment, primarily due to their computational demands.

RLC-MAC

Virtualizing the RLC layer yields enhanced resource utilization for storing RLC PDUs. During the operation of the DL RLC layer in the acknowledged mode (AM), RLC PDUs are buffered pending acknowledgment, enabling the potential optimization and scaling of the storage buffer predicated on operational mean values [For16].

The intricate scheduling tasks performed at the gNB necessitate a recurrent interaction, on a millisecond basis, between the RLC and MAC layers for the exchange of buffer occupancy status and RLC PDUs. Such temporal requisites may lead to potential protocol and packetization overheads. Considering similar challenges manifest in deeper layers of the LTE stack, the precedence for centralizing the MAC layer is evident, owing to the manifold benefits it proffers.

MAC-PHY

While the centralization of the PDCP layer inherently facilitates coordination mechanisms to augment user mobility, the MAC layer's centralization can arguably be recognized as the quintessential configuration that emboldens coordination strategies to enhance data traffic quality.

This configuration situates the MAC layer of multiple gNBs within a singular physical locale, fostering swift and dependable communication amongst MAC layers. This, in turn, aids the orchestrated scheduling tailored for inter-cell interference mitigation. However, an intrinsic challenge of this delineation is the mandatory operation of the HARQ processing within the centralized unit, which is concomitant with the HARQ's stringent temporal constraints. To elucidate, post the reception of a subframe by the gNB (DU), the MAC layer (CU) is bound by a 4 ms deadline to execute HARQ-specific protocols. A more exhaustive discussion on HARQ timing analysis is forthcoming in this section.

The adoption and practicality of this demarcation have garnered substantial interest within the industry, motivating standardization initiatives culminating in the introduction of FAPI [For17].

Although this configuration produces the fronthaul traffic bandwidth to variate in proportion to the user traffic, a non negligible overhead appears as result of the isolation of the PHY layer. The physical layer needs both user information and control information for the codification of physical channels. User information is provided as MAC PDUs and it is a result of the user traffic, whereas control information is provided as DCI and it is the result of the MAC scheduling decisions (number and distribution of physical resource blocks PRB, modulation and coding scheme MCS etc).

It is essential to delve into the unique realm of industrial mobile networks, recognizing the challenges and increasing demands of industries.

2.3 Industrial Mobile Network

Industrial environments, characterized by their demand for precision, reliability, and real-time communications, pose a unique set of challenges and requirements for mobile networks.

Unlike conventional RAN, Industrial Mobile Networks prioritize ultra-reliability and low-latency, especially vital for applications like automated vehicles, robotics, and real-time monitoring. Given these stringent requirements, the traditional mobile network designs often fall short, paving the way for specialized architectures and solutions tailored for industrial use cases.

One of the cornerstones of industrial mobile networks is the need for deterministic communication. In this type of environment, data packets need to be delivered not only in a timely manner, but also with a guaranteed timeline to ensure the synchronicity and accuracy of operations. These networks must also be robust enough to operate in harsh environments, which can include electromagnetic interference, physical obstructions, or even extreme temperatures.

5G technology, with its promise of reliable, high-speed communication, is a leading contender for meeting strict industrial requirements. Its integration of numerous sensors, machines, and control systems allows for seamless real-time data exchange and analytics. This interplay drives predictive maintenance, optimized resource usage, and improved production quality.

The relationship between 5G RAN technology and AGVs lies in the role that high-speed, low-latency connectivity plays in efficiently automating AGV operations. AGVs require a fast and reliable connection to communicate among themselves and with centralized control systems [LA23]. 5G RAN delivers data speeds significantly faster than previous generations, enabling real-time communication and faster decision-making for AGVs. As such, low latency is pivotal to ensure operations are carried out safely and efficiently, especially in settings where AGVs interact with humans and other equipment.

On the other hand, industrial and logistics environments typically feature a multitude of connected devices beyond just AGVs. It is imperative that this increased density of connected devices can be managed to ensure AGVs can operate seamlessly. In settings where multiple AGVs work collaboratively, coordination is vital. 5G connectivity allows for continuous communication between vehicles, facilitating route planning, collision avoidance, and overall operational optimization. Simultaneously, 5G connectivity also supports real-time data collection and analysis from AGVs, which can be employed for predictive maintenance [TPY20]. Onboard sensors on the vehicles can transmit data about performance and the network's state, leading to improved maintenance and better resource management for the network.

Within this specialized industrial landscape, the path taken by automated vehicles plays a crucial role in network efficiency and reliability. This seamlessly leads us to explore path optimization algorithms.

2.4 Path Optimization Models

Automated vehicles, with their inherent need for continuous and reliable connectivity, necessitate sophisticated path optimization algorithms. These algorithms determine the most efficient route for a vehicle, ensuring uninterrupted communication while minimizing potential interference and maximizing resource utilization.

Path optimization models constitute a pivotal domain within the realms of computer science, transportation engineering, and operations research. These models are designed to uncover the most efficient routes or paths between two points in a given network, be it a road network, communication network, or any interconnected system. Their primary aim is to minimize a certain cost or objective function while adhering to specific constraints.

The fundamental function is to determine the optimal route that entails the least cost or offers the highest utility. In the context of transportation, these models facilitate efficient navigation, ensuring that travel distances, time, or costs are minimized. Beyond navigation, path optimization finds application in various domains like logistics, robotics, and even data routing in computer networks. Its importance lies in enhancing resource utilization, reducing operational costs, and improving overall efficiency. Two prominent algorithms in the field of path optimization are Dijkstra's algorithm and the A* algorithm.

Dijkstra's algorithm is a method designed to identify the shortest route between nodes in a graph when given non-negative edge weights. It works by systematically choosing the node with the minimum distance and updating its adjacent nodes based on a principle of greediness. Despite its ability to ensure the shortest path, Dijkstra's algorithm efficiency declines when confronted with negative edge weights [GMJU16].

The A* algorithm combines Dijkstra's methodology with a heuristic approach, making it highly effective for graph-based pathfinding. By using a heuristic that estimates the remaining distance to the goal, A* can prioritize which nodes to examine. This results in a balance between accuracy and computational efficiency, making it ideal for situations requiring both speed and precision [LHW14].

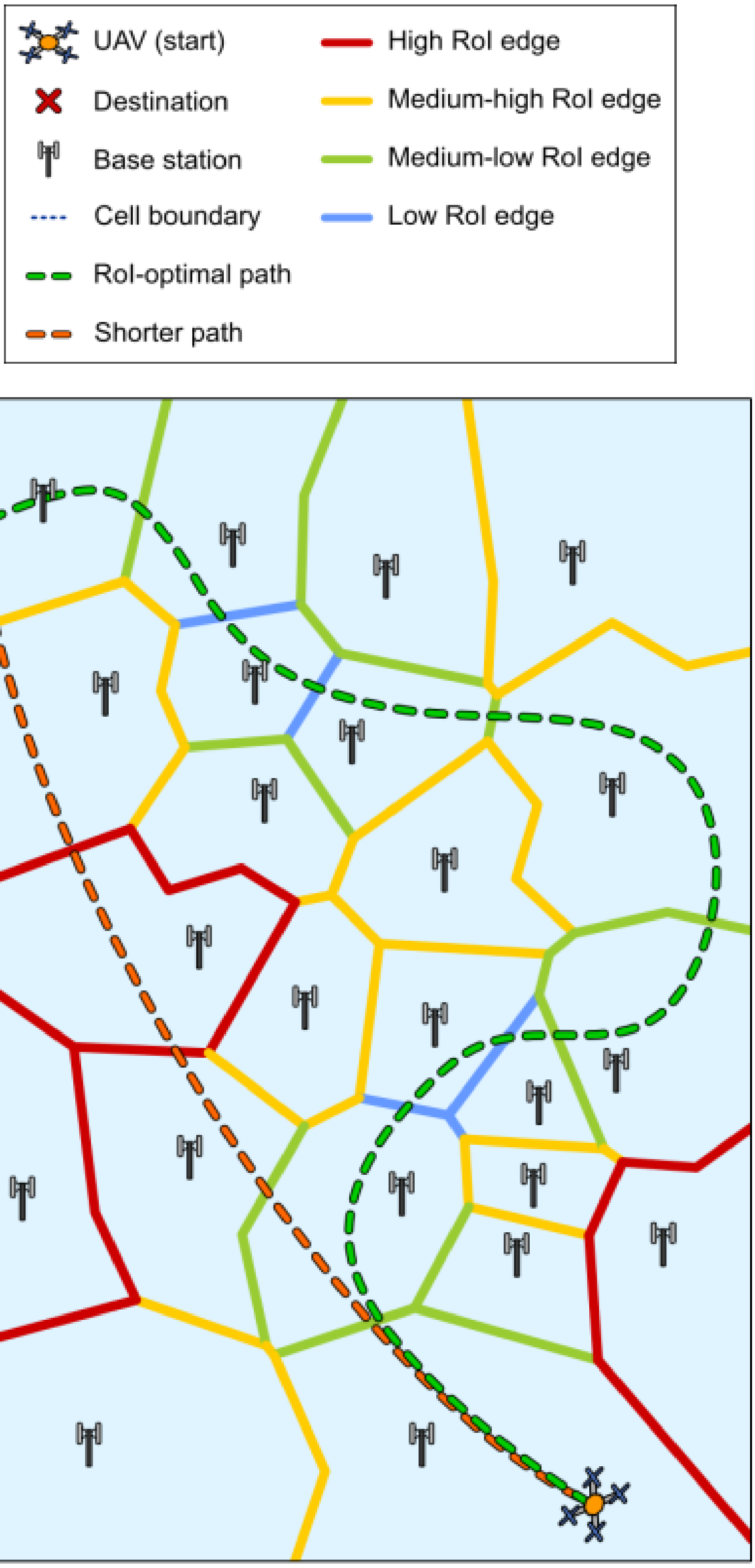


Figure 2.4: RoI-optimal path calculated compared to a possible shorter path.

Within the domain of graph theory and optimization, Dijkstra's and the A* algorithm are leading strategies for determining the shortest route between nodes in a graph. Each algorithm serves specific purposes, addressing diverse scenarios and concerns. Here, we will explore the fundamentals of both algorithms, explicate their applications, and compare their nuances.

Some of the more advanced methods for addressing these and other related challenges involve mathematical optimization techniques, in particular MILP and MIQP.

2.5 Mixed-integer Linear/Quadratic Problem

Mixed Integer Linear Programming (MILP) and Mixed Integer Quadratic Programming (MIQP) are valuable optimization tools. These methodologies enable the modeling of complex decision-making problems where certain variables are limited to integer values. In particular, in the field of network design and path optimization, MILP and MIQP provide a strong framework to obtain optimal or nearly optimal solutions, while considering various constraints and objectives.

Mixed Integer Linear Programming is a form of linear programming that places constraints on some or all variables to only take integer values. MILP is crucial in scenarios where solutions require distinct decisions; for instance, in supply chain management, production scheduling, or facility location issues. While linear programming can be solved efficiently using various algorithms, the introduction of integer constraints in MILP makes it NP-hard, resulting in increased complexity and longer solution times [Wol07].

On the other hand, Mixed Integer Quadratic Programming is an extension of MILP that allows for a quadratic objective function. This enables a wider range of applications, accommodating problems with nonlinear relationships between variables while still being representable as quadratic expressions. MIQP can address complex optimization landscapes, offering advanced modeling capabilities despite increased computational complexity [BV04].

Its relevance in modeling multiple aspects of RAN, such as resource allocation and path selection for automated vehicles, highlights its significance in this thesis.

2.5.1 Gurobi

Gurobi Optimization emerges as a beacon in this intricate landscape of optimization. Renowned for its efficiency, Gurobi is an optimization solver that can handle a wide variety of programming problems, including MILP and MIQP. Its robustness, paired with cutting-edge algorithms, positions it as a leading tool for professionals and researchers alike seeking optimal solutions to complex problems [Gur23].

To understand Gurobi's relevance, it is crucial to briefly introduce the concept of optimization. Mathematical optimization is concerned with finding the most suitable value (maximum or minimum) of a function called the *objective function*, subject to certain constraints.

The Gurobi Optimizer excels in addressing large-scale and complex problems through advanced algorithms, consistently outperforming many market solvers in solution timeframes. While it adeptly evaluates constraints to determine solution bounds, its near 100% success rate and ability to integrate seamlessly with languages, especially Python, have cemented its reputation as a leading tool in modern optimization techniques.

2.5.2 Glover's Constraints: Zero-One Polynomial Problem

The *Zero-One Polynomial Problem* is a significant challenge in computational complexity. The task is to determine if a multivariate polynomial with integer coefficients can evaluate to zero through a binary assignment, where variables are set to 0 or 1. Despite the apparent simplicity of the problem, the computational intensity escalates as the number of variables and the complexity of the polynomial increase.

Despite the apparent simplicity of the problem, the computational intensity escalates as the number of variables and the complexity of the polynomial increase. This problem is notable for its NP-hard status. In terms of computational complexity, problems classified as NP are those whose solutions can be verified in polynomial time. This problem is categorized as NP-hard, indicating its complexity in terms of computation. NP problems are those for which solutions can be verified in polynomial time.

Though theoretical in nature, this problem serves as a benchmark for algorithm designers and computer theorists. Although techniques such as exact algorithms and branch-and-bound methods exist, their effectiveness decreases as the problem size increases.

The restrictions necessary for obtaining accurate values are as follows:

$$\sum_{j \in Q} x_j - x_Q \leq q - 1, \tag{2.1a}$$

$$\sum_{j \in Q} x_j + q \cdot x_Q \leq 0, \tag{2.1b}$$

subject to

$$x_Q \in \{0, 1\} \tag{2.1c}$$

where q denotes the number of elements in Q [GW74].

As we delve deeper into the intertwined complexities of Functional Split adaptation and path selection in the subsequent chapters, this foundational understanding of mobile RAN, interference mitigation techniques, industrial network requirements, path optimization strategies, and the mathematical rigor of MILP and MIQP will serve as our guiding beacon.

Chapter 3

Theory

3.1 Problem Statement

The contribution of this thesis is twofold. First, we define a new RAN entity which interfaces both BSs and AGVs so as to compile the required information to optimize the paths of the moving vehicles and the functional split configuration of the whole RAN. Second, we describe a strategy to calculate both vehicle paths and functional splits and to forward the former to the AGVs, while meeting connectivity and network capacity constraints. In the following, we describe the considered scenario and the detailed procedure of the thesis.

We consider an industrial indoor scenario, where AGVs are connected via a mobile radio access network to an industrial core network, which in turn is connected to the rest of networking equipment and/or the internet. The RAN consists of a set of base stations that transmit to and receive from AGVs and other user equipment. Following the architecture of the 5G RAN, each base station (BS) is divided into a centralized unit, a distributed unit, and a radio or remote unit (RU). The RU hosts the radio equipment (including antennas). The DU is deployed near the RU, whereas the CU is deployed in a central location. Each DU has the capacity to run a large portion or the whole functionality of its corresponding base station (except for the radio processing associated to the RU). If required, the CU can take over the functionality of one or more DUs. This is referred to as the functional split, which can present several possibilities, illustrated in Figure 2.3 from the previous chapter. When functions are centralized at the CU, and, consequently, run on the same machine, they can quickly communicate with each other, for instance to implement interference coordination or cancellation. A depiction of the considered RAN architecture is shown in Figure 3.1.

Although full function centralization at the CU of all base stations would be beneficial from a performance perspective, we assume that this is not feasible in our considered scenario. Either the CU lacks the computational capacity for taking over the processing of all base stations or the network connecting CU and DUs does not have enough link

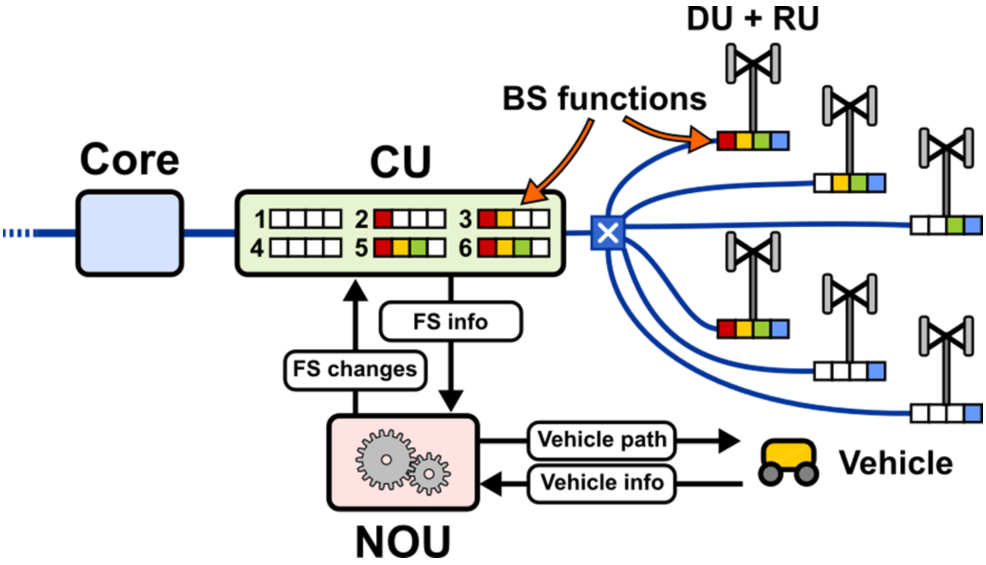


Figure 3.1: Architecture of the considered RAN, including centralized, distributed, and remote units, core, NOU, and vehicles.

capacity to support the high data rates needed for full centralization. However, partial function centralization, understood as the centralization of not all functions for all or some base stations, is indeed possible. Furthermore, it is possible to dynamically change the FS of each base station, by using pre-existing techniques to that effect. The AGVs in the network perform (semi-)autonomous tasks, which entails autonomously moving from their current position to a desired one, as demanded by the application layer. The AGVs require connectivity to navigate and perform their work, which is provided by the RAN. Owing to potential inter-cell interference at the cell edges, it is possible that AGVs experience bad or lost connectivity, which negatively impacts their current task.

We propose the definition of a Network Optimizing Unit (NOU), whose task is to compute both the paths to be followed by AGVs when they need to move and the sequence of optimal FS in the RAN such that the constraints of AGV connectivity, trip duration, and link capacities are met, and/or overall connectivity and operating cost is optimized. As a result, the NOU can instruct the base stations in the RAN (and thus the CU and DUs) to change their functional split. The NOU can be collocated with the CU or deployed independently. When following the O-RAN architecture, the NOU can be realized as a third-party application (rApp) of the Non-Real-Time RAN Intelligent Controller (Non-RT RIC) as part of the Service Management and Orchestration (SMO) framework. New interfaces may be needed to accommodate information exchange between BSs, AGVs, and NOU, or existing ones can be repurposed with new messages and information elements.

The following mathematical notation is used to formalize the description of the scenario and procedure of the thesis. We consider a network consisting of G base stations, and a vehicle moving around our scenario. The received signal power from base station $g \in \{1, \cdots, G\}$

at point p within the considered area is denoted as $r(g,p)$. The serving base station at point p is, in general, the one providing the greatest signal power at that point. We thus use function $h(p) := arg\max_g r(g,p)$ to refer to the index of the serving base station at point p. For any point p, signal power received by base station $h(p)$ is useful, whereas that received from other base stations is regarded as interference.

We model the functional split of base station g with the index $v_g \in \{1, \cdots, V\}$, where V is the total number of functional split options (also known as centralization levels). We assume that the values of c_g are sorted such that the effect of the interference coordination/cancellation capabilities between base stations featuring the same functional split increases with c_g. In case that base stations do not have the same functional split, they can only coordinate to the extent allowed by the base station with the lowest c_g. The vector of functional splits for all base stations is denoted as $c =< c_1, \cdots, c_G >$. Since the NOU has the capability to change c over time, we denote as $c(t)$ the vector of functional splits at time t.

The overall goal of this thesis is to provide an efficient means for calculating and communicating efficient paths for moving vehicles served by a mobile communication network, while guaranteeing connectivity constraints and optimize the overall network operation and resource consumption (via adapting the functional split). The procedure of this thesis is as follows:

1. An automatically guided vehicle receives a new command from its application layer that involves moving from its current location *origin* to a new location *destination*. During the trip, the AGV may need a minimum quality of connectivity (in terms of minimum throughput, maximum latency, etc.). To this end, the AGV needs to figure out an optimal path from *origin* to *destination* that avoids interference-prone areas whose traversal may result in degraded quality of connectivity. This is shown as step 1 in Figure 3.2.
2. The AGV sends a *PATH REQUEST* message to the NOU. In this message, the AGV provides origin and destination points (A,B) as well as additional information to assist path calculation, such as maximum linear and rotational speed, maximum duration of the trip, minimum required throughput during the trip (average or absolute), maximum allowed latency, etc. This is shown as step 2 in Figure 3.2. Alternatively, the *PATH REQUEST* message can be sent by a central fleet management unit on behalf of the AGV, which later internally forwards the received path to the corresponding AGV. Notwithstanding this possibility, in the remaining of this procedure we assume that the AGV interacts directly with the NOU.
3. Upon receiving the *PATH REQUEST* message from the AGV, the NOU retrieves the current functional split vector c, containing the configuration of all base stations in the RAN, and the current position, activity, and requirements of all AGVs in the network. This is shown as step 3 in Figure 3.2.
4. From this information, the NOU computes the most adequate functional split se-

quence $c*(t)$ and the associated optimal vector of trip functions $x*(t)$ for all vehicles. For instance, $c*(t)$ and $x*(t)$ can be obtained by solving the following optimization problem. This is shown as step 4 in Figure 3.2.

5. The optimal trip function found by the NOU is discretized into a sequence of checkpoints $\mathbb{X}_v$. This is shown as step 5 in Figure 3.2.
6. The NOU builds and sends a *PATH RESPONSE* message for vehicle indicating the sequence of checkpoints $\mathbb{X}_v$ and any other relevant information. This is shown as step 6 in Figure 3.2.
7. The NOU sends a *PATH UPDATE* message with content like PATH RESPONSE to those AGVs whose path and/or timestamps have changed because of the new path request. This is shown as step 7 in Figure 3.2
8. After receiving the new or updated paths, the AGVs follow instructions and move towards the checkpoints within the allowed time intervals. At the same time, the NOU instructs the base stations (CU and DUs) to change their functional split in accordance with the time-dependent vector of functional splits $c*(t)$ selected in step 4. This is shown as step 8 in Figure 3.2.
9. If an AGV realizes that it has not or will not be able to comply with the path and/or timestamps provided by the NOU, it sends a *PATH ERROR* message informing about the problem. The NOU will recalculate a new set of paths and timestamps for the given situation and send a *PATH UPDATE* message to the involved AGVs with the corresponding modifications. This is shown as step 9 in Figure 3.2.
10. The process repeats whenever an AGV is commanded to move by the application layer.

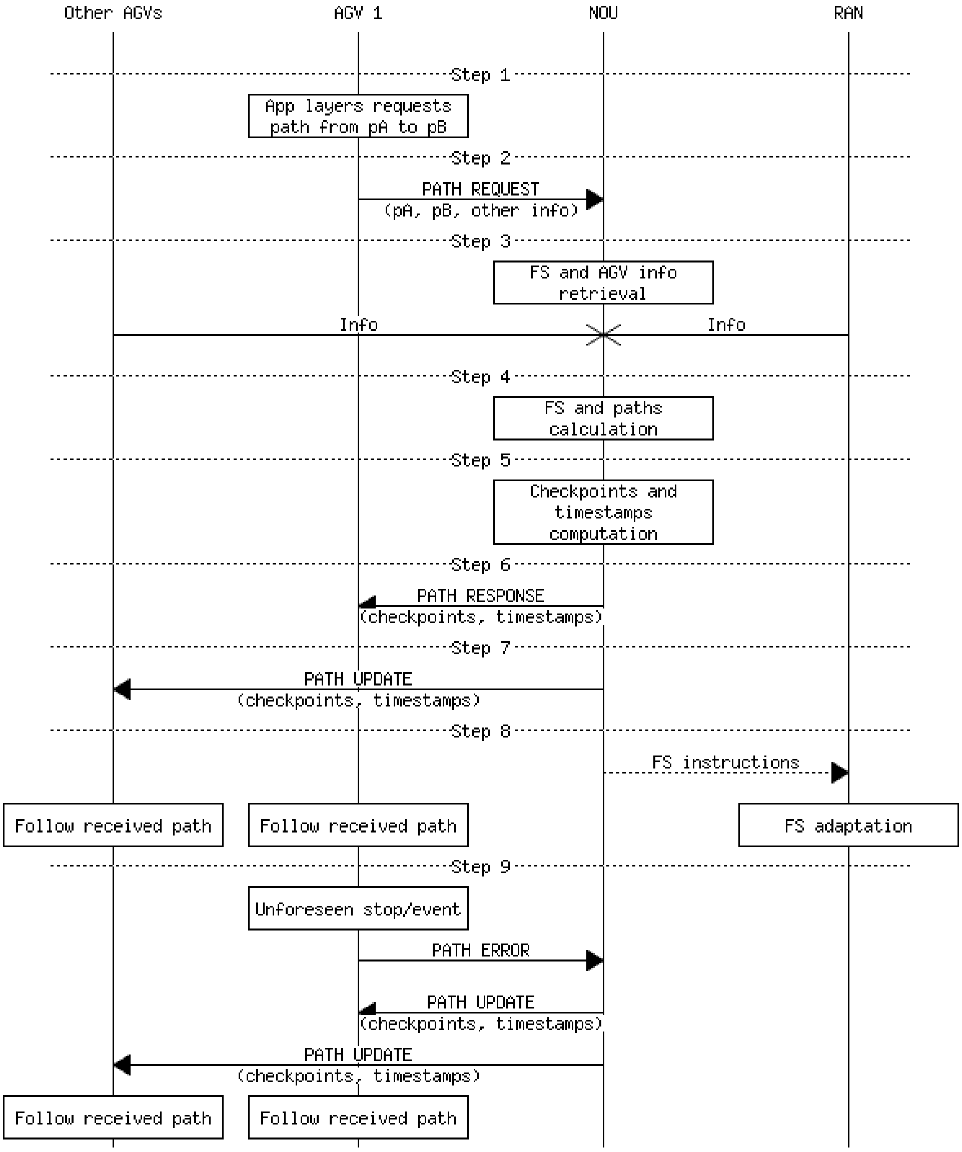
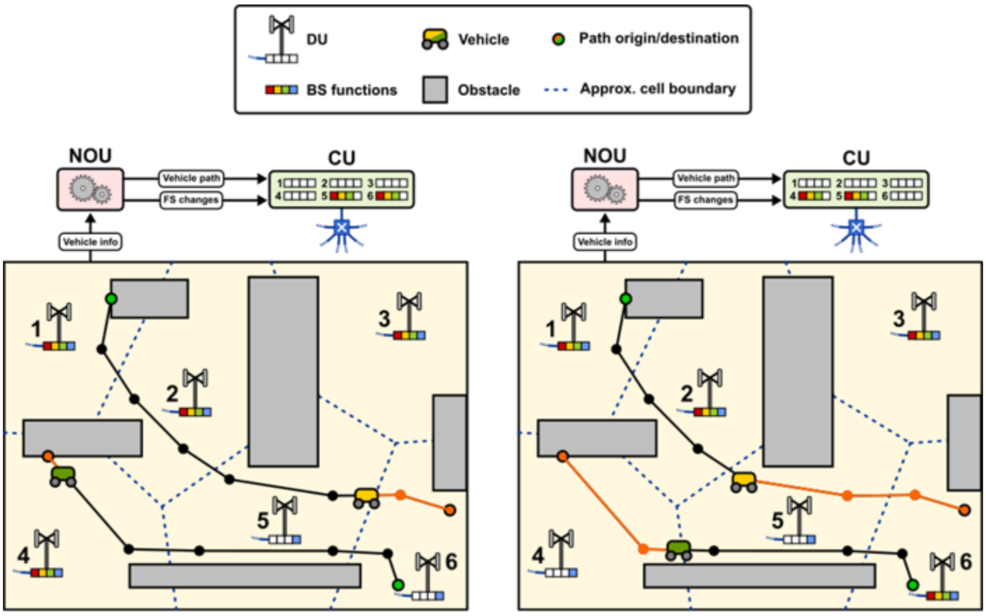
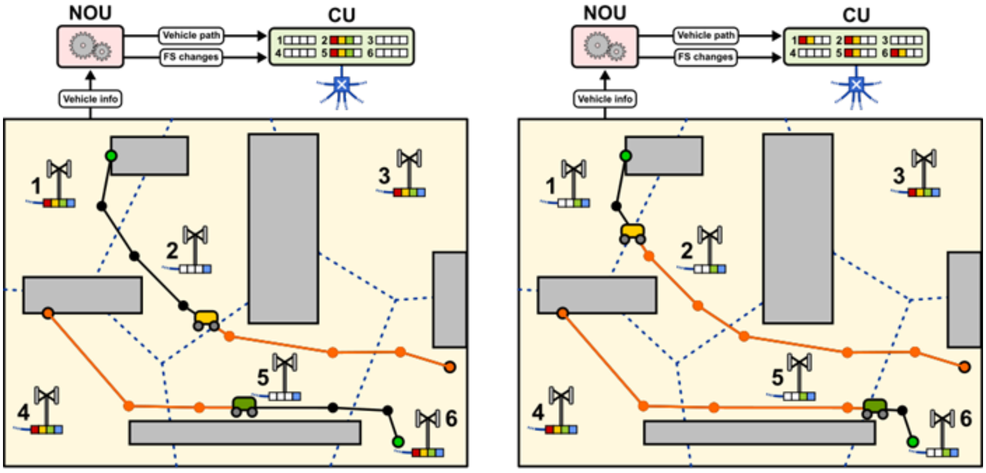


Figure 3.2: Summarized procedure and message sequence.

An example application of the described thesis is shown Figure 3.3. In subfigure a) (time $t = 0$), the yellow AGV is crossing the boundaries of cells 5 and 6, whose operation is centralized to minimize interference, whereas the green AGV has started to move after requesting and receiving a path from the NOU. In subfigure b), the green AGV is moving between its second and third checkpoint, which is associated with the centralization of functions of cells 4 and 5 (and the decentralization of 6, as it may be impossible to centralize more than two base stations due to network limitations). In the meantime, the yellow AGV waits at its third checkpoint, until the time to continue is reached. In subfigure c), the yellow AGV can continue its trip as cells 2 and 5 are now centralized. Finally, in subfigure d), both AGVs are crossing cell boundaries. For this event, the NOU decides to partially centralize cells 1, 2, 5, and 6 in order to limit the interference, since full centralization is not possible and making AGVs wait further may violate trip duration constraints.



a) Time $t = 0$ b) Time $t = t_1$



b) Time $t = t_2$ b) Time $t = t_3$

Figure 3.3: Example operation of the described problem.

In the following sections, we will delve deeper into the formulation of the problem we've just examined, presenting two distinct methodologies. The first approach offers a streamlined and simplified formulation, designed to facilitate understanding and pave the way for an intuitive implementation strategy. Its primary benefit is to provide a clear foundation, free from potential complications that could arise from a more detailed approach. The second methodology, conversely, is a more intricate and detailed formulation. While it incorporates a greater number of variables, making it more challenging to dissect, its strength lies in its closeness to real-world scenarios. Despite its evident complexity, this approach ensures a more robust and comprehensive solution, effectively capturing the nuances and intricacies of the practical situation at hand.

To the best of our knowledge, there is no state-of-the-art solution proposing the joint and mutually dependent optimization of functional split adaptation and vehicle path planning, as we do in this thesis report.

3.2 Initial Problem Simplified

The initial phase under scrutiny is the most straightforward in nature. This preliminary stage entails the alteration of resources within the 5G network through a two-step process. Initially, the primary objective is to ascertain the quickest trajectory from the starting point to the destination, with minimal regard for the prevailing condition of the generated network in the environment.

To accomplish this, as a first step in this process, the Dijkstra's Algorithm will be employed. This algorithm facilitates the computation of the swiftest route, guiding the AGV through the nodes strategically placed upon the map. Once the optimal path for the AGV to reach the designated destination expeditiously is established, the subsequent undertaking revolves around the resource optimization within the network. This optimization aims to mitigate the potential for interference when the AGV traverses the cell boundaries. In essence, this entails the coordination of resources among pairs of base stations.

Once we know which cells we have to cross thanks to the paths calculated by Dijkstra, for each node, the cost value that must be accounted for at the boundary is the maximum value v_g associated with the constituent base stations. To illustrate, if a gNB is associated with three functions while the other gNB is linked to only one, the RoI value will be dictated by the latter gNB, which acts as the limiting factor. Consequently, the resultant RoI value will be 3, representing the worst-case scenario. As a result, for a lower RoI value, the gNBs within cells that share an edge must be synchronized. In essence, the Centralized Unit must possess equivalent functions associated with those base stations. The approach to resolving this quandary involves an optimization problem that necessitates a well-structured formulation to enable Gurobi to expediently and effectively address it.

When addressing the puzzle of determining the optimal value, the value of an edge is identified as the highest v_g value among all its constituent stations. We can write this problem as the need to find a cost value c for a node such as

$$\mathbf{c} = \max(v_g, v_h)$$

Both v_g and v_h are members of $\mathbf{v}$, which represents the Cell Functional Split value vector. To ensure appropriate management in the central unit, the sum of base station resources cannot exceed its capacity.

However, Gurobi cannot effortlessly solve such problems that involve selecting between higher or lower values. Hence, it becomes imperative to devise a parameter transformation that converts this problem into a potentially linear or quadratic issue, which Gurobi is adept at resolving. The most feasible avenue to implement this involves a variable transformation, as illustrated in Table 3.1, wherein the v_g value is reconfigured as a sum of three w_{gi} values, each of which can assume the values 0 or 1. The result of this formulation is:

$$\max_{\mathbf{w}} \quad \mathbf{w}^T\mathbf{Q}\mathbf{w}, \tag{3.1a}$$

subject to

$$\begin{aligned}
&\mathbf{w} \in \{0,1\}^{3G} && \qquad (3.1\text{b})\\
&w_{gi} \geq w_{gj} \qquad\qquad \forall g, i < j, && \qquad (3.1\text{c})\\
&\sum_{g}\sum_{i=1}^{3} w_{gi} \leq M && \qquad (3.1\text{d})
\end{aligned}$$

where $\mathbf{w}$ denotes the vector of coefficients w_{gi} to be optimized and $\mathbf{Q}$ signifies the adjacency coefficients matrix and M denotes the number of functions in the network, as these functions are not unlimited. This is a parameter that can be adjusted according to on the network configuration.

This restructuring ensures that upon summation, the values of v_g are accurately represented, spanning the range from 0 to 3. This novel perspective allows v_g to be visualized as an array composed of three values, namely, w_{g0}, w_{g1}, and w_{g2}, effectively paving the way for optimization within Gurobi's framework.

The proposed restructuring ensures that, during the addition operation, the values of v_g are represented with utmost precision within the range of 0 to 3. This innovative perspective allows us to conceive v_g as a vector or array composed of three distinct values: w_{g0}, w_{g1}, and w_{g2}. The vector representation of these values not only offers an intuitive and clear visualization of the variables involved but also highlights the mathematical and computational interrelations between them. By breaking down v_g into its individual components, the analysis and diagnostic process is facilitated. Moreover, this vectorial representation not only provides a clear and detailed understanding of the variables at play but also sets a precedent for implementing more advanced optimization techniques. Specifically, this structure aligns perfectly with the capabilities and functionalities of the Gurobi framework, allowing for seamless integration and effective optimization of the problems being addressed.

Addressing the constraints of the problem is of paramount importance. Two specific restrictions demand close scrutiny. The primary constraint relates to the interrelationships among w_{g0}, w_{g1}, and w_{g2}. Specifically, w_{g0} should always be greater than or equal to w_{g1}, and the same holds true for w_{g2}, as can be seen in the constraint (3.1c). A more comprehensive understanding of this relationship can be observed in Table 3.1. These constraints are pivotal in guaranteeing that when the dot product between two v_g vectors is computed, the resultant value is precise and accurate.

v_g	w_{g0}	w_{g1}	w_{g2}
0	0	0	0
1	1	0	0
2	1	1	0
3	1	1	1

Table 3.1: Value Conversion for Functional Split

A thorough comprehension of these constraints allows for a more nuanced appreciation of the intricate dynamics that dictate the behavior of these vectors. The essence of these constraints hints at an underlying mathematical structure that seeks to maintain the integrity and consistency of the resultant computations. This also implies the possibility of more complex interactions between the vectors that might become evident as we delve deeper into the problem. The secondary constraint aligns with the one described above. Specifically, the cumulative sum of all $\mathbf{w}$ values should never surpass F. This mirrors the analogous constraint imposed upon v_g since the summation of v_g inherently comprises the set $\mathbf{w}$.

With a comprehensive understanding of the newly introduced variables and their associated constraints, the reformulated problem is primarily centered on the optimization of the $\mathbf{w}$ values. The core objective is to maximize the result of the vector product that involves the transposed $\mathbf{w}$ vector, $\mathbf{Q}$, and $\mathbf{w}$ itself. This setup not only resonates with the guiding principles inherent in the problem's constraints but also establishes a solid foundation for future optimization efforts. Moreover, by focusing on maximizing the outcome of the vector product, we are essentially homing in on the crux of the optimization problem. The interaction between $\mathbf{Q}$ and $\mathbf{w}$ serves as a crucial component in this optimization landscape. By understanding the nuances and intricacies of this relationship, one can better navigate the potential challenges and pitfalls that might emerge during the optimization process. This foresight, combined with a clear objective, provides a strategic advantage in the search for optimal solutions. Furthermore, this configuration not only aligns perfectly with the principles governing the problem's constraints but also paves the way for subsequent optimization pursuits, particularly within the ambit of the maximization problem at hand. This two-tiered optimization technique serves as a pivotal foundation for the subsequent phases of study, progressively delving into more complex network interactions and dynamics, thereby positioning the research on the trajectory towards achieving comprehensive and pragmatic solutions within the realm of industrial network orchestration.

3.3 Optimization algorithm formulation

To proceed with our research, we must unify the previous process, which is the next step we need to take. Our aim is to optimize both the Functional Split of the network and the route that the vehicle takes to its destination simultaneously. This dual optimization enhances operational efficiency and promotes a more harmonious and synchronized interaction between network scheduling and vehicular routing. The incorporation of these two crucial components has the potential to decrease operational expenses, expedite response times and create a more robust structure. By collectively addressing these aspects, we are prepared to deliver a comprehensive solution that meets the requirements of the network and the dynamics of vehicular traffic. The task at hand is to devise a framework or algorithm that can effectively manage both aspects with equal importance, without giving preference to one over the other.

Up to this point, the variables we're familiar with from the previous formulation are $\mathbf{v}$, where $\mathbf{v} =< v_1, \cdots, v_g >$, $|\mathbf{v}| = G$, and $\mathbf{w}$, with $\mathbf{w} =< w_{11}, \cdots, w_{g3} >$ and $|\mathbf{w}| = 3G$. The first variable is dependent on the second. Both are coefficient vectors that have been employed to optimize the FS of the 5G industrial network, where G is the number of gNBs, as discussed earlier in this document. However, to also compute and simultaneously optimize the AGV route, we must introduce new variables to the problem.

The decision to tackle this new challenge is based on conceptualizing it as a node flow problem. We will convert our network of nodes and edges into a bidirectional network. In this framework, the cost of each edge will signify the "effective cost" that the AGV must bear when transitioning from one cell to another, contingent upon the synchronization level of the cells. A more in-depth explanation of these costs will be provided in the ensuing sections of this chapter.

Understanding the complex interplay between the AGV's route and the synchronization of the cells is crucial. The synchronization dictates not only the efficiency of the AGV's path but also impacts the overall performance of the industrial network. By transforming our network into a bidirectional model, we can more accurately capture these intricate dynamics and design optimization strategies that are more aligned with real-world challenges.

Therefore, the new variable we are introducing will be termed $\mathbf{f}$, pertaining to the node flow. This vector will store binary values to determine whether a node is selected to be part of the route. Hence, we aim to define a function that considers the vehicle's path, corresponding to $\mathbf{f}$, and the FS vector $\mathbf{v}$. The objective with this function is to minimize the "effective cost" of each edge, which is associated with two base stations since, in a bidirectional diagram, they are directed edges reaching a node. This node represents the intersection between two base stations. Thus, the cost linked to an edge e, belonging to the entire set of edges (where $e \in \mathbb{E} \Rightarrow (g, h) \in \mathbb{G}^2$), given an FS, is equal to a function that provides the cost of the minimum of v_g and v_h. Consequently, the desired optimization function is:

$$\min_{\mathbf{v},\mathbf{f}} \quad \Gamma(\mathbf{v},\mathbf{f}), \tag{3.2a}$$

subject to

$$\mathbf{v} \in \{0,\cdots,3\}^{\mathbb{G}} \tag{3.2b}$$

$$\mathbf{f} \in \{0,1\}^{\mathbb{E}} \tag{3.2c}$$

$$\sum v_g = \mathbf{v}^T\vec{1} \leq M \tag{3.2d}$$

$$\sum_{e\in\mathbb{E}^+(n)} f_e - \sum_{e\in\mathbb{E}^-(n)} f_e = 0 \qquad \forall n \in \mathbb{N} \quad | \quad \{n_o, n_d\}, \tag{3.2e}$$

$$\sum_{e\in\mathbb{E}^+(n)} f_e - \sum_{e\in\mathbb{E}^-(n)} f_e = F \qquad \textit{for} \quad n_o, \tag{3.2f}$$

$$\sum_{e\in\mathbb{E}^+(n)} f_e - \sum_{e\in\mathbb{E}^-(n)} f_e = -F \qquad \textit{for} \quad n_d, \tag{3.2g}$$

From our observations, equation (3.2a) inherits the restriction (3.2d) from the previous problem (3.1a) (3.1d). Likewise, it's essential to highlight the restrictions concerning the flows. The restrictions (3.2f) and (3.2g) for the origin nodes (n_o) and destination nodes (n_d) specify that the outgoing and incoming flows should have values F and $-F$, respectively. In our scenario, $F = 1$. For the intermediate nodes, the constraint is that the incoming flow should equal the outgoing flow (3.2e) for the entire set of nodes, excluding the origin and destination [AMO93].

With this information at hand, we can rewrite (3.2a) as

$$\gamma(e,\mathbf{v}) = \Delta(\min(v_{g(e)}, v_{h(e)}))$$

This function, given an edge e, retrieves the indices of the base stations g and h linked to that edge, checks the FS by taking the minimum of the two, and directly converts that FS into a cost. This is the function we've termed Δ. For the function Δ, given a value v_g, it calculates the cost of that gNB as

$$\Delta(v) = c_0 + c_1\delta[v \leq 1] + c_2\delta[v \leq 2] + c_3\delta[v \leq 3]$$

The values c_i correspond to a vector $\mathbf{c} =< 1, -0.4, -0.4, -0.19 >$ used to compute the effective cost assigned to different FS values. As in the previous case, we must make a conversion to the vector $\mathbf{w}$ to streamline the optimization process. Therefore, the new function is written as

$$\hat{\Delta}(v) = c_0 + c_1 w_{g(e),1} w_{h(e),1} + c_2 w_{g(e),2} w_{h(e),2} + c_3 w_{g(e),3} w_{h(e),3}$$

and, in turn,

$$\Gamma(\mathbf{v},\mathbf{f}) = \sum_{e\in\mathbb{E}(n)} f_e \cdot \Delta(\min(v_{g(e)}, v_{h(e)}))$$

With all these functions defined, we ultimately obtain:

$$\min_{\mathbf{w},\mathbf{f}} \quad \sum_{e\in\mathbb{E}(n)} f_e \left(c_0 + \sum_{i=1}^{3} c_i w_{g(e),i} w_{h(e),i} \right), \tag{3.3a}$$

subject to

$$\mathbf{w} \in \{0,1\}^{3G} \tag{3.3b}$$

$$w_{gi} \geq w_{gj} \qquad \forall g, i < j, \tag{3.3c}$$

$$\sum_{g} \sum_{i=1}^{3} w_{gi} \leq M \tag{3.3d}$$

In addition to being subject to these constraints, this problem also adheres to constraints from (3.2c) to (3.2g). This problem constitutes a cubic optimization challenge due to the involvement of three variables (f_E, w_g and w_h). Given that Gurobi can't address this problem directly, alternative methodologies must be sought to reformulate the problem in a manner that harnesses the capabilities of the Gurobi software.

To reformulate and revert to a quadratic problem, we need to replace the product of the variables w_g and w_h with a single variable, also binary, which we will denote as z. This new variable will then be multiplied by the f variable of that edge. To ensure that the z value resulting from the multiplication is desired, specific constraints must be applied, such as Glover constraints. This allows the correct solution to be derived from a multiplication between 0 and 1 (table 3.2). The result of this reformulation is:

$$\min_{\mathbf{z},\mathbf{f}} \quad \sum_{e\in\mathbb{E}(n)} f_e \left(c_0 + \sum_{i=1}^{3} c_i z_{g(e),h(e),i} \right), \tag{3.4a}$$

subject to

$$z_{g,h,i} \geq w_{gi} + w_{hi} - 1 \qquad \forall g \in \mathbb{G}, g < h, i \in \{1,2,3\}, \tag{3.4b}$$

$$z_{g,h,i} \leq \frac{w_{gi} + w_{hi}}{2} \qquad \forall g \in \mathbb{G}, g < h, i \in \{1,2,3\}, \tag{3.4c}$$

$$z_{g,h,i} \in \{0,1\}^{\frac{3G(G-1)}{2}} \tag{3.4d}$$

Where the constraints (3.4b) and (3.4c) belong to Glover's formulation [GW74] and also apply the constraints from the previous formulations, from (3.2c) to (3.3c), both included. With this equation, the problem posed in this thesis would be formulated. Additionally, the above problem can be written in a canonical form understandable to the Gurobi optimization software. In this regard, we must concatenate the three vectors $\mathbf{w}, \mathbf{z}$ and $\mathbf{f}$ into a single vector, which we will denote as $\mathbf{x}$. This vector will include all the variables to be optimized in one line, with a length X corresponding to the sum of the lengths of the other vectors. Finally, this problem is written as:

$$\min_{\mathbf{x}} \quad \mathbf{x^T Q x} + \mathbf{x^T q}, \tag{3.5a}$$

subject to

$$x \in \{0, 1\}^X \tag{3.5b}$$

$$\tag{3.5c}$$

Where $\mathbf{Q}$ is the matrix containing the necessary coefficients for the correct implementation of this problem, $\mathbf{q}$ is the vector for linear terms, and $X = 3G + \frac{3G(G-1)}{2} + E$. In this way, Gurobi will analyze and correctly solve the problem we want to optimize.

w_{gi}	w_{hi}	(3.4b)	(3.4c)	z_i
0	0	≥ -1	≤ 0	0
0	1	≥ 0	≤ 0.5	0
1	0	≥ 0	≤ 0.5	0
1	1	≥ 1	≤ 1	1

Table 3.2: Value Conversion from $\mathbf{w}$ to $\mathbf{z}$

Closing this chapter, these mathematical formulations offer a robust framework for optimizing the node flow while considering the intricate nuances of the network's FS. By integrating these functions into the optimization process, we gain a deeper understanding of how the FS and AGV route interact. This synergy provides a holistic perspective, enabling us to pinpoint potential inefficiencies and address them proactively. Through this iterative and integrative process, the solution's robustness and efficiency are continually refined, paving the way for a comprehensive optimization strategy that addresses both the network and vehicular requirements.

Chapter 4

Implementation and Evaluation

Having laid the theoretical foundations of our study, this chapter is entirely devoted to the tangible essence of our work: the implementation and the outcomes. Throughout the subsequent sections, readers will grasp not just the ”what” and the ”why”, but also the ”how” of our project. As we progress, the results obtained will be unveiled, serving as a pivotal component to ascertain the effectiveness and efficiency of the proposed solutions.

4.1 Implementation

In formulating the problem to be addressed by this Master’s Thesis, it was determined that the development of a scenario simulator was essential in order to achieve a simple solution that conserves resources while retaining the potential for wider societal application. The simulator serves as a platform for carrying out the necessary tests and analyses without increase the production cost.

Primarily, an investigation was undertaken to evaluate the feasibility of employing two distinct programming languages for conducting simulations: Matlab and Python. The conclusive decision leaned towards Python as the preferred language. In contrast to Matlab, Python offers an extensive array of libraries that prove invaluable in the construction of the simulator, further bolstered by its consistent and recurrent updates. A pivotal consideration was the aspect of licensing. Python, uniquely, does not necessitate licensing, which not only enables Siemens and other prospective stakeholders to employ the final iteration of the simulator without the need for a Matlab license, but also permits customization of the code to cater to diverse exigencies.

Once the programming language had been decided upon, an extensive exploration of the various options for the design of the simulator took place, involving a careful study of the numerous packages encapsulated within the Python libraries. The tool that was found to be remarkably suited to the aim of emulating the authentic characteristics of an industrial network was the use of Voronoi diagrams to create network cells [AK00]. The Voronoi diagram, often referred to as a Voronoi tessellation, manifests itself as a geometric construct that intricately divides spatial regions into Voronoi cells based on their proximity to a set of generating points, which act as surrogates for the distributed base stations on the map. Each individual cell surrounds its nearest generating point and includes all points that are closer to that particular point than any other point in the set. This architectural approach embodies the complex interplay between geographical distribution and network efficiency, and provides a sound basis for subsequent simulation efforts.

Of course, Python provides powerful tools for visualizing spatial relationships, such as the *Voronoi* function and *voronoi_plot_2d* from the *science.spatial* library [VGO+20]. These functions allow the creation of a cellular diagram where each cell represents the area closer to its corresponding base station than to any other. For our project, we chose a stochastic approach after evaluating several methods. We randomly generated the locations of these base stations using the *np.random.rand* function from the *numpy* library [HMvdW+20]. This approach provided both flexibility and wide coverage in our simulations.

Furthermore, the decision to randomly generate base station locations using *np.random.rand* underlines the flexibility and adaptability of this approach. Random base station placement allows a wide range of network scenarios to be explored, promoting a comprehensive understanding of system behavior under different conditions. This approach, when integrated into network planning and optimization processes, contributes to the development of robust and efficient wireless communication systems.

The next step involves the strategic placement of a series of reference points on the map. These points serve a dual purpose: they provide users with insight into the associated Risk of Interference that must be assumed when a vehicle crosses the boundary between cells. Distinguished by different colors, the Risk of Interference (RoI) assumes different magnitudes. In addition, these points are connected to each other, highlighting the multiple paths available to the AGV for navigation. It is important to note that the AGV is not required to pass exactly through these points; rather, this representation is used for ease of understanding and to provide a clear overview of the simulator's mechanics. In the problem formulation, these points are the nodes of the diagram. Figure 4.1 shows a scenario randomly generated by our simulator for a number of 10 gNBs, where the cells are reflected, the two boundaries are marked, the starting point is marked with a black square and the destination is marked with a red cross.

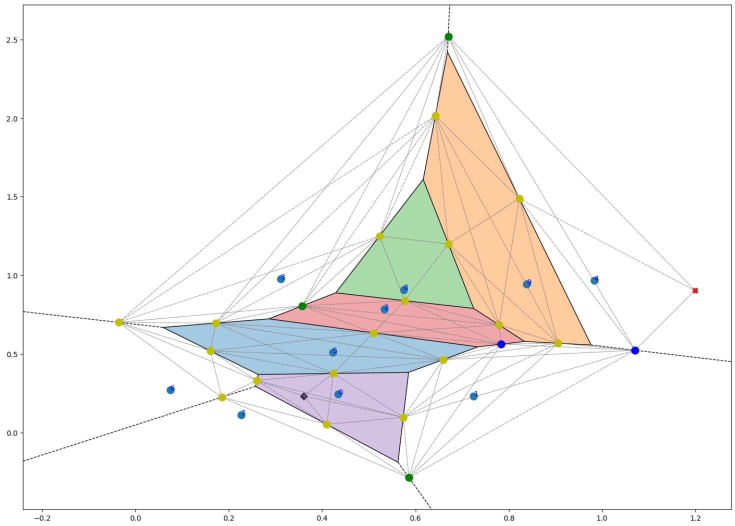


Figure 4.1: Scenario for 10 gNBs.

Once the scenario has been configured, the next step is to define the destination the AGV should reach from its starting position. This destination is symbolically marked by either a red cross or a black square. With these elements in place, we've created a comprehensive and adaptable framework, ready for detailed analysis. Armed with our data set, our focus shifts to processing this information to derive meaningful insights. This newly created scenario paves the way for us to tackle the problem we outlined in the previous chapter.

4.1.1 Simplified Problem

Certainly, here's the translation wiAs detailed in the theoretical section where this simplified problem is introduced and formulated, the solution is approached in two distinct stages. Initially, we apply the Dijkstra algorithm to identify the quickest route to the end point. Subsequently, we optimize the network resources.

With the start and end points defined, and having already represented the graph of our simulation environment, we employ the *nx.dijkstra_path* function from the *networkx* library to compute the set of nodes for the shortest path, termed *shortest_path_nodes*. This function returns an array containing the sequence of nodes that the vehicle must traverse to reach its destination in the most distance-efficient manner. To better visualize this route, a node diagram is generated where the path is highlighted in yellow, as depicted in figure 4.2. This image is an adaptation of 4.1, presented earlier.

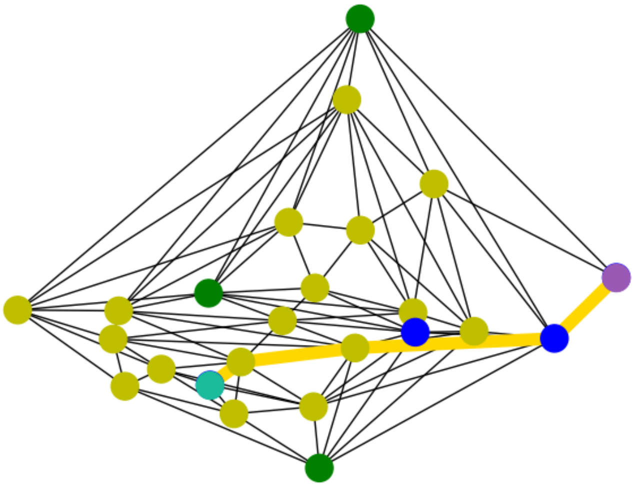


Figure 4.2: Shorter path using Dijkstra's Algorithm.

Once the path along which the AGV must move is known, using the newly acquired data, we convert to $\mathbf{w}$ and calculate the adjacency matrix $\mathbf{Q}$, which will be provided to Gurobi. To integrate Gurobi into Python, we must import *GRB* from the *gurobipy* library. In addition, we need to define the constraints (3.1c) and (3.1d) and initiate the optimization. After the process is complete, the previous diagram is represented again, replacing the initial values of the base stations with those optimized by Gurobi. This adjustment seeks to minimize the interference generated by the AGV as it crosses the boundaries between cells, as depicted in Figure 4.3.

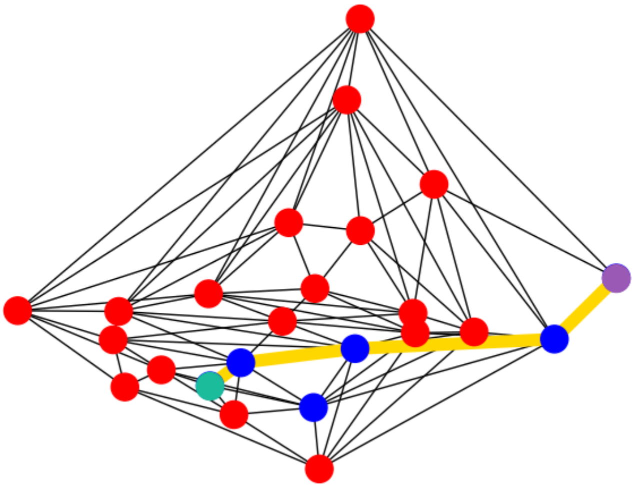


Figure 4.3: FS changes using Gurobi.

This method offers an in-depth visualization of the entire simulation process, from the initial scenario setup to the fine-tuning of the base station FS values, underlining the AGV's journey and its strategy for reducing interference. The optimization process facilitated by Gurobi ensures that the AGV operates in the most efficient manner possible within its environment. By focusing on the reduction of interference, the AGV not only preserves the integrity of the base station's communication but also ensures its smooth operation within the network.

4.1.2 Path and Functional Split Optimization Algorithm

Similar to the previous scenario, we begin with a representation of the environment we have constructed, either by randomly generating the positions of the gNBs or by inputting the coordinates of the desired base stations. From these positions, various cells are created, and the nodes through which the AGV must pass at the borders of the edges are defined using Voronoi diagrams. The starting and destination coordinates can also be manually set, allowing us to observe how, for a given scenario, the AGV leverages different possible paths based on varying parameters.

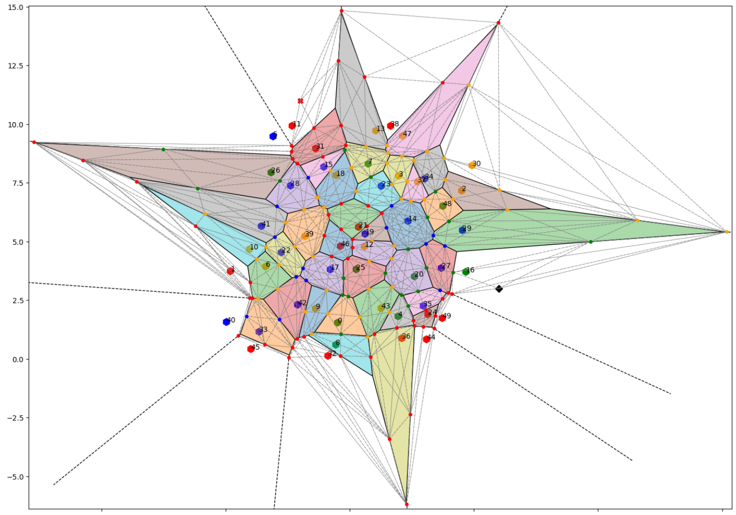


Figure 4.4: Scenario for 50 gNB.

In this process, as a starting point, we can disregard the distances separating the nodes from one another for both calculation and optimization purposes. This omission would not alter the outcome. In a more realistic simulation, once familiar with all the specifics of a real industrial network environment, we can incorporate the exact locations of the gNBs and the distances between the cells. This approach will be evident in the results presented throughout the thesis.

Once we have transformed the desired simulation scenario into the new bidirectional graph, it becomes evident that each incoming edge to the node has a distinct color, as depicted in Figure 4.5. This color-coding represents the efficient cost the AGV must assume for the network when choosing that particular node as its destination. Taking the terminal node as an example, it is noticeable that the edges leading to it are colored black, signifying a cost of zero. It is designed this way to ensure that whenever the vehicle has an opportunity, it travels to that node without incurring any additional cost. The rationale behind this is that the destination points are always located within a cell; hence, when the AGV travels to this point, it is already positioned within the cell, ensuring no interference. Table 4.1 delineates the relationship between the colors of the bidirectional edges and their associated Risk of Interference at that cell boundary. This systematic color coordination offers a clear perspective on route decision-making, ensuring the AGV's efficient navigation while minimizing network interference. The table serves as a quick reference, simplifying the complex interactions within the network and emphasizing the importance of strategic AGV movement within the cellular framework.

RoI	**Bidirectional Edge Cost**	**Color**
0	0.01	Blue
1	0.2	Green
2	0.6	Yellow
3	1	Red

Table 4.1: Conversion for RoI in bidirectional graph

With the updated costs of the bidirectional edges, the programmed code establishes the necessary constraints to allow Gurobi to optimize the problem unhindered, achieving both the optimal route to the destination and an ideal network resource allocation to minimize interference during the journey. The results of this simulation can be viewed in Figure 4.6b. This approach, combining refined edge costs with strategic optimization techniques, underscores the robustness of our model in navigating challenges and ensuring optimal AGV performance.

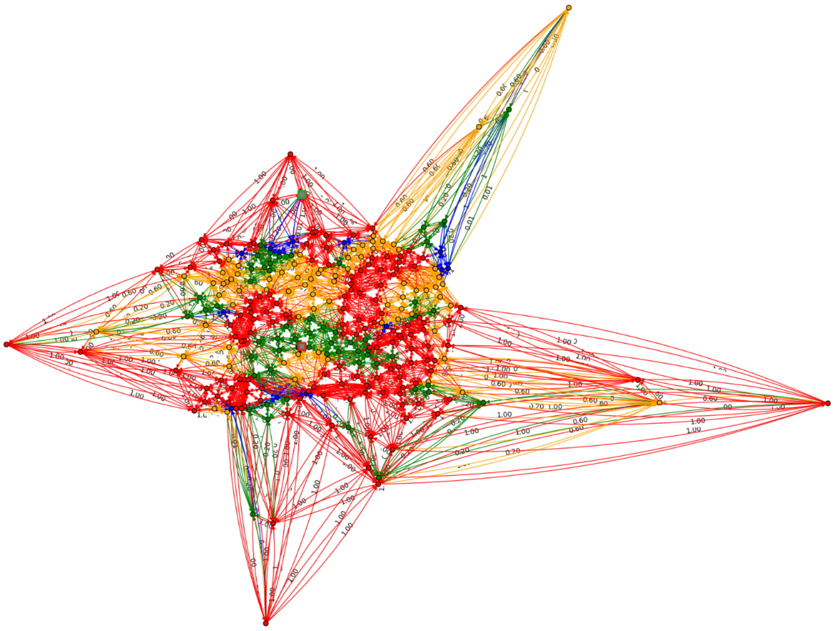
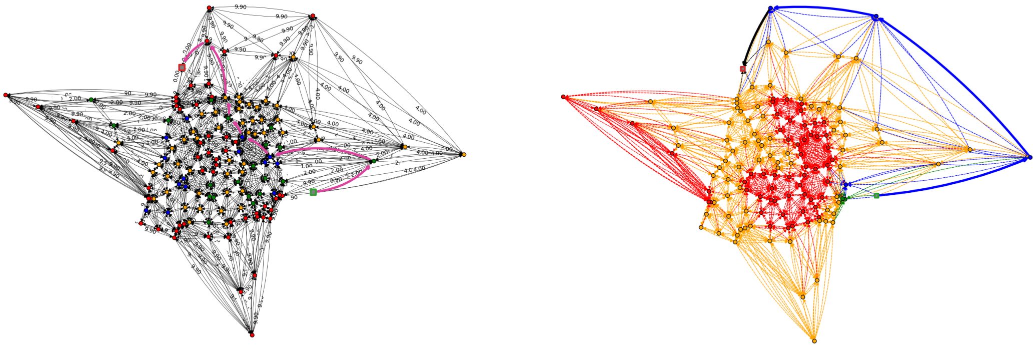


Figure 4.5: Bidirectional graph for scenario depicted in Figure 4.4.



(a) Path using Dijkstra's algorithm.

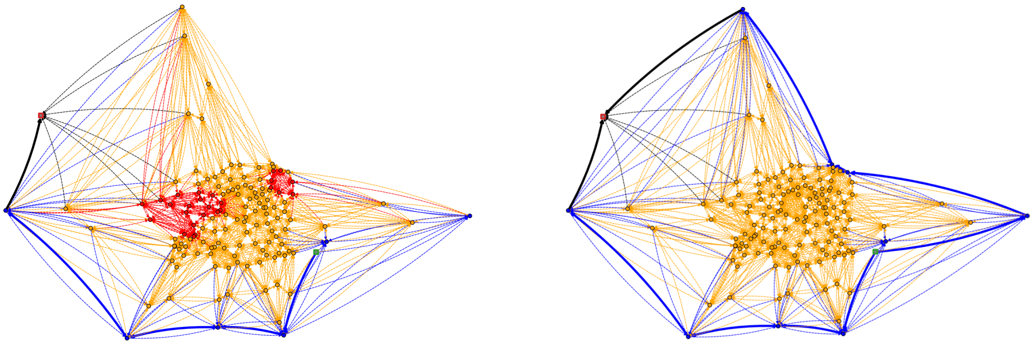
(b) Path using our algorithm.

Figure 4.6: Comparison of paths according to algorithm.

Upon comparison of the two latest images, it is apparent that the algorithm we have developed outlines a route that differs from the one calculated using Dijkstra's algorithm. After performing multiple simulations and subsequent comparisons, we deduce that our algorithm favors traversing through the smallest number of nodes. This methodology prioritizes crossing the least number of cells, which leads to reduced interference in the 5G network.

Alternative Paths to Destination

As an update to our algorithm, we can offer the AGV multiple alternative routes to its destination. This can be achieved by adjusting the flow parameters that leave and enter the source and destination nodes. For example, if we aim to provide two alternative routes to the destination, the flow leaving the initial node will be 2, and the flow entering the final node will be -2. The outcome for this instance can be viewed in figure 4.7.



(a) One-path network configuration. (b) Two-path network configuration.

Figure 4.7: Alternative paths from origin.

In the aforementioned figure, the two paths computed during optimization are evident. It's especially noteworthy to observe the distinct network resource management in contrast to when only one path is available. In this context, the need to coordinate more base stations results in the allocation of fewer resources per gNB. This demonstrates the intricacies involved in managing resources when multiple paths come into play, underscoring the significance of a comprehensive optimization strategy.

4.1.3 Many AGVs are Present in the Given Scenario.

As a final contribution to addressing this challenge, we will discuss the scenario encountered when there are two or more AGVs present in our setting. This proposal is the one that best represents the reality that we want to reach with this work.

Regardless of the number of users within the 5G network environment, both the number of w variables and z variables remain consistent with the previous case. Consequently, it's the flow array that ought to be adjusted, as the number of f variables will increase proportionally with the number of AGVs present in the scenario. This increment implies a rise in computational complexity to optimize the problem; however, as of now, this is the only solution we have identified to address this challenge.

If all AGVs were headed to the same destination, irrespective of their initial positions, the solution would be as straightforward as increasing the incoming flow to the destination node by the number of AGVs, represented as $-nAGV$, while always maintaining the outgoing flow from the initial nodes at 1. However, given that each robot will take a unique route, if we were to increase the flow for nodes or scale the flow for each start-end pair, we could encounter one of three potential scenarios, not all of which are necessarily correct.

- Each robot accurately determines its route to its destination.
- The robot with higher priority or a larger flow finds its correct route but might also mistakenly identify a route to a destination not intended for it. This mistake could leave a robot with ”lesser” priority or flow without its intended endpoint.
- Neither robot discerns the correct route to its designated endpoint.

After several weeks of research, code adjustments, and considering alternatives, this objective could not be attained. Every time modifications were made to the previous algorithm to incorporate the option of adding more vehicles to the scenario, unexpected errors arose, which seemed to have been previously resolved. Adding to this challenge was the significant computational cost associated with this rigorous calculation process. On occasions, this caused the code to become unresponsive, hindering our progress in the study.

As a side note, our intended study did not aim to increase the number of users beyond five. A high number of vehicles within the network would complicate resource optimization, potentially compromising the provision of optimal service to users. This highlights the intricate balance needed between efficient network usage and ensuring quality service delivery.

4.2 Results

To conclude this Master's Thesis, this section will present the results of the research conducted over the six-month study period. Various scenarios have been considered, where, by varying the number of base stations, we will examine the comparison between the two methods presented in this document. For these scenarios, we will generate random starting and ending points. After obtaining the optimized data, we will depict them in diagrams for comparative analysis and discussion. This approach ensures a comprehensive understanding of the system's performance under different configurations, offering insights into the strengths and limitations of each method.

4.2.1 Comparison Between Two Presented Methods

The aim of this study is to divide it into three categories, each representing different quantities of base stations: 10, 50, and 100 gNB. For any scenario, a series of randomized starting and ending points will be assigned. Using the previously discussed algorithms, we will compare and analyze the routes, identifying their strengths, weaknesses, and their ability to optimize network resources. The simulations aim to gather data on the distribution of resources among base stations and the average cost efficiency of each node per iteration. We will investigate how the number of network resources influences the network configuration by altering the value of M during optimization.

10gNB Scenario

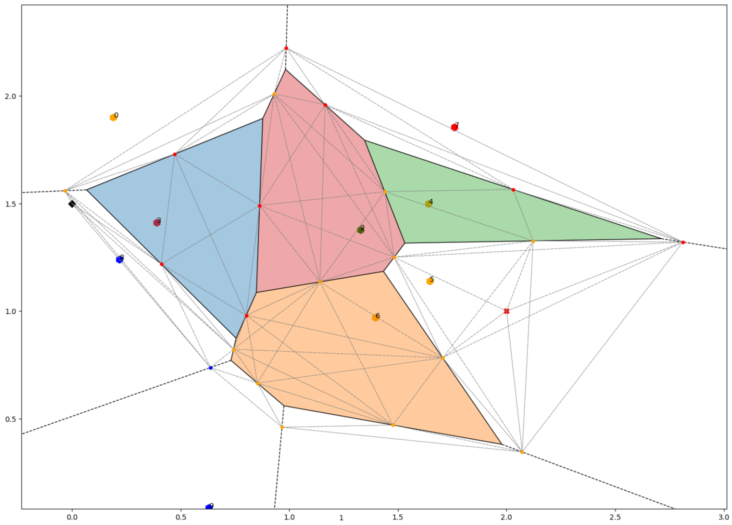


Figure 4.8: Scenario for simulations with 10 gNBs.

For this study, we initiated simulations for the 10 gNB scenario, utilizing the distribution illustrated in Figure 4.8. Because of the relatively small scale, the trajectories calculated by both algorithms will likely be very similar. This is because the AGV has fewer movement options compared to larger environments. This controlled environment offers a clear insight into the basic functions of the algorithms, without the complexities of larger network structures. After analyzing the data acquired from the simulations in the situation with 10 gNBs, Figure 4.9 shows a box plot that provides a detailed interpretation of the results.

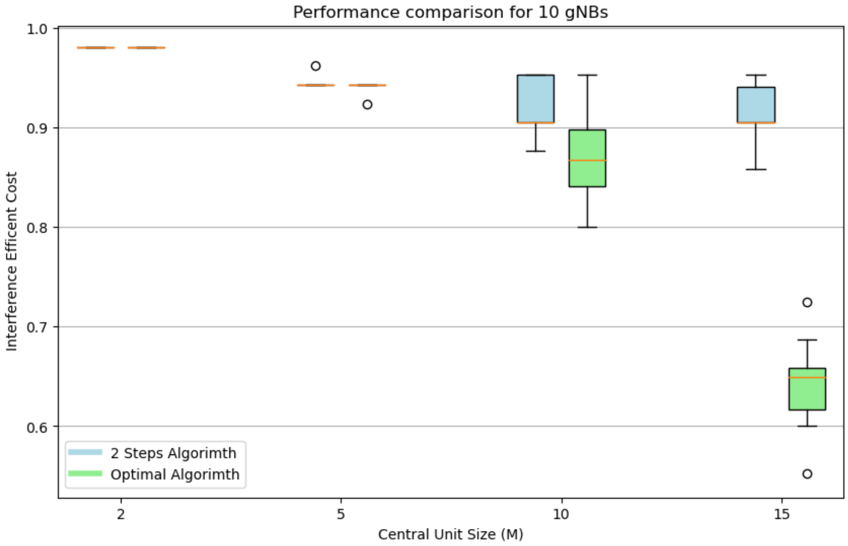


Figure 4.9: Average interference efficient cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 10 gNBs.

As can be observed, when the number of functions in the Central Unit is limited for the coordination of base stations, the fewer these functions, the more comparable the results between the two algorithms become. This is because they do not have significant flexibility to optimize the network. The subtle variations that can be distinguished between them are their choices of paths to the destination point; however, this does not significantly influence the average efficient interference cost of the network (Figure 4.11). On the other hand, as the network becomes more flexible in assigning functions and coordinating base stations, the average cost difference between the two methods begins to widen. The *Optimal Algorithm* consistently shows a lower average interference cost over the entire network configuration because it distributes resources more effectively during optimization. The average interference value in the network for the *Optimal Algorithm* is 0.86 and 0.65 when M values are 10 and 15, respectively. When compared to the values obtained from the *2 Steps Algorithm*, which are both 0.91, it indicates that the network resource management is inferior to the algorithm's design. Furthermore, it should be noted that the first two scenarios result from a small network with limited resources for optimization, having only 2 and 5 available slots for functions, respectively.

However, our main focus is to compare the paths computed by both algorithms. In Figure 4.10, the average interference values produced by the optimal route are illustrated.

It is apparent that in this small network, distinguishing variances between the two algorithms is difficult. Although they may generate distinct paths, the network's centralization configuration is nearly identical.

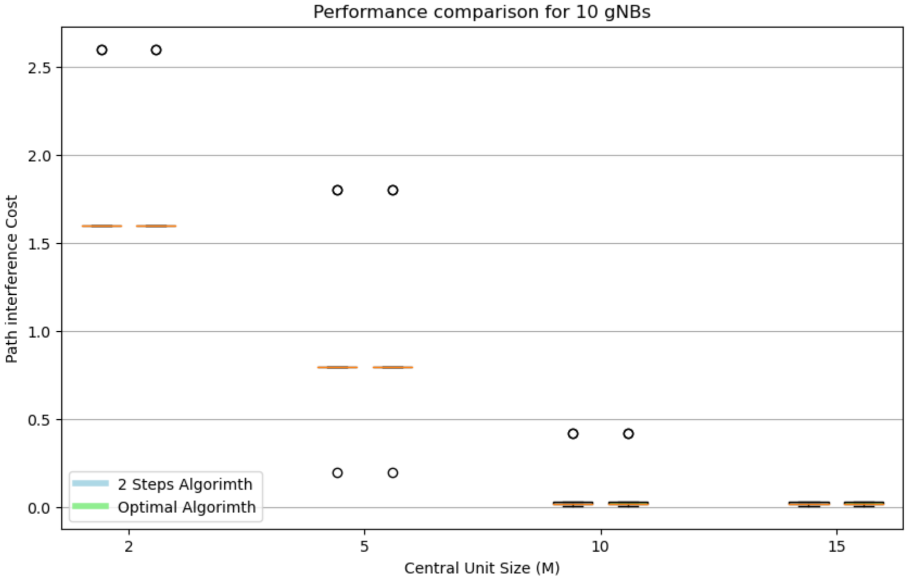


Figure 4.10: Average path interference cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 10 gNBs.

As previously noted, due to the small size of the network, it is unlikely for both algorithms to generate a different network configuration. This accounts for the minimal variability shown in Figure 4.10. For the initial three values of M, the boxplots illustrate the presence of several points indicating higher or lower costs in some of the paths compared to the rest of the averages.

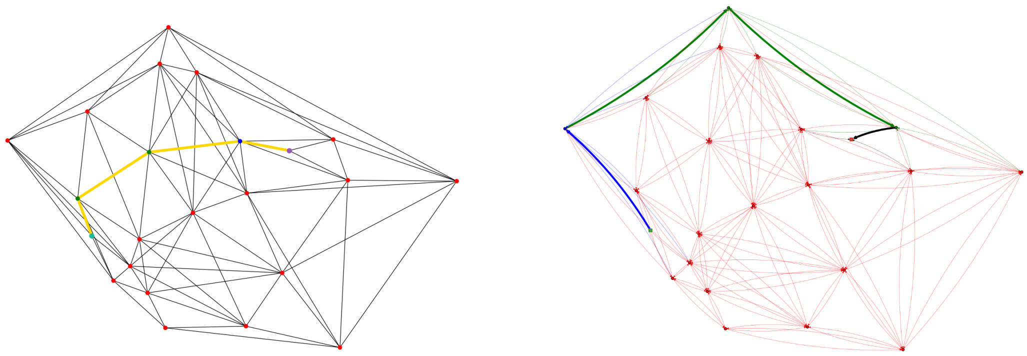


Figure 4.11: Same configuration to different path.

50gNB Scenario

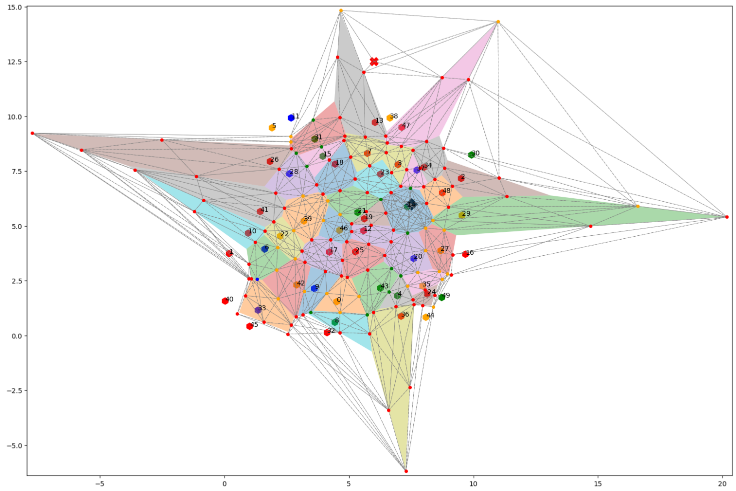


Figure 4.12: Scenario for simulations with 50 gNBs.

Continuing with the subsequent analysis, the following section explores a medium-sized network which comprises of 50 gNBs, as shown in Figure 4.12. This analysis proceeds in a similar manner as that of the previous scenario, wherein the optimal path's interference cost is determined and supported with network-wide total interference data. During the initial simulations, we found little difference between the outcomes produced by the *2 Step Algorithm* and the *Optimal Algorithm*. This phenomenon arises from the initial configuration used to derive results whereby the M value scales proportionally with the number of base stations; this implies that the functions centralized in the CU can be optimized more effectively without posing significant challenges. For our subsequent results, we have further limited the centralization of the network. For the network consisting of 50 gNBs, 5, 10, 12, and 20 centralized functions will be employed as the values of M.

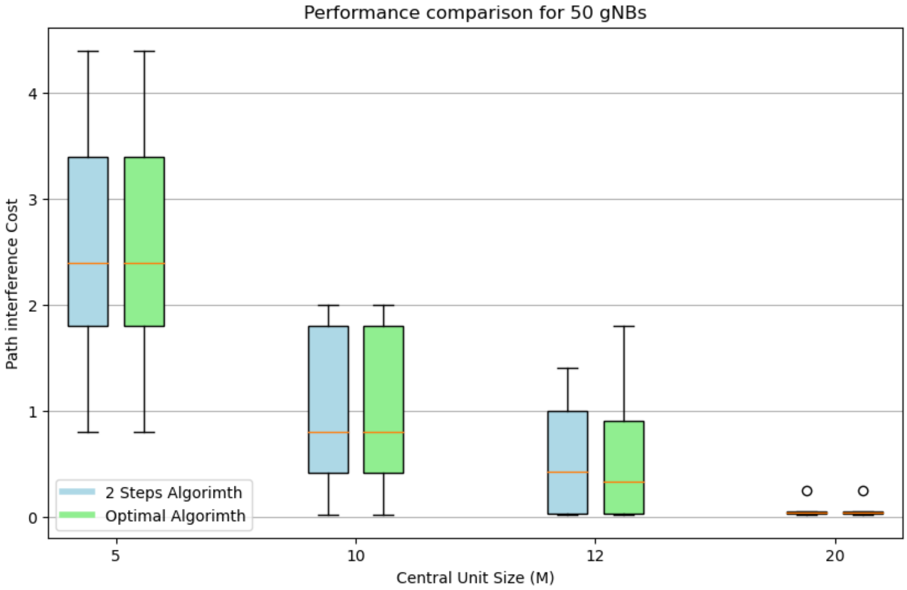


Figure 4.13: Average path interference cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 50 gNBs.

The variability of the results in Figure 4.13 regarding the interference cost for the path selected for the AGV is intriguing. In the previous instance, with network flexibility concerning large M values, both algorithms performed similarly with an average of approximately 0.01 when $M = 20$. However, when flexibility is limited, the *Optimal Algorithm* outperforms the *2 Steps Algorithm*. The greatest disparity becomes apparent when $M = 12$. Here, the mean interference expense is less than 0.5, a value that we deem satisfactory and commendable. However, when the centralized network is more limited, the data between the two algorithms can once again be compared, but with greater variability than in a configuration with more flexibility.

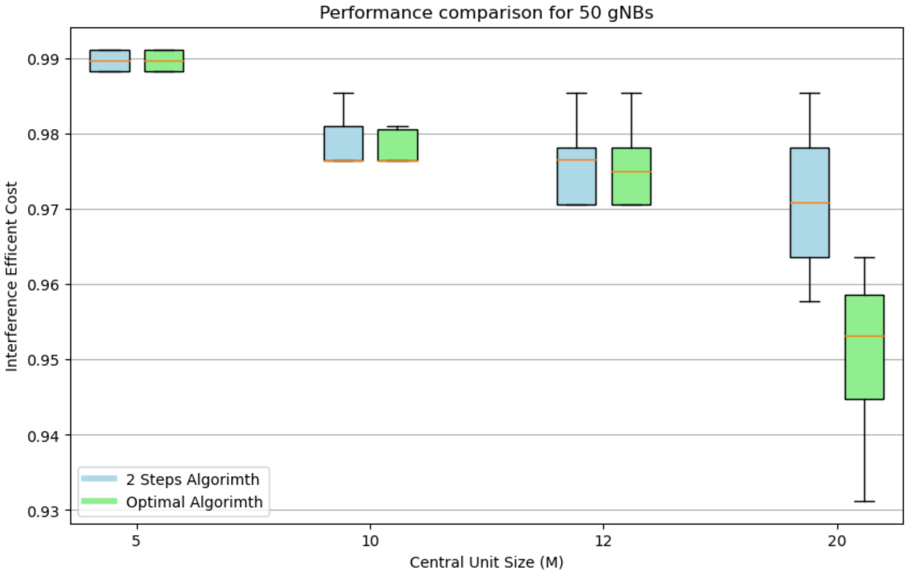


Figure 4.14: Average interference efficient cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 50 gNBs.

Finally, it should be emphasized that in managing the resources of the entire network, the *Optimal Algorithm* delivers consistently superior performance than the *2 Steps Algorithm* for less restricted values of M (refer to Figure 4.14).

100gNB Scenario

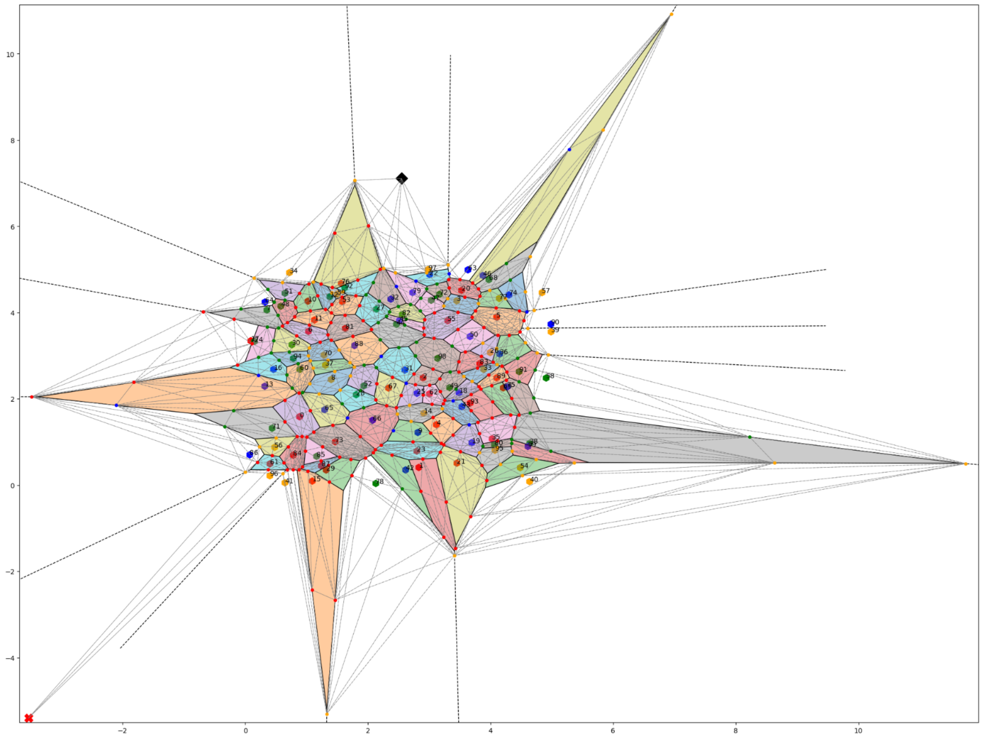


Figure 4.15: Scenario for simulations with 100 gNBs.

Finally, in the most complex scenario with 100 gNBs, we will follow the same study steps as before. For this configuration, we found similar results for the entire network management. This enables us to assert that the *Optimal Algorithm* outperforms the *2 Steps Algorithm*, even when faced with limited resources in a rather extensive network (see Figure 4.16). Given its scale, this scenario provided valuable insights into the network's behavior during high traffic and the effectiveness of the implemented methods to mitigate interference.

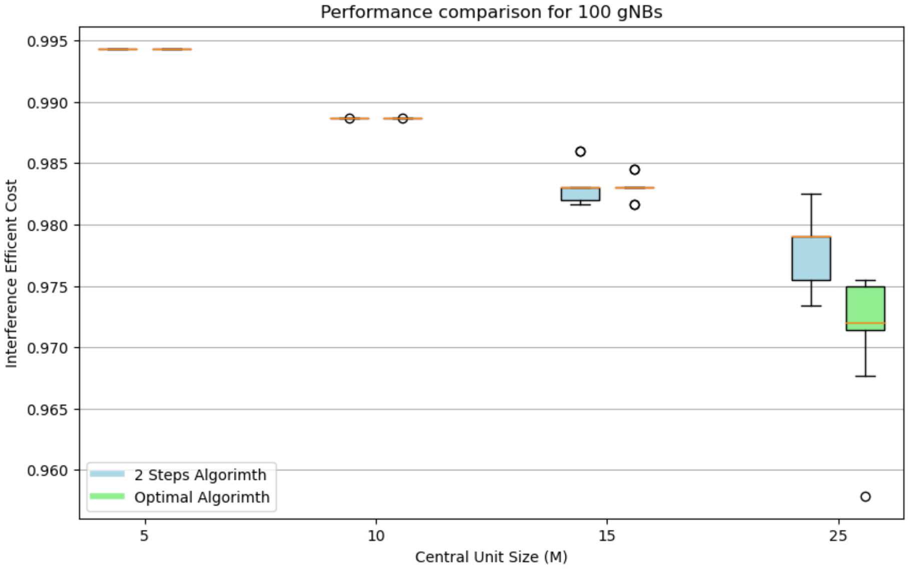


Figure 4.16: Average interference efficient cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 100 gNBs.

Now focusing on the analysis of the vehicle route, we selected four values of M: 5, 10, 15, and 25. The largest value represents 25% of the total number of base stations. Figure 4.17 displays the values, which initially appear to be quite similar, offering approximately the same average costs for all routes. However, the total cost can only be reduced for a small number of nodes in paths with a high amount of nodes to be visited. However, slight variability was observed during the simulations for the $M = 15$ configuration, where the first differences between the two algorithms were found. Figure 4.18 represents both the average values per simulation and the whole set, providing clearer insight into the matter. It shows that in five out of fifteen simulations, the algorithms demonstrate varying costs in their calculated optimal path for the AGV. In simulations 1, 3, 12, and 13, the *Optimal Algorithm* yielded a 0.6 improvement in interference cost compared to the *2 Steps Algorithm*. However, it should be noted that in simulation 4, the *2 Steps Algorithm* performed better than the *Optimal Algorithm*. Remember that these paths may not cross the same nodes, as we have seen in previous scenarios where each algorithm used a different path. However, they do intersect in the number of nodes they must cross. Therefore, comparing the average of these values is always for the same number of nodes.

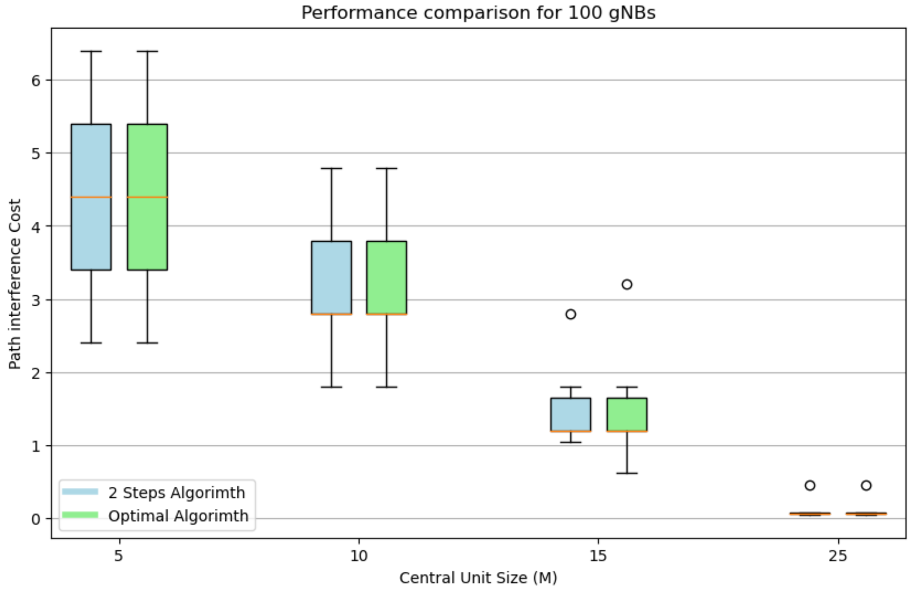


Figure 4.17: Average path interference cost achieved by *2 Steps Agorithm* and *Optimal Algorithm*, for different values of M, in a network of 100 gNBs.

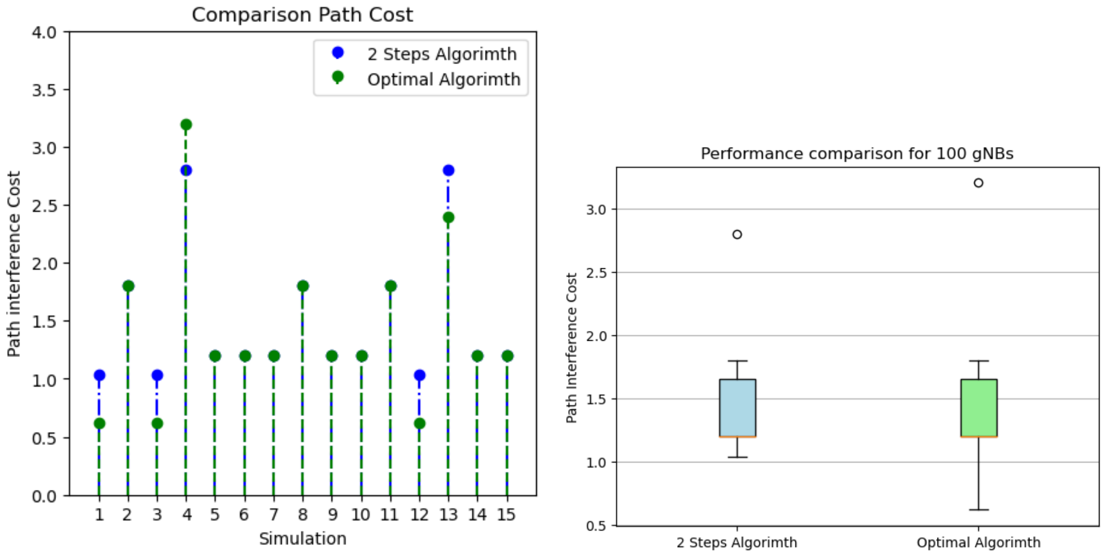


Figure 4.18: Path interference cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for each simulation when $M = 15$, in a network of 100 gNBs.

During our thorough examination of network optimization algorithms, clear patterns in behavior and performance were uncovered through objective observations. When assessing the *2 Steps Algorithm* and *Optimal Algorithm* across a range of network sizes, significant differences in efficiency were evident. However, when network capacity is limited, the variance in performance between the two algorithms is generally negligible. However, as the network's complexity and restrictions increased, the *Optimal Algorithm* consistently demonstrated superior adaptability and performance, especially with limited resources. This was evident even in extensive networks with up to 100 gNBs. Although both algorithms had their strengths and sometimes generated similar interference cost outputs, the *Optimal Algorithm* frequently handled interference more efficiently. These findings provide a solid foundation for further research and improvement endeavors in developing network algorithms, specifically in the realm of large-scale executions.

Chapter 5

Conclusions and Outlook

This Master's Thesis addressed the challenging integration of the Functional Split adaptation and the path selection for automated vehicles in industrial radio access networks. Through this exploration, several key lessons were unearthed that will lay the groundwork for future research and practical deployment.

Several notable conclusions have emerged from this study. The first, discussed in Chapter 4, is that when the network has the capacity to manage a significant number of functions in the central unit, the path obtained by applying Dijkstra, and thus the *2 Steps Algorithm*, will always be shorter in terms of distance. On the other hand, the *Optimal Algorithm* gives priority to crossing the fewest possible cells, that is, to visiting the fewest number of nodes on the way to the destination. It's also worth noting that there is an intermediate point where the differences between the two algorithms are maximized in terms of network optimization. If the problem is too restrictive, with small values of M, both techniques give almost identical results. We have observed that for a specific value of M, which typically approaches 20% of the number of gNBs comprising the network, the *Optimal Algorithm* consistently displays a lower average cost for the ideal path identified for the AGV and improved resource management in the centralized unit.

On the other hand, this study is initiated with the statement of three primary objectives. Two of these objectives have been met successfully, resulting in significant outcomes. The major accomplishment of the study is the creation of an effective algorithm designed for Functional Split adaptation. The algorithm proves advantageous in industrial settings that incorporate automated vehicles, as it simplifies the coordination and management between the Centralized Unit and the Distributed Unit. Furthermore, this study results in the presentation of an enhanced model exclusively designed for selecting optimal paths for these vehicles. The model is not merely a hypothetical concept; it thoroughly incorporates the intricacies of network topology and sophisticated details of industrial radio communication. Accordingly, it streamlines the efficiency of path selection while at the same time strengthening the connectivity for vehicles that are in motion.

Despite making considerable progress, we are unable to achieve specific objectives. More specifically, we encounter significant challenges in integrating new vehicles into the simulator. As discussed in Chapter 4, our inability to introduce new AGVs into the simulator necessitates a fresh perspective. Future research must further evaluate this issue to continue. It is possible to gain valuable insights into enhancing strategies and overcoming obstacles through the feedback and challenges faced. Given adequate time and resources, it is feasible to revisit and expand on the idea of implementing an assertive strategy.

Moreover, it is essential to move away from purely theoretical frameworks and focus on actual practical implementations. To achieve this goal, conducting small-scale experiments in genuine industrial contexts, such as those at Siemens Campus, is imperative. Practical trials can provide valuable insights into the algorithm's performance, limitations, and potential areas of improvement. With the rapid advances in 5G and vehicular communications, it is crucial to explore how emerging technologies can complement the algorithms and models presented in this study. This approach guarantees solutions to current challenges while increasing the efficiency and resilience of industrial radio access networks.

In conclusion, our study examined various scenarios and identified a direct correlation between the number of gNBs, the interference cost, and network resource management efficiency. The results we obtained lay a strong foundation for future research and optimization efforts in 5G networks, highlighting the crucial role of adaptive strategies as network infrastructure expands.

List of Figures

2.1 5G RAN Architecture Evolution [GV18]. 11
2.2 Dynamically centralized radio access networks [MA22]. 13
2.3 Possible Functional Split options along the protocol stack in a 5G RAN [MA22]. 14
2.4 RoI-optimal path calculated compared to a possible shorter path. 18

3.1 Architecture of the considered RAN, including centralized, distributed, and remote units, core, NOU, and vehicles. 22
3.2 Summarized procedure and message sequence. 25
3.3 Example operation of the described problem. 26

4.1 Scenario for 10 gNBs. 36
4.2 Shorter path using Dijkstra's Algorithm. 37
4.3 FS changes using Gurobi. 37
4.4 Scenario for 50 gNB. 38
4.5 Bidirectional graph for scenario depicted in Figure 4.4. 40
4.6 Comparison of paths according to algorithm. 40
4.7 Alternative paths from origin. 41
4.8 Scenario for simulations with 10 gNBs. 43
4.9 Average interference efficient cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 10 gNBs. 44
4.10 Average path interference cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 10 gNBs. 45
4.11 Same configuration to different path. 45
4.12 Scenario for simulations with 50 gNBs. 46
4.13 Average path interference cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 50 gNBs. 47
4.14 Average interference efficient cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 50 gNBs. 48
4.15 Scenario for simulations with 100 gNBs. 49
4.16 Average interference efficient cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for different values of M, in a network of 100 gNBs. . . . 50

4.17 Average path interference cost achieved by *2 Steps Agorithm* and *Optimal Algorithm*, for different values of M, in a network of 100 gNBs. 51
4.18 Path interference cost achieved by *2 Steps Algorithm* and *Optimal Algorithm*, for each simulation when $M = 15$, in a network of 100 gNBs. 51

List of Tables

3.1 Value Conversion for Functional Split 29
3.2 Value Conversion from **w** to **z** 33

4.1 Conversion for RoI in bidirectional graph 39

Appendix A

Notation und Abkürzungen

3GPP	3rd Generation Partnership Project
5G	Fifth Feneration networks
AGV	Autonomous Guided Vehicle
BS	Base Station
CU	centralized Unit
DU	Distributed Unit
FS	Functional Split
gNB	evolved eNodeB
GSM	Global System for Mobile Communications
IoT	Internet of Things
LP	Linear Programing
LTE	Long Term Evolution
MILP	Mixed-integer Linear Problem
MIMO	Multiple-Input and Multiple-Output
MIQP	Mixed-integer Quadratic Problem
mmWaves	Millimetre Waves
NOU	Network Optimizing Unit
PRB	Physical Resource Block
RAN	Radio Access Network
RoI	Risk of Interference
RU	Remote Unit
UE	User Equipment
URLLC	Ultra-Reliable Low-Latency Communication

Bibliography

[AK00] Franz Aurenhammer and Rolf Klein. Voronoi diagrams. *Handbook of computational geometry*, 5(10):201–290, 2000.

[AK19] Alberto Martinez Alba and Wolfgang Kellerer. A dynamic functional split in 5g radio access networks. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2019.

[AMO93] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, 1993.

[BV04] Stephen P Boyd and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

[For16] Small Cell Forum. Small cell virtualization functional splits and use cases, 2016.

[For17] Small Cell Forum. Fapi and nfapi specifications, 2017.

[GMJU16] Nitin Gupta, Kapil Mangla, Anand Kumar Jha, and Md Umar. Applying dijkstra's algorithm in routing process. *Int. J. New Technol. Res*, 2(5):122–124, 2016.

[Gur23] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023.

[GV18] Jorge Humberto Gomez Velasquez. *A flexible RLC-MAC placement for the next-generation RAN.* PhD thesis, Technische Universität München, 2018.

[GW74] Fred Glover and Eugene Woolsey. Converting the 0-1 polynomial programming problem to a 0-1 linear program. *Operations research*, 22(1):180–182, 1974.

[HMvdW$^+$20] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph

Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[LA23] Anbalagan Loganathan and Nur Syazreen Ahmad. A systematic review on recent advances in autonomous mobile robot navigation. *Engineering Science and Technology, an International Journal*, 40:101343, 2023.

[LHW14] Daniel R Lanning, Gregory K Harrell, and Jin Wang. Dijkstra's algorithm and google maps. In *Proceedings of the 2014 ACM Southeast Regional Conference*, pages 1–3, 2014.

[MA22] Alberto Martínez Alba. *Optimal Selection and Adaptation of a Flexible Functional Split in 5G Radio Access Networks.* PhD thesis, Technische Universität München, 2022.

[MAK22] Alberto Martínez Alba and Wolfgang Kellerer. Dynamic functional split adaptation in next-generation radio access networks. *IEEE Transactions on Network and Service Management*, 19(3):3239–3263, 2022.

[MZB^{+}18] Nikos Makris, Christos Zarafetas, Pavlos Basaras, Thanasis Korakis, Navid Nikaein, and Leandros Tassiulas. Cloud-based convergence of heterogeneous rans in 5g disaggregated architectures. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2018.

[TPY20] Cristian Tatino, Nikolaos Pappas, and Di Yuan. Multi-robot association-path planning in millimeter-wave industrial scenarios. *IEEE Networking Letters*, 2(4):190–194, 2020.

[VGO^{+}20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[Wol07] Laurence A Wolsey. Mixed integer programming. *Wiley Encyclopedia of Computer Science and Engineering*, pages 1–10, 2007.