TÉCNICO LISBOA

Railway planning: optimization approaches for the rolling stock rotation problem

Mickaël Anthony Da Silva

Thesis to obtain the Master of Science Degree in

Industrial Engineering and Management

Supervisor: Prof. Daniel Rebelo dos Santos

Examination Committee

Chairperson: Prof. José Rui De Matos Figueira
Supervisor: Prof. Daniel Rebelo dos Santos
Member of the Committee: Dr. Rita Portugal

November 2023

Dedicated to my mother.

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

En premier lieu, je tiens à remercier ma maman, Brigitte, pour toute la confiance qu'elle m'a accordée. Depuis le début de mes études universitaires, elle m'a acompagnée, aidée, soutenue dans toutes mes decisions, toujours cru en moi et en mon travail et je lui suis extrêmement reconnaissant pour cela. Je la remercie d'avoir fait de son mieux pour me fournir toutes les conditions nécessaires afin que je puisse travailler sur ma dissertation dans les meilleures conditions de travails possibles.

Em segundo lugar, quero muito agradecer aos meus avós, Salvador e Vitalina, por me terem acompanhado nestes 5 anos de jornada académica. Orgulharam-se sempre muito do meu trabalho e das minhas conquistas, e sinto-me genuinamente grato pela presença deles. Estiveram presentes ao meu lado desde o primeiro dia de faculdade e até então proporcionaram-me as melhores condições de trabalho, inclusive para o correto desenvolvimento da minha dissertação.

Um grande agradecimento à minha namorada, Beatriz, que se mostrou sempre disponível para celebrar as pequenas vitórias comigo ao longo da dissertação, para me ouvir desabafar nos momentos mais nervosos e difíceis ao longo destes meses e que conseguia sempre animar-me e ajudar-me a reconquistar a confiança e motivação necessárias para continuar a dissertação com entusiasmo. Sem a sua ajuda, não conseguiria desenvolver este trabalho em boas condições como foi possível. Sou-lhe extremamente grato por todo o seu acompanhamento e ajuda que foram imprescindíveis para esta dissertação, especialmente na reta final.

Agradeço imenso ao professor Daniel Santos, o meu orientador de dissertação. Sem ele, não conseguiria enfrentar muitos dos obstáculos colocados à minha frente no decorrer desta dissertação. Foi um grande gosto e extremamente enriquecedor aprender e aprofundar conhecimentos de investigação operacional a seu lado. Agradeço-o também pelo investimento e entusiasmo feito na minha dissertação, foram longas horas a partilhar e a discutir ideias. O professor Daniel mostrou-se sempre disponível para me ajudar, o que foi um grande fator decisivo para a elaboração da dissertação.

À minha família e próximos, agradeço profundamente por me terem acompanhado ao longo destes últimos cinco anos. Foram um apoio constante e mostraram-se sempre disponíveis para me ajudar fosse para o que fosse. Estes útlimos anos teriam sido muito mais difíceis se não tivesse tido o apoio deles, e por isso, um grande obrigado.

Por fim, quero muito agradecer a todas as pessoas que fizeram parte do meu percurso académico e que considero agora como amigos. Sem eles, estes cinco anos não teriam o mesmo significado e não seriam tão importantes e memoráveis. Agradeço também pelas tardes intermináveis passadas nos gabinetes 1.4.22 a 1.4.26, que ajudaram a tornar estes últimos meses mais leves e alegres.

Resumo

O material circulante é o conjunto de unidades ferroviárias e é um dos maiores ativos que uma empresa ferroviária pode ter. Ao escalonamento destas carruagens chama-se planeamento da rotatividade das locomotivas de comboio e é um problema muito complexo e demoroso, sendo que tem de ser feito incorrendo os menores custos operacionais possíveis. O escalonamento deve garantir que as carruagens operam todas as viagens planeadas, respeitando os requisitos ferroviários. Este trabalho incorpora várias condições relativas ao sistema ferroviário, tais como a composição dos veículos, a capacidade das estações, a bonificação para viagens regulares, a penalização para viagens em vazio, e as manobras de acoplar, desacoplar e virar carruagens. Resolvemos o problema aplicando o modelo dos hiperarcos e a nossa contribuição foca-se na adição de uma nova função objetivo, passando a ter um modelo bi-objetivo. Esta nova função objetivo visa minimizar as emissões produzidas ao longo das operações envolvidas. Para resolver este modelo, usamos o método da ε-restrição aumentada. Mostrámos que a complexidade do nosso modelo deve-se principalmente aos hiperarcos das operações de acoplamento e desacoplamento e que, variando a bonificação de regularidade, alteramos as rotas de solução, enquanto a variação da penalização de viagens em vazio não se altera tanto. Além disso, quando se considera a minimização das emissões, as soluções tendem a ter menos viagens em vazio do que as soluções fornecidas pela minimização dos custos. Por fim, para validar e testar o modelo, utilizámos a base de dados pública do sistema de metro de Chicago.

Palavras-chave: Material circulante, Composições de veículos, Acoplamento, Desacoplamento, Escalonamento, Emissões produzidas

Abstract

Rolling stock is the set of train units and is one of the most expensive assets that a railway company can own. The rolling stock rotation planning aims to schedule the rotation of the rolling stocks at minimal costs and is a very complex and time-consuming problem when properly integrated. The rotation must ensure that it covers all trips already defined in the timetable, and respects the requirements imposed by the railway network. This work integrates a significant number of industrial railway requirements such as vehicle compositions, station capacity, regularity bonification, deadhead trip penalization, shunting, and turn operations. We solved the problem through the hypergraph model methodology, and our contribution is the addition of a new objective function, implying the model to be considered as bi-objective. The new objective function aims to minimize the damaging emissions produced within the operations concerning the rolling stock rotation. To solve the model, we used the augmented ε-constraint method. We showed that the complexity of our model is mainly due to shunting operations hyperarcs and that by varying the regularity bonification, we would change the solution routes while the variation of the deadhead trip penalty would not change as much. Moreover, when considering the minimization of the emissions, the solutions tend to have fewer deadhead trips than the solutions provided by the cost minimization. Lastly, to validate and test the model, we used the Chicago railway subway's open data.

Keywords: Rolling stock, Vehicle composition, Shunting, Scheduling, Emissions produced

Contents

Acknowledgments vii
Resumo ix
Abstract xi
List of Tables xv
List of Figures xv
Glossary 1

1 Introduction **1**
1.1 Problem background 1
1.2 Motivation 2
1.3 Objectives and deliverables 3
1.4 Dissertation's structure 3

2 Literature Review **5**
2.1 Rolling stock rotation planning models 7
2.2 Scope 8
2.3 Requirements integration 8
2.4 Objective Function 9
2.5 Solution method 10
2.6 The environmental approaches 11
2.7 Summary in contribution 12

3 The rolling stock rotation planning problem **13**
3.1 Definition 13
3.2 Indexes 14
3.3 Parameters 14
3.4 Periodic Timetable 15
3.5 Constraints 16
3.5.1 Vehicle compositions 16
3.5.2 Capacity constraint 18
3.5.3 Regularity constraint 18
3.5.4 Orientation constraint 19

3.5.5 Station constraint . . . 20
3.6 Emissions of CO_2 equivalent . . . 21

4 Hypergraph Model **23**
4.1 Mixed Integer Programming Formulation . . . 23
4.2 Generation of hyperarcs . . . 24
4.3 Multi-objective approach . . . 28
4.3.1 Multi-objective methods . . . 28
4.3.2 Augmented ε-constraint method . . . 29

5 Chicago subway data **33**
5.1 The RSRP for subway railway planning . . . 33
5.1.1 Chicago Transit Authority description . . . 33
5.1.2 Chicago Transit Authority demand . . . 34
5.1.3 Chicago Transit Authority vehicles . . . 37
5.1.4 Chicago Transit Authority stations . . . 41
5.1.5 Chicago Transit Authority estimations . . . 42

6 Results **45**
6.1 Instances description . . . 46
6.2 Model's validation . . . 46
6.3 Complexity of the hypergraph model . . . 48
6.4 Addition of the emissions minimization objective function . . . 52
6.5 Bonification and penalizations . . . 55

7 Conclusions **58**

Bibliography **61**

List of Tables

2.1 Classification of this work and literature regarding RSRP optimization, based on Thorlacius et al.(2015) and Schlechte et al.(2023). 6

3.1 Indexes. 14
3.2 Station parameters' description. 14
3.3 Vehicle parameters' description. 15
3.4 Trip parameters' description. 15

5.1 Stations information. 34
5.2 Trips information. 37
5.3 Lines information. 38
5.4 Chicago vehicles' length and capacity. 38
5.5 Chicago's operating expenses. Source: Authority (2020). 39
5.6 Chicago vehicle's costs. 40
5.7 CTA's vehicles information. 41
5.8 Distance matrix between stations. 43
5.9 Computation of the average speed. 43
5.10 Bonification and penalization values. 44

6.1 Instances description. 46
6.2 Computational results. 53
6.3 Comparison between solution_1 and solution_2. 55

List of Figures

1.1 Evolution of the passenger-kilometers between 2015 and 2021. 1
1.2 Levels of railway planning. 2

3.1 Periodicity of the timetable. 16
3.2 Hypergraph model representation. 17
3.3 Regular trips. 19
3.4 Possible compositions considering orientation and position for two vehicles. 20
3.5 Tracks in terminal stations. 21

4.1 Diagram representing the generation of h_c between h_i and h_j, considering equal compositions. 26
4.2 Diagram representing the generation of h_c between h_i and h_j, considering shunting operations. 26

5.1 Demand distribution of the Beijing Yizhuang line. 35
5.2 Demand distribution throughout the day. 35
5.3 Chicago subway lines. 36
5.4 Chicago terminal stations. 42

6.1 Instance_0 solution for cost minimization. 47
6.2 Instance_0 route solution for cost minimization. 47
6.3 Route three's rotation. 48
6.4 Numerical and percentage increased of the number of vehicles per composition. 50
6.5 Numerical and percentage increased of the number of trips for the same feasible compositions. 50
6.6 Resolution times for the several instances scenarios. 51
6.7 Pareto Front from instance_1. 54
6.8 Variation of the number of regular and deadhead trips with the variation of the regularity bonification. 56
6.9 Variation of the number of regular and deadhead trips with the variation of the deadhead trips cost penalization. 57

6.10 Variation of the number of regular and deadhead trips with the variation of the deadhead trips emissions penalization. 57

Chapter 1

Introduction

1.1 Problem background

Over the years, the railway companies have more and more passengers to satisfy their travel needs, as can be seen in Figure 1.1. Even if the pandemic COVID-19 significantly decreased the passenger flow (46%), within the next year (2021) the passenger flow increased by 16.5% which is a considerable rise. Due to this continuous increase in railway passenger demand, it is imperative for railway companies to enhance their readiness to satisfy the ongoing emergence of passengers.

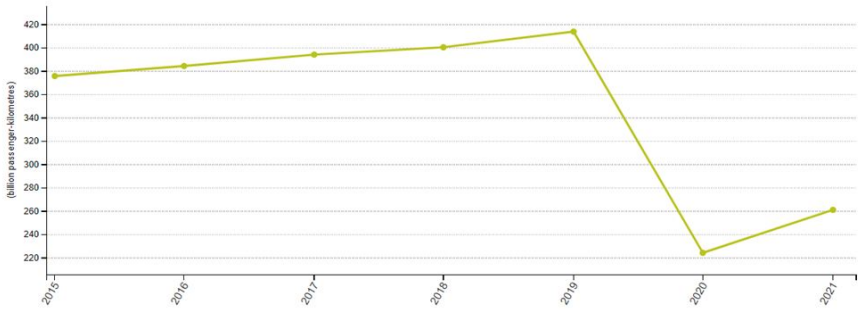


Figure 1.1: Evolution of the passenger-kilometers between 2015 and 2021. Source: Eurostat (2022).

Due to the complexity of operating railway transportation, the use of mathematical models and optimization methods results in gains regarding both railway customers and operators either for customers through improvement of service quality or operators in terms of reducing the costs. Indeed, one of the first applications for mathematical optimization and operations research was for railway planning (Schrijver, 2002).

Railway planning is a complex system involving the planning of several operations that can be divided into several levels with different time horizons (Liebchen and Möhring, 2014; Lusby et al., 2017). Starting from the general to the particular, first, there is the strategic level which is a stage in which the planning horizon is developed for several years, then the tactical level which has a planning horizon of days to years, and finally, the operational level where it deals with real-time planning management. This schema

is displayed in Figure 1.2. The problem aboard on this work is the rolling stock rotation and it finds itself at the tactical level (blue box).

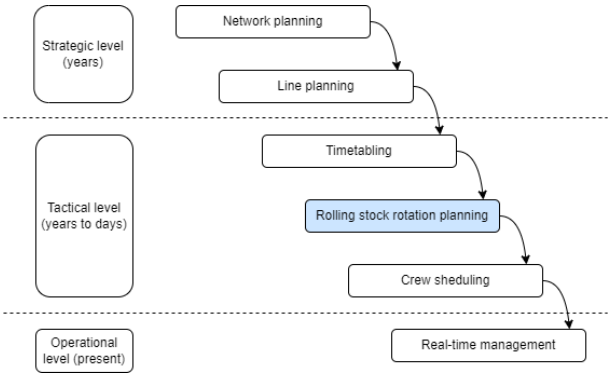


Figure 1.2: Levels of railway planning. Adapted from Lusby et al. (2017).

The rolling stock rotation planning (RSRP) problem is an essential part of railway planning, because it involves the scheduling of the train locomotives, in order to fulfill the timetable trips previously planned, satisfying the passengers' demand, while minimizing the operational costs (Thorlacius, 2015). One can see in Figure 1.2 that the trip timetable planning is performed just before the RSRP.

With the continuous increase in the number of passengers using the railway mode of transportation, the pollution associated with the railway network could also rise. Krezo et al. (2018) stated that of the several transportation modes, the railway is known for being one of the greenest modes of transportation, once that it is the transportation mode that produces less pollution. Nevertheless, railway transportation is not as flexible as other modes, leading to the transportation of significant amounts conveyed in massive train units (Demir et al., 2015). Consequently, with the increase of the railway network, and the encouragement from governments to embrace railway transportation, this work will also consider the emissions produced by the operations related to the rolling stock rotation problem.

1.2 Motivation

The RSRP is a major goal to have an efficient railway undertaking. The operations regarding the RSRP are complex and can change over the years due to timetable alterations, acquisition of new rolling stock, creation of new railway lines or destruction of existing ones, and significant changes in demand, for example. Additionally, some extraordinary circumstances can occur like the collision of two freight trains in November 2022 that ended closing the railway line between Hanover and Berlin (Freight, 2022), which lead to an inevitable re-scheduling of the rolling stock rotation in a very short period of time.

Hence, the ability to solve the RSRP in the most optimal and efficient way offers railway companies a significant competitive advantage in the railway market (Schlechte et al., 2023). For this reason, even if the rolling stock rotation planning is a tactical level problem (see Section 1.1), there can be cases in which it is an operational level problem.

Here lies the importance of the RSRP and the motivation for optimizing this problem of the many problems that railway companies have to deal with. Moreover, considering the emissions produced in the operations concerning the rolling stock rotation planning, and thus, confronting the results with the ones resulting from the cost optimization brings a new vision to the RSRP existing in the literature.

1.3 Objectives and deliverables

The goal of this work is to solve the rolling stock rotation planning problem using the hypergraph model proposed by Borndörfer et al. (2011), i.e., to find the optimal composition of vehicles for each trip, and their rotation, considering the constraints that the railway companies have. The optimal composition of the vehicles will take into consideration the minimization of the costs of this problem as well as the minimization of the emissions emitted by the railway system within the operations required. The duality between the two objective functions will be studied and we aim to find which features have more impact on the costs and which one have more impact on the emissions.

The hypergraph model is a complex model, and the more characteristics we integrate within the model, the more complex it is. Hence, we also have as an objective the understanding of the complexity of this model and provide possible suggestions to simplify the model.

1.4 Dissertation's structure

This work is composed of seven chapters, in which every step of the work will be properly explained and detailed.

The first chapter regards the introduction with a small overview followed by an explanation and contextualization of the rolling stock rotation planning problem. We introduce the parameters that we aim to analyze, and the goals that we will try to reach at the end of the dissertation.

The second chapter regards the literature review in which there are details of what is already known and what was already done for this subject, such as which models exist to solve the RSRP and their differences, or even what operations are involved within the rolling stock rotation planning. It will allow us to better understand the problem and decide which path to follow.

In the third chapter, there will be a detailed explanation, description, and definition of the rolling stock rotation planning problem, as well as a description of the characteristics integrated into our version of the RSRP model.

Within chapter four, we develop more about the hypergraph model, once the mixed integer programming formulation is presented and that we define the generation of the decision variables proper to the hypergraph model. Moreover, the augmented ε-constraint method is briefly explained once it is used to analyze and solve the RSRP considering it as a bi-objective problem.

Chapter five describes the Chicago subway network, and we will use the Chicago data in order to validate our model. This chapter includes all computations required to solve the RSRP using the hypergraph model as well as all assumptions made, regarding the Chicago real-world case.

By the sixth chapter, we will present and discuss the obtained results from our version of the model, as well as draw conclusions and reach the goals set in this first chapter,

Finally, chapter seven presents the conclusion of this dissertation and it discusses an overview of the achievements reached with our version of the RSRP model, as well as some suggestions for future work.

Chapter 2

Literature Review

The rolling stock rotation planning problem is a particular case of vehicle scheduling, but for railways. This last scheduling problem is well-known in the literature and has been studied by several authors (see survey Löbel,1997). The RSRP can be composed of several levels addressing various detail levels of this scheduling, and the biggest goal is to cover all the timetabled trips by the existing rolling stock, at the lowest operational costs possible (Thorlacius et al., 2015), respecting all the necessary requirements.

Initially, the operations research (OR) approaches would solve several specific problems within the railway undertakings, but individually, which are briefly described in many surveys (Cordeau et al., 1998; Ahuja et al., 2005; Caprara et al., 2007). More recently, the railway utilization of the OR techniques is no longer used to solve individual problems at a time, but to find optimal solutions to the issues that can be connected between them, which they are called integrated models (Thorlacius et al., 2015). This leads to the elimination of sub-optimal solutions, once for integrated models the solution takes into consideration way more parameters and requirements. This approach of the integrated models has not only been used for railway problems but has been used in the airline industry at the moment (Saddoune et al., 2012). Rolling stock rotation planning requires the integration of several features. As Thorlacius et al. stated in 2015, ”rolling stock planning is often performed in a step-by-step manner, taking only some of the many requirements into consideration in each step”. This means that not all requirements can be taken into account, or at least they should be progressively involved in the model. However, the more features implemented into the model, the more complex and time-consuming is to perform and compute.

In Table 2.1, there is a detailed description of the features implemented within the rolling stock rotation models presented in the literature from 2005 until 2023. The table is divided into the following characteristics: scope, topic, process, requirements integration, objective, and solution method.

Table 2.1: Classification of this work and literature regarding RSRP optimization, based on Thorlacius et al.(2015) and Schlechte et al.(2023).

Authors	Year	Scope		Requirements integration							Objective					Solution method			
		Cyclic	Acyclic	Train composition order	Maintenance by time	Maintenance by distance	Personnel on duty	Depot capacity	Depot topology	Regularity	Minimise cost	Minimize penalties	Minimize of train units	Maximize benefits	Minimize emissions	Commercial (MIP) solver	Heuristics	Column generation	Branch and price
Freling et al.	2005		✓	✓				✓	✓			✓				✓		✓	
Alfieri et al.	2006	✓		✓							✓					✓	✓		
Maróti and Kroon	2007	✓			✓	✓						✓				✓	✓		
Jha et al.	2008		✓								✓	✓				✓	✓		
Kroon et al.	2008	✓		✓				✓	✓			✓				✓			
Peeters and Kroon	2008	✓		✓							✓					✓			✓
Vaidyanathan et al.	2008	✓			✓						✓	✓				✓			
Cadarso and Marín	2011		✓					✓	✓		✓	✓				✓			
Borndörfer et al.	2011	✓				✓				✓	✓	✓				✓	✓		
Borndörfer et al.	2012	✓				✓		✓		✓	✓	✓				✓	✓		
Cacchiani and Toth	2012	✓	✓										✓				✓		
Giacco et al.	2014	✓			✓	✓							✓			✓			
Thorlacius et al.	2015		✓	✓	✓	✓	✓	✓	✓		✓	✓		✓			✓		
Haahr et al.	2016		✓					✓			✓	✓				✓		✓	
Borndörfer et al.	2016	✓		✓	✓		✓		✓	✓	✓	✓				✓	✓	✓	
Borndörfer et al.	2017	✓		✓	✓	✓		✓		✓	✓	✓				✓	✓	✓	✓
Lusby et al.	2017		✓	✓				✓	✓		✓	✓				✓		✓	✓
Grimm et al.	2019	✓		✓	✓	✓		✓		✓	✓	✓				✓	✓	✓	
Hoogervorst et al.	2021		✓	✓				✓		✓	✓	✓		✓		✓	✓		
Borndörfer et al.	2021	✓		✓	✓		✓		✓	✓	✓	✓				✓	✓	✓	
Gao et al.	2022	✓		✓	✓	✓					✓	✓					✓	✓	✓
Zhao et al.	2023	✓		✓	✓	✓					✓	✓					✓	✓	✓
Schlechte et al.	2023	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓		✓			✓	✓	
This work	2023	✓		✓					✓	✓	✓	✓			✓	✓			

2.1 Rolling stock rotation planning models

To solve the rolling stock rotation planning, many authors choose an arc-based multi-commodity flow or some similar model, therefore denoted as flow-based models (Alfieri et al., 2006; Peeters and Kroon, 2008; Vaidyanathan et al., 2008; Cadarso and Marín, 2011). This type of model focuses on the conservation of the flow, i.e., each vertex ensures that the incoming flow is equal to the outgoing flow, a typical approach in vehicle routing problems. It allows significantly lower complexity associated with the models. However, the more integrated the model is the more complexity the model gains, such as the implementation of regularity trips or maintenance requirements. Another model type used by the authors (even if less popular) is the path-based multi-commodity model, which from now on is referred to as path-based approaches (Jha et al., 2008; Thorlacius et al., 2015; Haahr et al., 2016). For this approach, every singular sequence of displacement of the vehicles is modelled, through the association of a path to the individual vehicles.

In 2011, Borndörfer et al. presented the hypergraph model, whose main difference and benefit is to model changes in the train's compositions at terminal stations, for long-distance railway passenger transport, in a different way. The authors handled the railway's necessary requirements using hyperarcs, rather than the arcs generally used until the moment. In this way, the representation of the train unit displacements is more detailed, leading to a better granularity level. The authors also proved that the model provides high-quality solutions in reasonable times, using the rapid branching and column generation techniques, for real-world instances. The first version of the model did not include the maintenance constraints. For this purpose, the authors Borndörfer et al. published in 2016, an improvement of their hypergraph model, considering the maintenance requirements.

The three models presented above are suitable for the RSRP and are more appropriate depending on the main features to integrate into the model. As a matter of fact, although integrated models consider several characteristics, due to the vastness of railway operations, some models enhance some requirements more than others. For instance, on the one hand, the hypergraph model presents itself as being more flexible and detailed in the representation of the train unit displacements, allowing a better understanding of the flows within the rotations, and is useful to lead with complex requirements. However, on the other hand, due to its intricacy is harder to model, and leads to higher computational times. Therefore, an alternative to simplify the hypergraph model is the flow-based approach, which is usually simpler to implement and less time-consuming to compute. Nevertheless, it is not as detailed as the hypergraph model, implying that it cannot capture all the complexities regarding the interactions between the rolling stock. Lastly, the path-based models have an intermediate quality between the hypergraph and the flow-based models, providing varied results' precision, once that it depends on how well the rolling stock rotation problem is modelled. Hence, depending on the author's expectations, different approaches can be exploited.

2.2 Scope

For the models that consider timetables that are presented as cyclic (also called periodic timetable, or regular timetable), there is an incentive for the train fleet to follow regular trips over the horizon, which can be defined within days, weeks, and sometimes months. This implies that each trip sequence can be defined a priori through a set of simple rules (Lusby et al., 2017). A significant part of the authors present models with a regular timetable to operate.

It can also happen that the timetable to operate and schedule the rolling stocks is acyclic, meaning that the trips are not regular. For these situations, there is a necessity to optimize each trip sequence, and the trip sequence cannot be provided a priori. The fact that the timetable is not regular leads to only considering a lower number of train compositions when compared with regular timetables and the train fleet position and order are commonly ignored (Lusby et al., 2017).

When comparing both cyclic and acyclic timetables, each one of them has its advantages and disadvantages. For example, the cyclic timetable allows passengers to get used to the departure time of their usual train at their usual departure station. Moreover, it also allows us to consider a significant number of passenger transfers. On the other end, the acyclic timetable is more flexible than the periodic timetable. Furthermore, the cyclic timetable can be inefficient, once it can operate trips on days of the week in which the demand is significantly lower than the rest of the days, which does not happen with the acyclic timetable (Kroon et al., 2008).

2.3 Requirements integration

The requirements integration is one of the most important topics presented in Table 2.1, once they allow us to know what kind of requirements the models integrate into their solution. As we discussed, the rolling stock rotation planning is an intricate problem due to several limitations imposed by the layout of the rail. Hence, the more constraints are integrated into the model, the more complex the model, thus more time-consuming whether on the formulation problem or in the resolution time (Cadarso and Marín, 2011; Haahr et al., 2016; Borndörfer et al., 2021).

Regarding the train composition order, the addition of these requirements leads to a significant complexity of the model, once the model needs to take into account way more possibilities than if the model did not consider it (Thorlacius et al., 2015; Borndörfer et al., 2021). Thus, the integration of this feature is an important decision: some authors prefer to ignore this characteristic such as Maróti and Kroon (2007) and Jha et al. (2008); and others authors implement this feature considering both the composition order and the orientation of the fleet like Haahr et al. (2016) and Borndörfer et al. (2016).

Furthermore, in 2021, Borndörfer et al. suggested dividing this problem into three layers using the coarse-to-fine (C2F) approach. The authors used a hierarchical column-generation method that first performs the calculations on a coarse level and verifies the orientation and position on a more fine level. The first layer is the coarse one, the vehicle layer, in which neither the position nor orientation of the

fleet is considered. Then, there is the middle level that considers the vehicle position and it is called the configuration layer. Finally, the fine layer is the composition layer in which both the position and orientation of the fleet are taken into account. This method allows the evaluation of the train compositions in several levels of complexity. Moreover, as we discussed previously in Section 2.2, it can be proved that the models considering acyclic timetables do not integrate within their models the train composition order.

Maintenance requirements are essential to railway fleets. Table 2.1 separates the models integrating the maintenance by time and maintenance by distance. The difference between them is the necessity to maintain the fleet after a certain number of hours or kilometres travelled by the train unit. It is less complex to add maintenance requirements for the path-based models, once each possible movement sequence of a certain train unit is modeled. However, when it comes to flow-based models, the addition of the maintenance requirements adds more complexity to the model. Moreover, it is difficult to integrate the maintenance restrictions for acyclic models.

For the flow-based models, in which the maintenance increases the complexity of the model, the daily train units are generated to comprise a certain number of arcs related to the maintenance operation, meaning that the fleet can be assigned with sufficient maintenance operations. In this way, Gao et al. (2022), divided the problem into two levels: the first level is to generate the trip sequences while ignoring the maintenance requirements; and the second level is to assign each fleet to its maintenance necessities, according with the optimization of the maintenance operations.

The depot capacity is considered in the majority of the models, and it represents the maximum number of fleets that the depot can take. This feature is easily combined with the depot topology once the topology aims to specify how the shunting and turn operations are performed (Freling et al., 2005; Kroon et al., 2008; Cadarso and Marín, 2011; Thorlacius et al., 2015; Lusby et al. 2017; Schlechte et al., 2023).

The regularity constraint comes directly from the cyclic timetable. In 2011, Borndörfer et al. introduced the regularity notion in their hypergraph model. This constraint enhances the model to return rolling stock rotations in which the train compositions are the same, for regular trips. This is achieved through a reduction of the costs if the train composition is constant for all regular trips. The authors kept this requirement (Borndörfer et al., 2021), and it aims to provide the passengers with some comfort, in terms of knowing what to expect when the train arrives. For example, for the same regular trips, the passengers would be already expecting the train to have a certain length, and for that reason, they already know which part of the platform the train will take over.

2.4 Objective Function

When it comes to optimization purposes, many models use several types of objective functions. The main objectives functions used are mainly concerning minimizing operational costs and penalties, as can be seen in Table 2.1. Minimizing costs is one of the most used approaches by the authors,

which allows the company to reduce its actual costs in the operations performed, thus saving money. It is to be highlighted that whether the author minimizes the penalties, the train units, or maximizes the benefits, all of these objective functions implemented in their respective models, are intrinsically related to the operational costs. For example, Jha et al. (2008), integrated into their model penalizations for paths that deviate from the due date, in the form of costs. The penalization is considered in the cost unit, considering the path miles, the deviation, and the due date. Another example is the model presented by Borndörfer et al. in 2016 that penalized the deadhead trips (empty trips) through a time penalization, that was then converted into additional costs. Regarding the minimization of the train units, the same happens, once that Cacchiani and Toth (2012), and Giacco et al. (2014) presented their models minimizing the train units, however, this was performed through the minimization of the costs directly related to the train units.

A distinct approach was presented in 2015, by Thorlacius et al. which integrated within their model the benefit approach. The authors introduced the benefit to counterbalance the operational costs and penalties involving the rolling stock rotation planning. Thus, they were able to compute the net value of the operations, by subtracting the costs and penalizations to the benefits. This model allows the authors to have a different and more financial understanding of the operations.

As far as we know, the models presented in the literature are solved through one objective function. As mentioned, these objective functions can be detailed and itemized into arc costs, turn operation costs, shunting costs, penalization costs, and maintenance costs (Borndörfer et al., 2012; Cadarso and Marín, 2011). However, all of these parameters are set on the same scale, which is the cost. Our contributions presented in this work is to explore the impact of having a bi-objective rolling stock rotation planning problem, through the minimization of the costs and the emissions. When it comes to the minimization of the emissions, to the best of our knowledge, there are no traces of RSRP models implementing this objective function, or even considering some features regarding the pollution integrated within the models. Hence, this work, by considering this new objective function, brings a new perspective to the rolling stock rotations planning problems.

2.5 Solution method

Regardless of the approach chosen to solve the rolling stock rotation planning problem (hypergraph, flow-based, and path-based models), due to the intricacy of the real-world data, the majority of the models implemented a heuristic method to solve the problem. Indeed, all solution methods used in the literature apply commercial solver or heuristics, and, in some cases, the authors use both methods (Thorlacius et al., 2015).

Furthermore, some authors also applied decomposition techniques in their models such as column generation and branch and price techniques, for some cases. By doing this, the authors were able to reduce the resolution time for the integration of some requirements within the model (Borndörfer et al., 2021) and the resolution time of the solution computation (Lusby et al., 2017; Grimm et al., 2019).

Moreover, the authors were able to prove that the application of heuristics methods reduced the solving time and returned good solutions, without having a large increase in the costs in the solutions found.

Through the heuristics and decomposition applications, the models presented by the authors show significantly low-resolution times considering real-world cases. Furthermore, on average, the instances solved by the authors do not consider timetables with more than 2,000 trips. This has led to a lack of studies regarding instances considering more than 2,000 trips (Gao et al., 2022).

2.6 The environmental approaches

As we saw in Section 2.4, the main focus of the RSRP's models presented in the literature is the minimization of costs, regardless of the pollution impact. This is due to the fact that the authors essentially use their version of the rolling stock rotation problem for real-world instances, in collaboration with real-world railway companies. Giving some examples, Freling et al. (2005) used their model to optimize the Netherlands shunting operation for the Zwolle station, Borndörfer et al. (2016) applied the hypergraph model to the Deutsche Bahn in Germany, and Hoogervorst et al. (2021) implemented their RSRP's model for the Netherlands Railways.

The environmental concerns are not implemented in the rolling stock rotation models, leading to models that do not aim to minimize the operations emissions produced. As a matter of fact, since the beginning of the railway development, technology has considerably evolved, making it possible to start improving other factors, beyond the cost reduction. By focusing on cost reduction at first, advancements in energy efficiency needed to be made. According to the International Energy Agency (IEA) and the International Union of Railways (UIC), between 1975 and 2012, the energy consumed per passenger-kilometer decreased by 62%, a decrease that was accompanied by a 60% reduction in CO_2 emissions for the rail passenger transport (Agency, 2023; Railways, 2023). The case of China, India, and Russia support these accomplishments. These are three countries with extensive railway networks, and yet the ones with the lowest CO_2 emissions per passenger-kilometer. China is distinguished for its most efficient energy consumption per passenger-kilometre (67 KJ/passenger-km), in their high-speed rail lines. India is highlighted for having the lowest CO_2 emissions produced by passenger-kilometer (10 g of CO_2/passenger-kilometer), and the most efficient consumes of energy per tonne of merchandise transported, with 102 KJ/tonne-km. Finally, Russia stands for the country with the lowest emissions of CO_2 produced in goods transportation (9 g of CO_2/ tonne-km).

The International Energy Agency considers that the railway is one of the least pollutants modes of transportation, once that when compared with other transportation means, the railway tends to produce significantly less emissions of CO_2, especially when compared with private cars (Profillidis et al., 2014). IEA explains this due to a high percentage of electric railway systems, counting for more than 85% of the railway operations. Nevertheless, even if the railway mode of transportation presents to be less pollutant than the remaining modes, it does not mean that its negative impacts should not be taken into consideration. Moreover, it is to be noted that the dominance of the railway in terms of emissions produced is

related to the type of locomotives built and not to the operations management. In other words, the rolling stock rotation planning problem usually optimizes the costs regarding railway operations, which is not the case for the emissions produced. IEA states that railway transportation is less pollutant due to the railway network and not the operations management.

2.7 Summary in contribution

As mentioned, even if the hypergraph model is the one that leads to higher computational times and is more complex to implement, it allows us to better understand the flows within the rotations, and it is more integrated in the sense that allows us to lead with complex requirements. Moreover, in our version of the model, the implementation of the emissions of CO_2 equivalent produced is essential, hence we want to implement it in a more integrated model. For these reasons, we will choose the hypergraph model to solve our version of the RSRP further presented in Chapter 3.

When it comes to our contribution, to the best of our knowledge, there are no models implementing the sustainability feature in the literature. Thus, we will integrate it into our model through the implementation of the emissions of CO_2 equivalent produced throughout the operations related to the rolling stock rotation. This will allow us to oppose the solutions obtained with the cost-minimization approach and the emissions minimization.

Chapter 3

The rolling stock rotation planning problem

3.1 Definition

The rolling stock rotation planning problem is one of the most complex problems in railway planning (Borndörfer et al., 2016). This section aims to properly define the RSRP and describe all mathematical structures implemented in our version of the model.

The RSRP aims to cover all timetable trips, respecting the necessary requirements in order to properly operate the trains. A trip $t \in T$ belongs to the timetable trips planned in the standard week and has departure and arrival times, in its respective departure and arrival stations $s \in S$. Let F be the set of the fleet, also referred to as wagons, locomotives, or vehicles, and C as the set of the compositions that are a group of vehicles $f \in F$. Take V as the set of vertexes, therefore referred to as nodes. The nodes represent the departure and arrival points of the vehicles that operate the trip t. A standard arc $a \in A$ is an arc that connects two nodes. A standard arc $a = (u, v) \in A$ operates trip t if and only if $u \in V$ express the departure node of trip t and $v \in V$ stands for the arrival node for that same trip t. A hyperarc $h \in H$ covers trip t if each standard arc $a \in h \subseteq A$, operates trip t. The set of all hyperarcs h that covers trip t is defined as trip hyperarcs $h \in H(t)$, and the set of all hyperarcs connecting two trips is denoted by connection hyperarcs $h \in H(c)$. Additionally, the set of hyperarcs h that enters and exits a node v is denoted as $H(v)^{in} := \{h \in H | \exists a \in h : a = (u, v)\}$, and $H(v)^{out} := \{h \in H | \exists a \in h : a = (v, w)\}$, respectively, and considering nodes $u, v, w \in V$. Finally, the hypergraph model is denoted by $G = (V,\ A,\ H)$.

To solve the rolling stock rotation planning problem the constraints regarding the trips and stations must be well-known and be a part of the model. In the further sections, it will be explained all the constraints taken into consideration for the subsequent application of the hypergraph model and its respective costs. Note that the maintenance requirement was left aside from our model. As mentioned in Section 2.3, the implementation of the maintenance constraints is more complex for flow-based models, therefore, we focused our analysis on other requirements.

3.2 Indexes

In order to build the model, firstly we have to define the indexes used for the parameters and variables considered. These sets of indexes are described in Table 3.1.

Table 3.1: Indexes.

Set	Range	Description
F	$f = 1, \ldots, n_f$	Indices related to the vehicles;
S	$s = 1, \ldots, n_s$	Indices related to the stations;
T	$t = 1, \ldots, n_t$	Indices related to the trips;
V	$v = 1, \ldots, n_v$	Indices related to the nodes;
C	$c = 1, \ldots, n_c$	Indices related to the vehicles compositions;
H	$h = 1, \ldots, n_h$	Indices related to the hyperarcs.

3.3 Parameters

The parameters used in our version of the RSRP model are detailed and described in the following tables, presented below. Table 3.2 contains the parameters related to the stations, Table 3.3 describes the vehicles parameters, and Table 3.4 details the trip parameters.

Table 3.2: Station parameters' description.

Station parameters	
$dist_{s_k s_l} \in \mathbb{Q}^+$	Distance (kilometers) between station s_k and s_l, using the railway path, where s_k and s_l are any stations belonging to the set of stations;
$sLeng_s \in \mathbb{Q}^+$	Length of the platform of the s-th station;
$turn_s \in \{0, 1\}$	Binary parameter that is defined 1 when station s is capable of performing turn operations, and 0 otherwise;
$tTurn_s \in \mathbb{Q}^+$	Time required to operate the turn operation in the s-th station, if the turn exists into the solution;
$cTurn_s \in \mathbb{Q}^+$	Cost associated to the turn operation in the s-th station, if the turn exists into the solution;
$shunt_s \in \{0, 1\}$	Binary parameter that is defined 1 when station s is capable of performing shunting operations, and 0 otherwise;
$tShunt_s \in \mathbb{Q}^+$	Time required to operate the shunting operation in the s-th station, if the shunting exists into the solution;
$cShunt_s \in \mathbb{Q}^+$	Cost associated to the shunting operation in the s-th station, if the shunting exists into the solution;
$reqTime_s \in \mathbb{N}$	Required time that a train has to remain in the terminal station between operating two trips.

Table 3.3: Vehicle parameters' description.

Vehicle parameters	
$cap_f \in \mathbb{N}$	Passenger capacity of the f-th vehicle;
$fLeng_f \in \mathbb{N}$	Length (meters) of the f-th vehicle;
$util_f \in \mathbb{Q}^+$	Utilization cost of the f-th vehicle. The utilization cost is considered a fixed cost only if the vehicle f is in the solution of the model;
$trav_f \in \mathbb{Q}^+$	Travel cost of one kilometer traveled by the f-th vehicle. This cost is variable and increases as the number of kilometers travelled by vehicle f also increases;
$emis_f \in \mathbb{Q}^+$	Emissions of CO_2 equivalent (kilograms) produced and emitted by one kilometer travelled by the f-th vehicle. The emissions are variable, thus the value will increase with the increase of the kilometers travelled by vehicle f.

Table 3.4: Trip parameters' description.

Trip parameters	
$d_t \in \mathbb{N}$	Passenger demand for the t-th trip;
$tLeng_t \in \mathbb{N}$	Maximum length that a train composition can take for the t-th trip;
$cRegBonif \in \mathbb{Q}^+$	Cost regularity bonification. It decreases the operational costs for regular trips, in order to enhance the same composition train to perform regular trips;
$cDhPen \in \mathbb{Q}^+$	Cost deadhead trips penalty. It increases the operational costs for deadhead trips, in order to disfavour the existence of deadhead trips in the solution;
$eDhPen \in \mathbb{Q}^+$	Emissions deadhead trips penalty. It increases the emissions produced for deadhead trips, in order to disfavour the existence of deadhead trips in the solution.

3.4 Periodic Timetable

When it comes to managing the RSRP, the periodicity considered can be daily or weekly. Once that in most cases, the trips repeat themselves weekly, the weekly periodicity is considered. In Figure 3.1, the repeatability is represented through a loop that connects the end of the week with the beginning of the following week (Borndörfer et al., 2016). Once the trips planned for a certain week are the same every week, one can say that the end of the standard week n connects appropriately with the beginning of week $n + 1$.

One relevant limitation is that the cyclic hypergraph model does not cover the planning of weeks that do not have the same trips planned as the rest of the year. This can happen during holidays, like Christmas, Summer, or Easter, for example. In these weeks, the planned trips are not the same, which means that the optimization of the hypergraph model will not consider all the trips planned for these holidays. This implies that the model needs to be concerted for the cases in which the planned trips are not the same. Furthermore, by considering the timetable cyclic, we cannot satisfy exceptional trips planned on an operational level. These cases are acyclic, so they are not considered in the solution.

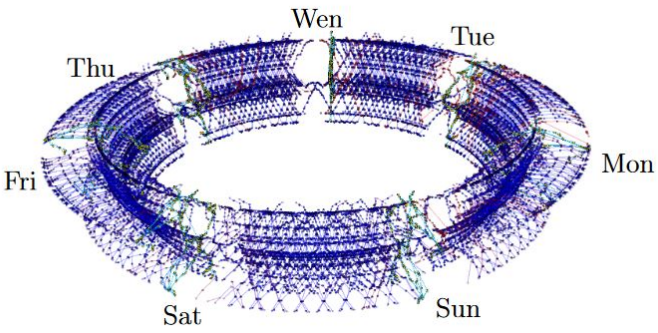


Figure 3.1: Periodicity of the timetable. Source: Borndörfer et al. (2016).

3.5 Constraints

3.5.1 Vehicle compositions

A vehicle $f \in F$ is a single unit fleet that allows the passengers to travel a trip $t \in T$. A set of grouped vehicles is called a vehicle composition c, that belongs to the set of compositions C. It is to be noted, that a singular vehicle is also considered a composition, but in that case, it is a composition composed by a singular vehicle. As we will explore further across this chapter, not all compositions $c \in C$ are feasible for all trips $t \in T$. The set of the feasible compositions is denoted by $C(t) \subseteq C$, and they allow to operate all the timetable $t \in T$.

Figure 3.2 shows how the hypergraph model works in order to solve the RSRP and how the vehicles and compositions interact within the model. In the figure, there are represented three trips $t_1, t_2, t_3 \in T$, all bellowing to the timetable table to be operated. A hyperarc $h \in H$ is a connection between two nodes, we can separate them into two categories. The first set is the set of the trip hyperarc $h \in H(t) \subseteq H$. The hyperarcs within this set have the purpose of covering a trip $t \in T$, i.e., they start at the departure node of the trip t of a composition $c \in C(t)$ and end at the arrival node of that same trip and composition. The second set of hyperarcs is denoted the connection hyperarcs $h \in H(c) \subseteq H$. These hyperarcs connect the trips among them. They can represent the deadhead trip that a composition c may have to travel to join two trips or represent the shunting operations, or turn operations. The two categories of hyperarcs represent the overall set of hyperarcs, $H(t) \cup H(c) = H$. Moreover, a hyperarc h cannot belong to the set of trip hyperarcs $H(t)$ and the set of connection hyperarcs $H(c)$, thus these two categories are complementary $H(t) \cap H(c) = \emptyset$.

The hypergraph model uses flow conservation to solve the RSRP. To ensure that, we need to consider the outgoing and ingoing hyperarcs for each node $v \in V$. Take Figure 3.2 as an example, representing two rotations in a hypergraph model $G = (V, A, H)$, with $|T| = 3$ and $|V| = 28$. For hyperarc h_4, we have two nodes on the left, representing the departure of the compositions represented by h_4, and two nodes representing the arrival. As we can see, all nodes conserve the flow. For example, the let top node has the connection hyperarc h_{13} ingoing and the trip hyperarc h_4 outgoing. For the example provided, both these hyperarcs belong to the solution, and thus if hyperarc h_{13} is in the solution, hyperarc

h_4 will also be, due to the conservation flow.

The feasible compositions $c \in C(t)$ offer the model a large number of degrees of freedom for four main reasons. The first one is due to the shunting operations. Within the RSRP, several shunting operations can be done, and the more vehicles a feasible composition has, the more shunting operations that composition can have. To limit the degrees of freedom for this characteristic, we only allow the model to perform one shunting operation per feasible composition, i.e., one composition cannot decouple in more than two sub-compositions, and three compositions cannot couple into one singular composition in one operation. This example is presented in Figure 3.2 with hyperarc $h_{11} \in H$. The second reason is the orientation of the vehicles. Consider a composition $c \in C$ formed by the vehicles {Blue, Blue}. This composition in the model corresponds to four possible feasible compositions that are represented in Figure 3.2 by hyperarcs $h_3, h_4, h_5, h_6 \in H(t)$. This characteristic will be further explained in Section 3.5.4. The third reason is the position of the vehicles within the composition. Our version of the RSRP considers the position of the vehicle in a composition c, but does not distinguish particular vehicles; it only considers the type of vehicle position. For example, in our model, the composition {Blue_1, Yellow, Blue_2} is the same as {Blue_2, Yellow, Blue_1}, as we only consider the type of vehicles without specifying each particular vehicle. Returning to the example, the composition considered is {Blue, Yellow, Blue}. Finally, the fourth reason is the size of a feasible composition $c \in C(t)$. Its size varies based on several factors, such as the demand per trip $t \in T$, the capacity of the vehicles cap_f, and the maximum length, $leng_t$, that a composition c can take for a trip t. If a composition c composed of two vehicles can satisfy the demand and the maximum length for a trip t is six, there are a significant number of possible feasible compositions, considering the order and position of the vehicles.

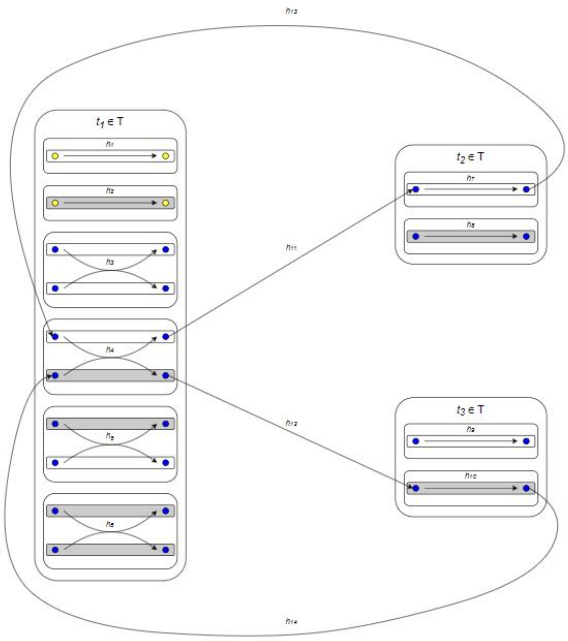


Figure 3.2: Hypergraph model representation.

There is an associated cost with the vehicle utilization, $util_f$, that is added to the operations if the vehicle f, represented by hyperarcs, is to be selected in the solution. As mentioned in Table 3.3, there are two types of costs regarding the rolling stocks utilization, that need to be taken into consideration to solve the RSRP: fixed and variable costs. These costs are essential once they are mandatory to operate the timetable in the standard week, at the minimal operational costs possible.

As we discussed, the connection hyperarcs $h \in H(c)$ can represent deadhead trips, if the arrival station of trip t_i is not the same as the departure station of trip t_j, shunting operations, or turn operations. These three operations incur additional costs for the model. From these three operations, only the turn operation can be mandatory to perform, if the arrival station of trip t_i does not allow the turn. Regarding the remaining operations, the model will tend to not include them in the solution, once they are associated with non-necessary costs. As a matter of fact, the deadhead trips are not only expensive to the railway companies, but they are an opportunity lost to use the trip in order to satisfy the demand. For this reason, a deadhead trip penalization is included in the model, $cDhPen$, to ensure that this type of trip is discouraged by the model, optimizing the trips that directly serve the passengers.

3.5.2 Capacity constraint

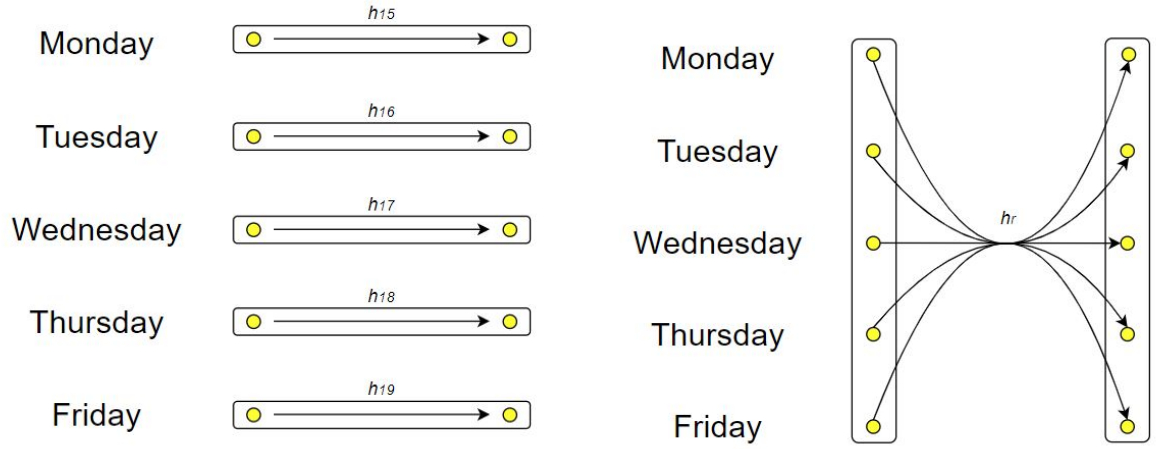
The capacity constraint lies in the fact that each trip $t \in T$ is created knowing the forecasted demand d_t, which is the total number of passengers that will travel on trip t. So, for each trip t, the demand must be satisfied, i.e., the sum of the capacity vehicles, cap_f, for a composition $c \in C$, operating a certain trip t needs to be able to satisfy the demand.

3.5.3 Regularity constraint

The RSRP is considered a cyclic problem (see Section 3.4). This means that the timetable set of trips is defined over a standard week and that the solution is applicable to the following weeks. However, we can find the presence of a cyclic timetable within the week in the analysis. This usually happens in the work days in which part of the trips are the same for the five days of the working week. These trips occur every day at the same departure and arrival times, usually with the same demand, and pass through the same stations in the same direction. Thus, this set of trips is called regular trips. Regular trips have the advantage of creating comfort for the passengers, once they are aware of the expectations in terms of the train´s composition. For instance, they already know where they prefer to wait on the station platform, or where on the platform they will have direct access to the train or not. This comfort needs to be registered in the model, and for that, a bonification is included in our version of the RSRP.

Figure 3.3 (a) presents the set of trips having the same departure and arrival stations and times and the same train compositions. We presented an example considering only the working days, but regular trips are any trips with the same set of characteristics, already defined, regardless of the day of the week. Therefore, these trips are referred to as regular, and we can create a regular hyperarc, h_r that has the total operational cost as the sum of all operational costs involving hyperarcs $h_i, i = 15, 19$ minus

a reduction, the regularity bonification, $cRegBonif$. The generation of h_r is displayed in Figure 3.3 (b) and as we can see, this new hyperarc represents the same as the ones in Figure 3.3 (a), excluding their costs. Section 4.1 will discuss better the objective functions, but we can already state that when minimizing the costs, the regularity hyperarcs will be chosen over the normal trip hyperarcs. In this way, the benefits of the regular trips will be encouraged in our RSRP solution.



(a) Separate hyperarcs

(b) Joined through hyperarc h_r

Figure 3.3: Regular trips.

3.5.4 Orientation constraint

The orientation of the vehicles was also considered, which means that for each composition of vehicles considering the orientation of a trip $t \in T$, there are $2^{\text{number of vehicles}}$ compositions considering the orientation. It can be seen in Figure 3.4 that for a composition $c \in C$ composed by two vehicles, the number of possible compositions is sixteen, considering the orientation and position of the vehicles. In 2021, Borndörfer et al. designated the two possible orientations of the vehicles in the following way. If the first class of the vehicle is in the front of the vehicle, considering its direction, the orientation is called $tick$. However, if the first class finds itself on the back of the vehicle, once again considering the trip direction, then the orientation is said to be $tock$. By the same logic, we call a vehicle´s orientation $neutral$ if this one does not have first or second class, as is the case for the subway fleet, i.e., the $neutral$ orientation is equivalent to not considering the orientation. However, we make a point about this distinction, because our model is prepared to consider or not the orientation of the vehicles.

However, not all the compositions considering the orientation are feasible arcs for the solution. If a trip does not specify anything, it is considered that all compositions are admissible, but if there is some specification such as considering only the compositions with the fleet in $tick$ orientation, the remaining compositions are not feasible. This can happen for night trips in which the vehicles with first class can have beds and, in this case, typically the railway companies prefer to have the fleet with first class in front at the beginning of the train, eliminating the compositions that had the fleet with second class at the front. However, for our version of the RSRP, this is not necessary, even if we decided to implement

the possibility of restricting the compositions according to their orientations.

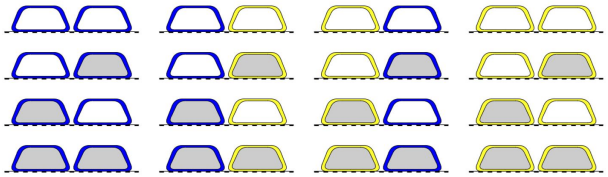


Figure 3.4: Possible compositions considering orientation and position for two vehicles.

In Figure 3.4, there are represented all sixteen possible compositions regarding the orientations and positions of two vehicles blue and yellow. Consider that the vehicles with the white background are the ones with the orientation $tick$ and the ones with the grey background with orientation $tock$. The driving direction is from left to right.

3.5.5 Station constraint

Each station $s \in S$ has a length of its platform, $sLeng_s$, and it is noted that if a trip $t \in T$ passes through station s, then station s belongs to $S(t)$. For that reason, some compositions, $c \in C$ cannot be considered, because they are too long to fit in the platform. Moreover, once each trip t is associated with one line, one can calculate the maximum length that a composition c serving a trip t can have, i.e., the feasible compositions $c \in C(t)$. For that, first, we have to define the length of a trip t which is defined as the minimum length of station $s \in S(t)$.

Furthermore, terminal stations required other conditions, such as the possibility to turn the train or not. Consider that terminal stations are stations s belonging to the set $S(d)$. When it comes to terminal stations $s \in S(d)$, i.e., stations in which the trip $t \in T$ starts or ends, the station can force the train to switch its orientation or not. This can lead to a change in the composition and orientation of the train, which needs to be taken into account in the RSRP.

Figure 3.5, represents two possible situations for terminal stations. On the left of Figure 3.5, the train will need to change its orientation of the composition, once the fleet orientation arriving at the departure station will not be the same as the one when departing, once no turnaround is allowed. On the other hand, in the right image in Figure 3.5, the turnaround is available which means that the train will perform its next trip with the same orientation. These situations are not mutually exclusive, once it is possible to have a terminal station s with both types of tracks, i.e., in which the train is allowed to turn of not. Hence, the variable $turn_s$ is defined according to the terminal station tracks.

The third requirement considered for the stations is once again, regarding the terminal stations $s \in S(d)$, and it is related to the shunting operations. The shunting operations were already explained in more detail in Section 3.5.1. These operations are time-consuming and they only can be performed in terminal stations, although this operation is not allowed to be performed in all terminal stations. Thus, the set of terminal stations allowing shunting operations needs to be defined accordingly with the structure of the terminal stations.

(a) Tracks not allowing the train to turn. (b) Tracks allowing the train to turn.

Figure 3.5: Tracks in terminal stations.

Moreover, some terminal stations can have a park in which the trains are allowed to stay from one day to another (or more time), while they wait for their next trip. In these cases, the limitation of the park needs to be taken into consideration. However, for our version of the RSRP, it was assumed that the park has unlimited capacity.

3.6 Emissions of CO_2 equivalent

As we mentioned in Section 2.6, the literature does not present models with the integration of CO_2 emissions, hindering the understanding of how emissions can be assessed and allocated within the operations, as we did for the costs, throughout the current chapter.

Once we aim to formulate a model that minimizes the emissions produced by the operations related to the rolling stock rotation, we decided to evaluate the emissions through a parameter that is associated with the travel distance traveled by the train compositions, $emis_f$. This parameter is closely correlated to the energetic efficiency of each vehicle, and it is set to be different for each type of vehicle. For instance, it is expected that electric vehicles to have a lower coefficient of emissions of CO_2 equivalent produced than diesel vehicles. As well as it is expected that the most recent is the fleet, the most energetically efficient it is.

Pollution is produced within several operations, and it can be estimated in different ways within the railway system. For instance, it can be evaluated in kilo Jules per passenger-kilometer or in kilograms of CO_2 equivalent emitted per passenger-km. Our parameter $emiss_f$ integrates the pollution in kilograms of CO_2 equivalent produced by kilometer. So, this implies that the emissions registered for the operations regarding the RSRP need to be converted into this unit. Once again, due to the lack of literature integrating this feature, and since this dissertation does not count on the collaboration of a railway company, we are significantly limited to the poor detailed and itemized available data for the integration of this feature in our model.

In alignment with the deadhead trip cost penalty, we implemented the deadhead trip emissions penalty, $eDhPen$. This parameter has the same goal as the cost penalty, which is the disadvantage of the model in choosing solutions involving deadhead trips, or at least reducing them. Similarly, as the

cost penalty that was created to represent the cost of opportunity for performing an unnecessary trip (see Section 3.5.1), the emissions penalty serves to specify to the model that the empty trips do not have the same value as trips that serve passengers. Hence, they should be penalized. Nevertheless, we did not include a regularity bonification through the emissions, once that in terms of emissions, it is not important to compositions of regular trips have the same compositions. Furthermore, a rolling stock rotation solution can be less pollutant by switching compositions for regular trips.

Chapter 4

Hypergraph Model

4.1 Mixed Integer Programming Formulation

In this section, we present the mixed integer programming formulation implemented to solve the rolling stock rotation planning problem. The MIP uses the hypergraph model and thus is focused on the flow conservation for each node $v \in V$, implying that $v^{out} = v^{in}$. To remember, each trip $t \in T$ is operated by a feasible train composition $c \in C(t)$, and all trips are connected among them through the connection hyperarcs $h \in H(c)$.

A binary decision variable x_h is defined for each hyperarc $h \in H$. For $x_h = 1$, indicates that the hyperarc belongs to the RSRP's solution, and thus it has its cost and emissions produced associated. The solution of current RSRP is solved through a mixed integer programming formulation of the hypergraph model, as it is presented above.

$$\text{minimize} \quad \sum_{h \in H} c_h x_h \tag{4.1}$$

$$\text{minimize} \quad \sum_{h \in H} e_h x_h \tag{4.2}$$

$$\text{s.t.} \quad \sum_{h \in H(t)} x_h = 1 \qquad \forall t \in T, \tag{4.3}$$

$$\sum_{h \in H(v)^{in}} x_h = \sum_{h \in H(v)^{out}} x_h \qquad \forall v \in V, \tag{4.4}$$

$$x_h \in \{0, 1\} \qquad \forall h \in H. \tag{4.5}$$

This version of the RSRP is presented with two linear objective functions. The first one is Objective Function 4.1 and it aims to minimize the total cost of the RSRP's operations. On the other hand, the second linear objective function aims to minimize the emissions of CO_2 produced throughout the operations of the RSRP (Objective Function 4.2). Equation 4.3 indicated that for each trip $t \in T$, one hyperarc in H(t) covers the trips, i.e., one hyperarc h is associated with one and only one trip t, ensuring

that each trip is considered in the hypergraph model. Additionally, the flow needs to be conserved, and this is achieved through Equation 4.4. This equation ensures that for each node $v \in V$ that is contained in the solution, i.e., for each node v in which the rotation is considered, it is ensured that the number of outgoing hyperarcs is the same as the ingoing hyperarcs. Finally, the Equation 4.5 guarantees the binarity of the decision variable x_h that represents the hyperarc h. The binary decision variable will be one of the hyperarc h belonging to the RSRP's solution and zero otherwise.

4.2 Generation of hyperarcs

The hypergraph model relies on the creation of hyperarcs. As mentioned before, a hyperarc $h \in H$ can be divided into two groups, the trip hyperarcs $h \in H(t)$ and the connection hyperarcs $h \in H(c)$. The first subset represents the hyperarcs that cover the trips, while the second subset contains the hyperarcs connecting the trip hyperarcs. So, in order to generate the trip hyperarcs, we first have to generate the feasible compositions for a trip $t \in T$. The feasible composition algorithm is displayed in Algorithm 1, and it takes as an argument the set of all possible compositions (see Section 3.5.1). The set of possible compositions is a priori defined with a maximum number of vehicles allowed for each composition $c \in C$. The algorithm verifies for each trip $t \in T$ if the compositions $c \in C$ comply with the trip conditions, such as the passenger demand of trip t, that composition c needs to satisfy and the maximum length that trip t allows composition c to have. Therefore, Algorithm 1 verifies if composition c is able to operate in the trip line. In this scenario (that will be explained in Chapter 5), the only feasible compositions for the red line are the ones with 5000 series vehicles. Finally, for every composition c fulfilling the requirements, it can be added to the set of feasible compositions that trip t can have, $C(t)$.

Algorithm 1 Generation of feasible compositions.

```
for all t in T do
    for all c in C do
        if c.Length() ≤ tLeng_t & c.Capacity() ≥ d_t then
            if t.getRouteId() == "Red" then
                int verification = 0
                for all f in c.getComposition() do
                    if f == "5000 series" then
                        verification++
                    end if
                end for
                if verification == c.getSize() then
                    t → addFeasibleComposition(c);
                end if
            end if
        end if
    end for
end for
```

Now that we have defined how to choose the feasible compositions $c \in C(t)$ for each trip $t \in T$, it is possible to generate the trip hyperarcs $h \in H(t)$, as it is displayed in Algorithm 2. Throughout the generation of the trip hyperarcs h, the nodes $v \in V$ will also be created, once that for each vehicle $f \in F$,

of belonging to each feasible composition c, two nodes are created. The first node is referred to as the outgoing node, v^{out}, and the second as the ongoing node, v^{in}, referencing the departure and arrival of the vehicle f. Moreover, for each feasible composition c, we create trip hyperarcs h, that indicate the displacement of composition c between the departure and the arrival stations. Finally, we defined a set of outgoing and a set of ingoing nodes (v^{out}, and v^{in}, respectively) that are allocated to each trip t. This will be essential to connect the trips t among them, and therefore, to generate the connection hyperarcs $h \in H(c)$.

Algorithm 2 Generation of trip hyperarcs and vertices.

```
for all t in T do
    for all c in C(t) do
        for all vehicles in c do
            create v^out
            create v^in
        end for
        Create trip hyperarc h_t
        Add h_t to v^out
        Add h_t to v^in
    end for
end for
```

Algorithm 3, generates the connection hyperarcs $h \in H(c)$ that connects the trip hyperarcs $h \in H(t)$. To simplify, the connecting hyperarcs will be called h_c, the feasible compositions for trip i are represented by h_i and the ones for trip j are referred to as h_j. The presented algorithm generates all the connection hyperarcs h_c that have the same position order of the vehicles $f \in F$ within the compositions $c \in C(t)$, i.e., Algorithm 3 only generates hyperarcs for stations in which it is allowed the turn operation, as it can be seen in condition (3). As mentioned, Algorithm 3, generates the connection hyperarcs between trips with the same composition orientation. Therefore, to link trips that do not need shunting operations, the condition (4) is considered. It only fails to know if the connection is feasible in terms of the time required to perform all the required operations. For this purpose the function $hasTimeToTravel(h_i, h_j)$ verifies if it is possible to travel between the arrival station of h_i and the departure station of h_j, considering the respective arrival and departure times (condition (5)). It also takes into account the required time for h_i to remain in its arrival station, and the turn and shunting times if the stations allowed it. Figure 4.1 represents the generation of a hyperarc h_c between trips with the same compositions, considering that the time requirements are respected. It is to be noted that the creation of h'_c is possible, when h_i changes to h_j and the other way around, due to the cycles (1) and (2).

When it comes to the shunting operations, Algorithm 3 was created to only allow one possible shunting operation at a time. In the first place, we can only consider the shunting operations if the terminal stations allow it (condition (6)). Then, we created the following functions to consider the shunting operations: *doesComp_aDecouplesIntoComp_b_Begin(h_a, h_b)*, and *doesComp_aDecouplesIntoComp_b_End(h_a, h_b)*. These functions take as argument two hyperarcs and verify if the first one can be decoupled into the second one, but only consider the beginning of composition h_a or the end of composition h_b. Take as an example the hyperarcs h_i and h_j: if h_j's composition is the same as the beginning of h_i's composition,

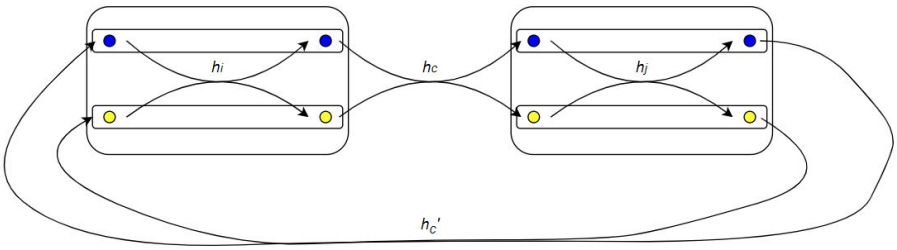


Figure 4.1: Diagram representing the generation of h_c between h_i and h_j, considering equal compositions.

the function *doesComp_aDecouplesIntoComp_b_Begin(h_i, h_j)* is true. It only lacks to know if it is possible to connect h_i with h_j, in order to create h_c, as it is seen in condition (8). Moreover, it can be also possible to connect h_j with h_j, if h_j has time for it, creating h'_c, as it is shown in Figure 4.2. The same process is applied to the *doesComp_aDecouplesIntoComp_b_End(h_a, h_b)* function.

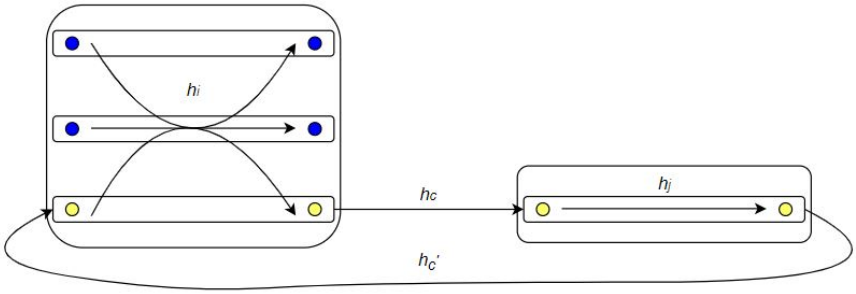


Figure 4.2: Diagram representing the generation of h_c between h_i and h_j, considering shunting operations.

Despite Algorithm 3 generating the connection hyperarcs, it only generates one type of them. When a trip $t_i \in T$, arrives at the departure station, the train composition can turn, keeping its compositions, or it can be forced to not turn, due to the layout of the station. For the arrival stations, in which the train composition is forced to not turn, it implies that the composition ongoing for the next trip $t_j \in T$ is the opposite reverse of composition for trip t_i. So, we developed another algorithm that is very similar to Algorithm 3, that generates the hyperarcs considering these situations. The differences between these two algorithms lie in the fact that the conditions instead of checking if the composition order is the same, verify if the composition orders between h_i and h_j are reversed. Thus, the generation of these connection hyperarcs h_c is quite similar.

Algorithm 3 Generation of connection hyperarcs

```
for all h_i in H(t) do                                                          ▷ (1)
    for all h_j in H(t) do                                                      ▷ (2)
        if h_i.arrivalStation.doesTurn() == 1 and h_j.departureStation.doesTurn() == 1 then   ▷ (3)
            if h_i.getCompositionId() == h_j.getCompositionId() then            ▷ (4)
                if hasTimeToTravel(h_i, h_j) == 1 then                          ▷ (5)
                    Create connection hyperarc h_c
                    Add h_c to h_j.get_v^out
                    Add h_c to h_i.get_v^in
                end if
            end if
        else if h_i.arrivalStation.Shunt() == 1 or h_j.departureStation.Shunt() == 1 then   ▷ (6)
            if doesComp_aDecouplesIntoComp_b_Begin(h_i.getComp(), h_j.getComp()) then
                if hasTimeToTravel(h_i, h_j) == 1 then                          ▷ (7)
                    Create connection hyperarc h_c
                    Add h_c to h_j.get_v^out
                    Add h_c to h_i.get_v^in
                end if
                if hasTimeToTravel(h_j, h_i) == 1 then                          ▷ (8)
                    Create connection hyperarc h_c
                    Add h_c to h_i.get_v^out
                    Add h_c to h_j.get_v^in
                end if
            end if
            if doesComp_aDecouplesIntoComp_b_End(h_i.getComp(), h_j.getComp()) then
                if hasTimeToTravel(h_i, h_j) == 1 then
                    Create connection hyperarc h_c
                    Add h_c to h_j.get_v^out
                    Add h_c to h_i.get_v^in
                end if
                if hasTimeToTravel(h_j, h_i) == 1 then
                    Create connection hyperarc h_c
                    Add h_c to h_i.get_v^out
                    Add h_c to h_j.get_v^in
                end if
            end if
        end if
    end for
end for
```

4.3 Multi-objective approach

4.3.1 Multi-objective methods

In order to solve the RSRP through the mixed programming formulation, the multiplicity of the linear objective functions must be taken into account. Because of that, it is expected that no single optimal solution will be found that optimizes both objective functions 4.1 and 4.2. The multiplicity of the objective functions causes the loss of the notion of the optimal solution and the notion of the Pareto optimality is introduced (Mavrotas, 2009). The set of all Pareto optimal solutions, also named Pareto non-dominant solutions composes the Pareto front. These solutions cannot improve one objective function without jeopardizing at least one of the other objective functions. In this context, these solutions are the ones that cannot minimize even more the costs (Equation 4.1), without increasing the emissions (Equation 4.2) or the other way around. Within the Pareto front lies all the efficient solutions. Consider the following example to define the efficient solutions: without loss of generality, it is assumed that all the n objective functions f_i, i = 1, ..., n, are for minimization. A feasible solution x for the multi-objective mathematical problem (MOMP) is efficient if no other feasible solution x' such that $f_i(x') \leq f_i(x)$, for i = 1, ..., n.

For the MOMP, it fits the decision-maker (DM) to choose the most preferred solutions. There are categories enhancing how much the decision maker is involved in expressing his preferences (Hwang and Masud, 1979): the a priori methods, the interactive methods, and the a posteriori or generation methods. In a priori methods, the DM indicates his preferences before the solution process through the setting of weights to the objective functions, for example. The significant downside of this method is that is very hard to quantify weights to the objective functions precisely before even starting the process. The interactive methods mix steps calculating the solutions with discussion with the DM until there is a convergence of the most preferred solution. This method needs some time and consecutive iterations, in which the decision-maker conducts the process accordingly with his answers. The disadvantage is that the DM never sees the Pareto front which leads to a lack of knowledge of the whole paradigm. This can lead the decision-maker to choose a preferred solution according to his preferences in comparison with other solutions, ignoring some possible solutions not found that he could prefer. Finally, in the generation methods, the Pareto front is first calculated and then, the DM is involved in the process so that he can choose the preferred solutions, which is a significant advantage. Once he is informed of the whole paradigm, he can not only be required in the final step of this method (the efficient solutions need to be calculated first and the decision-maker is not required for that) and when he is required, the DM is aware of all the possible efficient solutions, which leads to a most conscious final decision about his preferences. Due to a higher time-consuming work, the a posteriori method is sometimes non-preferred in opposite to the other methods.

Usually, the weighting and ε-constraint methods are preferred when solving a MOMP, once they are allowed to provide a significant subset of the Pareto front, which generally is appropriate to choose the preferred solutions. Nevertheless, the ε-constraint presents several significant advantages in comparison with the weighting method, as Mavrotas, discussed in 2009.

a) When it comes to solving linear problems, the ε-constraint method is able to alter the original feasible region and to produce non-extreme efficient solutions, while the weighting method is applied to the original feasible region and is only able to provide efficient extreme solutions. Thus, in almost every new iteration, with the ε-constraint method it is possible to find a different efficient solution, whereas the weighting method can have several redundant iterations, once a lot of matches of weights can return the same efficient extreme solution. Hence, the set of efficient solutions found in the ε-constraint method is wider than using the weighting method.

b) For multi-objective integer and mixed integer programming problems, the ε-constraint method can return non-supported efficient solutions, while the weighting method cannot (Steuer, 1986; Miettinen, 1998).

c) In order to use the weighting method, it is necessary to scale the objective functions, while it is not necessary for the ε-constraint method.

d) With the ε-constraint method it is possible to control the number of efficient solutions calculated by adjusting the number of grid points for the several objective function ranges. The weighting method does not allow the achievement of a desirable number of efficient solutions so easily, as can be proven by confronting point a).

Therefore, to solve the multi-objective function for the RSRP proposed, the ε-constraint method is chosen for the reasons explained above.

4.3.2 Augmented ε-constraint method

The ε-constraint method has some flaws when it comes to building the Pareto front. The solutions found can be dominated by other solutions non-dominated. To avoid this issue, Mavrotas (2009) suggested the augmented ε-constraint method to solve this issue. The method was later improved by the same author in Mavrotas and Florios (2013). The method suggested helps build the Pareto front ensuring that only non-dominant solutions are found, which increases the reliability of the method. In order to perform that, as opposed to other ε-constraint methods that only use the objective function of $z_1(x)$, the augmented ε-constraint method through the utilization of surplus variables in the objective function, a component that is related to the remaining objective functions is included. In this section, the version adopted by Almeida et al. in 2023 of the augmented ε-constraint method is used and described.

Ideal and approximation nadir points

The first step is to compute the ideal point for all objective functions $z_i, i = 1, ..., n_z$, using Equation 4.6. As mentioned, using the augmented ε-constraint method, when calculating the objective function it is necessary to consider a component related to the remaining ones. Hence, the first component corresponds to the minimization of one of the objective functions $z_i(x), i = 1..., n_z$, and the second component is the sum of all the remaining objective functions $z_j(x), j = 1, ..., n_z, j \neq i$, multiplied by 10^{-E},

which is an adequate number that guarantees that, independently the value of the second component, called the perturbation, the value of $z_i^{\rho_i^*}$ is the same as we would only calculate the minimization of $z_i(x), i = 1, ..., n_z$, as presented in Equation 4.7.

$$z_i^{\rho_i^*} = \min_{x \in X} \left\{ z_i(x) + (10^{-E}) \sum_{j=1, i \neq j}^{n_z} z_j(x) \right\}, \quad i = 1, \ldots, n_z \tag{4.6}$$

$$z_i^* = \min_{x \in X} z_i(x), \quad i = 1, \ldots, n_z \tag{4.7}$$

The perturbation of $z_i(x), i = 1, ..., n_z$ is defined as ρ_i and is descripted in Equation 4.8, where 0 $\leq \rho_i <$ 1.

$$\rho_i = (10^{-E}) \sum_{j=1, i \neq j}^{n_z} z_j(x) \quad i = 1, \ldots, n_z \tag{4.8}$$

Once that $z_i(x) \in \mathbb{N}$, then ρ_i does not interfere with the value of $z_i(x)$, and it is now possible to define the ideal point for $z_i(x)$ by removing the perturbation from Equation 4.6, as it is described in Equation 4.9.

$$z_i^* = z_i^{\rho_i^*} - \rho_i \quad i = 1, \ldots, n_z \tag{4.9}$$

It is now possible to define all the ideal points for each $z_i(x), i = 1, ..., n_z$. The set of all ideal points is defined as z^{id} and it is described in Equation 4.10.

$$z^{id} = (z_1^*, ..., z_i^*, ..., z_{n_z}^*) \tag{4.10}$$

Now that the ideal points are found for each $z_i(x), i = 1, ..., n_z$, it is possible to find the approximate nadir points for each $z_i(x)$. When minimizing $z_i(x)$ in Equation 4.6, the value value obtained for the objective function $z_j(x), j = 1, ..., n_z, j \neq i$ is denoted z_j^{i*}. So, each scalar component, z_i^{nadir}, is a part of the approximate nadir vector, called z^{nadir}, and it is defined as in Equation 4.11.

$$z_i^{nadir} = \max_{j=1,\ldots,n_z} \left\{ z_i^{j*} \right\}, \quad i = 1, \ldots, n_z \tag{4.11}$$

Once each scalar component is calculated, the approximate nadir vector is defined by the Equation 4.12.

$$z^{nadir} = (z_i^{nadir}, ..., z_i^{nadir}, ..., z_{n_z}^{nadir}) \tag{4.12}$$

Build the Pareto front

At this point, both ideal and approximate nadir points are found for each $z_i(x), i = 1, ..., n_z$. However, to find the remaining non-dominant solutions desired, using the ε-constraint method is required to first calculate the surplus variables denoted by $s_i(x), x \in X$, for each objective function $z_i(x), i = 1, ..., n_z$. The surplus variables are calculated through the ratio of the difference between the value of the objective

function $z_i(x)$ and its correspondent ideal point, z_i^*, by the difference of its nadir point, z_i^{nadir}, and its ideal point, z_i^*, as it is displayed in Equation 4.13. The surplus variables talked values between 0 and 1: $0 \leq s_i(x) \leq 1$.

$$s_i(x) = \frac{z_i(x) - z_i^*}{z_i^{nadir} - z_i^*}, \quad i = 2, \ldots, n_z, \quad x \in X \tag{4.13}$$

Once the surplus variables are defined, it is possible to calculate the objective function denoted by z^{ε}, as it is presented in Equation 4.14. Once again, the objective function is divided into two components, the first one is related to the objective function $z_1(x)$, and the second one is the perturbation. This perturbation is calculated through the sum of all surplus variables, $s_i(x)$ and it ensures that the solutions to be found are non-dominant, as it is pretended. Additionally, the utilization of the surplus variables also guarantees that the value found for the objective function $z_i(x)$ is the same as it would be if we would only minimize the objective function $z_1(x)$ separately, like in the Equation 4.7, for $i = 1$. The necessity of adding the perturbation (Equation 4.16 is because it is necessary to also consider the remaining objective functions $z_j(x), j = 2, ..., n_z$ and that they are subjected to an upper bound denoted ε, as it is displayed in Equation 4.15.

$$z^{\varepsilon} = \max_{x \in X} \left\{ z_1(x) + \left(n_z - \frac{1}{0.9} \sum_{i=2}^{n_z} s_i(x) \right) \right\} \tag{4.14}$$

$$z_j(x) \geq \varepsilon_j \quad j = 2, \ldots, n_z \tag{4.15}$$

The perturbation displayed below in Equation 4.16 is related to the objective function $z_1(x)$ and once the surplus variables take values from 0 to 1, the perturbation will fit between $0 \leq \phi < 1$. So, ϕ will not interfere with the minimization of the objective function $z_1(x)$, as long as the remaining objective functions $z_j(x), j = 2, ..., n_z$ is restricted by the upper bounds vector ε, once that $z_1(x) \in \mathbb{N}$.

$$\phi = (n_z - 0.9)^{-1} \sum_{i=2}^{n_z} s_i(x) \tag{4.16}$$

The upper bounds will decrease over the augmented ε-constraint method and are defined in Equation 4.17.

$$\varepsilon_i = z_i^{nadir} + v_i p_i (z_i^* - z_i^{nadir}), \quad i = 2, \ldots, n_z \tag{4.17}$$

However, the upper bound, ε_i needs to be updated over the process, and for that the first step is to define a parameter k, that is subjected to $\frac{1}{k_i} \in \mathbb{N}$. This is the gap percentage between the ideal, z_i^*, and nadir, z_i^{nadir} points. This percentage is incremented over each iteration for a specific objective function $z_i(x), i = 2, ..., n_z$. In order to keep the count of the incrementations, a vector v of dimensions n_z - 1 is created, so that each component, $v_i, i = 1, ..., n_z - 1$, indicates the number of iterations that ε_i suffered and incrementation through its corresponding gap percentage, k_i. Throughout the iterations, all parameters are constant, except for v_i which is altered across iterations, and finds itself between

$0, ..., 1/p_i$. In addition, note that the maximum value of the upper bound is the $z_i^{nadir}, i = 2$, and the minimum value is the same as $z_i^*, i = 2$.

Through the continuous iterations of v_i, we are able to find a maximum of $1/p_i$ solutions that allow us to build the Pareto Front, and therefore, analyze with more detail the implications of the addition of the Objective Function 4.2 in our version of the RSRP.

Chapter 5

Chicago subway data

5.1 The RSRP for subway railway planning

The RSRP model can be used for several railway planning types, such as intercities train trips, or urban subways. Then, to validate the model, we had to find real data or simulate it, from a railway system. Once the literature's works do not provide sufficient data to validate the model due to confidentiality constraints, the real data from the Chicago Transit Authority[1] (CTA) was used. This section aims to explain how the data was found and how was it used in order to validate the model.

5.1.1 Chicago Transit Authority description

CTA operates the second-largest public transportation system in the United States of America. CTA serves the city of Chicago and 35 suburbs and is able to provide 81% of Chicago's public transportation. For the model, only the subway data is necessary. They operate 1,492 rail cars, distributed over 8 lines. The total sum of the rail tracks is 360.65 kilometers, and the total number of stations operated by CTA is 145. Additionally, the transit authority operates 2,318 trips per day, on average. The CTA data was chosen for one main reason, which is the diversity of available data provided on the website, leading to a lower lack of needed data, compared with other open data available from other railway companies.

The CTA publishes several databases and three of them have real data that can be used to solve the RSRP. The first database regards the daily demand [2] for each station between 01/01/2001 and 30/06/2022, which is a large period of time that can be analyzed. However, due to the propriety of the Periodic Timetable (see Section 3.4), only one week needs to be chosen to solve the RSRP. The second database concerns the trip timetable [3], as the available period of time with real data is between 10/12/2020 and 28/02/2021. For that reason, the week that will be analyzed and for which the RSRP will be solved is the one between 10/12/2020 and 16/12/2020, in which both databases have real data.

[1] Chicago Transit Authority's open data: `https://www.transitchicago.com/data/`.

[2] Chicago Transit Authority's daily demand: `https://data.cityofchicago.org/Transportation/CTA-Ridership-L-Station-Entries-Daily-Totals/5neh-572f`.

[3] Chicago Transit Authority's trip timetable: `https://www.transitchicago.com/downloads/sch_data/`.

The third database does not change over time, once it indicates which stations belong to which lines [4], and will also be used in the process.

This section is now divided into different explanations of how the data was obtained or calculated. We will first start with the calculation of the demand per trip, then the cost calculations, and after the emissions computations.

5.1.2 Chicago Transit Authority demand

Accordingly, with the provided data, the demand per trip is not given. However, it can be calculated through some assumptions. CTA provides the list of the 145 stations with the respective number of entries per day (Table 5.1). The rides are the number of entries at each station, and the "Day type" column indicates if the trip occurs on a working day (w), Saturday (A), or on a Sunday or holiday (U). This raises a problem, which is the lack of information related to the passengers that change the line when traveling. Once CTA did not have any information about them, they were ignored and only considered in their first train.

Table 5.1: Stations information.

Station id	Station name	Date	Day type	Rides
40010	Austin-Forest Park	10/12/2020	W	4450
40010	Austin-Forest Park	11/12/2020	W	4480
40010	Austin-Forest Park	12/12/2020	A	2380
40010	Austin-Forest Park	13/12/2020	U	2220
40010	Austin-Forest Park	14/12/2020	W	4070
40010	Austin-Forest Park	15/12/2020	W	3930
40010	Austin-Forest Park	16/12/2020	W	3980
40020	Harlem-Lake	10/12/2020	W	12170
40020	Harlem-Lake	11/12/2020	W	11430
40020	Harlem-Lake	12/12/2020	A	7140
40020	Harlem-Lake	13/12/2020	U	5880
⋮	⋮	⋮	⋮	⋮

Moreover, CTA also provides the station stops for each trip[5]. So, knowing this, we can calculate the demand per trip as long as we know which part of the daily demand enters in which trip. For that, we need to know the distribution of the demand per day, and for that, we adopted the daily demand distribution of the Beijing Yizhuang line studied by Wang et al. (2018) presented in Figure 5.1.

Figure 5.1 indicates the number of passengers entering the train for each hour of the day. We used this distribution and transformed it into a percentage, as can be seen in Figure 5.2. The histogram presented indicates which percentage of the total daily station demand is allocated to among the daily schedule, and therefore allocated to a certain trip. This is a piece of relevant information, once that CTA only provided the daily demand and not the demand per trip, as the RSRP requests.

[4]Chicago Transit Authority's stations' information: `https://data.cityofchicago.org/Transportation/CTA-System-Information-List-of-L-Stops/8pix-ypme`.

[5]Chicago Transit Authority's stations stops for each trip: `https://www.transitchicago.com/downloads/sch_data/`.

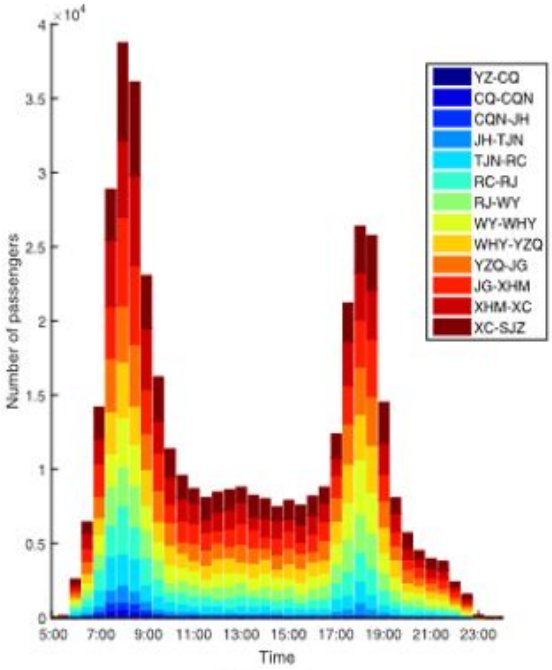


Figure 5.1: Demand distribution of the Beijing Yizhuang line.

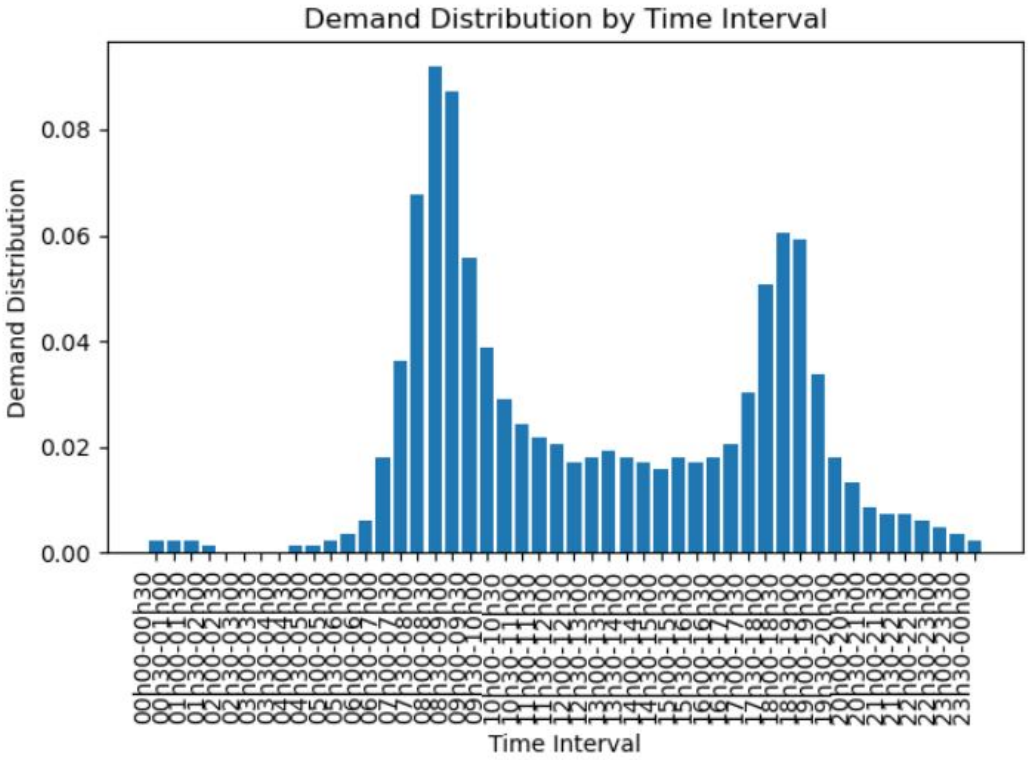


Figure 5.2: Demand distribution throughout the day.

Considering the daily demand of each station, and using the daily distribution of Figure 5.2, it is possible to simulate the distribution of the Chicago subway along the day, for each station. To calculate the demand per trip, more information is needed. The first one is the number of trips per station in a certain schedule, and the second one is the distribution of the passengers considering the trip direction.

To solve the first problem, we computed the number of trips passing through each station for each half hour. To solve the second problem, we need to know the percentage of passengers going in each direction. However, contrary to the Beijing Yizhuang line (Wang et al., 2018), the direction of the trip is not important to the computation of the trip demand. Indeed, it is expected a bigger demand for the trips going towards the city during the morning and then a higher demand for the trips departing from the city at the end of the day. In Chicago's lines, this issue can be ignored, due to the lines' structure that are displayed in Figure 5.3. There are two groups of possibilities for the eight Chicago subway lines:

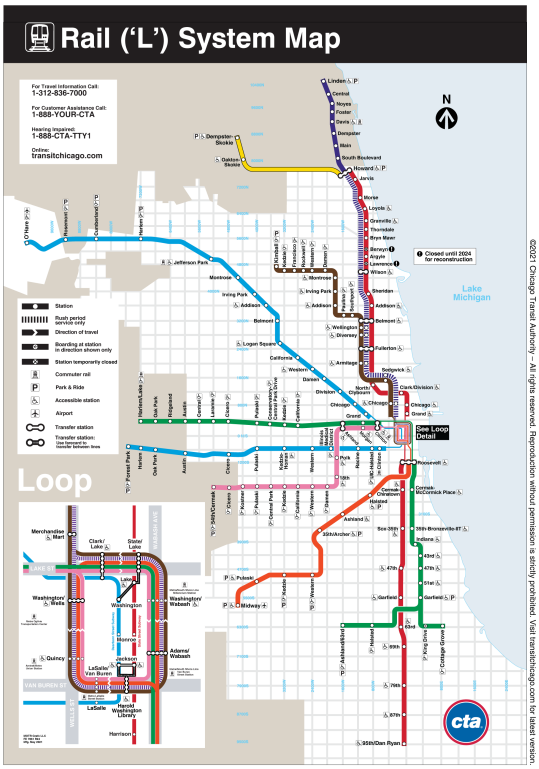


Figure 5.3: Chicago subway lines.

- Red, blue, and green lines - once these tree lines pass through the city center and then continue to the suburbs, it does not matter the direction of the trip. Once neither of these lines starts or ends at the city center, in both directions (North or South), the train will pass through the city and leave it. Let´s see the case in which a trip in the red line goes to the south: this trip starts in the north, so, for the stations at the north of the city, it is expected a high quantity of passengers going to the south (city center) and low going to the north (suburbs). However, when the same trip passes through the city center, it is expected that the passengers for each station aim to go north, so there would be a low percentage allocated to the south direction, after passing the city center. Due to this reason, we assume that the demand is symmetrical when confronting both directions, which means that it was assumed that the demand can be allocated at 50% of the total demand for the station, to simplify;
- Yellow, purple, pink, orange, and brown lines - these five lines are cyclic, which means that they depart and arrive at the same station. In the same logic as the other three lines, the trips that are

performed in these five lines also start and end outside of the city center, which means that the direction does not affect the demand. The same reasoning was applied, which led to the same conclusion: the demand was divided equally (50%) for all stations.

At this point, it is known the total passenger entries, for each trip considering its direction, at a certain station. To compute the trip demand, it is only necessary to sum the station's demand in which each trip $t \in T$ passes through. The only lacking information is the existing passengers throughout the trip. This information is essential, because if we only sum the total demand, the calculated demand would be way bigger than the actual demand, once the passengers leave along the trip. Once that was not possible to find information in the literature relative to this, and we only needed to know the maximum demand at the same time for each trip t, the following assumption was considered: the total demand of the trip is 50% of the sum of all the demand´s stations in which trip t goes through. Now, the trip demand d_t is known for each trip t.

Now that we have all the data related to the CTA trips, an extract of the overall information of the trips input in our RSRP model is described in Table 5.2.

Table 5.2: Trips information.

Trip id	Date	Line	Departure station	Arrival station	Departure time	Arrival time	Length	Demand
62189359694	10/12/2020	Org	Clark/Lake	State/Lake	48480	48450	118	281
62189360473	10/12/2020	Pink	54th/Cermak	State/Lake	47520	49260	132	83
62189360473	16/12/2020	Pink	54th/Cermak	State/Lake	565920	567660	132	83
62188840736	10/12/2020	Red	Dan Ryan	Howard	57840	61740	118	193
62188780868	12/12/2020	Red	Howard	Dan Ryan	230400	234120	118	140
62189359940	10/12/2020	Blue	Jackson	O'Hare	28920	31440	118	688
62189359942	10/12/2020	Blue	LaSalle	Forest Park	28950	30660	118	352
62189359940	10/12/2020	Blue	Jackson	O'Hare	28920	31440	118	688
62189359942	10/12/2020	Blue	LaSalle	Forest Park	28950	30660	118	352
62189360133	10/12/2020	Blue	O'Hare	Jackson	30420	33180	118	569
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

5.1.3 Chicago Transit Authority vehicles

CTA provides information about the station's maximum capacity to receive the train, and which fleet is assigned to which station. This information is displayed in Table 5.3. To exemplify, let´s consider the red line. One can see that the red line can only be operated by 5000 rail cars and that its maximum train car capacity is 8, which means that in this line compositions with more than 8 trains are not allowed. However, the information regarding the maximum train car capacity is not provided for every line, so some assumptions had to be made for the purple, pink, and yellow lines. The values assumed are around the maximum capacity of 8 cars for the other lines, so we assumed a maximum capacity of 9 vehicles for lines purple and pink and 7 vehicles for line yellow (values in red).

Table 5.3: Lines information.

Line	2600 rail cars	3200 rail cars	5000 rail cars	7000 rail cars	Maximum train car capacity
Red	No	No	Yes	No	8
Blue	Yes	Yes	No	No	8
Brown	Yes	Yes	No	Yes	8
Orange	Yes	No	No	No	8
Purple	No	No	Yes	No	9
Pink	No	No	Yes	No	9
Green	No	No	Yes	No	8
Yellow	No	No	Yes	No	7

Thus, with the information available in Table 5.3, it is now possible to allocate the maximum train length for each trip, considering its line and vehicle lengths. For the Chicago subway, the vehicle length is not relevant, once all vehicles have the same length, despite their types, as it is presented in Table 5.4. Furthermore, CTA also provides the capacity of the types of vehicles used in their operations. Nevertheless, they do not provide all the required information, requiring to compute some calculations. CTA only provides the seated capacity of their vehicles, with one exception: the 5000-series whereby the total capacity (seated and stand-up) is also provided. From these values, one can say that the percentage of seated passengers is 27.64% (34/123). Knowing this value, it is possible to compute the total number of passengers that each type of vehicle can satisfy (total capacity). Consider the example of the 2600-series vehicles: if the number of seated places is 46, then the total capacity will be 166 (46/27.64%). All computed values all represented in red, and their calculations followed the same logic.

Table 5.4: Chicago vehicles' length and capacity.

Vehicle type	Year of construction	Length (meters)	Capacity (seated)	Capacity (stand-up)	Capacity (total)
2600-series	1981–1987	14.63	46	120	166
3200-series	1992–1994	14.63	39	102	141
5000-series	2009–2015	14.63	34	89	123
7000-series	2019–TBD	14.63	38	99	137

From this point, the remaining information about the Chicago subway fleet lacks to know the operational costs and emissions produced. Starting with the operational costs: CTA does not provide a detailed and allocated distribution of their operational costs, regarding their fleet, however, through the financial report for December 2020 (because the week in the analysis is from 10/12/2020 to 16/12/2020), one can compute the daily operational costs (Table 5.5).

Now that the operating expenses are known, one can calculate the daily fixed and variable costs. The fixed costs are considered to be the *Labor and fringe benefits*, *Purchase of security services*, *Maintenance and repairs, utilities, rent and other* (we only assumed 80%, leaving aside 20% for maintenance, once that the maintenance is not been consider in this version of the RSRP), and *Provision for deprecia-*

Table 5.5: Chicago's operating expenses. Source: Authority (2020).

Daily operating expenses (€)	
Labor and fringe benefits	3,137,041.43
Materials and supplies	199,606.74
Fuel	99,069.52
Electric	65,795.51
Purchase of security services	53,306.74
Maintenance and repairs, utilities, rent and other	289,928.79
	3,844,748.74
Provision for injuries and damages	61,016.15
Provision for depreciation	1,336,884.83
Total operating expenses	5,242,649.72

tion, leading to fixed daily operating expenses of 4,759,176.04 €. Once CTA states that they operate, on average, 2,318 trips per day, the fixed cost per trip is 2,053.14€. Now, we only have to divide this value by the average number of needed fleets, per trip. We computed the average the average demand per trip, which is 257 passengers, on average; and also computed the average capacity that Chicago's fleet can carry on: $\frac{166+141+123+137}{4} \approx 142$. Dividing the 257 average passengers by the 142 average capacity of one vehicle, we reach the average number of compositions for the Chicago subway operations, which is two vehicles. Now that we found the average number of vehicles within a composition per trip, we can conclude that the fixed cost of a vehicle is 2,053.14€/2 = 1,026.57€. However, in order to differentiate the utilization costs of the four types of vehicles, we calculated the costs assuming variations of 25% and 15% from this base value (positive variations for 2600-series and 3200-series fleet, and negative variations for 5000-series and 7000-series fleet). Taking as an example the 2600-series fleet, the fixed utilization cost assumed is 1,026.57€ plus a variation of 25%, reaching 1,283.21€. The oldest vehicles (2600-series and 3200-series fleet) are associated with higher costs than the latest vehicles (5000-series and 7000-series fleet).

In order to compute the variable costs, we add up the operations whose costs depend on the activity: *Materials and supplies*, *Fuel*, *Electric*, and *Provision for injuries and damages*, reaching a daily variable cost of 425,487.92€. By doing the same calculations as for the fixed operating costs, the value of the allocated costs for one vehicle is 91.78€. In opposite to the fixed costs, we still have to divide it by the average number of kilometers traveled, and for that, we know that the total CTA tracks are distributed by 360,65 km for the eight existing lines. So, considering the average number of kilometers traveled as 45.08km (360.65 km/8 lines), we can assume that the variable cost of the vehicles is 2.04€/km (91.78 €/45.08 km). Once again, the vehicles were constructed in different periods, and for that reason, we assumed that the older vehicles would have higher variable costs, and the younger fleet would have lower variation costs. Take as an example the case of the 5000-series fleet, we calculate its value, by negatively varying the base costs by 15%, reaching a variable cost of 1.73€/km.

The total fixed and variable costs allocated to the four vehicles operating the Chicago subway are

displayed in Table 5.6.

Table 5.6: Chicago vehicle's costs.

Vehicle type	Utilization cost (€)	Travel cost (€/km)
2600-series	1,283.21	2.54
3200-series	1,180.55	2.34
5000-series	872.58	1.73
7000-series	769.93	1.53

The only remaining parameter needed is the emissions of CO_2 equivalent produced by each vehicle of the Chicago subway system.

When it comes to the emissions produced, the report presented by the Congressional Budget Office (CBO) indicates that the average CO_2 equivalent produced by the Chicago Transit Authority subway over the year 2019 is 0.320 pounds of CO_2 per pax-mile (for Transportation, 2022). Even if the week used to solve the RSRP is between 10/12/2020 and 16/12/2020, the value of 0.320 is used by analogy. Nevertheless, this value needs to be adapted to the four types of vehicles the CTA uses. Indeed, the value of 0.320 is an average. Once the Chicago subway lines are operated by four vehicles, this value will be allocated over the four vehicles, considering their year of construction, but keeping the average. Once again, we consider that the older the fleet is, the more technologically limited is, thus it is considered to pollute more. Considering that, the older railway vehicles are leading to higher indices of CO_2 equivalent per pax and miles, the generated values for $emis_i, i = 1, 2, 3, 4$ depend on the fabrication year of the vehicles. Using as a reference, the average value of 0.320, by increasing this value by 15% and 25%, for older vehicles, and decreasing by 15% and 25% for younger vehicles, we reached the following values: 0.4, 0.368, 0.272, and 0.24, for the 2600-series, 3200-series, 5000-series, and 7000-series vehicles, respectively. The values need to be converted from pounds to kilograms and miles to kilometers. The calculation performed is explained through the Equation 5.1, using the conversations: 1 pound (lb) is approximately equal to 0.453592 kilograms and 1 mile (mi) is approximately equal to 1.60934 kilometers (km).

$$emis_i(CO_2(kg)/pax/km) = emis_i(CO_2(lb)/pax/mi) \times \frac{0.453592\frac{kg}{lb}}{1.609344\frac{km}{mi}} \tag{5.1}$$

Considering, for example, the estimated value of the emissions of CO_2 for the 2600-series vehicle - vehicle $f_i, i = 1$, the value of 0.4 lb of CO_2/pax/mi is converted in Equation 5.2.

$$emis_1(CO_2(kg)/pax/km) = 0.40CO_2(lb)/pax/mi \times \frac{0.453592\frac{kg}{lb}}{1.609344\frac{km}{mi}} \approx 0,1099\ CO_2(kg)/pax/km \tag{5.2}$$

At this point, we have the CO_2 emissions produced by kilogram and person for each vehicle. Once we know the capacity of each vehicle, we can estimate the total emissions produced by each fleet,

multiplying the CO_2 produced per person, by the number of passengers that each vehicle can take, leading us to the estimated value of the CO_2 produced per kilometer, for each vehicle. Equation 5.3, calculates this value, continuing the example for the 2600-series fleet.

$$emis_1(CO_2(kg)/km) = 0,1099\ CO_2(lb)/pax/mi \times 166\ pax \approx 18.71\ CO_2(kg)/km \tag{5.3}$$

Hence, the total information needed to solve the RSRP regarding the Chicago subway vehicles is displayed in Table 5.7.

Table 5.7: CTA's vehicles information.

Vehicle type	Length (m)	Capacity (# passengers)	Utilization Cost (€)	Travel Cost (€/km)	Emissions of CO_2 eq (kg/km)
2600-series	14.63	166	1,283.21	2.54	18.71
3200-series	14.63	141	1,180.55	2.34	14.62
5000-series	14.63	123	872.58	1.73	9.43
7000-series	14.63	137	769.93	1.53	9.27

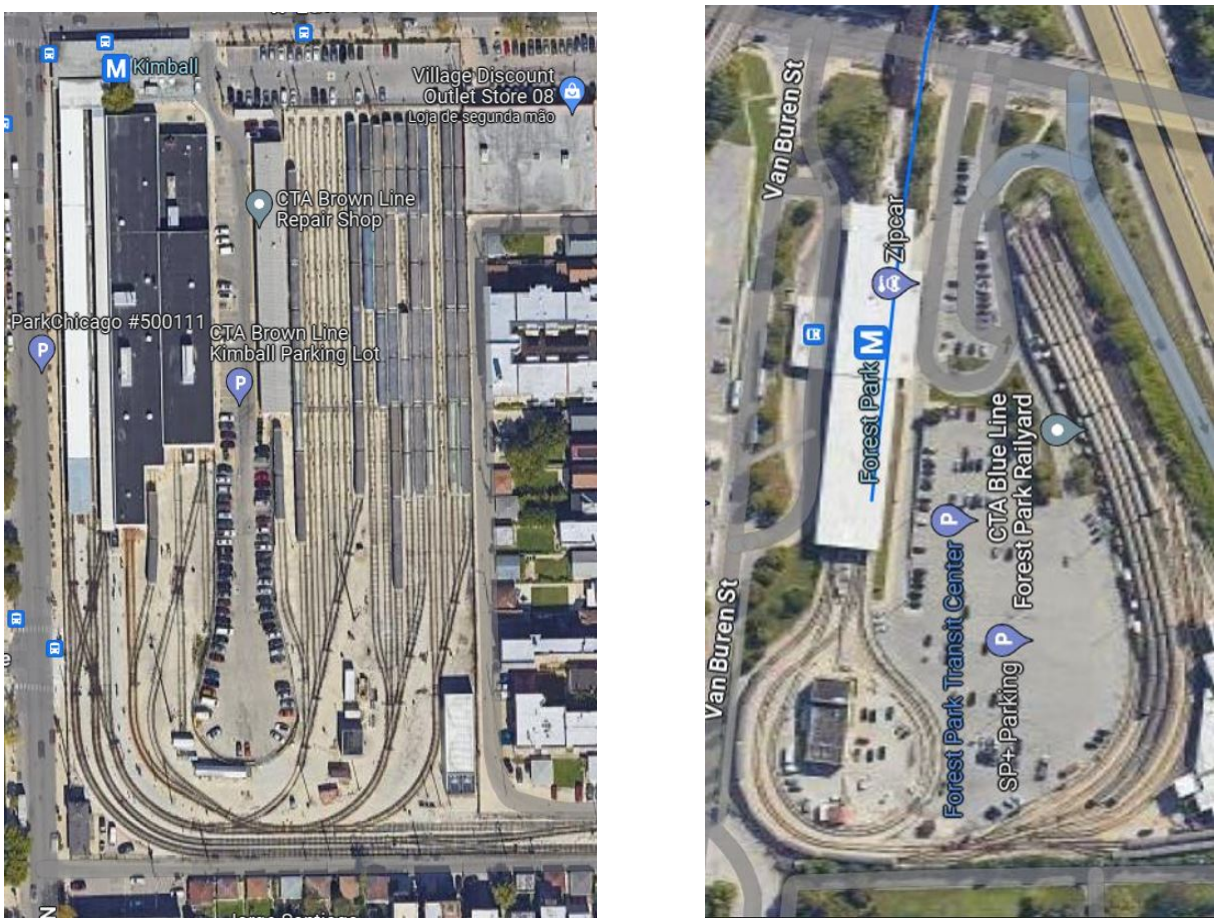
5.1.4 Chicago Transit Authority stations

From the design of the Chicago railway terminal station, the parameter $turn_i$ is obtained. For terminal stations like Kimball (Figure 5.4 (a)), there is no track reserved for the vehicles to turn, so the $turn_i$ is 0. On the other hand, the stations that have the same design as Forest Park (Figure 5.4 (b)) own a track reserved for the vehicles to turn around, so the $turn_i$ is set to 1. When $turn_i$ is 1, it will allow all the trains to turn, but the orientation of each particular vehicle ($tick$ and $tock$), despite the fact that it changes the orientation, is not relevant to the model. Once the instances in the analysis are regarding the Chicago railway subway, the orientation of the vehicles is considered to always be the same, $neutral$, once the urban subway vehicles do not have first-class or second-class, this is reserved for the vehicles operating in intercities lines, for example. To know the parameter $shunt_s$, we used a similar approach, by observing the structure of the Chicago subway terminal stations. According to their layout, it was decided if the shunting operations were allowed or not, for the train compositions.

It was also possible to calculate the distance between the several stations using the Google API Distance Matrix Service[6]. The API allowed us to determine both distance and time between the set of all stations. Even if we only needed the time in order to solve the rolling stock rotation planning, we decided to opt for the distance. As a matter of fact, the API returned the time between the stations considering the stop time of each station, hence to know the travel time between two terminal stations, through the API, we would get an erroneous value. To solve this issue, we used the distance between stations, which is presented in Table 5.8, and then computed the average speed to get the travel time.

Now that the distance between the stations is known, we can calculate a more accurate travel time if we know the average speed of the train. CTA does not provide that information, therefore we had to

[6]Distance Matrix Service: `https://developers.google.com/maps/documentation/javascript/examples/distance-matrix`.



(a) Kimball.

(b) Forest Park.

Figure 5.4: Chicago terminal stations.

estimate it. To this end, we analyze three sections between two consecutive stations relatively distant from each other (so that the train had time to reach its average speed) and calculate the average speed. Table 5.9 displays the information regarding these three sections, and as one can see, the average speed computed for the Chicago subway trains is 0.01373 km/sec (49.43 km/h). For the purpose of our model, we used the speed in this unit, once that the distances between the stations are in kilometers, and that the departure and arrival times are in seconds.

5.1.5 Chicago Transit Authority estimations

Despite the large amount of data available provided by the Chicago Transit Authority, some parameters essential to solving the rolling stock rotation planning problem are not available in the CTA's open data. So, in order to solve the RSRP some assumptions had to be made. The estimated data is a compilation and adaptation of other real railway cases around the world. For instance, between two trips, the train composition has a minimum required time in which it needs to stay in the terminal station. Even if CTA does not provide this information, Chang et al. (2000) indicates that the trains need 45 minutes to stay within the terminal station. However, for the terminal stations analyzed by the authors, the train needed to travel to the depot, which is 15 km from the terminal station. For the Chicago subway, the de-

Table 5.8: Distance matrix between stations.

Stations (distance in km)	Austin	Harlem	Pulaski	Quincy	Davis	Belmont	Jackson	...	Cermak-McCormick	Washington/Wabash
Austin	0	4.4	5.8	15.3	34.2	23.5	15.4	...	18.2	14.8
Harlem		0	6.5	15.4	34.6	23.9	15.9	...	18.5	15.2
Pulaski			0	9	27	17.5	9	...	12.1	8.8
Quincy				0	22.4	10.8	0.8	...	5.2	1.5
Davis					0	29.7	21.6	...	24.3	21.0
Belmont						0	10.5	...	13.7	10.1
Jackson							0	...	3.4	0.9
⋮								⋱	⋮	⋮
Cermak-McCormick									0	3.4
Washington/Wabash										0

Table 5.9: Computation of the average speed.

	Yellow line	Blue line	Green line
	Oakton - Sokie	Harlem - Jefferson	Halsted - Garfield
Distance (km)	7	4,7	3,8
Time (minutes)	6	7	6
Speed (km/min)	1,167	0,671	0,633
Speed (km/sec)	0,019444	0,011190	0,010556
Average speed (km/sec)		0,01373	

pot is in the same location as the terminal station, so we need to subtract the travel time for the terminal instances analyzed by the authors. They say that the depot is at 15 km, considering an average speed of 0.01373 km/sec (0.8238 km/min) and that the required time at the terminal station is 45 minutes minus the travel time to and from the depot (2 $\times$ 15 min), the required time for the terminal station estimated is approximately 8 minutes and a half, which is 510 seconds (45 min $-$ 2 $\times$ $\frac{15\ km}{0.8238\ km/min}$).

CTA does not provide any information regarding the costs or the execution times, when it comes to the operations performed within the terminal stations, such as turn and shunting operations operations. Nevertheless, in 2021, Schwerdfeger et al. indicated that the required time to perform a shunting operation for the German railway operator Deutsche Bahn is, on average, 30 minutes (1800 seconds). Not having any information about the turn operation, we assumed that its required time would be twice the time required to be at the terminal station between trips, 17 minutes - 1020 seconds (2$\times$8.5 min). Although it was not possible to find the operational costs for both operations, we estimated it using as a reference the average vehicle utilization cost (1,026.57 €). Once the turning operation was estimated to take less time than the shunting operation, we assumed the turning operation to be one-third of the reference cost and the shunting operation to be half. Therefore, the turning cost is 342.19 € (1,026.57 € / 3), and the shunting cost is 513.28 € (1,026.57 € / 2).

Finally, the only required parameters that remain to be estimated are the bonification and penalizations, associated with the regular and deadhead trips, respectively. This value would be preferentially provided by the decision-maker, after considering and quantifying the benefits of considering the regularity within the timetable, and the disadvantages of the existence of the deadhead trips in the RSRP's solution. Therefore, we had to assume these values without discussing them with the decision-maker, in our case, we are not working with a decision-maker. The values estimated are presented in Table 5.10.

Table 5.10: Bonification and penalization values.

Cost	Regular trips bonification	20%
	Deadhead trip penalization	10%
Emissions	Deadhead trip penalization	10%

As it can be seen in Table 5.10, we considered relatively low percentages of bonification and penalizations. In this way, we are enhancing train compositions to operate regular trips, and discouraging the model to operate deadhead trips. Moreover, we estimated the same penalization for both costs and emissions regarding the deadhead trips, so that there are differences between the two objective functions. The bonification of the regularity is higher than the deadhead trip's penalization once the model already tends to disadvantage deadhead trips, once they are costly and are not necessary to satisfy the demand.

Chapter 6

Results

The hypergraph model used to solve the RSRP was implemented in C++, using the mixed integer programming solver CPLEX 22.1.0. The computations were performed on a computer with an Intel(R) Core(TM) i5-8250U CPU with 8,00 GB of memory RAM, with a base clock speed of 1,60 GHz, and a maximum turbo frequency of 1,80 GHz. All computed instances are related to the Chicago Urban Subway railway system, USA. It is to be noted that for the several analyses throughout this chapter, various instances were created in order to take a deeper look into the understanding of the model. Due to the low time reserved to test the model, the created instances are relatively small, yet enable us to draw pertinent conclusions regarding our version of the rolling stock rotation planning problem.

In this section, there are some questions that aim to be answered in order to evaluate the complexity of the hypergraph model and the implications of the addition of a new objective function, such as the following:

a) What is more time-consuming for the hypergraph model: more possibilities of compositions or more trips for a given timetable?

b) How does the addition of the new objective function (4.2) related to the emissions impact the cost minimization (4.1)?

c) How do the regularity bonification and deadhead trips penalizations impact the RSRP considering both objective functions 4.1 and 4.2?

6.1 Instances description

In order to obtain the results achieved throughout this chapter, we had to create several instances, varying their characteristics, by the analysis desired. Table 6.1 displays the several instances that will be used according to the maximum number of vehicles allowed per feasible composition, the number of trips and if the shunting is or is not considered. As the instances will be used, a deeper explanation will be provided.

Table 6.1: Instances description.

Instances	Maximum number of vehicles per composition	$\|T\|$	Shunting allowed?
instance_0	2	13	✓
instance_c	2	49	✓
instance_c.3	3	49	✓
instance_c.4	4	49	✓
instance_c.5	5	49	✓
instance_c.6	6	49	✓
instance_c.50T	2	50	✓
instance_c.59T	2	59	✓
instance_c.79T	2	79	✓
instance_c.109T	2	109	✓
instance_c.149T	2	149	✓
instance_c.249T	2	249	✓
instance_1	5	34	✓
instance_2	2	222	✓
instance_3	2	222	
instance_4	2	106	
instance_5	2	396	
instance_6	2	325	
instance_7	2	307	
instance_8	2	408	
instance_9	3	273	
instance_10	2	177	

6.2 Model's validation

The interpretation of the hypergraph was already explored in Section 3.5.1. So, given that, we can explore the instance_0 solution and how the vehicle planning can be done. The instance_0 was created considering the 13 trips from the Chicago red subway line that have a departure time between 16h00 and 16h15 on 10/12/2020 and 12/10/2020 (Thursday and Saturday, respectively). Instance_0 was deliberately created to be a small instance with its only purpose of showing how the hypergraph model optimizes the rolling stock rotation and validating the model.

For the 13 trips considered in instance_0, a diagram of the RSRP's solution is displayed in Figure 6.1.

The trip direction is indicated through the background color: the white background indicates the direction as Howard to 95th/Dan Ryan, while the grey background illustrates the direction as 95th/Dan Ryan to Howard. Moreover, for all trips, the composition that minimizes the costs is {5000-series fleet, 5000-series fleet}.

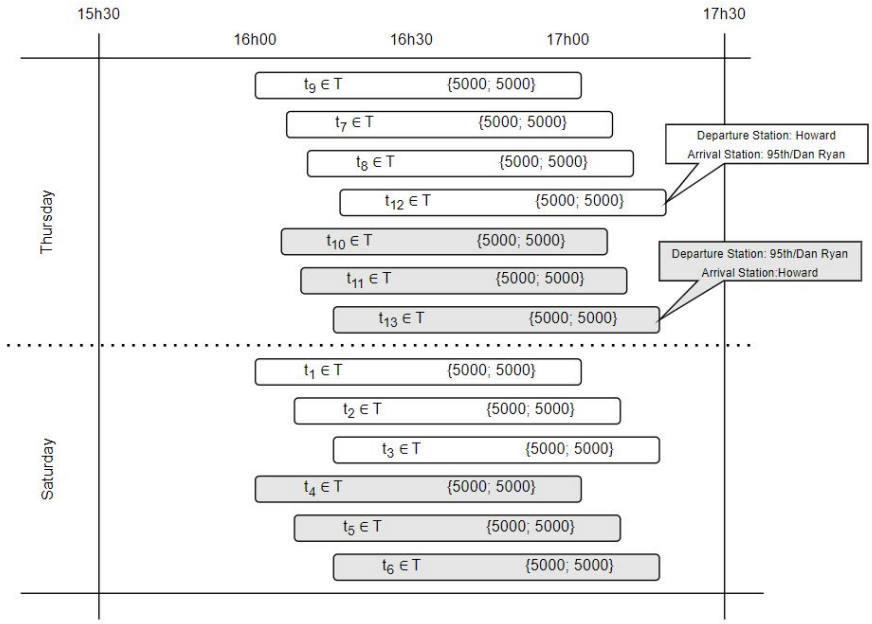


Figure 6.1: Instance_0 solution for cost minimization.

The diagram present in Figure 6.1 indicates the composition of each trip, and the departure and arrival stations and time, but it does not present the rotation solved by the model. For that, take Figure 6.2 that illustrates the trips to which each composition train operates.

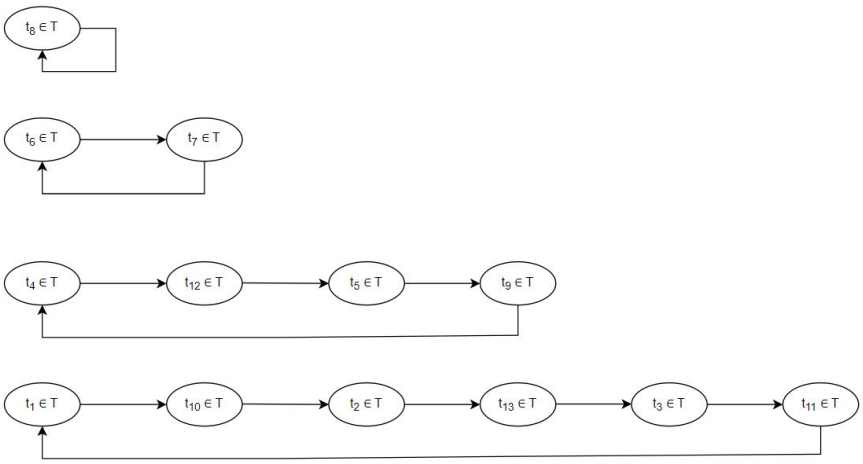


Figure 6.2: Instance_0 route solution for cost minimization.

As it can be seen in Figure 6.2, there are four rolling stock routes created. The first one indicates that the one train composition operates only trip t_8.The solution indicates that this particular composition only travels trip t_8 each week and when the trip ends, it returns to the initial departure station to operate the

trip in the next week. This means that a deadhead trip is performed for this composition, each week. The second route indicates that the train compositions operate trip t_6, and then trip t_7. As can be observed in Figure 6.1, these two trips are on different days, and they can be performed by only one composition train. Moreover, the arrival station of t_6 is Howard, and the departure station of trip t_7 is also Howard, meaning that no deadhead trips are performed. The same happens when the train goes from trip t_7 to trip t_6. The third route present in the solution ensures four trips, which means that one train cannot operate all within the same week, once trips t_4 and t_5 occur at the same time, and trips t_9 and t_{12} also occur at the same time. For this reason, there is the necessity of having two trains operating these trips, due to the cyclicity of the timetable (see Section 3.4). To better understand this situation, Figure 6.3 illustrates the need to use two train compositions. As it can be seen, both train compositions have the same compositions, which is $\{$5000-series car, 5000-series car$\}$, and the train compositions operate the same trips, but in different orders: assume that train composition 1 starts in trip t_4, and train composition 2 starts in trip t_5. In this way, all timetable trips are covered by the train compositions, respecting all the necessary requirements for the Chicago subway's red line. Finally, the fourth route covers six trips, and comparable to what happened for the third route, this path cannot be ensured by only one train composition. This route has a maximum of three trips to cover at the same time, meaning that it needs three train compositions to ensure the path. Once again, the three train compositions start at different trips but perform the same route, to cover all six trips.

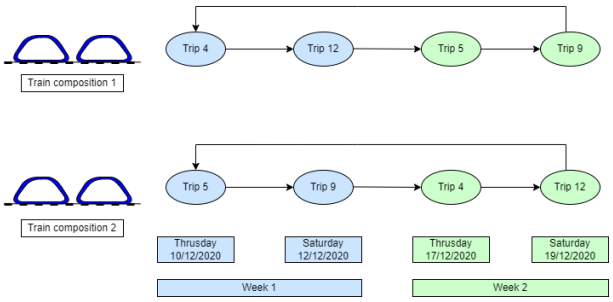


Figure 6.3: Route three's rotation.

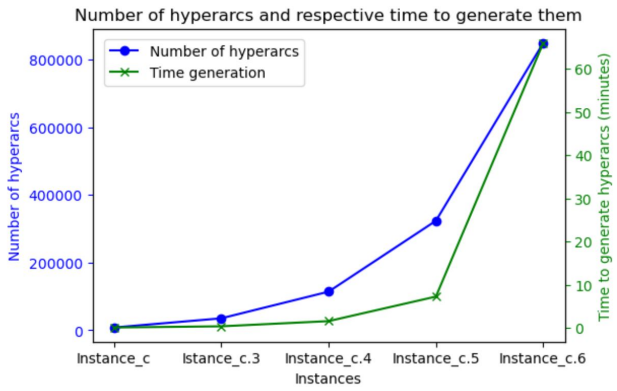
6.3 Complexity of the hypergraph model

As explored throughout this work, the hypergraph model is well-known for its complexity, once the number of generated hyperarcs increases significantly with the addition of new feasible compositions and trips. In this section, we will explore what is most time-consuming for the model. For that, we will analyze the increment of the feasible compositions for one instance against the increment of additional trips on that same instance. The instance in use is called instance_c and considers the 49 trips departing between 10h00 and 10h15 and between 15h00 and 15h15 from the Chicago subway red, blue, and green lines on 10/12/2020.

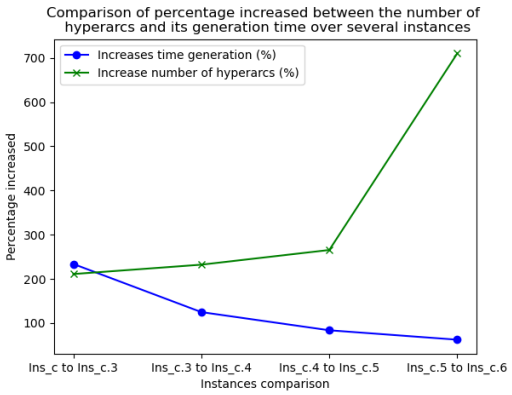
Instance_c was created considering two constraints. The first is the necessity of starting with a few feasible compositions. For instance_c, the minimum number of vehicles for the feasible compositions is 2. In this way, we can increase the number of feasible compositions by adding one vehicle at a time. The second constraint is the necessity of starting with a considerably low number of trips, in this case, 49 out of the 26,730 trips provided by the CTA for the week in analysis. In order to understand what is more time-consuming to our RSRP model and why, we created two scenarios: 1 and 2. Scenario 1 considers instances between instance_c.3 until instance_c.6, i.e., it maintains the number of trips and increases the maximum number of vehicles for the feasible compositions $c \in C(t)$ for each trip $t \in T$. Scenario 2 considers the instances from instance_c.1T until instance_c.200T, which means that scenario 2 keeps constant the maximum number of vehicles for the feasible compositions c for each trip t but progressively increases the number of trips t.

In Figure 6.4 (a), there is represented the increase of the number of hyperarcs and its respective generation time for each instance - scenario 1. Instance_c considers 49 trips and the maximum number for each feasible composition is 2. Instance_c.3 considers 49 trips also, and the maximum number for each feasible composition is 3, until instance_c.6 considers 49 trips and a maximum number for each feasible composition of 6 vehicles. From this initial instance, the maximum number that a feasible composition can take is 8 vehicles, but when we ran the model for 7 cars the needed time to generate hyperarcs was 18 hours and a half, which was significantly high. Hence, we decided to only analyze until instance_c.6. As can be seen, by adding one more vehicle to the set of feasible compositions, the number of hyperarcs and their respective time to generate them also increases, as would be expected. Even if the number of hyperarcs and generation time increase, they do not increase similarly. As it is shown in Figure 6.4 (b), the number of hyperarcs tends to not increase as much over the tested instances. It means that as we increase the set of feasible compositions, the number of hyperarcs will increase at a lower rate as we test this scenario in further instances. However, even if the number of hyperarcs created for the instances does not increase at high rates when compared with the last instance, its generation time does. This means that, no matter how fast the number of hyperarcs augments, its generation time will tend to exponentially increase. To narrow this issue, we suggest computing limited feasible compositions. For the case of instance_c, there is no need to consider feasible compositions with more than 3 vehicles, once the minimum number of vehicles needed is 2. Consequently, by limiting the degrees of freedom for trips, we can significantly decrease the number of feasible cars, and consequently the number of hyperarcs and their generation time. Our suggestion is to limit the number of vehicles in composition, by one more than needed, i.e., if a trip $t \in T$ can satisfy its demand d_t using 3 cars, there should be only necessary to consider as feasible compositions, trains with 3 or 4 vehicles. Even if by doing this, the a priori work is more time-consuming, it can save significant time a posterior. In our example, if we limit instance_c by 3 vehicles in each composition (the minimum needed fleet is 2), we generate the hyperarcs in 22 seconds. Still, if we consider at most 6 vehicles, the generation time will be 66 minutes. It is to be noted that in terms of operational costs or emissions produced, both instances returned the same solutions. Even if this could not be the case for instances with more trips.

Figure 6.5 displays the impact of the addition of more trips on the hyperarcs´ number (a) and gen-



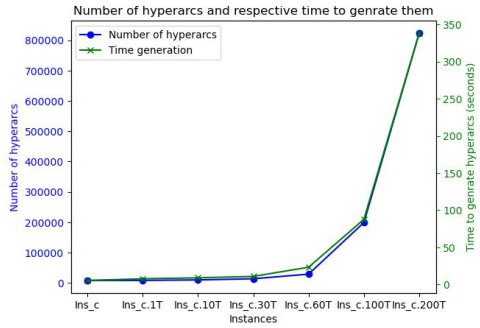
(a) Increased of number of vehicles.



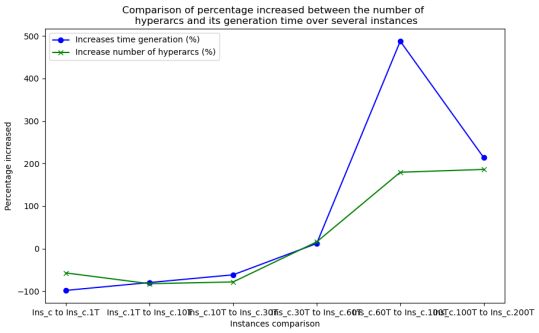
(b) Percentage increased between instances.

Figure 6.4: Numerical and percentage increased of the number of vehicles per composition.

eration time (b) - scenario 2. The starting instance is the same as the previous case (instance_c), and we progressively increased the number of trips. Firstly, we increased the number of trips by 1 (instance_c.1T), then 10 (instance_c.10T), 30 (instance_c.30T), 60 (instance_c.60T), 100 (instance_c.100T), and finally 200 (instance_c.200T). It is to be noted that for these seven instances, the maximum number of vehicles within the feasible composition is two so that we can compare directly the difference between the increase of feasible compositions with the trip increase.



(a) Increased of number of trips.



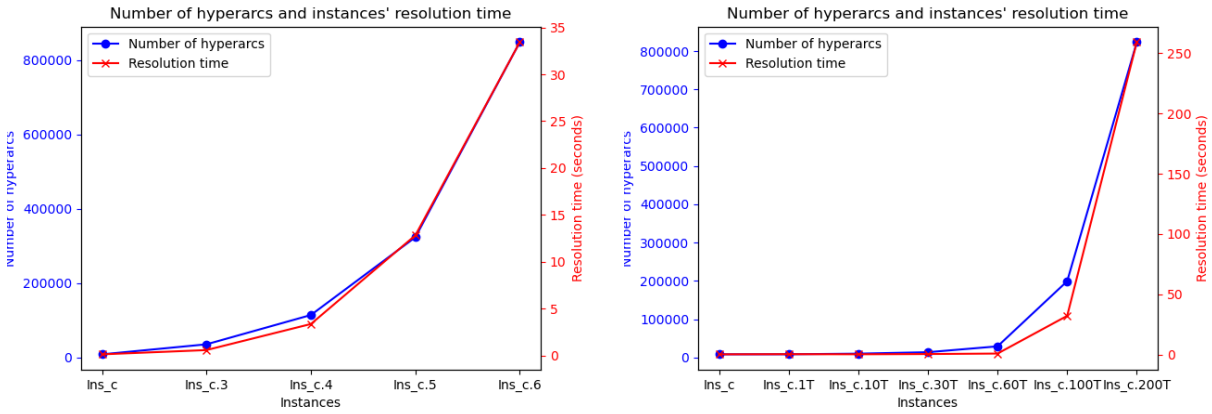
(b) Percentage increased between instances.

Figure 6.5: Numerical and percentage increased of the number of trips for the same feasible compositions.

With the increase in trips, the hyperarcs' number and time generation also increase. However, Figure 6.5 (b) shows that the variation between the instances regarding the increase of trips is not as predictable as the one for the number of vehicles within the feasible compositions (Figure 6.4 (b)). However, we can see that the hyperarcs' number for instance_c.6 is 848,904, and for instance_c.200T is 823,459. This means that we had to add 200 additional trips all with two vehicles at maximum for the creation of the feasible compositions to achieve roughly the same number of hyperarcs for an instance of 49 trips, but with six as the maximum number of vehicles for the feasible compositions. However, even if the number of hyperarcs is almost the same, the corresponding generation time is very different. For instance_c.6 the generation time is 66 minutes, while the generation time for instance_c.200T is almost

6 minutes (eleven times inferior to instance_c.6). This can be explained by the shunting operations. The more sizes of feasible compositions $c \in C(t)$ allowed for a trip $t \in T$, the more shunting operations are allowed, i.e., it is possible to decouple and couple the train in several ways. To remember, this was already forecasted in Section 5.1.4, hence the limitation of shunting operations by one for each feasible composition c. Indeed, this feature can be more optimized than we did (see Section 4.2). However, this leads us to the same conclusion drawn when analyzing the hyperarcs' number and time generation between instance_c and instance_c.6: the limitation of the feasible compositions c, for each trip $t \in T$, seems to be an interesting way to optimize the generation time of the hyperarcs. By comparing these two cases, we saw that by limiting the degrees of freedom of the generation of feasible compositions c, we can optimize the model and significantly decrease the hyperarcs' generation time.

So far, we only compared the hyperarcs' number and time generation. However, another interesting parameter to analyze is the resolution time in all instances. Figure 6.6 displays the results for our RSRP model's resolution, for both scenarios. It is to be noted that, at the moment, the resolution times presented are related to the minimization of the operation costs (Objective Function 4.1).



(a) Increase of number of vehicles.

(b) Increase of number of trips.

Figure 6.6: Resolution times for the several instances scenarios.

The first notable difference between both scenarios is the progression of the resolution time between scenarios. The resolution time of scenario 2 increases in a more exponential way than scenario 1, which aligns with the progression of both hyperarcs' number and generation time found in Figure 6.4. Furthermore, it stands out that the model's resolution times are not correlated to the generation time of the hyperarcs. Indeed, the resolution time for instance_c.6 is 33 seconds, while the one for instance_c.200T is 4 minutes and 18 seconds (almost 8 times higher). Regarding the resolution time, it seems that the addition of trips is more time-consuming than the increase of the maximum number of vehicles for the feasible compositions.

All in all, the hypergraph model is a complex and time-consuming methodology to solve the rolling stock rotation planning problem. However, through the control and limitation of the degrees of freedom, it is possible to make it less time-consuming. That is why the Borndörfer et al. (2016) used the column generation method to create the feasible compositions. It allows to progressively generate columns,

instead of solving the problem from the beginning. Nevertheless, even if Algorithm 3 is optimized, it can be expected that the proportions of the generation time remain, i.e., as we progressively implement new hyperarcs, we exponentially increase the computational time.

Another relevant factor that increases the complexity of the model is the non-optimization of Algorithm 3. This algorithm causes the model to solve the problem by considering all possibilities at once. This is emphasised by the fact that the model resolution time from scenario 2 is lower.

6.4 Addition of the emissions minimization objective function

Typically, the RSRP is focused on the minimization of the operational costs. Our version of the RSRP also includes the minimization of the emissions produced by railway operations. This section is dedicated to understanding how this new objective function (4.2) impacts the usual minimization of the operational costs (4.1). In the first place, Table 6.2 describes the results of both solutions considering only the objective function related to the cost minimization (4.1) and the one regarding the emissions minimization (4.2), respectively. It is to be noted that, in order to solve larger instances, we reduced the number of degrees of freedom, so that the number of hyperarcs generated would not be so big that it would end by jeopardizing the more the resolution time that it would be able to find a better solution.

To obtain our computational results, we used from instance_0 until instance_c.9. All of these instances (with one exception) were created for considering different departure times, different days of the week, and different sets of lines. The exception is between instance_c.2 and instance_c.3, once these instances are almost identical. The only difference is that instance_c.2 includes the generation of hyperarcs for the shunting operations while instance_3 does not have them so we could be able to see the difference with and without the shunting operations hyperarcs. It was found that the difference is very meaningful, once that the hyperarcs' generation time for instance_2 is almost six hours, while the hyperarcs' generation time for instance_3 is only seventeen minutes. Moreover, the number of hyperarcs generated is more than four times higher when considering shunting operations. For this reason, and once we did not have solutions considering shunting operations due to the associated costs, we decided to ignore this operation to continue with the computational results. To read the table, consider the first line of each instance related to the results from the costs' minimization and the second line related to the results from the emissions' minimization.

A difference between cost minimization and emissions minimization is the number of deadhead trips traveled. For emissions minimization, the model tends to avoid deadhead trips due to the penalty associated, more than the cost minimization avoids. On the other hand, the regular trips do not seem to be more enhanced for any approach. Due to the regularity bonification, it was expected to observe a significantly higher number of regular trips for cost minimization, which is not the case, as can be observed for instance_2 and instance_9.

Table 6.2: Computational results.

Instances	$\|T\|$	# Shunting operations	# Turn operations	$\|V\|$	$\|H(t)\|$	# Regularity hyperarcs	$\|H(c)\|$	#Deadhead trips	Time to generate hyperarcs (hh:mm:ss)	Time to solve (hh:mm:ss)	Gap (%)	Cost (€)	Emissions of CO_2 eq (kg/km)
instance_0	13	0	1	270	45	2	2,061	1	00:00:03	00:00:01	0	46,585	9,562
	13	0	1	270	45	0	2,061	1	00:00:03	00:00:01	0	47,331	9,561
instance_1	34	0	4	14,384	1,563	0	2,615	34	00:09:22	00:00:01	0	319,603	77,508
	34	0	4	14,384	1,563	0	2,615	32	00:09:22	00:00:01	0	329,729	76,461
instance_2	222	0	0	13,114	2,117	44	3,022,555	172	05:51:16	00:07:40	0	581,327	35,953
	222	0	0	13,114	2,117	90	3,022,555	170	05:51:16	00:07:20	3,96E-05	758,114	14,123
instance_3	222	-	0	13,114	2,117	44	727,267	172	00:17:18	00:05:27	0	581,327	35,953
	222	-	0	13,114	2,117	90	727,267	170	00:17:18	00:04:44	4,09E-05	758,108	14,123
instance_4	106	-	0	1,434	423	0	18,057	46	00:00:11	00:00:01	0	253,191	25,152
	106	-	0	1,434	423	0	18,057	47	00:00:11	00:00:01	0	256,442	24,875
instance_5	396	-	0	5,354	1,582	0	292,626	158	00:02:12	00:00:14	0	935,934	91,240
	396	-	0	5,354	1,582	0	292,626	150	00:02:12	00:00:13	0	939,962	85,421
instance_6	325	-	0	4,792	1,396	0	456,524	51	00:03:17	00:00:14	0	915,158	134,304
	325	-	0	4,792	1,396	0	456,524	22	00:03:17	00:01:34	0	937,755	127,280
instance_7	307	-	0	3,220	996	164	943,120	21	00:04:03	00:04:03	0	704,475	135,101
	307	-	0	3,220	996	119	943,120	9	00:04:03	00:06:20	0	748,165	123,816
instance_8	408	-	0	4,728	1,468	297	2,093,236	26	00:11:15	00:20:24	0	855,512	151,455
	408	-	0	4,728	1,468	245	2,093,236	16	00:11:15	00:23:40	0	922,316	139,182
instance_9	273	-	0	5,110	1,208	54	375,090	88	00:02:51	00:01:09	0	685,434	68,863
	273	-	0	5,110	1,208	78	375,090	85	00:02:51	00:01:54	0	740,863	50,251

We have already seen above how the solutions change in terms of operational costs and emissions produced among the presented instances. However, to better understand the solutions achieved through our bi-objective function, we have to take a deeper analysis, and for that, we will analyze the Pareto Front of instance_1, which is presented in Figure 6.7.

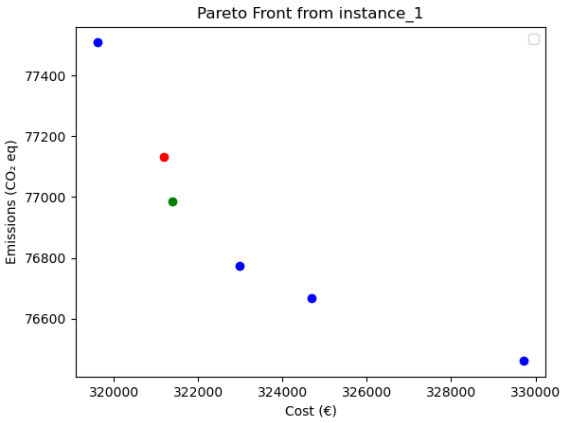


Figure 6.7: Pareto Front from instance_1.

From Figure 6.7, it can be seen six solutions found using the augmented ϵ-constraint method (see Section 4.3.2). Once we do not have a decision-maker, we cannot discuss with them in order to properly find a solution that would fit their expectations and objectives. However, we can better understand the solutions found, using a bi-objective function. First of all, the extreme solutions are the ones that only minimize the costs and emissions. One of the many advantages of being able to see the Pareto front is the ability to compare two solutions. For instance, take a decision-maker who would like to consider a decrease in their emissions produced, but still keep the costs reduced, and let z_1 be the solution for the costs and z_2 for the emissions. Let´s say that he is willing to increase its costs by 0.5% (compared with the minimum costs) if it implies a reduction of the emissions produced. If we didn't have the whole picture that the Pareto front allows us, a solution that fits this description is the one with z_1 = 321,188€ and z_2 = 77,133 CO_2 eq (red point). This solution increases the costs by 0.496% while reducing the emissions by 0.484%, which complies with the decision-maker's expectations. However, this solution increases more the costs, than it decreases the emissions. By taking a deeper look, we can see that there is another interesting solution, the one represented by the green point. Even if this solution increases the costs by 0.556%, which is more than the decision-maker is willing to accept, it decreases the emissions by 0.673%, which is significantly more than the previous solution, and consequently competitive with the red solution. Hence, the decision-maker could be tempted to choose this rotation solution, even if the costs are slightly superior than initially expected.

Assuming that the decision-maker chose the solution represented by the green point, we will compare this solution with the one when only minimizing the costs, once this is the solution that the companies' RSRP models typically would use. Table 6.3 displays the differences between both solutions in terms of deadhead and regular trips, and shunting and turn operations. Take solution_1 as the solution for the minimization of the costs, and solution_2 as the solution presented in the green point of Figure 6.7. The

only difference between both solutions is the number of deadhead trips present in the solution. Indeed, the more importance we give to the minimization of emissions, the less we provide to the minimization of the costs, and the implications stated above, can already be seen between these two solutions. Hence, a reduction in the number of deadhead trips is expected once the emissions penalization is quite significant when compared with the cost penalization of the deadhead trips.

Table 6.3: Comparison between solution_1 and solution _2.

	Shunting	Turn	Deadhead	Regular
solution_1	0	4	34	0
solution_2	0	4	31	0

In short, the addition of the minimization of the emissions' objective function brings new possible rotations for the RSRP. The objective function related to the emissions tends to benefit rotations that reduce the number of deadhead trips.

6.5 Bonification and penalizations

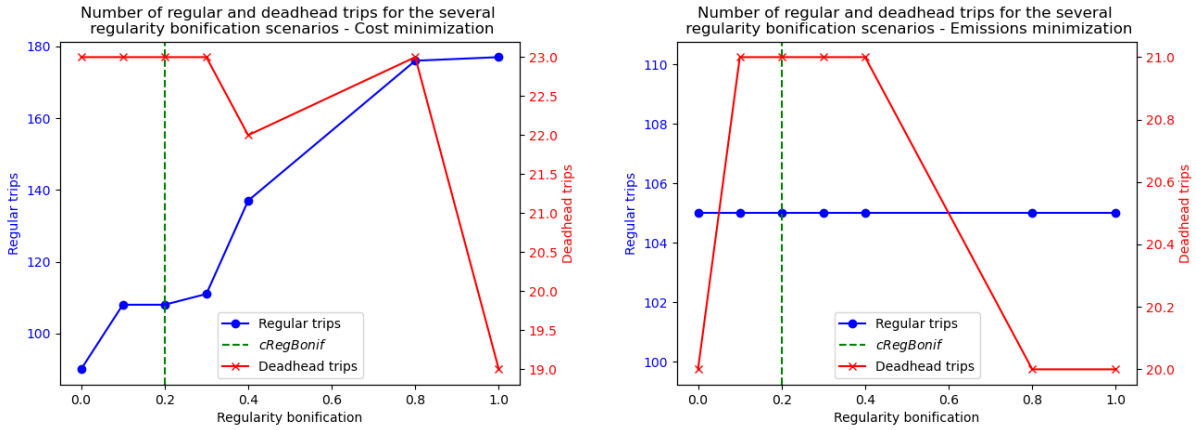
As explained in Section 3.5.1, Section 3.5.3, and Section 3.6, this version of the RSRP model includes the regularity bonification and the deadhead trips penalizations. The regularity bonification is present through the reduction of the operational trips if, for regular trips, the train composition is the same. The deadhead penalization exists for both Objective Function 4.1 (costs minimization) and Objective Function 4.2 (emissions minimization), through the increase of the costs and emissions in the deadhead trips, respectively. This leads the model to return solutions that preferentially choose the same train compositions for regular trips and exclude deadhead trips. However, this implementation can lead to divergent solutions found. Throughout this section, we will explore how these parameters impact the solutions found in our version of the RSRP via a scenario analysis.

To properly explore and analyze the three parameters in the study, we created instance_10 which is described in Section 6.1, that considers the 177 trips departing between 10h00 and 11h00 of 10/12/2020, 14/12/2020, and 16/12/2020 in the red, yellow, and green lines. For the values implemented in the model (regularity bonification = 20%; deadhead trip cost penalty = 10%; deadhead trip emissions penalty = 10%), the solution found for the cost minimization proposes 23 deadhead trips and 108 regular trips, while for the emissions minimization, the solution return 21 deadhead trips and 105 regular trips. Now that we established this, we can observe and analyze how the variation of the parameters modifies the solutions. It is to be noted that we will only analyze the variations of deadhead trips and regularity trips, once turn operations do not vary over the analysis, and that if we had considered shunting operations, the time required for the computational results would be massive.

Figure 6.8 represents the variation of the regularity bonification, $cRegBonif$, for both costs and emissions minimization, and the results obtained from these two objective functions are not the same. First

of all, the range of the regular trips in Figure 6.8 (a) is wider than for the regular trips in Figure 6.8 (b), once that for all solutions found considering the emissions minimization, the number of regular trips remain the same. Thus, the higher the regularity bonification, the more the solution (for cost minimization) contains regular trips. Once we consider a periodic timetable (see Section 3.1), the variation of the parameter is significantly present. Moreover, as the number of regular trips increases with the increase of the regularity bonification, it is to be noted that the deadhead trips tend to decrease. This shows that by varying the regularity bonification, we are also varying the number of deadhead trips for both objective functions, even if the impact is more visible for cost minimization.

Furthermore, we can analyze the difference in the solutions found for both objective functions related to this scenario analysis. For example, if we vary the regularity bonification from 20% to 30%, the number of regular trips increases, for the cost minimization. However, for the emissions minimization, the solution remains the same. We have to decrease the bonification until 10% or increase it to 40% to return solutions different than the ones with the original regularity bonification (20%). The increase and decrease of the bonification implies a decrease in the deadhead trips.

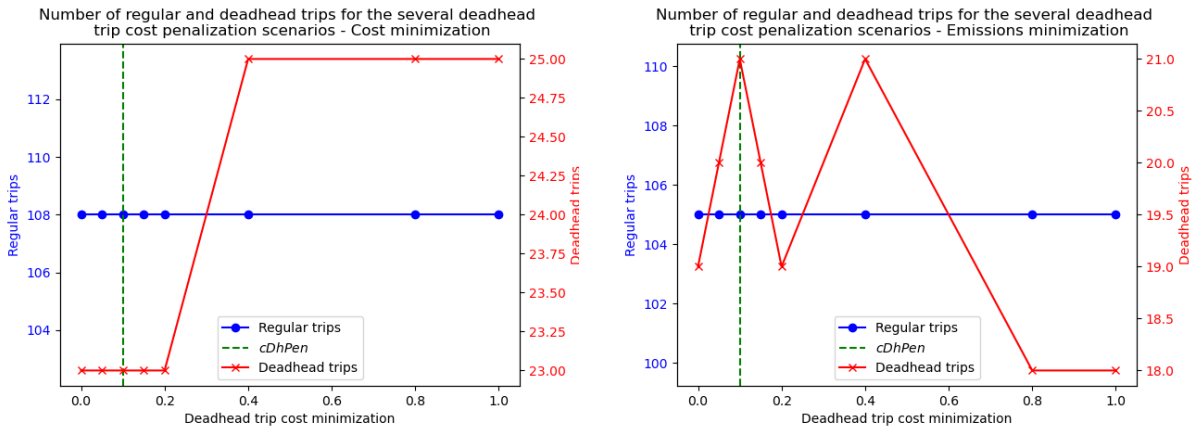


(a) Cost minimization.

(b) Emissions minimization.

Figure 6.8: Variation of the number of regular and deadhead trips with the variation of the regularity bonification.

When it comes to the analysis of the deadhead trips cost penalization's variation, the same results reached for the regularity bonification cannot be applied. From Figure 6.9 (a) and Figure 6.9 (b), the number of regular trips distinguishes itself. It appears the variation of the deadhead trips penalization does not have an impact on the solutions found in terms of regular trips. Contrary to what could be expected, the number of deadhead trips does not decrease when increasing the deadhead trips' cost penalization. We can assume that the solution found a way to increase the number of deadhead trips while decreasing the value of the costs. This can be explained if, for example, the deadhead trips are performed in such small paths that the penalization does not influence the solution. Even if the variation of the number of deadhead trips is quite irregular for the emissions minimization (Figure 6.9), it can be observed that the number of deadhead trips tends to decrease as we increase the value of the deadhead trips penalization.

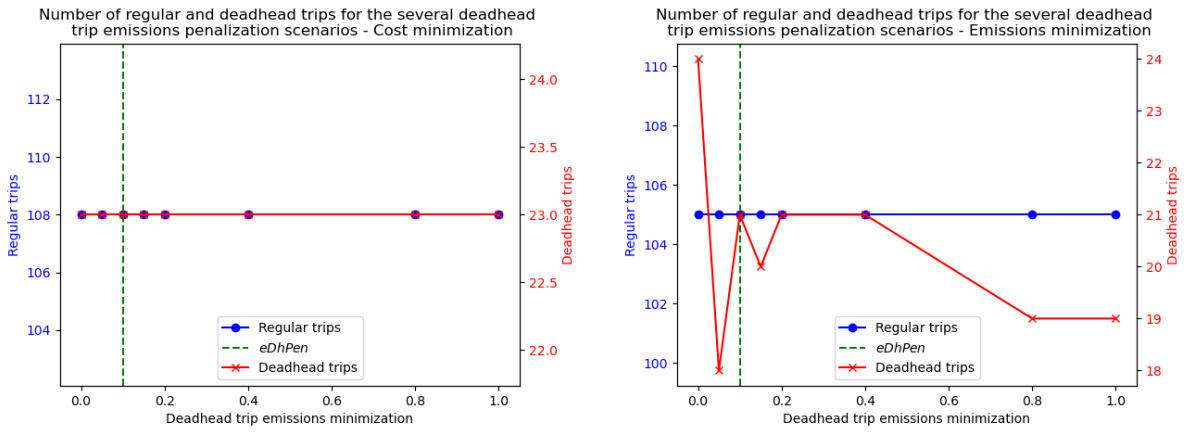


(a) Cost minimization.

(b) Emissions minimization.

Figure 6.9: Variation of the number of regular and deadhead trips with the variation of the deadhead trips cost penalization.

The deadhead trip emissions penalization is the only penalization parameter directly connected to the emissions objective function, thus it is expected that it will impact more the number of deadhead trips than the cost penalization. As it can be observed in Figure 6.10 (b), this happens, once the range for the number of deadhead trips is wider than the one for the cost penalization (Figure 6.9 (b)). The figure indicates that the variation of the deadhead emissions penalty varies with the number of deadhead trips. Even if the variation is not regular, it can be observed that the higher the penalty is, the fewer deadhead trips the solution contains, as would be expected. Once again, the number of regular trips does not change by varying the deadhead trips emissions penalty. Moreover, Figure 6.10 (a) indicates that no matter the value of the penalty considered, neither the number of deadhead trips nor the number of regular trips will vary. This outcome could be expected, once the variations for the deadhead cost penalty were already few, and thus, it is expected that when analyzing the emissions penalty, it has no impact on the solution obtained through the cost minimization.



(a) Cost minimization.

(b) Emissions minimization.

Figure 6.10: Variation of the number of regular and deadhead trips with the variation of the deadhead trips emissions penalization.

Chapter 7

Conclusions

Rolling stock rotation planning is a complex problem that deals with several requirements. It only makes sense to solve it considering a considerable number of conditions, otherwise, the solutions proposed do not fit with reality. However, the more requirements implemented, the more intricate the model and the more time-consuming is to build it.

To solve the RSRP, we used the hypergraph model, proposed by Borndörfer et al. (2011), which allows us to have a deeper understanding of the rolling stock flow, once the model enhances the modeling of the several changes in train compositions. Our version of the RSRP integrated the emissions of CO_2 equivalent produced throughout the operations related to this problem, as a second objective function, making the problem bi-objective. This feature was not implemented in any other work that we found in the literature and thus, brings a new perspective in the RSRP. Moreover, this implementation enables us to understand which differences exist between the minimization of the costs and the minimization of the emissions. To implement this new objective function in our bi-objective problem, we used the augmented ε-constraint method proposed by Mavrotas (2009).

Our version of the RSRP model is able to optimize the rotation of the rolling stock considering the requirements integrated. However, it is an intricate model, which leads to high computational times. We found that is possible to reduce the computational time, by restricting the maximum number of vehicles per train composition. The more vehicles can a train composition consider, the more complex and time-consuming the model. By using the hypergraph model, we considered every possible and feasible composition of each trip, however, it was revealed that this was unnecessary and even adverse to the computational time. Once demand is a priori known, by reducing the number of possible vehicles for the train composition, the degrees of freedom are reduced, thus the computational time is also decreased.

Moreover, there was found another advantage of reducing the maximum number of vehicles per composition, is the reduction of the shunting operations. The more possibilities of feasible compositions, with different sizes, the more hyperarcs to perform the several possible shunting operations are generated. Even with our initial limitation of one shunting operation per feasible composition, we found that hyperarcs related to the shunting operations still represented the majority of the hyperarcs, and were not present in our solution. Thus, by eliminating these possibilities, we were able to reduce even more

the degrees of freedom. Nevertheless, it is to be noted that the shunting operations are relevant in the rolling stock rotation planning. However, one way to reduce the complexity would be ignoring them or only considering them for certain stations, for example.

Furthermore, we were able to properly integrate the objective function regarding the emissions produced. Even if we had to state some assumptions it was possible to understand the implications and changes of having this new objective function. First of all, when the emissions are minimized, the model tends to avoid more deadhead trips than when only minimizing the costs. Secondly, through a bi-objective analysis, we are prepared to show a potential decision-maker the results and find unexpected solutions that can lead to low increases in the operational costs, but higher decreases in the emissions produced by the operations, which is an interesting point of view.

When it comes to the parameters (regularity bonification, deadhead trip cost penalization, and deadhead trip emissions penalization), on the one hand, it was observed that the higher the regularity bonification, the more regular trips there are, and fewer deadhead trips. On the other hand, both deadhead penalizations did not show to vary the number of regular trips, but only the number of deadhead trips. Nevertheless, the number of deadhead trips varied more for the emissions penalization than the cost penalizations. This aligns with the fact that there are several costs associated with different operational activities, which does not happen for the emissions. Thus, to achieve better and more realistic results for the emissions it is necessary to have a deeper understanding of the emissions that can be allocated to the several operations within the rolling stock rotation planning problem.

Even if our version of the rolling stock rotation planning problem using the hypergraph model works, it can be improved. To start, it was found that Algorithm 3 is not optimized, which leads to hyperarc generation times significantly higher than the resolution time. Therefore, an improvement of Algorithm 3 is a suggestion for the reader. Furthermore, due to the long hyperarc generation times, it was not possible to compute significant large instances for the Chicago subway. However, as many authors have shown, the integration of real-world data leads to extensive computational times. To solve this issue, heuristics approaches can be integrated into the model. The same can be applied in this version of the model, once the generation of the connection hyperarcs is optimized.

Another limitation that we had to deal with was regarding the data. Even if the Chicago subway open data is revealed to be satisfactory and appropriate for the RSRP, it was not enough to really understand and model all the potential requirements that the Chicago subway must have. Another limitation was the lack of data and information regarding the allocation and description of the emissions produced for rolling stock rotation operations. It is strongly recommended to collaborate with a railway company to better explore the impact of the emissions produced when facing the operations' costs, and thus, developing further this new version of the rolling stock rotation planning.

Bibliography

[1] Agency, I. E. (2023). *Rail*. Retrieved 12 September, 2023, from https://www.iea.org/energy-system/transport/rail.

[2] Ahuja, R. K., Cunha, C. B., and Şahin, G. (2005). Network Models in Railroad Planning and Scheduling. In Greenberg, H. J. and Smith, J. C., editors, *Emerging Theory, Methods, and Applications*, pages 54–101. INFORMS.

[3] Alfieri, A., Groot, R., Kroon, L., and Schrijver, A. (2006). Efficient Circulation of Railway Rolling Stock. *Transportation Science*, 40(3):378–391.

[4] Almeida, J., Santos, D., Figueira, J. R., and Francisco, A. P. (2023). A multi-objective mixed integer linear programming model for thesis defence scheduling. *European Journal of Operational Research*, page S0377221723005003.

[5] Authority, C. T. (2020). *Financial Statements and Supplementary Information*. Retrieved 26 MAY, 2023, from https://www.transitchicago.com/downloads/sch$_{d}ata/$.

[6] Borndörfer, R., Eßer, T., Frankenberger, P., Huck, A., Jobmann, C., Krostitz, B., Kuchenbecker, K., Mohrhagen, K., Nagl, P., Peterson, M., Reuther, M., Schang, T., Schoch, M., Schülldorf, H., Schütz, P., Therolf, T., Waas, K., and Weider, S. (2021). Deutsche Bahn Schedules Train Rotations Using Hypergraph Optimization. *INFORMS Journal on Applied Analytics*, 51(1):42–62.

[7] Borndörfer, R., Grimm, B., Reuther, M., and Schlechte, T. (2011). A hypergraph model for railway vehicle rotation planning.

[8] Borndörfer, R., Grimm, B., Reuther, M., and Schlechte, T. (2017). Template-based re-optimization of rolling stock rotations. *Public Transport*, 9(1-2):365–383.

[9] Borndörfer, R., Reuther, M., Schlechte, T., Waas, K., and Weider, S. (2016). Integrated Optimization of Rolling Stock Rotations for Intercity Railways. *Transportation Science*, 50(3):863–877.

[10] Borndörfer, R., Reuther, M., Schlechte, T., and Weider, S. (2012). Vehicle rotation planning for intercity railways.

[11] Cacchiani, V. and Toth, P. (2012). Nominal and robust train timetabling problems. *European Journal of Operational Research*, 219(3):727–737.

[12] Cadarso, L. and Marín, (2011). Robust rolling stock in rapid transit networks. *Computers & Operations Research*, 38(8):1131–1142.

[13] Caprara, A., Kroon, L., Monaci, M., Peeters, M., and Toth, P. (2007). Chapter 3 Passenger Railway Optimization. In *Handbooks in Operations Research and Management Science*, volume 14, pages 129–187. Elsevier.

[14] Chang, Y.-H., Yeh, C.-H., and Shen, C.-C. (2000). A multiobjective model for passenger train services planning: application to Taiwan's high-speed rail line. *Transportation Research Part B: Methodological*, 34(2):91–106.

[15] Cordeau, J.-F., Toth, P., and Vigo, D. (1998). A Survey of Optimization Models for Train Routing and Scheduling. *Transportation Science*, 32(4):380–404.

[16] Demir, E., Huang, Y., Scholts, S., and Woensel, T. V. (2015). A selected review on the negative externalities of the freight transportation: Modeling and pricing. *Transportation Research Part E: Logistics and Transportation Review*, 77:95–114.

[17] Eurostat (2022). *Railway passenger transport statistics - quarterly and annual data*. Retrieved 20 June, 2023, from https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Railway_passenger_transport_statistics_-_quarterly_and_annual_dataoldid=585131.

[18] for Transportation, C. (2022). *CBO Releases Overview of GHG Emissions in Transportation*. Retrieved 29 June, 2023, from https://enotrans.org/article/cbo-releases-overview-of-ghg-emissions-in-transportation/.

[19] Freight, R. (2022). *Collision: Berlin-Hanover line closed at least until the end of the weekend*. Retrieved DIA 03 June, 2023, from https://www.railfreight.com/railfreight/2022/11/18/collision-berlin-hanover-line-closed-at-least-until-the-end-of-the-weekend/?gdpr=accept.

[20] Freling, R., Lentink, R. M., Kroon, L. G., and Huisman, D. (2005). Shunting of Passenger Train Units in a Railway Station. *Transportation Science*, 39(2):261–272.

[21] Gao, Y., Xia, J., D'Ariano, A., and Yang, L. (2022). Weekly rolling stock planning in Chinese high-speed rail networks. *Transportation Research Part B: Methodological*, 158:295–322.

[22] Giacco, G. L., Carillo, D., D'Ariano, A., Pacciarelli, D., and Marín, G. (2014). Short-term Rail Rolling Stock Rostering and Maintenance Scheduling. *Transportation Research Procedia*, 3:651–659.

[23] Grimm, B., Borndörfer, R., Reuther, M., and Schlechte, T. (2019). A Cut Separation Approach for the Rolling Stock Rotation Problem with Vehicle Maintenance. page 12 pages. Artwork Size: 12 pages Medium: application/pdf Publisher: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany Version Number: 1.0.

[24] Haahr, J. T., Wagenaar, J. C., Veelenturf, L. P., and Kroon, L. G. (2016). A comparison of two exact methods for passenger railway rolling stock (re)scheduling. *Transportation Research Part E: Logistics and Transportation Review*, 91:15–32.

[25] Hoogervorst, R., Dollevoet, T., Maróti, G., and Huisman, D. (2021). A Variable Neighborhood Search heuristic for rolling stock rescheduling. *EURO Journal on Transportation and Logistics*, 10:100032.

[26] Hwang, C. and Masud, A. (1979). *Multiple Objective Decision Making. Methods and Applications: A State of the Art Survey*, volume 164. Springer-Verlag.

[27] Jha, K. C., Ahuja, R. K., and Şahin, G. (2008). New approaches for solving the block-to-train assignment problem. *Networks*, 51(1):48–62. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/net.20195.

[28] Krezo, S., Mirza, O., Kaewunruen, S., and Sussman, J. (2018). Evaluation of CO2 emissions from railway resurfacing maintenance activities. *Transportation Research Part D: Transport and Environment*, 65:458–465.

[29] Kroon, L., Maróti, G., Helmrich, M. R., Vromans, M., and Dekker, R. (2008). Stochastic improvement of cyclic railway timetables. *Transportation Research Part B: Methodological*, 42(6):553–570.

[30] Liebchen, C. and Möhring, R. H. (2014). The modeling power of the periodic event scheduling problem: Railway timetables — and beyond. In *Lecture Notes in Computer Science*, pages 3–40. Springer Berlin Heidelberg.

[31] Lusby, R. M., Haahr, J. T., Larsen, J., and Pisinger, D. (2017). A Branch-and-Price algorithm for railway rolling stock rescheduling. *Transportation Research Part B: Methodological*, 99:228–250.

[32] Löbel, A. (1997). Optimal vehicle scheduling in public transit.

[33] Maróti, G. and Kroon, L. (2007). Maintenance routing for train units: The interchange model. *Computers &amp Operations Research*, 34(4):1121–1140.

[34] Mavrotas, G. (2009). Effective implementation of the -constraint method in Multi-Objective Mathematical Programming problems. *Applied Mathematics and Computation*, 213(2):455–465.

[35] Mavrotas, G. and Florios, K. (2013). An improved version of the augmented -constraint method (AUGMECON2) for finding the exact pareto set in multi-objective integer programming problems. *Applied Mathematics and Computation*, 219(18):9652–9669.

[36] Miettinen, K. (1998). *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston.

[37] Peeters, M. and Kroon, L. (2008). Circulation of railway rolling stock: a branch-and-price approach. *Computers & Operations Research*, 35(2):538–556.

[38] Profillidis, V. A., Botzoris, G. N., and Galanis, A. T. (2014). Environmental effects and externalities from the transport sector and sustainable transportation planning – a review. *International Journal of Energy Economics and Policy*, 4(4):647 – 661.

[39] Railways, I. U. (2023). *UIC Activity Report*. Retrieved 12 September, 2023, from https://uic.org/?gad_source=1gclid=Cj0KCQjwhfipBhCqARIsAH9msbn_743Fbr36wnx4JGif6EKCb42Abkmp7FFg7_KZnEY81QZhZqvdXSIaAk36EALw_wcB.

[40] Saddoune, M., Desaulniers, G., Elhallaoui, I., and Soumis, F. (2012). Integrated Airline Crew Pairing and Crew Assignment by Dynamic Constraint Aggregation. *Transportation Science*, 46(1):39–55.

[41] Schlechte, T., Blome, C., Gerber, S., Hauser, S., Kasten, J., Muller, G., Schulz, C., Thuring, M., and Weider, S. (2023). The Bouquet of Features in Rolling Stock Rotation Planning.

[42] Schrijver, A. (2002). On the history of the transportation and maximum flow problems. *Mathematical Programming*, 91(3):437–445.

[43] Schwerdfeger, S., Otto, A., and Boysen, N. (2021). Rail platooning: Scheduling trains along a rail corridor with rapid-shunting facilities. *European Journal of Operational Research*, 294(2):760–778.

[44] Steuer, R. (1986). *Multiple Criteria Optimization, Theory, Computation and Application*.

[45] Thorlacius, P. (2015). Processes, Cost Structures and Requirements.

[46] Thorlacius, P., Larsen, J., and Laumanns, M. (2015). An integrated rolling stock planning model for the Copenhagen suburban passenger railway. *Journal of Rail Transport Planning & Management*, 5(4):240–262.

[47] Vaidyanathan, B., Ahuja, R. K., and Orlin, J. B. (2008). The Locomotive Routing Problem. *Transportation Science*, 42(4):492–507.

[48] Wang, Y., D'Ariano, A., Yin, J., Meng, L., Tang, T., and Ning, B. (2018). Passenger demand oriented train scheduling and rolling stock circulation planning for an urban rail transit line. *Transportation Research Part B: Methodological*, 118:193–227.

[49] Zhao, Y., Li, D., and Yin, Y. (2023). Integrated optimization of train formation plan and rolling stock scheduling under Time-dependent demand. *Computers & Operations Research*, 150:106049.