Programa de Doctorado "Matemáticas"

PhD Dissertation

RAPID TRANSIT NETWORK DESIGN AND LINE PLANNING

Author

Alicia De Los Santos Pineda

Supervisor

Prof. Dr. *Juan Antonio Mesa López-Colmenar*

Co-supervisor

Dr. *Gilbert Laporte*

September 7, 2014

UNIVERSIDAD DE SEVILLA
Departamento de Matemática Aplicada II

Diseño de Redes de Transporte Rápido y Planificación de Líneas

Rapid Transit Network Design and Line Planning

Alicia De Los Santos Pineda

Tesis doctoral

A mi hija.
A Antonio.
A mis padres.

Agradecimientos

Durante los años en los que he realizado mi tesis son muchas las personas que me han prestado su ayuda y apoyo de forma desinteresada, a quienes quiero expresar mi gratitud.

En primer lugar quiero agradecer el apoyo, dedicación y ayuda que me han brindado mis directores de tesis Juan Antonio Mesa y Gilbert Laporte. Me gustaría agradecer a Juan Antonio el haberme trasmitido su pasión por la investigación, por sus sugerencias e ideas que tanto me han servido, por las interminables correcciones, por la confianza depositada en mí, y por su respaldo y amistad. No sólo me ha enseñado a nivel profesional sino también a nivel personal, me ha mostrado que siempre hay un camino correcto para hacer las cosas, a afrontar los problemas con honestidad y a creer en mí misma. I am grateful to Gilbert for his enthusiasm and optimism, for his suggestions and ideas, for his time and dedication. I also wanted to thank his hospitality and support during my stay at the CIRRELT and HEC, providing all the facilities and resources required to develop my work. During that time I always felt at home and I have fond memories.

Me gustaría extender mi agradecimiento a los miembros del grupo de investigación LOGRO y a los del proyecto de Excelencia TEP-5022 por el apoyo mostrado, muy especialmente a David Canca, por su disponibilidad, colaboración y ayuda que tan desinteresadamente me ha brindado en los últimos años, y, por supuesto, por su amistad. También me gustaría agradecer de manera especial a Federico Perea por todo lo que he aprendido trabajando con él, por su acogida y recibimiento en su departamento, por su amistad. Gracias al Departamento de Estadística e Investigación Operativa Aplicadas y Calidad, de la Universitat Politècnica de València por su acogida y los medios recibidos.

I also want to thank Anita Schöebel and Jonas Harbering for the welcome, for effective collaboration, support and resources received during my stay at the Institute for Numerical and Applied Mathematics at the University of Göttingen.

A los miembros del Departamento de Matemática Aplicada II, de los que me gustaría destacar aquellos con los que he tenido un especial trato. A Ana Beatriz Sánchez, por su amistad y apoyo con la que he compartido despacho en estos últimos meses; ha sido un placer tenerte como compañera. Es sorprendente que haya personas que, sin apenas conocerte, te estén dando la mano y prestando su ayuda, como Nieves Jiménez, quien desde el primer día se ofreció tan desinteresadamente a ayudarme con mis clases. A los miembros del grupo “café”, con los que he tenido agradables desayunos. A Mª Carmen Sánchez, por tener siempre hueco para atenderme y por su trato tan cordial. A todos los doctorandos/as y nuevos doctores; Inés Gallego, Soledad Fernández, José Manuel

Gallardo y Elísabeth García.

No puedo olvidar a mis compañeras o, mejor dicho, amigas; Elísabet Vela, Mónica Molina y Eva Barrena. Ha sido un placer tener como compañera de despacho a una gran amiga como Eli, siempre atenta y sincera. Gracias por estar en todo momento. A Mónica por su apoyo y sobretodo por su amistad, no puedo olvidar lo buena amiga que ha sido en los momentos tan difíciles que he pasado. A Eva, con la que he compartido despacho e incontables horas de trabajo. Gracias por escucharme, por tu humor, por tu generosidad y sobretodo, por esas risas que hemos echado. A Fernando Fernández por sus consejos, que tanto me han ayudado y por su opinión siempre tan acertada. Gracias por describir con tanto entusiasmo ese viaje que tengo pendiente. A mis amigas Guadalupe, Amanda, Yoli y Mª del Mar, que siempre han estado a mi lado a pesar de la distancia.

Para terminar, el agradecimiento más profundo y sentido va para mi familia. A mi hija Laura, por todo el tiempo que me he perdido de ella, por soportar tantas horas sin la compañía de su madre, por tener que entender el tenerme al lado sin poder jugar o hacer planes juntas. Para la persona que me ha enseñado a ser quien soy, quien, a pesar de no poder estar físicamente en estos últimos años, siempre lo he sentido cerca, recordando las veces que me decía "tú puedes, ya verás cómo lo consigues, creo en tí". A mi madre por todo el sacrificio y apoyo incondicional que me ha brindado; sin tí esto no hubiera sido posible, gracias por hacer que sea lo que hoy en día soy. A Antonio, por su paciencia, por tenderme su mano cada vez que he caído, por sacrificarse tanto por mí, sobretodo, por darme ánimos usando ejemplos de fútbol. Quisiera expresar también mi agradecimiento a mis hermanos y hermanas por todo lo mucho que me han apoyado y animado. Extiendo mi agradecimiento a mis suegros, siempre disponibles y atentos; sin vuestra ayuda no podría haber seguido adelante.

Introduction

The transport sector is a key factor for a society continuously growing. The transport systems can be oriented to move passengers or goods, or both cases. Transport provides mobility of people, access to employment, development, improving the welfare of a society. An effective transport makes accessible isolated regions and easy the day-to-day life. Immersed in a world of constant evolution is difficult to think a future without an efficient and ecologic transport.

Operations research can help transport planning process. These problems can be described and analyzed by means of mathematical models and efficient approaches.

This thesis focusses on Rapid Transit Systems (RTS) which includes metro, bus rapid transit (BRT), light rail transit, monorail, etc. Bus rapid transit is a RTS special case which should be studied in a different way. So, some models in this thesis cannot be applied on a BRT. In the area of passenger transport, much effort has been devoted to improve the mobility of people, so reducing the traffic congestion, energy consumption and pollution. In the railway context, the rapid transit planning process has traditionally been decomposed into a succession of stages, namely, network design, line design, timetabling, rolling stock, and personnel planning. In RTS with the exception of railway systems, network design and line planning (without frequency) define an only stage. During last years it can be observed a certain trend to integrate several stages of the railway planning process. Recently contributions in this field, integrating network design and line planning have been proposed. Several authors Guihaire and Hao (2008), Goerigk et al. (2013), Michaelis and Schöbel (2009), Marín et al. (2009), Zhu (2011) consider that, if possible, the integration of several planning stages will conduce to better solutions. From an optimization point of view, the solution of an integrated multi-stage problem is preferable to a succession of optimal solutions to single-stage problems. Obviously, solving the whole problem is more difficult, but taking decisions stage by stage (design, line planning,..) will lead to worse solutions than if some of these phases are integrated.

One proposal of this thesis is to integrate network design and line planning problems. Under this perspective, we are interested in determining simultaneously the infrastructure network, line planning, train capacity of each line, fleet investment and personnel planning. Moreover, we incorporate the traffic assignment procedure in the optimization process. It can be observed that when the infrastructure network is built, it is difficult if not impossible to change it. Therefore, if the locations of the stations and of theirs connections are selected at the first stage, the solutions at the following stages (line configuration as well as the train capacity and frequencies) are conditioned by them. Another relevant aspect to take into account in the rapid transit network design, is the estimated demand. The main characteristics of an RTS are set according to the demand. So, it is important to introduce a competing mode in the model and determine rolling stock levels based on the captured demand. For these reasons, integrating line and station location as well as train capacities and frequencies at the planning stage leads to better solutions.

As mentioned, the resulting problems by the integration of network design and line planning are difficult to solve and they require efficient techniques for solving the problem. In this thesis we develop mathematical models, efficient techniques and algorithms. Another important contribution in this thesis is the realistic treatment of the problem. Thus, we present a rigorous analysis for the calibration of the aspects that appear as a consequence of the integration of network design and line planning. Moreover, in a realistic situation, several input data such as the origin-destination matrix, travel times by the alternative mode, costs can be uncertain (see Chapter 5).

In this thesis we are also interested in evaluating and analyzing networks by means of measures in the connectivity and robustness context as well as in studying RTS line planning problems.

The remainder of this thesis is organized as follows. In Chapter 1 we present a review of robustness measures in the transportation context. These measures can be used as objective functions in rapid transit network design and line planning problems. Moreover, robustness measures can be applied to determine where the network is more vulnerable and to compare different types of networks. We propose new measures based on passenger's perspective: passenger robustness measures, connectivity measures and passenger-oriented transferability measures.

In Chapter 2 we present a review of rapid transit network design problems and we analyze the different models found in the literature. The main novelty of this chapter is the consideration of a general model that contains as particular cases, all models related with this problem studied in the literature. Another important contribution in this chapter is the realistic treatment of the problem and the incorporation of a long term public economic support in the network design problem. Moreover, with respect to recent published works, this work goes one step further by considering train capacities and frequencies at the planning stage in order to adjust the number of users using the RTS and therefore, an adequate line planning and rolling stock. Specifically, the problem we deal with integrates network design and line planning including frequency and capacity of each line as decision variable. We assume a competing mode and integrate the traffic assignment procedure in the optimization process. So, passengers are assigned to each mode in a continuous way by means of a logit function and continuous variables. Chapter 3 is devoted to line planning problems. Specifically, we simultaneously select the frequency and the number of carriages for each line of the RTS maximizing the net profit. We distinguish two possible situations: an unlimited number of carriages and a maximum number of carriages. Last problem may yield to congested networks and it requires a special treatment. We develop a mathematical model for the first problem as well as efficient approaches such as linearization constraints and algorithms. For the capacitated problem, we propose an approach and an algorithm which include a congestion function measuring the level of in-vehicle crowding on each arc.

In Chapter 4 we review the different algorithms found in the literature about network design and line planning. We propose a model to solve our problem, considering only one route for each OD pair in the RTS. In this problem passengers choose between a competing mode and the shortest path in the RTS.

Then, in Chapter 5 we introduce some robustness aspects in the rapid transit network design problem. Concretely, we are interested in obtaining feasible solutions under uncertain circumstances in the infrastructure network design problem. We study different approaches of robustness that can be applied on such problem.

Finally, we end with some conclusions.

Contents

Agradecimientos **v**

Introduction **iii**

1. Rapid transit networks: robustness and related measure assessment **1**

1.1. Introduction . . . 1

1.1.1. Representation of transportation networks . . . 2

1.2. Measures in the infrastructure network . . . 5

1.2.1. Mean connectivity . . . 5

1.2.2. Pair disconnection measure . . . 6

1.2.3. Toughness . . . 7

1.2.4. Diameter . . . 8

1.2.5. Characteristic path length and Efficiency indicator . . . 9

1.2.6. Vulnerability indicator . . . 13

1.3. Measures in the metric network . . . 15

1.3.1. Diameter . . . 15

1.3.2. Average shortest distance . . . 17

1.3.3. Global and local Efficiency indicators . . . 17

1.4. Measures for networks with demand patterns in operation . . . 18

1.4.1. Measures for networks with demand patterns . . . 18

1.4.2. Measures in operations . . . 22

1.5. Transferability measures . . . 23

1.5.1. Representations of CTLNs by means of graph and hypergraphs . . . 24

1.5.2. Connectivity measures in $L(\mathbb{H})$ and $L^M(\mathbb{H})$ . . . 25

1.5.3. Connectivity measures in $\mathbb{H}$ . . . 27

1.5.4. Passenger-oriented transferability measures . . . 29

1.5.5. Passenger-oriented measures in $L(\mathbb{H})$ and $L^M(\mathbb{H})$ 29
1.5.6. Passenger-oriented measures in $\mathbb{H}$ 32
1.6. Conclusions 34

2. Modelling the rapid transit network design problem 35
2.1. Introduction 35
2.2. Objectives 36
2.3. A mathematical programming model 39
2.3.1. Data and notation 40
2.3.2. Variables 42
2.3.3. Objective functions 43
2.3.4. Constraints 49
2.4. Special cases 52
2.4.1. An integrated methodology for the rapid transit network 53
2.4.2. Analysis of the parameters of transfers in rapid transit 54
2.4.3. Simultaneous optimization of transit line configuration 55
2.4.4. An extension to rapid transit network design problem 57
2.4.5. Location of infrastructure in urban railway networks 57
2.4.6. Urban rapid transit network capacity expansion 58
2.4.7. An approach for solving a modification of the extended rapid transit 60
2.4.8. Urban rapid transit network design: accelerated Benders 61
2.4.9. Rapid transit network design using simulated annealing 62
2.4.10. Rapid transit network design for optimal cost 63
2.4.11. A game theoretic framework for the Robust Railway 63
2.4.12. Designing robust rapid transit networks with alternative routes . . 64
2.4.13. GRASP algorithms for the robust railway network design 65
2.4.14. Recoverable robustness in rapid transit network design 66
2.4.15. Summary table 66
2.5. Conclusions 67

A. Data, notation and variables 69
A.1. Data and notation 69
A.2. Variables 70

B. Tables 73

3. Simultaneous frequency and capacity problem 79
3.1. Introduction 79
3.2. The problem 82
3.2.1. Data and notation 82
3.3. Uncapacitated problem 84
3.3.1. A mathematical programming program 84
3.3.2. Efficient approaches 89
3.4. Capacitated problem 97
3.4.1. Two algorithms for our problem 101
3.5. Computational experiments 104
3.5.1. Parameter setting 105
3.5.2. Computational comments 109
3.6. Conclusions 111

C. Computational results 113
C.0.1. Computational experiments for the uncapacitated problem 113
C.0.2. Computational experiments for the capacitated problem 125

4. An adaptive neighborhood search heuristic for metro network design 135
4.1. Introduction 135
4.1.1. Literature review on algorithms for RTND 136
4.2. The mathematical programming program 149
4.2.1. Data and notation 149
4.2.2. Variable 149
4.2.3. Constraints 151
4.2.4. Efficient approaches 153
4.3. The heuristic. Adaptive large neighborhood search heuristic 156
4.3.1. Initial solution 160
4.3.2. Operators 160
4.4. Computational results 162
4.4.1. Instance generation 164
4.4.2. Parameters setting 166
4.4.3. Experiments on a small network 167
4.4.4. Experiments on a medium network 172
4.5. Conclusions 174

5. Robust network design **179**
5.1. Introduction . . . 179
5.2. A mathematical programming model . . . 180
5.2.1. Data and notation . . . 180
5.2.2. Variables . . . 181
5.2.3. Objective function . . . 181
5.2.4. Constraints . . . 182
5.2.5. Problem reformulation . . . 183
5.3. Robustness in rapid transit network design . . . 184
5.3.1. Possible uncertainty sets . . . 185
5.4. Robustness regarding uncertainty set $\mathcal{U}_1$ . . . 186
5.5. Robustness regarding uncertainty set $\mathcal{U}_1^{\Gamma}$ . . . 188
5.6. Light robustness . . . 192
5.6.1. On the uncertainty $\mathcal{U}_1$ . . . 193
5.6.2. On the uncertainty $\mathcal{U}_{1,2}^{\Gamma}$ . . . 194
5.7. Conclusions and further work . . . 195

Conclusions **197**

List of Figures **199**

List of Tables **201**

List of Algorithms **203**

Bibliography **205**

Index **215**

Curriculum Vitae **217**

Chapter 1

Rapid transit networks: robustness and related measure assessment

1.1. Introduction

In this chapter we will concentrate on describing and analyzing the different measures found in the literature in the transportation networks context. These measures can be oriented regarding to connectivity and robustness.

The well-known *Graph theory* provides a natural way to represent networks as has been shown in the past. Graph theory was first developed to solve a transportation problem: Euler (1741) famously analyzed the Seven Bridges of Königsberg and showed it was not possible to design a path crossing each bridge exactly once. The application of graph theory to road transportation systems emerged in the late 1950´s. It was mainly treated from the economics point of view. Garrison and Marble (1962, 1964) pioneered in the field by introducing three graph theory measures directly linked to network design (circuits α, degree of connectivity γ and complexity β). At the same time, Kansky (1963) defined new indicators related to complexity and network specificities. The application of graph theory to urban transit systems emerged in the early 1980´s. More recently, Gattuso and Miriello (2005) have applied Garrison and Marble´s and Kansky´s indicators and others to 13 metro networks.

The concept of **robustness** has been studied in both computer science and in engineering. According to *the Glossary of the Institute of Electrical and Electronics Engineers*

(IEEE) Geraci (1991), robustness can be defined as "the degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions." Gribble (2001) defined system robustness as "the ability of a system to continue to operate correctly across a wide range of operational conditions, and to fail gracefully outside of that range." Ali et al. (2003) have considered an resource allocation mapping to be robust if it "guarantees the maintenance of certain desired system characteristics despite fluctuations in the behavior of its component parts or its environment". In transportation systems, Immers et al. (2004) define the robustness as "the degree to which a system is capable of functioning according to its design specifications in the case of a serious disruptions".

According to Holmgren (2007): "Robustness signifies that the system will retain its system structure (function) intact (remain unchanged or nearly unchanged) when exposed to perturbations." In Nagurney and Qiang (2009) transportation network robustness has also been quantified in presence of degradable links.

In rapid transit system planning, robustness can be defined with respect to fluctuations in the input parameters, (i.e., the parameters are estimations), with respect to disturbances or disruptions (fails in links, drops in electrical power, trains breaking down, etc.), or with respect to integration with other subsequent planning phases. The topological configuration of associated to a transportation network may dramatically affect the system robustness.

In this chapter we will concentrated on the robustness of a RTS and we will consider the concept of robustness proposed by Immers et al. (2004). So, the robustness will be treated respect to disturbances or disruptions in the rapid transit system.

According to different aspects, we can define several measures. These measures can be used as objective functions in rapid transit network design and line planning problem. Moreover, robustness measures can be applied to determine where the network is more vulnerable and to compare different types networks.

1.1.1. Representation of transportation networks

As mentioned, Graph Theory provides a natural way to represent networks. A graph $G(N, E)$ is an abstract object defined by means of two sets: a set N representing a finite set of elements called *nodes*, and a set E formed by pairs of elements in N. Let n and m be the number of elements of N and E, respectively. Depending on the kind of relationship defined in E, a graph is classified as undirected or directed graph. In an undirected graph $G(N, E)$, each element of E called *edge*, represents a connection between two different nodes i, j of N. An edge is usually denoted by $\{i, j\}$, e_{ij} or e. In a directed graph $G(N, A)$ the order in each element of A is an important aspect. An element (i, j) of A called arc, has an origin $i \in N$ and a destination $j \in N$.

If an edge occurs several times in E, the copies of such edge are called parallel edges and the corresponding graph is known as multigraph.

For determined situations, it is useful to associate numerical values (weights) to edges or nodes of an undirected or directed graph. The weighted graphs are usually called *networks*. The weights defined on edges (resp. arcs) can be expressed as a function $\omega : E \rightarrow \mathbb{R}$ (resp. $\omega : A \rightarrow \mathbb{R}$) which assigns a weight $\omega(e)$ (resp a weight $\omega(a)$) to each edge $e \in E$ (resp. arc $a \in A$). Depending on the context, these weights can describe different aspects such as cost, travel time, distance, capacity, strength of interaction, etc. From a network perspective, three different embedded and overlapping layers can be distinguished:

- A first layer; the infrastructure network, in which only nodes (stations) and edges (links between adjacent stations) are considered.

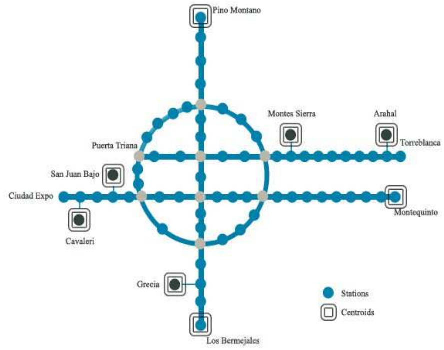
- A second layer; the line network, which represents the line set of a RTS. Each line is characterized by its itinerary and frequency. Note that this layer is defined on the first layer where trains run. In order to help passengers, this layer is usually shown at platforms by means of maps.

- A third layer; the passenger system, in which the mobility patterns of users are taken into account. At this case, passengers travel according to the itinerary of lines given in the second layer.

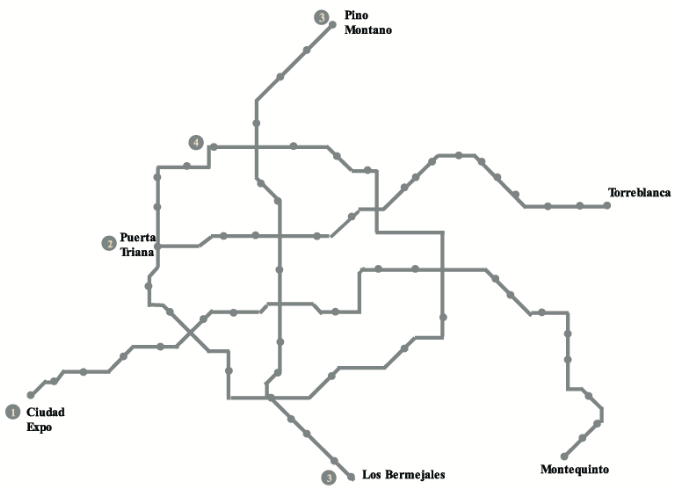
Two possible abstraction levels can be considered at the first layer: topological and metric. In Figure 1.1 we depict the different layers associated to metro of Seville.

The structure of this chapter is as follows. In Section 1.2 we describe measures related to the infrastructure network of rapid transit systems. The Section 1.3 is devoted to define and analyze measures on weighted networks. We have presented measures defined on the above mentioned third layer in Section 1.4. In Section 1.5 we formally describe representations of a collective transportation network as well as connectivity and transferability measures. We will end with some conclusions.

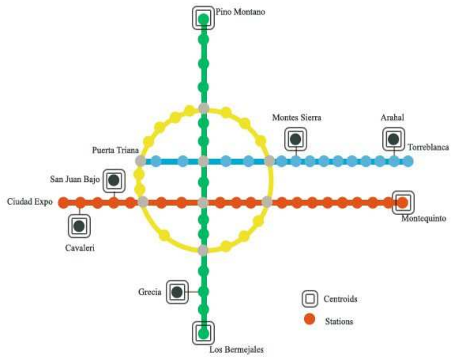
(a) Metro network of Seville.



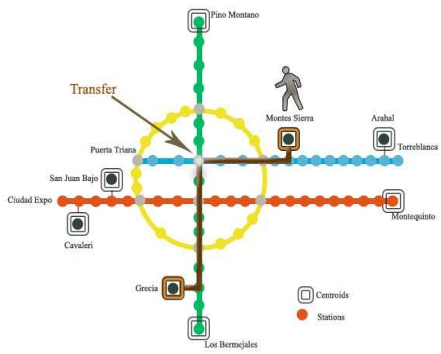
(b) Topological level abstraction (first layer).



(c) Metric level abstraction (first layer).



(d) Line network (second layer).



(e) Passenger routes (third layer).

Figure 1.1.: Different layers associated to the metro network of Seville.

1.2. Measures in the infrastructure network

Let $G = (N, E)$ be a network representing the infrastructure network of a rapid transit system, where N is the station set and E the set of links connecting adjacent stations. Let n and m be the number of stations and edge in G, respectively. Over this level, we consider the topological distance d_{ij} between two nodes i and j, i.e., the number of edges that the shortest path between i and j contains.

Following sections are devoted to describe the different measures that can be found in the literature related to robustness measures of transportation networks in the topological context.

1.2.1. Mean connectivity

The *mean connectivity measure* defined by Tainitier (1975), measures the probability of disconnecting a network after eliminating a set of edges. This measure can be useful in determining whether the connectivity of the network can be improved by adding some edges. Let $\varepsilon(G)$ be the set formed by the $m!$ possible orderings of the edge set and ϵ be an element of $\varepsilon(G)$ representing a possible ordering of the edge set. Starting with a network $G = (N, \emptyset)$, we will add edges in the order ϵ until G becomes a connected network. Let $\xi(\epsilon)$ be the position of the edge in ϵ that makes connected the network.

The mean connectivity of G is defined as follows:

$$\mathcal{M}(G) = m - \frac{1}{m!} \sum_{\epsilon \in \varepsilon(G)} \xi(\epsilon). \tag{1.1}$$

Tainitier proved that this measure satisfies several properties:

1. $\mathcal{M}(G') \leq \mathcal{M}(G)$, for $G' = (N, E')$ with $E' \subset E$.
2. $\lambda(G) - 1 \leq \mathcal{M}(G) \leq m - n + 1$, where $\lambda(G)$ is the edge-connectivity of G, i.e., is the smallest number of edges whose removal from G results in a non-connected network.

For the sake of clarification, we show the following example.

Example 1.2.1 *The following Figure 1.2 represents a network with mean connectivity equal to* 4. *It can be note that* $\varepsilon(G)$ *is formed by* 120 *elements as follows:*

- $\epsilon_1 = [(1, 2), (2, 3), (1, 4), (2, 4), (4, 5)]$. *Note that* $\varepsilon(\epsilon_1) = 5$*, since we need to add all edges to* $G(N, \emptyset)$ *to transform* G *in a connected network.*

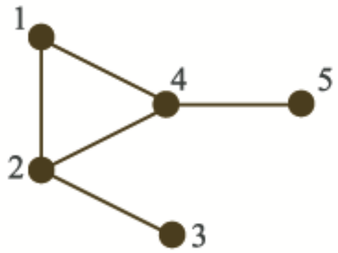


Figure 1.2.: Computation of the mean connectivity for a sample network.

- $\epsilon_2 = [(1,2),(2,3),(1,4),(4,5),(2,4)]$. *In this case,* $\varepsilon(\epsilon_2) = 4$*, since if we introduce the edges* $(1,2),(2,3),(1,4),(4,5)$ *the network is connected.*

- $\epsilon_3 = [(1,2),(2,3),(2,4),(1,4),(4,5)]$, $\varepsilon(\epsilon_3) = 5$. *As in* ϵ_1*, we need to add all edges to transform* G *in a connected network.*

- $\epsilon_3 = [(1,2),(2,3),(4,5),(2,4),(1,4)]$, $\varepsilon(\epsilon_4) = 4$. *In this case the network is connected with* 4 *edges.*

- $\vdots$

- $\epsilon_{120} = [(1,2),(2,3),(1,4),(2,4),(4,5)]$, $\varepsilon(\epsilon_{120}) = 5$. *We need to insert all edges in* G*.*

1.2.2. Pair disconnection measure

An important measure proposed by Ng and Efstathiou (2006) to evaluate how the nodes of a network are connected is the *network disconnectedness.* This measure can be applied for both connected and non-connected networks. Concretely, if G is a non-connected network, the *network disconnectedness* is the ratio between the number of pairs of unreachable nodes and the maximum number of possible pairs of nodes $(n(n-1)/2)$. In contrast, if G is connected, is interesting to study this ratio when a node or an edge is eliminated of G. The network disconnectedness is defined as

$$pd(i) = pnd(i)/[n(n-1)/2], \quad pd(e) = pnd(e)/[n(n-1)/2], \tag{1.2}$$

where $pnd(i)$ denotes the number of unreachable node pairs when the node i is eliminated and $pnd(e)$ is the number of unreachable node pairs when the edge e is interrupted.

From this measure, it can be defined the worst (wc) and average (aver) case for pair disconnection measure.

$$
\begin{aligned}
pd_N^{wc}(G) &= \max_{i \in N} pd(i), \qquad pd_E^{wc}(G) = \max_{e \in E} pd(e) \\
pd_N^{aver}(G) &= \frac{1}{n} \sum_{i \in N} pd(i), \quad pd_E^{aver}(G) = \frac{1}{m} \sum_{e \in E} pd(e)
\end{aligned}
\tag{1.3}
$$

We observe that this measure satisfies several desirable properties: it lies within a predefined range and satisfies a monotonicity property.

Proposition 1.1 *Let $G = (N, E)$ be a connected network representing a rapid transit system. The pair disconnection measure holds the following properties.*

1. $0 \leq pd_N^{wc}(G) \leq (n-2)/n$ *and* $0 \leq pd_E^{wc}(G) \leq 1$. *The nearer to* 0, *better communicated the nodes are.*

2. *It is monotone non-decreasing in the sense that, if G' is a network obtained when adding a new edge to G connecting two nodes, we have that $pd_N^{wc}(G') \leq pd_N^{wc}(G)$ and $pd_E^{wc}(G') \leq pd_E^{wc}(G)$.*

Proof 1.2

1. *Note that the denominator of $pnd(i)$ and $pnd(e)$ do not depended on i. Thus, $\max_{i \in N} pd(i) = \max_{i \in N} pnd(i)/[n(n-1)/2]$.*

   The worst case respect to $pnd(i)$ is obtained for a star network which has all nodes connected to a central node. If the central node i is eliminated, there not exists edge in $G - i$, i.e., $G - i = (N \setminus \{i\}, \emptyset)$ in whose case $pnd(i) = (n-1)(n-2)/2$. Analogously, the worst case to $pd_E(G)$ is obtained in a graph with two nodes and one edge: $N = \{i, j\}, E = \{e\}, e = \{i, j\}$, in whose case, $pd_E^{wc}(G) = 1$.

   The minimum value (best case) is reached in a completely connected network K_n with n nodes and all possible edges, in whose case, $pd_E^{wc}(K_n) = 0$ and $pd_N^{wc}(G) = 0$.

2. *Adding a new edge in G provides a new couple of connected nodes, which implies that $pd(i)$ and $pd(e)$ are non-decreasing.*

1.2.3. Toughness

The *toughness* of a network was introduced by Chvátal (2006). It measures the number of connected components in which the network can be decomposed by the failure of a certain number of nodes. The toughness is the minimum ratio between the cardinality of a subset S of N and the number of connected components resulting after eliminating the set of nodes S:

$$\mathcal{T}(G) = \min_{S \subset N, K(G-S)>1} \left\{ \frac{|S|}{K(G-S)} \right\} \tag{1.4}$$

where $K(G-S)$ is the number of connected components that G is split into when removing S.

The minimum value is $1/(n-1)$ obtained in a start network, i.e., a network in which all nodes are connected to a central and there not exists more connection between them. The maximum value can be ∞, for the case of a completely connected network.

Now, we compute the toughness in network shown in Figure 1.3.

Example 1.2.2 *For the network of 1.3 the toughness is* $\mathcal{T}(G) = \min\{1/2, 1, 1\} = 1/2$. *In this network, the family of subset* S *of* N *is the follow:*

$$\begin{aligned}&\{\{1\},\{2\},\{3\},\{4\},\\ &\{1,2\},\{1,3\},\{1,4\},\{2,3\},\{2,4\},\{3,4\},\\ &\{1,2,3\},\{1,2,4\},\{1,3,4\},\{2,3,4\}\}.\end{aligned}$$

For $S_1 = \{3\}, S_2 = \{1,3\}$ *and* $S_3 = \{2,3\}$, $K(G - S_i) = 2$. *In other case,* $K(G-S) = 1$.

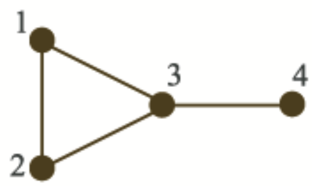


Figure 1.3.: Computation of the toughness for a sample network.

1.2.4. Diameter

The *diameter* $\mathcal{D}(G)$ of a network is the longest shortest path between any pair of nodes of G. Dekker and Colbert (2004) and Ng and Efstathiou (2006) consider that this measure is interesting from robustness point of view when G is a connected network.

It can be note that the diameter may increase if an edge or node fails and it may become infinite if it disconnects the network. Another interesting aspect in the transportation context is that longer diameters imply longer time for completing trips and therefore higher loads of the edges. If G is a non-connected network and $CG_1, \ldots CG_k$ are its connected components, we propose the following measure based on its connected components,

$$\overline{\mathcal{D}}(G) = \max_{i=1,\ldots k} \mathcal{D}(CG_i). \tag{1.5}$$

This measure gives information on the connectivity of networks. Let $G-e$ be the network that result after eliminating an edge e. From this measure, we define two interesting robustness measures: the maximum and average increase of the diameter when an edge is eliminated. So, the maximum increase is defined as

$$\Delta_E^{wc}\mathcal{D}(G) = \max_{e \in E} |\mathcal{D}(G-e) - \mathcal{D}(G)|, \tag{1.6}$$

and the average increase

$$\Delta_E^{aver}\mathcal{D}(G) = \frac{1}{n}\sum_{e \in E} |\mathcal{D}(G-e) - \mathcal{D}(G)|. \tag{1.7}$$

These measures can be extended to node eliminations taking into account that when a node is eliminated, all incident edges are also removed.

It can be observed that if $G-e$ is a connected network, its corresponding diameter increase is a positive number, i.e., $\Delta_E\mathcal{D}(G) = \mathcal{D}(G-e) - \mathcal{D}(G) \geq 0$. In contrast, if $G-e$ is a non-connected network, the diameter of $G-e$ is infinity. Last case, we propose to analyze the connected components as in (1.5).

1.2.5. Characteristic path length and Efficiency indicator

Watts and Strogatz (1998) introduced a class of networks named *small-world*, in analogy with the concept of small-world phenomenon developed in social psychology. They found that these systems can be highly clustered like regular lattices as well as that such networks have small characteristic path length like random graphs. This class of graphs interpolates between a regular lattice and random graph. The mathematical characterization of the small-world behavior proposed by Watts and Strogatz (1998) is based on the evaluation of two measures; the characteristic path length $\mathfrak{L}$ (see (1.8)) and the clustering coefficient $\mathcal{C}$ (see (1.10)) which will be defined in this section. By means of these mesures they studied the topological properties of real networks. However, these measures suffer several limitations: only the topological network is taken into account, in which information about the physical length of each link is unknown, multiple edges between the same couple of nodes are not allowed and, $\mathfrak{L}$ and $\mathcal{C}$ are ill-defined in some cases, as for example when the network is non-connected network.

In order to overcome the drawbacks previously commented, Latora and Marchiori (2001, 2002, 2004, 2008) and Crucitti et al. (2002) proposed new measures valid both for weighted and unweighted networks: the global and local efficiency. They argued that the global efficiency $\mathcal{E}_{glob}$ plays a role of the inverse of the characteristic path length $\mathfrak{L}$ and the local efficiency $\mathcal{E}_{loc}$ a similar one to the clustering coefficient $\mathcal{C}$. By means of these measures, the description of networks in terms of efficiency extends the small-world analysis to unconnected networks.

Barabási and Albert (1999), Albert et al. (1999), Barabási et al. (1999), Jeong et al. (2001) and Jeong et al. (2000) have studied the degree distribution of a network $P(k)$ defined as the proportion of nodes with k incident edges. The authors found that many large networks as the World Wide Web, Internet, metabolic networks and protein networks are classified as *scale-free*, i.e., they have a power-law degree distribution $P(k) \sim K^{-\gamma}$, where γ is an exponent that often varies between 2 and 3.

The connectivity of a scale-free network is concentrated in a few highly connected nodes. These networks are vulnerable to attack but not to random failure since the probability of failing a node is quite small. Thus, the scale-free networks are robust under random failures but they are extremely vulnerable to attacks. However, the small-world networks are robust to attacks and are vulnerable to random failures.

Following sections will be devoted to describe the measures above mentioned.

Characteristic Path Length and Global Efficiency

The average distance between stations, known as *the characteristic path length*, is defined as

$$\mathfrak{L}(G) = \frac{1}{n(n-1)} \sum_{i,j \in N} d_{ij}. \tag{1.8}$$

Note that $\mathfrak{L}(G)$ measures the average separation between two nodes in a connected network. It becomes interesting to study how a network is affected by the elimination of nodes or edges. However, at this analysis, the original network may be transformed into a non-connected network. In order to avoid this problem and to extend the analysis to non-connected networks, Latora and Marchiori (2001, 2002), Crucitti et al. (2002) have defined the global efficiency $\mathcal{E}_{glob}$.

The efficiency between node i and j, ϵ_{ij}, is assumed to be inversely proportional to the shortest path length, i.e., $\epsilon_{ij} = 1/d_{ij}$. When there is not path linking i and j it is assumed that $d_{ij} = \infty$ and $\epsilon_{ij} = 0$. So, the global efficiency of a network G (connected or non-connected) is the average measure of all possible ϵ_{ij}, that is,

$$\mathcal{E}_{glob}(G) = \frac{1}{n(n-1)} \sum_{i,j \in N} \epsilon_{ij}. \tag{1.9}$$

At the topological case, $\mathcal{E}_{glob}$ is a positive measure and it cannot exceed 1. As a consequence, this measure allows to compare different networks. Extension to weighted networks will be presented in Section 1.3.

Local Degree of Clustering and Local Efficiency

An important concept, which comes from social network analysis is the transitivity. In a social system there is a strong probability a friend of your friend is also your friend. The most common way to measure the transitivity of a network G is by means of the fraction of transitive triples, i.e. the fraction of connected triples of nodes which also form triangles of interactions. This measure can be written as Newman (2001):

$$\mathfrak{T}(G) = \frac{3 \times \text{ number of triangles in } G}{\text{number of possible triples of nodes in } G}.$$

$\mathfrak{T}(G)$ is a classical measure used in social sciences to indicate how much, locally, a network is clustered. Note that the number of possible triples of nodes in G is $\binom{n}{3}$.

Watts and Strogatz (1998) used another measure to evaluate the local degree of clustering. The authors defined the well-known *clustering coefficient* $\mathcal{C}$ as follows. For each node $i \in N$, the subgraph $G_i = (N_i, E_i)$ formed by all first neighbors of i is considered. In G_i, node i and all edges incidents to i are eliminated. So, if node i has k_i neighbors, then G_i will have k_i nodes and at most $k_i(k_i - 1)/2$ edges. Let $\mathcal{C}_i$ be the proportion of these edges that really exist. The clustering coefficient $\mathcal{C}$ is the average of $\mathcal{C}_i$, calculated over all nodes:

$$\mathcal{C} = \frac{1}{n} \sum_{i \in N} \mathcal{C}_i, \tag{1.10}$$

where

$$\mathcal{C}_i = \frac{\text{number of edges in } G_i}{k_i(k_i - 1)/2}.$$

Example 1.2.3 *We consider the network associated to Figure 1.4. In Figure 1.5 we*

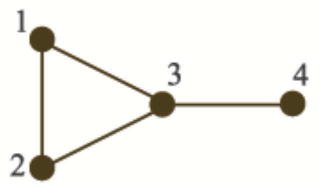


Figure 1.4.: Computation of the clustering coefficient for a small network.

depict the neighbor subgraph G_i *associated to each node* i*. In this way, we obtain* $\mathcal{C}_1 = 1$,

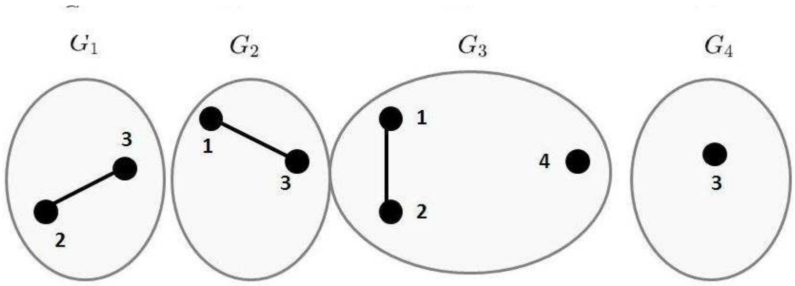


Figure 1.5.: Computation of the clustering coefficient for a small network.

$\mathcal{C}_2 = 1$, $\mathcal{C}_3 = 1/3$ *and* $\mathcal{C}_4 = 0/0$.

Latora and Marchiori (2002), Crucitti et al. (2002) have showed that $\mathcal{C}$ suffers several limitations: $\mathcal{C}$ is ill-defined in several cases. For instance, in a railway network $\mathcal{C}_i = 0/0$ in terminal stations of only one line. Another drawback is that it works only in the topological context, where the only required information is about the existence or absence of links, and nothing on the link length.

Latora and Marchiori (2001, 2002) have proposed an alternative way applicable on weighted and non-connected networks. The clustering coefficient can be substituted by the *local efficiency indicator*:

$$\mathcal{E}_{loc}(G) = \frac{1}{n} \sum_{i \in N} \mathcal{E}_{glob}(G_i), \tag{1.11}$$

where

$$\mathcal{E}_{glob}(G_i) = \frac{1}{|N_i|(|N_i|-1)} \sum_{j,k \in N_i} \frac{1}{d_{jk}},$$

and G_i denotes the subgraph neighbor associated to i.

Note that $0 \leq \mathcal{E}_{loc}(G) \leq 1$. In the transportation context, this measure gives information on how efficiently passengers can move in the network from a local point of view.

We think that several special cases must be specified. For instance, terminal stations of only one line in a rapid transit system yields to neighbor subgraphs formed by only one node and this measure has not sense. So, it is reasonable to think that at this case, $\mathcal{E}_{glob}(G_i)$ may be 0 or 1, depending on the context that is being applied. For instance, it is appropriated to consider $\mathcal{E}_{glob}(G_i) = 0$ when the connectivity of G is being analyzed and $\mathcal{E}_{glob}(G_i) = 1$ when G represents the primal graph associated to a hypergraph (see Section 1.5).

1.2.6. Vulnerability indicator

Latora and Massimo (2005) defined the vulnerability of networks G under a class of damages Λ. This kind of damages should be understood as a set of possible damages, such as failures in nodes or links on the infrastructure G. So, the vulnerability of G under a class of damages Λ is defined as

$$\mathcal{V}[G,\Lambda] = \frac{\mathcal{P}[G] - \mathcal{P}^{wc}[G,\Lambda]}{\mathcal{P}[G]}, \qquad (1.12)$$

where $\mathcal{P}[G]$ is the usual performance of G and $\mathcal{P}^{wc}[G,\Lambda]$ is the worst performance of G under the class of damages Λ. In this work, the goal is finding the critical components of the network respect to the drop in the network's performance caused by its deactivation. As a practical application, the authors considered communication networks as well as infrastructure transportation networks and they identified the performance of G with the global efficiency indicator.

In a previous work, Latora and Marchiori (2004) defined a measure of the centrality of a node i called *the importance* as the drop in the efficiency of G caused by the deactivation of such node i, this is,

$$\mathcal{V}_{loc}(i) \equiv \Delta\mathcal{E}_{glob} = \mathcal{E}_{glob}(G) - \mathcal{E}_{glob}(G-i),$$

where $\mathcal{E}_{glob}(G)$ denotes the global efficiency defined in (1.9) and $G-i$ represents the resulting network after eliminating the node i and all edges incident to i. The most important nodes i reaches the highest $\mathcal{V}_{loc}(i)$.

Criado et al. (2005) define the *global efficiency* as an arithmetic mean of the local vulnerabilities

$$\widetilde{\mathcal{E}}_{glob}^{aver}(G) = \frac{1}{n}\sum_{i\in N} |\mathcal{V}_{loc}(i)|$$

and as a maximum

$$\widetilde{\mathcal{E}}_{glob}^{wc}(G) = \max_{i\in N} \|\mathcal{V}_{loc}(i)\|.$$

In Criado et al. (2006, 2007) several efficiency functions are defined. Let $\mathbb{G}$ be the set of all networks with a finite number of nodes. The *efficiency function* $\mathbb{E}$ is a function $\mathbb{E} : \mathbb{G} \to [0, 1]$ such that

- $\mathbb{E}(G_\emptyset) = \mathbb{E}(V, \emptyset) = 0$.
- $\mathbb{E}(K_n) = 1$, where K_n is the completely connected network with n nodes.
- $\mathbb{E}(G) \leq \mathbb{E}(G')$ if G' is obtained from G by adding edges.
- $\mathbb{E}(\cdot)$ is invariant under isomorphism of $\mathbb{G}$, i.e., $\mathbb{E}(\cdot)$ does not vary if we transform $G \in \mathbb{G}$ into an other network $H \in \mathbb{G}$ without breaking or adding any edges, so that H and G are identical ignoring the labels on the nodes.
- $\mathbb{E}(G)$ is computable in polynomial time with respect to the number of nodes of G.

From this perspective, several alternative efficiency functions have been defined in the literature. For instance, the arithmetic efficiency defined in Latora and Marchiori (2001, 2004):

$$\mathbb{E}^{+}(G) = \frac{1}{n(n-1)} \sum_{i,j\in N} \frac{1}{d_{ij}}$$

and the geometric efficiency proposed by Criado et al. (2006)

$$\mathbb{E}^{\bullet}(G) = \left(\prod_{i,j\in N} \frac{1}{d_{ij}}\right)^{\frac{1}{n(n-1)}}.$$

The vulnerability function defined by Criado et al. (2005, 2007) is based on the network's performance caused by failures in nodes and edges. The *vulnerability function* $\mathbb{V}$ is a function $\mathbb{V} : \mathcal{G} \to [0, 1]$ holding the following properties.

- $\mathbb{V}$ invariant under isomorphisms.

- $\mathbb{V}(G') \geq \mathbb{V}(G)$ if G is obtained from G' by adding edges.
- $\mathbb{V}(G)$ is computable in polynomial time respect to the number of nodes of G.

In Criado et al. (2005) two vulnerability functions are defined as follows

$$\mathbb{V}_1(G) = \exp\{\frac{M^+ - M^-}{n} + n - m - 2 + \frac{2}{n}\} \tag{1.13}$$

and

$$\mathbb{V}_2(G) = \exp\{\frac{\sigma}{n} + n - m - 2 + \frac{2}{n}\}, \tag{1.14}$$

where $M^+ = \max\{gr(i); i \in V\}$ and $M^- = \min\{gr(i); i \in V\}$ are the maximum and minimum degree of nodes, respectively, and, $\sigma = (\frac{1}{n}\sum_{i=1}^{n}(gr(i) - 2m/n)^2)^{1/2}$ is the standard deviation of the degree distribution.

1.3. Measures in the metric network

In this section, we consider a metric $l(e)$ in the infrastructure network $G(N, E)$ associated to each edge e, representing; length, running time or generalized cost for traversing it. So, the distance between a pair of nodes is the sum, $l(e)$, of edges that the shortest path between such nodes contains.

1.3.1. Diameter

The diameter of a network $\widetilde{\mathcal{D}}(G)$ is the longest shortest path between any pair of nodes of G, that is,

$$\widetilde{\mathcal{D}}(G) = \max_{i,j \in N} d_{ij}. \tag{1.15}$$

We observe two basic properties for comparing networks: it belongs to a predefined range, and it is monotone decreasing.

Proposition 1.3 *Let G be a infrastructure network with a metric. We have that:*

1. $\max_{e \in E(K_n)} l(e) \leq \widetilde{\mathcal{D}}(G) \leq d_{1n}$, *where d_{1n} denotes the distance between first-node 1, and last-node n, in a chain graph formed by n-nodes.*

2. *It is monotone decreasing in the sense that, if G' is network obtained when adding a new edge to G connecting two nodes, we have that $\widetilde{\mathcal{D}}(G') \leq \widetilde{\mathcal{D}}(G)$.*

Proof 1.4 *1. It is easy to note that the minimum value of $\widetilde{\mathcal{D}}(G)$ is reached for a completely connected network K_n. At this case, $\widetilde{\mathcal{D}}(N, K_n) = \max_{e \in E(K_n)} l(e)$. Similarly, the maximum value is obtained for a chain graph with n-nodes. So, the maximum distance is reached for the shortest path between first and last node.*

2. Adding a new edge in G provides a new couple of connected nodes, which does not increase $\widetilde{\mathcal{D}}(G)$.

It is interesting to analyze networks when edges or nodes fail. It can be observed that if $G - e$ is a connected network, its corresponding diameter increase is a non-negative number, i.e., $\Delta_E \widetilde{\mathcal{D}}(G) = \widetilde{\mathcal{D}}(G - e) - \widetilde{\mathcal{D}}(G) \geq 0$. In contrast, if $G - e$ is a non-connected network, the diameter of $G - e$ is infinity. Last case, we propose two alternative ways to obtain information on the connectivity of networks, as follows:

1. To analyze the connected components as in (1.5).

2. To introduce a penalty $k \in \mathbb{N}$. If an edge or node failure is expected to be long enough, it is frequent that the transit company offers an alternative transportation mode, usually a bus, between the affected stations (see Figure 1.6). So, if $G - e$ is a non-connected network, we modify the distance associated to e, $l(e)$ by $l(e)$ times k. This aspect can be extended to interruptions in nodes considering a penalty on all edges incident to such node.

In order to illustrate this situation we consider what happened in the Barcelona metro in August 2008 (Figure 1.6). Due to the construction of the high-speed train tunnel

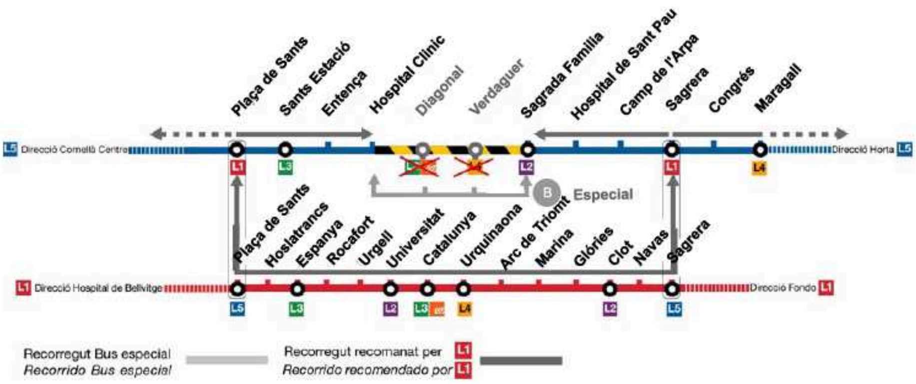


Figure 1.6.: The affected line of Barcelona metro network.

across Barcelona, the service in stations Diagonal and Verdaguer was interrupted during a determined number of days. So, the blue line was fragmented into two parts and the stretch Hospital Clinic-Sagrada Familia was disabled. The operator provides two choices to the affected passengers:

1. an alternative route: transferring from the blue1 line L5 to the red line L1 at station Plaça de Sants and transferring back to L5 at station Sagrera (or the other way around),

2. a bus between stations Hospital Clinic and Sagrada Familia (labelled as B Especial in the figure).

1.3.2. Average shortest distance

The average shortest distance between any pair of nodes has been suggested for measuring robustness (see Ng and Efstathiou (2006)). This measure is defined as

$$ASd(G) = \frac{1}{n(n-1)}\sum_{i\in N}\sum_{j\in N} d_{ij}. \tag{1.16}$$

Similar to the characteristic path length, this measure may be ∞ if the network is non-connected.

1.3.3. Global and local Efficiency indicators

Latora and Marchiori (2001, 2008) have defined the global and local efficiency for weighted networks. However, in this case, the efficiency belongs to a predefined range $[0, \infty)$. In order to compare different networks, this measure must be normalized. Let K_n be the completely connected graph . So, the global efficiency for a weighted network G is defined as

$$\widetilde{\mathcal{E}}_{glob}(G) = \frac{\mathcal{E}_{glob}(G)}{\mathcal{E}_{glob}(K_n)}, \tag{1.17}$$

where $\mathcal{E}_{glob}(G)$ represents the global efficiency defined in Section (1.2), according to the length of each edge.

Note that $0 \le \widetilde{\mathcal{E}}_{glob}(G) \le 1$, and that it measures how efficiently can passengers move in a global scale.

Now, we will introduce the local efficiency indicator on this layer. For the purpose, let $G_i = (N_i, E_i)$ be the neighbor subgraph of G with k_i-nodes formed by all first neighbors of node i. The local efficiency is defined as:

$$\widetilde{\mathcal{E}}_{loc}(G) = \frac{1}{n}\sum_{i\in N}\frac{\mathcal{E}_{glob}(G_i)}{\mathcal{E}_{glob}(G_i^{ideal})}, \tag{1.18}$$

where $\mathcal{E}_{glob}(G_i^{ideal})$ is the global efficiency of the ideal network ${G_i}^{ideal}$ which has k_i nodes and all possible edges. The local properties of G can be characterized by evaluating for each node i, the efficiency of its neighbor subgraph $G_i = (N_i, E_i)$.

1.4. Measures for networks with demand patterns in operation

In this section we will summarize the paper *Evaluating passenger robustness in a rail transit network. Transportation Research Part C: Emerging Technologies, 20(1):34–46, 2012 by A. De Los Santos, G. Laporte, J.A. Mesa and F. Perea.* First, we will describe the corresponding measures for networks with demand patterns.

1.4.1. Measures for networks with demand patterns

In this section we propose a measure for the robustness of a network by calculating the overall travel time in the entire system when some link fails. Let $t(e)$ denote the time required to traverse edge e. For the sake of simplicity it is assumed that only one type of train runs on the network. Let $H = (h_{ij})$ be the origin-destination matrix (denoted by OD) in the interval time that we are studying, where h_{ij} denotes the number of passengers going from i to j.

It must be noted that our model assumes that this matrix is not affected by failures, that is, all passengers will keep travelling to their normal destination regardless of disruptions in the normal functioning of the network. Otherwise, the number of passengers missed could be a primary index or passenger loss of robustness.

We will suppose that passengers choose their fastest alternative and that for each pair i, j there is only one shortest path on the network joining i and j (note that this is what happens in practice when dealing with real data). Therefore the flow of each edge is

$$f_e = \sum_{i,j\in N} f_e^{(i,j)}, \tag{1.19}$$

where $f_e^{(i,j)} = h_{ij}$ if the shortest path from i to j contains edge e and zero otherwise. The maximum direct effect of an edge failure (without considering secondary delays) in the network service is observed on the edge having the maximum flow. Measures of robustness are the maximum and the average number of passengers affected by the disruption of the service in an edge:

$$f_{E\max} = \max_{e \in E} f_e, f_E = \frac{1}{m} \sum_{e \in E} f_e.$$

If the disruption happens in station i, the number of passengers affected f_i is the sum of those arriving to and departing from the station. Other cases (e.g. only passengers departing from the station) could also be considered. Since the interruption time affecting the passengers can be diverse, an indicator in terms of time loss for using alternative routes or even a different mode of transportation is a measure of the overall inconvenience to the passengers. The total travel time of the network $G(N, E)$ is

$$T(N, E) = \sum_{e \in E} f_e t(e). \tag{1.20}$$

If there is a disruption in edge $\overline{e}$, then the flow through edge e can be larger because of the addition of passengers for which second shortest time route contains e. This flow will be denoted by $f_e(\overline{e})$. Then the total travel time is: $TT(\overline{e}) = \sum_{e \neq \overline{e}} t(e) f_e(\overline{e})$ and the difference: $TTL(\overline{e}) = TT(\overline{e}) - TT$, is the total time loss. Measures of robustness in terms of ridership time are

$$TTL = \max_{\overline{e} \in E} TTL(\overline{e}), \overline{TTL} = \frac{1}{m} \sum_{\overline{e} \in E} TTL(\overline{e}).$$

In a similar way the robustness of the network for disruptions in nodes can be measured. A related measure has recently been suggested for highway planning in Scott et al. (2006). In De-Los-Santos et al. (2012) we have performed a more complex analysis and we have considered secondary delays and different types of interruptions in the system. In order to evaluate the robustness of the network, we have computed its total travel time when one link fails, taking into account the possible changes in passenger routes and the delays induced by such changes. In order to measure robustness, measures relative to the overall time of a network when links fail are introduced for two different cases: without-bridging interruptions and with-bridging interruptions. In the first case, passengers either have to wait for the failure to be repaired or find an alternative route in the network, whereas in the second case, a bus service is provided between the affected stations and only the edge failing link is disrupted.

When an edge fails, other edges of the network may also be affected by having their flow increased or decreased. If $\bar{e}$ fails, let $f_e(\bar{e})$ be the new flow through edge e, and let $f_{\bar{e}}^{ALT}$ be the number of passengers who decide to take the alternative mode provided by the operator to traverse edge $\bar{e}$, when this alternative mode is actually provided. Let $\tilde{t}(\bar{e})$ be the sum of the travel time through edge $\bar{e}$ plus the waiting time until the disruption is repaired.

The effect of a without-bridging interruption of edge $\bar{e}$ on network $G(N, E)$ is defined as the ratio between the total travel time of the ideal completely network and the total travel time of network $G(N, E)$ in case edge $\bar{e}$ suffers a disruption without bridging :

$$R((N,E),\bar{e}) = \frac{T(K_n)}{DT((N,E),\bar{e})}, \tag{1.21}$$

where

$$DT((N,E),\bar{e}) = f_{\bar{e}}(\bar{e})\tilde{t}(\bar{e}) + \sum_{e\neq\bar{e}} f_e(\bar{e})t_{\bar{e}}(e),$$

$t_{\bar{e}}(e)$ is the time needed to traverse edge e when edge $\bar{e}$ is interrupted. In the implementation of this index three models have been considered to calculate $t_{\bar{e}}(e)$. For more details we refers the reader to De-Los-Santos et al. (2012).

Let $t'(\bar{e})$ be the time needed to traverse edge $\bar{e}$ by the alternative mode. The effect of a with-bridging interruption of edge $\bar{e}$ on network $G(N, E)$ is defined as the ratio between the total travel time of the ideal completely network and the total travel time of network $G(N, E)$ if edge $\bar{e}$ suffers a disruption with bridging (see De-Los-Santos et al. (2012)):

$$BR((N,E),\bar{e}) = \frac{T(K_n)}{BDT((N,E),\bar{e})}, \tag{1.22}$$

where

$$BDT((N,E),\bar{e}) = f_{\bar{e}}^{ALT}t'(\bar{e}) + \sum_{e\neq\bar{e}} f_e(\bar{e})t(e).$$

The next proposition shows that both measures R and BR satisfy three desirable properties. The first one is the scale-invariance, which ensures that the value of the index is not affected by the scale on which the number of passengers or the time are measured. The second one states that if the network becomes more dense the measure increase. The third property ensures that both measures are positive and cannot exceed 1.

Proposition 1.5 *Let $G(N, E)$ be a network and let $\bar{e} \in E$. Then, the following properties* *hold.*

1) *Scale-Invariance: Both* $R((N,E),\bar{e})$ *and* $BR((N,E),\bar{e})$ *are invariant with respect to scale changes in the OD matrix and in travel times.*

2) *Monotonicity: Let* $E^+ = E \cup \{e^+\}$*. Then* $BR((N,E^+),\bar{e}) \geq BR((N,E),\bar{e})$ *and* $R((N,E^+),\bar{e}) \geq R((N,E),\bar{e})$*.*

3) *Membership in (0,1]:* $0 < R((N,E),\bar{e}) \leq 1$ *and* $0 < BR((N,E),\bar{e}) \leq 1$*.*

The detailed proofs are in De-Los-Santos et al. (2012).

The measures: definition and properties

Measures R and BR, see (1.21) and (1.22), enable us to measure the effect of a disruption on a particular edge. In order to give a passenger robustness measure for a network, the following measures are introduced:

- Measures of passenger robustness against without-bridging disruptions:

$$\begin{aligned} \delta R(N,E) &= \frac{T(K_n)}{\max_{\bar{e}\in E} DT((N,E),\bar{e})}, \\ \mu R(N,E) &= \frac{T(K_n)}{\sum_{\bar{e}\in E} DT((N,E),\bar{e})/|E|}. \end{aligned} \tag{1.23}$$

  In (1.23), $\delta R(N,E)$ is the ratio between the total travel time of the ideal completely network and the travel time of the constructed network when the edge whose failure most increases this total travel time fails, whereas $\mu R(N,E)$ includes in the denominator the average total travel time in case of failures. Note also that $\max_{\bar{e}\in E} DT((N,E),\bar{e})$ yields the edge that most affects the functioning of network (N,E), called *critical edge*, in the event of a without-bridging disruption.

- Measures of passenger robustness against with-bridging disruptions:

$$\begin{aligned} \delta BR(N,E) &= \frac{T(K_n)}{\max_{\bar{e}\in E} BDT((N,E),\bar{e})}, \\ \mu BR(N,E) &= \frac{T(K_n)}{\sum_{\bar{e}\in E} BDT((N,E),\bar{e})/|E|}. \end{aligned} \tag{1.24}$$

The interpretations of δBR and μBR are similar to those of δR and μR. Again, $\max_{\bar{e}\in E} BDT((N,E),\bar{e})$ yields the critical edge of network (N,E) in the event of a with-bridging disruption.

From now on, both δR and δBR may be called *maximum measures*, whereas μR and μBR are *mean measures*. Following the traditional terminology in the robustness literature (Réka and Barabási (2002)) the mean or μ measures refer to robustness against random failures and the maximum or δ measures to robustness against intentional attacks. Random failures and intentional attacks have been dealt with in Laporte et al. (2010) the latter being treated as a non-cooperative game theory application.

The following theorem is a direct consequence of proposition 1.5 and of the definition of our passenger robustness measures in (1.23) and (1.24). It states that these are scale-invariant, monotone and their values lie in $(0,1]$.

Theorem 1.1 *$\delta R, \mu R, \delta BR$ and μBR satisfy the following properties: 1) scale-invariance, 2) monotonicity, and 3) membership in $(0,1]$.*

These properties allow us to use our measures to compare the passenger robustness of different networks.

Definition 1.6 *Let (N,E) and (N',E') be two transportation networks. Network (N,E) is more robust than network (N',E') against a without-bridging (respectively with-bridging) interruption if $\delta R(N,E) \geq \delta R(N',E')$ (respectively $\delta BR(N,E) \geq \delta BR(N',E')$). Alternatively the mean measures can be also used to compare networks.*

1.4.2. Measures in operations

In De-Los-Santos et al. (2012), the previous model is extended introducing lines in the infrastructure network, assuming their origins, itineraries, stops, destinations and frequencies are fixed in periodic timetables.

The network representing the lines is non-connected, but adding pedestrian edges corresponding to transfers in the stations makes it connected. A route going from an origin station to a destination station in the network can use edges of several lines as well as the edges of the transfers.

The overall travel time is defined when a without-bridging (with-bridging) failure occurs on edge $\bar{e} \in E$ as the sum of $DT((N,E),\bar{e})$ ($BDT((N,E),\bar{e})$), plus the new transfer times at stations, which results in *Transfer Disruption Time* and *Transfer Bridging Disruption Time*, respectively:

$$
\begin{aligned}
TrDT((\cup N^i, E'), \bar{e}) &= DT((N,E), \bar{e}) \\
&+ \sum_{i \in V} \sum_{j,k: i(l_j), i(l_k) \in N^i} t(e_i(l_j, l_k))(\bar{e}) f_{e_i(l_j,l_k)}(\bar{e}), \\
TrBDT((\cup N^i, E'), \bar{e}) &= BDT((N,E), \bar{e}) \\
&+ \sum_{i \in N} \sum_{j,k: i(l_j), i(l_k) \in N^i} t(e_i(l_j, l_k))(\bar{e}) f_{e_i(l_j,l_k)}(\bar{e}).
\end{aligned}
\tag{1.25}
$$

where $t(e_i(l_j, l_k))$ is the sum of the time needed to change platforms at (possibly) different levels (from the stop of line l_j to the stop of line l_k), plus the average waiting time for line l_k and $f_{e_i(l_j,l_k)}(\bar{e})$ is the flow through edge $e_i(l_j, l_k)$ (which represents the platform changes), if $\bar{e} \in E$ fails. The detailed description appears in De-Los-Santos et al. (2012).

From these overall travel times in case of disruptions, the measures of the passenger robustness of a network considering transfer times is defined, considering two types of interruptions.

- Measures of passenger robustness against without-bridging interruptions:

$$
\begin{aligned}
\delta TrR(\cup N^i, E') &= \frac{T(K_n)}{\max_{\bar{e} \in E} TrDT((\cup N^i, E'), \bar{e})}, \\
\mu TrR(\cup N^i, E') &= \frac{T(K_n)}{\sum_{\bar{e} \in E} TrDT((\cup N^i, E'), \bar{e})/|E|}.
\end{aligned}
\tag{1.26}
$$

- Measures of passenger robustness against with-bridging disruptions:

$$
\begin{aligned}
\delta TrBR(\cup N^i, E') &= \frac{T(K_n)}{\max_{\bar{e} \in E} TrBDT((\cup N^i, E'), \bar{e})}, \\
\mu TrBR(\cup N^i, E') &= \frac{T(K_n)}{\sum_{\bar{e} \in E} TrBDT((\cup N^i, E'), \bar{e})/|E|}.
\end{aligned}
\tag{1.27}
$$

1.5. Transferability measures

First, we will summarize the paper *Analyzing connectivity in collective transportation line networks by means of hypergraphs. The European Physical Journal Special Topics, 215(1):93–108, 2013. by E. Barrena, A. De Los Santos, J.A. Mesa, F. Perea.* as follows.

In Barrena et al. (2013) we have analyzed the performance of a Collective Transportation Line Network (CTLN) with respect to the number of transfers. For this purpose, we have represented the line network associated to a CTLN by means of hypergraphs and their associated graphs.

Hypergraphs are the natural extension of graphs allowing edges (called hyperegdes) with more than two elements. This fact enables to describe a CTLN in a simplified way, representing each line as one hyperedge. Associated to a hypergraph, there exists two well-known graph: the primal and the linear graph. In this work, we consider three different ways to describe a CTLN: using the hypergraph, the linear graph and the linear multigraph (for more details, see Barrena et al. (2013)). On each structure, we have introduced topological connectivity indicators, which give a measure of how easy or hard it is to transfer from one line to another. Concretely, we have concentrated on the characteristic path length, the clustering coefficient, the local efficiency and the global efficiency. In order to define the clustering coefficient and the local and global efficiency on the hypergraph, we have used the primal graph associated to a hypergraph.

In the next section the graphs and and the hypergraph associated to one CTLNs are introduced.

1.5.1. Representations of CTLNs by means of graph and hypergraphs

Now, we present the formal description of all structures previously mentioned.

We assume the existence of a set of stations, $N = \{1, \ldots, n\}$ and a set of lines $\mathcal{L} = \{\ell_1, \ldots, \ell_{|\mathcal{L}|}\}$ in the CTLN. Each line $\ell \in \mathcal{L}$ is characterized by its set of nodes and itinerary. In this way, a CTLN $\mathcal{G}$, can be defined as $\mathcal{G} = (N, \mathcal{L})$. Associate to this CTLN, we can define the following structures.

- Transit hypergraph

  We call transit hypergraph to the hypergraph associated to $\mathcal{G}$, which is defined as $\mathbb{H} = (N(\mathbb{H}), E(\mathbb{H}))$, where $N(\mathbb{H}) = \{1, \ldots, n\}$ is the node set of $\mathcal{G}$ and the hyperedge set $E(\mathbb{H}) = \{\ell_1, \ldots, \ell_{|\mathcal{L}|}\}$, contains subsets of $N(\mathbb{H})$, each of them representing the set of stations that itinerary of each line contains.

- The linear graph and linear multigraph

  The linear graph $\mathrm{L}(\mathbb{H}) = (N(\mathrm{L}(\mathbb{H})), E(\mathrm{L}(\mathbb{H})))$ associated to hypergraph $\mathbb{H}$ is a graph in which each hyperedge in $\mathbb{H}$ is a node in $\mathrm{L}(\mathbb{H})$ and, the edge set $E(\mathrm{L}(\mathbb{H}))$ is the set of transfer edges connecting lines with intersections between them. This graph is assumed to be simple and, therefore, two nodes may be connected at most by an edge.

To define the linear multigraph $\mathrm{L^M}(\mathbb{H})$, we take into account the number of intersection stations between two lines in $\mathbb{H}$, and represent each intersection in $\mathrm{L^M}(\mathbb{H})$ by an edge. So, the linear multigraph is a graph described as $\mathrm{L}(\mathbb{H})$ in which multiples edges are allowed.

1.5.2. Connectivity measures in $\mathrm{L}(\mathbb{H})$ and $\mathrm{L^M}(\mathbb{H})$

Over this level of abstraction, we will introduce the characteristic path length, the clustering coefficient, the global and local efficiency that evaluate the connectivity of a CTLN.

Characteristic path length

As mentioned in Section 1.2.5, this measure yields us the average distance between nodes, which is computed taking into account the length of the shortest path over all pairs of nodes. Therefore, according to the definition of $\mathrm{L}(\mathbb{H})$ or $\mathrm{L^M}(\mathbb{H})$, this measure will give information on the average number of transfers between lines of a CTLN $\mathcal{G}$.

For the purpose, we consider the topological distance between each pair of nodes in the linear graph. Recall that a node in $\mathrm{L}(\mathbb{H})$ is a line ℓ of $\mathcal{G}$. Indeed, the topological distance $d_{pq}^{\mathrm{L}(\mathbb{H})}$ between two lines ℓ_p and ℓ_q, is the number of edges contained in the shortest path that connects such lines.

In order to clarify, the characteristic path length described in Section 1.2.5 on $\mathrm{L}(\mathbb{H})$ is

Definition 1.7 *The* characteristic path length *of the linear graph* $\mathrm{L}(\mathbb{H})$ *with* $|N(\mathrm{L}(\mathbb{H}))| > 1$ *is defined as the average distance in* $\mathrm{L}(\mathbb{H})$*, i.e.,*

$$\mathfrak{L}(\mathrm{L}(\mathbb{H})) = \frac{1}{|N(\mathrm{L}(\mathbb{H}))|(|N(\mathrm{L}(\mathbb{H}))| - 1)} \sum_{p \neq q} d_{pq}^{\mathrm{L}(\mathbb{H})}.$$

Next proposition shows that $\mathfrak{L}(\mathrm{L}(\mathbb{H}))$ satisfies three desirable properties: scale-invariance (trivially), belonging to a predefined range of variation and monotonicity.

Proposition 1.8 *Consider a CTLN $\mathcal{G}$, and let $\mathrm{L}(\mathbb{H})$ be its associated linear graph. Let $\mathfrak{L}(\mathrm{L}(\mathbb{H}))$ be the characteristic path length of the linear graph. We have that:*

1. $1 \leq \mathfrak{L}(\mathrm{L}(\mathbb{H})) \leq |\mathcal{L}| - 1$. *The nearer to 1, the more interconnected the lines are.* $\mathfrak{L}(\mathrm{L}(\mathbb{H})) = 1$ *means that for all pairs of lines of $\mathcal{G}$ there is a transfer station that directly connects them.* $\mathfrak{L}(\mathrm{L}(\mathbb{H})) = |\mathcal{L}| - 1$ *can only be achieved when the $\mathcal{G}$ consists of two lines that are connected (note that in this case $|\mathcal{L}| - 1 = 1$).*

2. $\mathfrak{L}(\mathrm{L}(\mathbb{H}))$ *is monotone decreasing in the sense that, if* $\mathcal{G}'$ *is obtained when adding a new link to* $\mathcal{G}$ *connecting two lines, we have that* $\mathfrak{L}(\mathrm{L}(\mathbb{H})') \leq \mathfrak{L}(\mathrm{L}(\mathbb{H}))$*, where* $\mathrm{L}(\mathbb{H})'$ *is the linear graph of* $\mathcal{G}'$*. Moreover,* $\mathfrak{L}(\mathrm{L}(\mathbb{H})') < \mathfrak{L}(\mathrm{L}(\mathbb{H}))$ *if and only if the new link connects two lines that were not directly connected in* $\mathcal{G}$*.*

Proof 1.9 *See Barrena et al. (2012)*

It can be observed that the characteristic path length on the linear multigraph $\mathrm{L}^{\mathrm{M}}(\mathbb{H})$ holds the same properties than $\mathrm{L}(\mathbb{H})$.

Thank to these properties, the characteristic path length provides a valuable information on CTLNs.

Clustering coefficient

The clustering coefficient defined in Section 1.2.5, can be extrapolated to CTLNs. Indeed, the clustering coefficient $\mathcal{C}$ on $\mathrm{L}(\mathbb{H})$ provides information from a macroscopic point of view on the number of transfers needed to travel between neighbors of a node when this is deleted. Note that, at this level of abstraction, the stations of each line is not taken into account. The number of possibilities to transfer is considered in the linear multigraph as follows

Definition 1.10 *Let* $\mathrm{L}_p^M(\mathbb{H})$ *be the neighbor multigraph associated to* $p \in N(\mathrm{L}^M(\mathbb{H}))$ *and let* U^{max} *be a threshold that represents the maximum number of transfer nodes that can exist between two lines. If node* p *has* k_p *neighbors, then* $\mathrm{L}_p^M(\mathbb{H})$ *will have at most* $U^{max}\frac{k_p(k_p-1)}{2}$ *multiedges.* $\mathcal{C}_p(\mathrm{L}_p^M(\mathbb{H}))$ *is the fraction of these edges that actually exist and the* clustering coefficient $\mathcal{C}(\mathrm{L}^{\mathrm{M}}(\mathbb{H}))$ *on the linear multigraph* $\mathrm{L}^{\mathrm{M}}(\mathbb{H})$ *is the average of* $\mathcal{C}_p(\mathrm{L}_p^M(\mathbb{H}))$*, calculated over all nodes:*

$$\mathcal{C}(\mathrm{L}^M(\mathbb{H})) = \frac{1}{|N(\mathrm{L}^{\mathrm{M}}(\mathbb{H}))|} \sum_{p \in N(\mathrm{L}^{\mathrm{M}}(\mathbb{H}))} \mathcal{C}_p(\mathrm{L}_p^M(\mathbb{H})),$$

where

$$\mathcal{C}_p(\mathrm{L}_p^M(\mathbb{H})) = \frac{\textit{number of edges in } \mathrm{L}_p^M(\mathbb{H})}{U^{max} k_p(k_p-1)/2}.$$

Local and global efficiency

The global and local efficiency can be defined on the linear graph by means of Equations (1.9) and (1.11), respectively. These measures give information on how efficiently passengers can move from one line to another from a global and a local point of view, respectively.

These measures satisfy several desirable properties that allow us to compare different CTLNs. The following proposition collects such properties.

Proposition 1.11 *Consider a CTLN $\mathcal{G}$, and let $\mathrm{L}(\mathbb{H})$ be its associated linear graph. We have that:*

$\frac{1}{|\mathcal{L}|-1} \leq \mathcal{E}_{glob}(\mathrm{L}(\mathbb{H})) \leq 1$. *The nearer to 1, the more interconnected the lines are. $\mathcal{E}_{glob}(\mathrm{L}(\mathbb{H})) = 1$ means that for all pairs of lines of $\mathcal{G}$ there is a transfer station that directly connects them. $\mathcal{E}_{glob}(\mathrm{L}(\mathbb{H})) = \frac{1}{|\mathcal{L}|-1}$ if and only if $|\mathcal{L}| = 2$.*

2. *$\mathcal{E}_{glob}(\mathrm{L}(\mathbb{H}))$ is monotone non-decreasing in the sense that, if $\mathcal{G}'$ is obtained when adding a new link that connects two lines in $\mathcal{G}$, we have that $\mathcal{E}_{glob}(\mathrm{L}(\mathbb{H})') \geq \mathcal{E}_{glob}(\mathrm{L}(\mathbb{H}))$, where $\mathrm{L}(\mathbb{H})'$ is the linear graph of $\mathcal{G}'$. Moreover, $\mathcal{E}_{glob}(\mathrm{L}(\mathbb{H})') > \mathcal{E}_{glob}(\mathrm{L}(\mathbb{H}))$ if and only if the new link connects two lines that were not directly connected in $\mathcal{G}$.*

Proof 1.12 *See Barrena et al. (2012)*

It can be seen that the same properties for the global efficiency indicator hold for the local efficiency indicator.

1.5.3. Connectivity measures in $\mathbb{H}$

Now, we will describe the connectivity measures on the transit hypergraph $\mathbb{H}$. On this structure, the distance $d_{ij}^{\mathbb{H}}$ on the elements of $N(\mathbb{H})$ is the length of the shortest ordinary (i, j)-chain. So, $d_{ij}^{\mathbb{H}}$ is the minimum number of different lines one needs in order to travel from station i to station j. According to this distance, the characteristic path length can be analogously defined as in Section 1.2.5.

Characteristic path length

Over this level of abstraction, the characteristic path length of a CTLN provides an average measure of how easy/hard it is to transfer between stations.

Similarly to the characteristic path length on the linear graph, this measure satisfies on the transit hypergraph the following properties:

Proposition 1.13 *Consider a CTLN $\mathcal{G}$, and let $\mathbb{H}$ be its associated transit hypergraph. We have that the characteristic path length on $\mathbb{H}$ satisfies the following two properties:*

1. $1 \leq \mathfrak{L}(\mathbb{H}) \leq \frac{1}{3}(|\mathcal{L}| + 2)$.

2. *Let $\mathcal{G}'$ be a CTLN obtained when adding one new link joining two lines of $\mathcal{G}$, and let $\mathbb{H}'$ be the associated hypergraph. Then $\mathfrak{L}(\mathbb{H}) \geq \mathfrak{L}(\mathbb{H}')$.*

Proof 1.14 *See Barrena et al. (2012)*

Clustering coefficient

The clustering coefficient over this level of abstraction, evaluates the level of connectivity between stations. The local clustering coefficient for hypergraphs can be expressed as the natural extension of $\mathcal{C}$ in graphs, considering hyperedges instead of edges. A major drawback of this definition is that the connectivity is analyzed according to the number of hyperedges but not regarding to the number of stations. Due to this fact, we will study the clustering coefficient on the associated primal graph but using hypergraph terminology in order to simplify the calculations.

The primal graph associated to a hypergraph is defined as follows

Definition 1.15 *For a hypergraph H and a set $X \subseteq N(H)$, the subhypergraph induced by X is the hypergraph $H[X] = (X, \{e \cap X : e \in E(H)\})$.*

The primal graph, also called the Gaifman and dual graph (see Dechter and Pearl (1989)) of a hypergraph is the graph with the same nodes as the hypergraph, and edges between all pairs of nodes contained in the same hyperedge.

To obtain the clustering coefficient in the primal graph requires a high computational effort, mainly due to the high number of edges in this graph. Barrena et al. (2013) propose a different methodology to calculate the clustering coefficient based on hypergraph properties.

Local and global efficiency

We now introduce the efficiency indicators for the transit hypergraph $\mathbb{H}$. The global efficiency and the local efficiency will give information on how efficiently passengers can move between stations. In this case, the definition of local and global efficiency is described using the distance $d_{ij}^{\mathbb{H}}$. These measures satisfy desirable properties that allow to evaluate and to compare different networks (see Barrena et al. (2013)).

1.5.4. Passenger-oriented transferability measures

Now, we summarize the work titled *Transferability of collective transportation line networks from a topological and passenger demand perspective. E. Barrena, A. De Los Santos, G. Laporte, J.A. Mesa recently submitted.*

We have concentrated on the passenger system level and data related to travel patterns are needed. We are interested in analyzing the performance of a CTLN with respect to the number of transfers carried out by all passengers traveling in the network.

We assume the following hypotheses:

- Passengers use their shortest paths.
- There is no capacity on stations (stops), nor on lines or edges.
- There is no other means of transportation competing with that of the CTLN, therefore demand is fixed.
- The number of passengers wishing to use of the CLTN is greater than one for each pair of different nodes.
- All transfers are considered similar.

For the purpose, we consider the passenger demand between stations, as well as between the lines of the network by means of origin-destination matrices. We assume the number of passengers traveling in the CTLN is known. Concretely, let g_{ij} be the expected number of passengers travelling from station $i \in N$ to station $j \in N$ and $\bar{g}_{pq}$ be the number of passengers traveling from line ℓ_p to ℓ_q. It can be observed that $\bar{g}_{pq}$ can be computed by means of the number of trips between stations. We assume that $g_{ij} \geq 1$, for all $i \neq j, i, j \in N$, and, therefore, $\bar{g}_{pq} \geq 1$. Let g be the total demand expressed as the sum of all demands g_{ij}, $i, j \in N$ and let g^L be defined as the sum of all demands $\bar{g}_{pq}$, $p \neq q$. Note that the total demand g can also be defined by means of linear graphs, i.e., $\mathsf{g} = \mathsf{g}^L + \sum_p \bar{g}_{pp}$, where $\bar{g}_{pp}$ represents the number of passengers travelling within ℓ_p.

In the following section we will show the passenger-oriented transferability measures on hypergraph, linear graph and linear multigraph.

1.5.5. Passenger-oriented measures in $\mathrm{L}(\mathbb{H})$ and $\mathrm{L}^{\mathrm{M}}(\mathbb{H})$

Over this level of abstraction, we will introduce the characteristic path length, the clustering coefficient, the local efficiency and the global efficiency that evaluate the transferability of a CTLN considering passenger demand.

Characteristic path length

In this section we define the characteristic path length incorporating passenger demand. Over this level, this measure gives information on how the number of transfers affects the passengers. The following definition is the natural extension of the $\mathfrak{L}(\mathrm{L}(\mathbb{H}))$.

Definition 1.16 *We define the* characteristic path passenger-oriented length *of the linear graph* $\mathrm{L}(\mathbb{H})$ *with* $|N(\mathrm{L}(\mathbb{H}))| > 1$ *as the average passenger-oriented distance in* $\mathrm{L}(\mathbb{H})$*, i.e.,*

$$\mathfrak{L}_{PO}(\mathrm{L}(\mathbb{H})) = \sum_{p \neq q} \frac{d_{pq}^{\mathrm{L}(\mathbb{H})} \bar{g}_{pq}}{\boldsymbol{g}^L},$$

where $\bar{g}_{pq}/\boldsymbol{g}^L$ *is the proportion of passengers transferring from line* ℓ_p *to line* ℓ_q *over all passengers who transfer.*

The next lemma proves that the characteristic path flow-weighted length above defined, is a natural extension of $\mathfrak{L}(\mathrm{L}(\mathbb{H}))$ defined in Barrena et al. (2013).

Lemma 1.17 $\mathfrak{L}_{PO}(\mathrm{L}(\mathbb{H}))$ *is an extension of* $\mathfrak{L}(\mathrm{L}(\mathbb{H}))$*, which obtains proportional results if the number of passengers between each pair of lines* ℓ_p*,* ℓ_q*,* $p \neq q$*, is the same, that is, all the elements* $\bar{g}_{pq}, p \neq q$ *are the same.*

This definition satisfy three interesting properties: invariance to scale changes, staying within a predefined range of variation and monotonicity. Similar definitions and properties hold for the linear multigraph $\mathrm{L}^{\mathrm{M}}(\mathbb{H})$.

Clustering passenger-oriented coefficient

The clustering coefficient in $\mathcal{L}(\mathbb{H})$ measures the number of transfers needed to travel between neighbors of a line when this is deleted, taking into account the number of passengers travelling between lines. Next definition is an extension of the classical clustering coefficient for graphs.

Definition 1.18 *Let* $\mathcal{G}$ *be a CTLN and let* $\mathrm{L}(\mathbb{H})$ *be its associated linear graph. We consider the passenger-oriented clustering coefficient* $\mathcal{C}_{PO}$ *on the linear graph* $\mathrm{L}(\mathbb{H})$ *as an extension considering demand of the clustering coefficient presented in Watts and Strogatz (1998). Therefore, for each node* $p \in N(\mathrm{L}(\mathbb{H}))$*, the subgraph* $\mathrm{L}_p(\mathbb{H})$ *formed by all first neighbors of* p *is considered. In this subgraph, node* p *and all edges incidents to* p *are eliminated. If node* p *has* k_p *neighbors, then* $\mathrm{L}_p(\mathbb{H})$ *will have* k_p *nodes and at most* $k_p(k_p-$

1)/2 edges. $\overline{\mathcal{C}}_{PO}(\mathrm{L}_p(\mathbb{H}))$ *is the fraction of these edges that actually exist and* $\mathcal{C}_{PO}(\mathrm{L}(\mathbb{H}))$ *is the average of* $\overline{\mathcal{C}}_{PO}(\mathrm{L}_p(\mathbb{H}))$*, calculated over all nodes:*

$$\mathcal{C}_{PO}(\mathrm{L}(\mathbb{H})) = \frac{1}{\sum_{p \in N(\mathrm{L}(\mathbb{H}))} g_p} \sum_{p \in N(\mathrm{L}(\mathbb{H}))} \overline{\mathcal{C}}_{PO}(\mathrm{L}_p(\mathbb{H})),$$

where

$$\overline{\mathcal{C}}_{PO}(\mathrm{L}_p(\mathbb{H})) = \frac{\textit{number of edges in } \mathrm{L}_p(\mathbb{H})}{k_p(k_p-1)/2} g_p,$$

where g_p *the total number of passengers traversing line* L_p*. Note that* $\mathcal{C}_{PO}(\mathrm{L}(\mathbb{H})) \in [0,1]$*. We consider that if* $|\mathrm{L}_p(\mathbb{H})| = 1$*, then* $\overline{\mathcal{C}}_{PO}(\mathrm{L}_p(\mathbb{H})) = 0$*.*

The clustering coefficient on the linear multigraph, in which the number of intersection nodes, that is, the number of possibilities to transfer between lines is taken into account, is defined as follows.

Definition 1.19 *Let* $\mathrm{L}_p^M(\mathbb{H})$ *be the neighbor multigraph associated to* $p \in N(\mathrm{L}^{\mathrm{M}}(\mathbb{H}))$ *and* U^{max} *a threshold that represents the maximum number of transfer nodes that can exist between two lines. If node* p *has* k_p *neighbors, then* $\mathrm{L}_p^M(\mathbb{H})$ *will have at most* $U^{max}(k_p(k_p-1))/2$ *multiedges.* $\overline{\mathcal{C}}_{PO}^M(\mathrm{L}_p^M(\mathbb{H}))$ *is the fraction of these edges that actually exist and the passenger-oriented clustering coefficient* $\mathcal{C}_{PO}^M(\mathrm{L}^M(\mathbb{H}))$ *on the linear multi-graph* $\mathrm{L}^{\mathrm{M}}(\mathbb{H})$ *is the average of* $\overline{\mathcal{C}}_{PO}^M(\mathrm{L}_p^M(\mathbb{H}))$*, calculated over all nodes:*

$$\mathcal{C}_{PO}^M(\mathrm{L}^M(\mathbb{H})) = \frac{1}{\sum_{p \in N(\mathrm{L}^{\mathrm{M}}(\mathbb{H}))} g_p} \sum_{p \in N(\mathrm{L}^{\mathrm{M}}(\mathbb{H}))} \overline{\mathcal{C}}_{PO}^M(\mathrm{L}_p^M(\mathbb{H})),$$

where

$$\overline{\mathcal{C}}_{PO}^M(\mathrm{L}_p^M(\mathbb{H})) = \frac{\textit{number of edges in } \mathrm{L}_p^M(\mathbb{H})}{U^{max} k_p(k_p-1)/2} g_p.$$

Note that $\mathcal{C}_{PO}^M(\mathrm{L}^M(\mathbb{H})) \in [0,1]$*.*

Passenger-oriented local and global efficiency

Now, we will introduce the demand pattern on the global and local efficiency. These measures will give information on how efficiently passengers can move between stations. All measures here defined, satisfy three interesting properties: invariance to scale changes, staying within a predefined range of variation and monotonicity. We will define the global and local efficiency on $\mathrm{L}(\mathbb{H})$, but these concepts are also applicable on $\mathrm{L}^M(\mathbb{H})$.

Definition 1.20 *We define the* passenger oriented global efficiency indicator *of the linear graph* L(ℍ) *as the average of the inverse of the passenger-oriented distances in* L(ℍ)*, that is,*

$$\mathcal{E}^{PO}_{glob}(\mathrm{L}(\mathbb{H})) = \sum_{p \neq q} \frac{\bar{g}_{pq}}{g^L d^{\mathrm{L}(\mathbb{H})}_{pq}},$$

where $\bar{g}_{pq}/g^L$ *is the proportion of passengers transferring from line* ℓ_p *to line* ℓ_q *over all passengers who transfer.*

Definition 1.21 *We define the* passenger-oriented local efficiency indicator *of the linear graph* L(ℍ) *as the average passenger oriented global efficiency of the subgraph* $\mathrm{L}_p(\mathbb{H}) = (N_p, E_p)$*, formed by all first neighbors of* ℓ_p *in* L(ℍ)*, where* $N_p = N(\mathrm{L}_p(\mathbb{H}))$ *and* $E_p = E(\mathrm{L}_p(\mathbb{H}))$*. Mathematically,*

$$\mathcal{E}^{PO}_{loc}(\mathrm{L}(\mathbb{H})) = \frac{1}{\sum_{p \in N} g_p} \sum_{p \in N} \mathcal{E}^{PO}_{glob}(\mathrm{L}_p(\mathbb{H}))\, g_p,$$

where g_p *is the number of passengers traveling within line* ℓ_p*.*

1.5.6. Passenger-oriented measures in ℍ

In this section we will introduce the characteristic path length, the clustering coefficient, the local efficiency and the global efficiency that evaluate the transferability of a CTLN considering passenger demand. Over this level we will have into account the passengers traveling between stations and not between lines.

Characteristic path passenger-oriented length

The characteristic path length on the hypergraph gives information on the average number of transfers of all passengers in a CTLN.

Definition 1.22 *We define the characteristic path passenger-oriented length of the transit hypergraph* ℍ*, with* $|N(\mathbb{H})| > 1$*, as the average passenger-oriented distance in* ℍ*, i.e.,*

$$\mathfrak{L}_{PO}(\mathbb{H}) = \sum_{i \neq j} d^{\mathbb{H}}_{ij} \frac{g_{ij}}{g}.$$

It can be seen that two different line configuration associated to a same CTLN, will have different $\mathfrak{L}_{PO}(\mathbb{H})$. Note that now the importance of an edge is also given by the number of passengers crossing such edge. This measure is also an interesting objective in the design of lines of a CTLN. So, if we consider the problem of designing a line network minimizing the characteristic path length, its solutions will tend to have better connections between lines with more passengers traveling between them. As in $\mathfrak{L}_{PO}(\mathbb{H})$, the characteristic path passenger-oriented length on $\mathbb{H}$ is a natural extension of $\mathfrak{L}(\mathbb{H})$ defined in Barrena et al. (2013) and it satisfies analogous properties.

Clustering passenger-oriented coefficient

As in Barrena et al. (2013), we will refer to the primal graph of a hypergraph to define the clustering coefficient on hypergraphs and all calculations will be based on the terminology of hypergraphs.

Definition 1.23 *Let $G_{\mathbb{H}}$ be primal graph associated to $\mathbb{H}$ and and $G_{\mathbb{H}_i}$ be the subgraph formed by all first neighbors of i. The passenger-oriented clustering coefficient of $G_{\mathbb{H}}$ is defined as follows:*

$$\mathcal{C}_{PO}(G_{\mathbb{H}}) = \frac{1}{\sum_{i \in N(G_{\mathbb{H}})} \tilde{g}_i} \sum_{i \in N(G_{\mathbb{H}})} \overline{\mathcal{C}}_{PO}(G_{\mathbb{H}_i}),$$

where

$$\overline{\mathcal{C}}_{PO}(G_{\mathbb{H}_i}) = \frac{\text{number of edges in } G_{\mathbb{H}_i}}{k_i(k_i-1)/2} \tilde{g}_i,$$

k_i being the number of nodes of $G_{\mathbb{H}_i}$ and $\tilde{g}_i$ the number of passengers traversing station i. Note that $\mathcal{C}_{PO}(G_{\mathbb{H}}) \in [0,1]$.

The passenger-oriented clustering coefficient on the primal graph $G_{\mathbb{H}}$ is a natural extension of $C(G_{\mathbb{H}})$ defined in Barrena et al. (2013).

Passenger-oriented local and global efficiency

The *passenger-oriented local and global efficiency on the hypergraph* $\mathbb{H}$ measure how the passenger are communicated between the first neighbors of a station when it is eliminated.

Definition 1.24 *Let $G_{\mathbb{H}}$ be primal graph associated to $\mathbb{H}$ and and $G_{\mathbb{H}_i}$ be the subgraph formed by all first neighbors of i. We define the passenger-oriented global efficiency indicator of $G_{\mathbb{H}}$ as the average of the inverse of the passenger-oriented distances in $\mathbb{H}$, that is,*

$$\mathcal{E}^{PO}_{glob}(G_{\mathbb{H}}) = \sum_{i \neq j} \frac{g_{ij}}{g \; d^{\mathbb{H}}_{ij}},$$

where g_{ij}/g *is the proportion of passengers travelling from line* i *to line* j *over all passengers who travel.*

Definition 1.25 *Let* $G_{\mathbb{H}}$ *be primal graph associated to* $\mathbb{H}$ *and* $G_{\mathbb{H}_i}$ *be the subgraph formed by all first neighbors of* i*. We define the passenger-oriented local efficiency indicator of* $\mathbb{H}$ *as the average passenger-oriented global efficiency of the subgraph* $G_{\mathbb{H}_i}$*, as*

$$\mathcal{E}_{loc}^{PO}(G_{\mathbb{H}}) = \frac{1}{\sum_{i\in N(\mathbb{H})} \tilde{\mathsf{g}}_i} \sum_{i\in N(\mathbb{H})} \mathcal{E}_{glob}^{PO}(\mathbb{H})\ \tilde{\mathsf{g}}_i,$$

$\tilde{\mathsf{g}}_i$ *the number of passengers traversing station* i*.*

1.6. Conclusions

In this chapter we have reviewed the existing literature on the rapid transit network design measures. We have represented a transportation network by means of Graph and Hypergraph theory. We have described measures found on transportation network in the topological context, measures in operations and with demand patterns as well as transferability measures. On these measures, we have analyzed properties and we have included some extensions.

Chapter 2

Modelling the rapid transit network design problem

2.1. Introduction

In recent years, much effort has been devoted to the construction, improvement or extension of rapid transit networks. This phenomenon is motivated by the increase in travel demand, traffic congestion, the growing length of trips and by the necessity of reducing energy consumption and pollution (Gendreau et al. (1995)). It is important to pay special attention to the investment needed in the building process of new networks or new lines due to their very high cost and because these infrastructures cannot easily be modified within a short time horizon.

Due to its complexity, the railway planning process has traditionally been decomposed into a succession of stages, namely, network design, line design, timetabling, rolling stock, and personnel planning (Guihaire and Hao (2008)). In this chapter we will focus on rapid transit network design, line planning and rolling stock. The main novelty of this chapter is the consideration of a general model that contains as particular cases, all models treated in the literature related with this problem. Another important aspect is the integration of the strategic and tactical phases into an optimization model that determines the location of stations and their connections, a set of lines, each one formed by two different terminal stations and a sequence of intermediate stations (an itinerary), the frequency of each line and the capacity of vehicles. Moreover, several aspects related to rolling stock and personnel planning as well as a long term public economic support for a network profitable operation are also considered as key factors in the network design problem. From now on, we will call to the network that is being designed *the rapid transit system (RTS)*. In order to model the problem realistically, it is appropriate to assume that there exists a different mode of transportation competing with the RTS (e.g. private car, bus, bicycle) which we will call *the alternative mode (ALT)*. Note that, although we have only considered one mode of transport, this aspect can be easily extended to the case of several competing transportation modes. We assume the trains used to operate a determined line are identical, that is, all trains employed for a line carry the same number of vehicles. Without loss of generality, we suppose a train is formed by a locomotive and several passenger carriages. The number of trains needed for each line is determined by their frequency and cycle time (for more details, see Section 2.3.3).

The main input data that we are considering are the underlying network, that is, the potential location for the stations and their connections, the distance matrix between pairs of stations of the underlying network, the travel patterns and the building, capacity and operational related costs.

The remainder of this chapter is structured as follows. In order to properly describe the main characteristics of the models in the literature, we need some notation valid for all models (see Appendix A). In Section 5.2.2 we introduce the variables needed in order to formulate the problem and discuss the different considered objectives. Section 2.3 presents a general model for the rapid transit network design problem. In Section 2.4 we review the literature related to rapid transit network design. Finally, this chapter ends with a summary table containing the main characteristics of each paper analyzed earlier, and with some conclusions.

2.2. Objectives

In this section we discuss different objective functions that can be found in the literature related to the rapid transit network design problem and line planning problem. The general objective in any transport system is to improve the population mobility by providing shorter travel times (Gendreau et al. (1995)). However, different perspectives are taken into account in the transportation network design and line planning. Vuchic (2005), Ceder (2007), Van Nes (2002) and Van Nes and Bovy (2000) state that the objectives for the transportation network design and line planning can be classified into three categories depending on the point of view that is considered. From the passenger's perspective the main characteristics that any transportation network must have in order to be attractive are the service offered, spending as little time as possible to reach the destination, direct connections, and the price of the service. Moreover, the operator wants to minimize costs and maximize revenues and, at the same time, provide a good service level to attract passengers. The community is the third point of view. It wants to find a service of quality for the traveler in order to improve the population's mobility. Moreover, an equilibrium between the preferences of the passengers and those of the operators is implicity contemplated. Therefore, the objectives for the transport network design are classified into three perspectives: community, passengers and operator, which we describe below.

The following criteria can be classified into the preferences of the community:

- Area coverage

  The area coverage (Vuchic (2005)) is defined by computing population of the served area as a percentage of the total urban area population. In Guihaire and Hao (2008), the area coverage is defined as the percentage of the estimated demand that can be served by the rapid transit.

- Trip coverage

  The number of passengers using the transportation system being build is a common objective in the network design and line planning. This number is usually estimated according to OD matrices and travel patterns.

- Social welfare

  The concept of social welfare is defined as the sum of consumer surplus and producer surplus (see Van Nes (2002)) . In Jansson (1996) consumer surplus are the benefits of all travellers who use the public transport since that the travel time is lower than their maximum acceptable travel time. The producer surplus is equivalent to profit. Note that the component expressing the preferences of the passengers is the consumer surplus and the preferences of the operator is the producer surplus. The consumer surplus incorporates the sensitivity of the demand to changes in the given network. So, the level of demand will depend on the quality of the services offered: travel times, comfort, transfers, etc.

- Service quality/passenger attraction

  A common characteristic between service requirements of passengers and operator is to provide a service of high quality in order to attract as many passengers as possible, representing enough mobility capacity and reducing the traffic congestion (see Gendreau et al. (1995)).

- System cost

  The total cost of the system is defined by means of the investments, building and operating costs (Gendreau et al. (1995), Van Nes (2002), Gallo et al. (2011)).

In order to guarantee a good service quality, passengers prefer the following characteristics:

- Short travel time

  The travel time can be broken down into the access time from the origin to the station and into the station, waiting time to take first train in the origin station, riding time, transfer time between platforms and the time to get the destination since alighting the train. This factor is very important to attract passengers since time has a direct influence on modal split.

- Low number of transfers/direct trips

  The number of transfers is also a criterion for the passengers. Users do not like transfers in order to get to the destination station. In general, a passenger uses rapid transit system if no more than two transfers are required (see Guan et al. (2006)). Travelers prefer direct connections in order to complete their trip.

- Good service frequency

  High frequency is a very important aspect for passengers; a slow service frequency implies long waiting times (see Gendreau et al. (1995)).

- Reasonable fares

  Travel costs represent a composite measure of different factors such as travel time, monetary cost and comfort. Transit passengers accept fares depending on the offered service.

From the operator's point of view the system designed must operate efficiently, both at the economical and technical levels. Thereby, the operator considers several factors, some of which are contrary to the preferences of the passengers. The following characteristics can be stated as the operator's point of view:

- Investment costs

  Investment costs are the costs of building the physical network. It depends on the total length as well as the characteristics of the network (see Van Nes (2002), Vuchic (2005)).

- Operating costs

  Operating costs are usually expressed defining two type of costs: fixed and variable costs (see Van Nes (2002), Vuchic (2005)). The fixed operating cost includes costs such as maintenance cost and overheads. The variable cost includes rolling stock operation and personnel costs.

- Fleet acquisition cost

  It represents the investment cost of the rolling stock.

- Revenues

  The revenue for a rapid transit system is determined by the fares paid by the travelers. It can be observed that if the fare is the same for all passengers independently of the length of their trips, this objective is proportional to the passenger attraction. By contrast, if the fare depends on the distance, the revenue is oriented towards the operator.

- Profit

  The net profit of the rapid transit network is expressed as the difference between revenue and total cost in terms of monetary units (see Li et al. (2011b)). From the point of view of operators, the total revenue is the income derived from the passengers who use the rapid transit system.

Note that the classification here presented is not strict. In fact, some objectives can be included into several perspectives at the same time and, therefore, depending on the author's point of view the classification can be different.

2.3. A mathematical programming model

We present a mathematical model that generalizes all models already considered in the literature. In the review we have focused on papers that deal with the rapid transit network design general problem.

As mentioned, the main innovative point of our model with respect to current literature is the simultaneous treatment of network design, line planning and fleet acquisition problems. Moreover, line frequency and capacity are also considered, as well as several aspects related to rolling stock and personnel planning.

Our general network design model also includes passenger transfers between the lines, flow conservation, an upper bound on costs, as well as location and allocation constraints, and a competing mode. As stated in the introduction, it is assumed that the mobility patterns are known but the demand captured by the RTS may vary according to the offered service (Kepaptsoglou and Karlaftis (2009), Ranjbari et al. (2011)). The main input data are the underlying network, that is, the potential location for the stations and their connections, the distance matrix between pairs of stations of the underlying network, the travel patterns as well as train capacities, building costs and operational related costs.

We describe the infrastructure network by means of graphs, where stations are nodes and the links between stations are edges. The model uses the notation introduced in Section 5.2.1 and the variables defined in Section 5.2.2. The objective function considered in the model is a general function that combines all perspectives defined above.

2.3.1. Data and notation

We assume the existence of a set $N = \{1, \ldots, n\}$ of nodes representing potential sites for locating stations and a set $A \subseteq N \times N$ of potential arcs. Each arc between two potential stations i and j will be represented by $a = (i, j)$. Let $E = \{\{i, j\} : i, j \in N,\ i < j,\ (i, j) \text{ or } (j, i) \in A\}$ be a set of edges linking the elements of N (potential rail stretches or sections). Let $G_E = G(N, E)$ be the graph which represents the underlying network (from which sections and stations of lines are to be selected). For each node i, $N(i) = \{j \in N : \{i, j\} \in E\}$ denotes the set of adjacent nodes to i. Let $d_{ij} = d_{ji}$ be the length of edge $\{i, j\} \in E$. The parameter d_{ij} can also represent the time as well as the generalized cost needed to traverse edge $\{i, j\}$. Times can be transformed into distances by using the parameter λ, which represents the average distance traveled by a train in a hour (commercial speed). The undirected graph $G_{E'} = G(N, E')$, represents the competing (private car, bus, etc.) mode network. The nodes are assumed to coincide with those of the rapid transit mode: they could represent origin or destination of the aggregated demands; however, edges are possibly different. Let d'_{ij} be the traversing time of edge $\{i, j\} \in E'$ by the competing mode. Therefore, the whole network is a graph $G = G(N, A')$, where $A' = E \cup E'$. In this work we assume all travels are concentrated at stations of the system, that is, the centroid of each transportation area is assumed to be a station. So, the travel time between census tracts and stations is not considered at the estimation of total travel time of each trip. This aspect can be easily considered by nodes representing transportation areas and defining new edges connecting these nodes among them and with those in N, with the corresponding lengths (times to traverse, generalized cost). Let $W = \{w_1, \ldots, w_{|W|}\} \subseteq N \times N$ be the set of ordered origin-destination (OD) pairs, $w = (w_s, w_t)$. For each OD pair $w \in W$, let g_w be the expected number of passengers per hour for an average day and u_w^{ALT} be the travel time of OD pair using the alternative mode.

The stations are connected by lines ℓ, each of them is characterized by two different terminal stations (initial and final stations), the intermediate stops, the frequency and the capacity of each train (number of carriages). In graph terminology, a line is defined as an elementary and simple path, that is, it is a path (a sequence of edges which connects a sequence of vertices) in which all edges and all nodes are different (except if the line is a circular line, at whose case the initial and final station are equals). The set of lines is denoted by $\mathcal{L}$. Note that $\mathcal{L}$ is not defined a priori, i.e., we do not use a line pool as input for our model, but assume that the number of lines is limited. A lower and upper bound, ℓ_{min} and ℓ_{max} on the length of each line, are considered. Moreover, a maximum number γ of lines can traverse the same edge. Let ψ^{min} and ψ^{max} be the minimum and maximum frequency of a line (number of services per hour). The cost structure is as follows.

- Building costs

  In order to describe this cost, we introduce two parameters c_{ij} and c_i, representing the cost of constructing and edge $\{i, j\} \in E$ and a node i, respectively. As is usual in the network design, there exists an upper bound C_{max} on the total construction of the RTS. All these parameters are expressed on the same time period.

- Operating cost

  This term is defined by means of two different costs: the fixed and variable costs. The fixed costs includes maintenance and overheads of rails and stations. Let OSC_i be the operating station cost for each station i and ORC_{ij} be the operating rail cost for each edge $\{i, j\}$. These two parameters are expressed in terms of of monetary unit per year.

  Concerning variable cost, we introduce rolling stock operation and personnel costs. We assume a train is composed by one or several locomotives and a determined number of carriages. Therefore, the rolling stock operation cost is defined by the cost for operating one locomotive per unit of length c_{loc} and the cost representing operating cost of one carriage c_{carr} per length unit. Both parameters take into account running costs such as fuel or energy consumption. With respect to the personnel costs, a cost c_{crew} per train and year is given.

- Rolling stock acquisition

  As mentioned, we assume a train is composed by a locomotive and several carriages. So, the rolling stock acquisition is determined by the acquisition of locomotives and carriages. Denote I_{loc} be the purchase price of the necessary locomotives per train and I_{carr} be the purchase price of one carriage.

We introduce two conversion factors which play the role of homogenizing the different terms that appear in the total cost. Concretely, let ρ be the total number of hours that a train is operating per year and $\hat{\rho}$ be the horizon of years to recover the total building cost and the rolling stock acquisition cost.

According to capacity, let Θ be the capacity of a carriage measured in number of passengers. Let $\delta^{\min}$ and $\delta^{\max}$ be two parameters representing the minimum and maximum number of carriages that can be included in a train, respectively. The capacity associated to a train can be defined by the capacity of a carriage (Θ) and the number of carriages forming the train. In other words, this term represents the maximum number of passengers that it can transport at any given time.

2.3.2. Variables

We require the following variables to formulate the problem:

- $y_i^\ell = \begin{cases} 1, & \text{if node } i \text{ is selected to be a station of line } \ell \in \mathcal{L} \\ 0, & \text{otherwise.} \end{cases}$

- $y_i = \begin{cases} 1, & \text{if node } i \text{ is selected to be a station in the RTS} \\ 0, & \text{otherwise.} \end{cases}$

- $x_{ij}^\ell = \begin{cases} 1, & \text{if edge } \{i,j\} \in E \text{ belongs to line } \ell \in \mathcal{L} \\ 0, & \text{otherwise.} \end{cases}$

- $x_{ij} = \begin{cases} 1, & \text{if edge } \{i,j\} \in E \text{ is included in the RTS} \\ 0, & \text{otherwise.} \end{cases}$

- $h_\ell = \begin{cases} 1, & \text{if line } \ell \in \mathcal{L} \text{ is included} \\ 0, & \text{otherwise.} \end{cases}$

- $f_{ij}^{w\ell} \in [0, 1]$ is the proportion of demand of w that traverses arc $(i, j) \in A$ using line ℓ.
- $f_i^{w\ell\ell'} \in [0, 1]$ denotes the proportion of demand of w that transfers in station i from line ℓ to line ℓ'.
- $\bar{f}_w^{RTS} \in [0, 1]$ represents the proportion of demand of w using the RTS.
- $f_w^{RTS} \in [0, 1]$ denotes the maximum proportion of demand of w that uses the RTS.
- $f_w^{ALT} \in [0, 1]$ denotes the proportion of demand of w that uses the alternative mode.
- The integer variable $\delta_\ell \in \{\delta^{min}, \ldots, \delta^{max}\}$ represents the number of carriages used by trains of line ℓ. Without loss of generality we assume all services of a line use the same composition.
- The integer variable $\psi_\ell \in \{\psi^{min}, \ldots, \psi^{max}\}$ denotes the frequency of line ℓ.
- $u_w > 0$ is the travel time of pair w using the RTS.
- b_w^ℓ is the number of passengers that directly travel from w_s to w_t at line ℓ.

2.3.3. Objective functions

In this section we formulate different objective functions that can be taken into account when designing a network. As already mentioned, depending on the considered perspective, objective as well as constraints can be different. Most models consider two different perspectives: those of the passenger and the operator. Concretely, if the model is geared towards passengers, that is, the objective function is based on preferences of the users, cost constraints appear in the mathematical formulation of the problem. In contrast, if the model is geared towards the operator (the objective function describes the perspectives of operator), the corresponding constraints ensure a minimum level of service quality.

As mentioned in Section 2.1, we consider the existence of public economic support for the building and operation of the RTS during certain planning horizon. This assumption is very common in the development of railway networks around the world. Usually, governments provide subsidies on the basis of the number of passengers or passenger-kilometer in order to guarantee certain positive margin to companies exploiting the transportation system. In this work we will consider a parameter τ defining the economic support per passenger.

We start by describing different objectives using the notation defined in Section 5.2.1 and then we formulate a general objective function combining all perspectives. First of all, the perspective of the community can be represented by means of the following objectives:

- Trip coverage

  Trip coverage is defined as the number of passengers who uses the RTS, and it is mathematically expressed as

$$z_{TrC} = \sum_{w \in W} g_w \bar{f}_w^{RTS}. \tag{2.1}$$

- System travel time

  The traveling time is a decisive factor to predict whether a passenger will take the RTS or the alternative transport. The traveling time of a passenger in the RTS is composed of the in-vehicle time (time of a passenger by train between his origin and his destination), the average waiting time to take the train in the origin station and an estimation time for each transfer. As in García-Ródenas et al. (2006), the transfer time is the sum of two terms: the time spent between platforms uc_i, and the average waiting time for taking the next train of the line to transfer. Last term can be calculated as one, divided by twice the frequency of the line to transfer. The average travel time of OD pair w using the RTS can be explicitly defined as follows:

$$\begin{aligned} u_w &= \{(60/\lambda) \sum_{\ell \in \mathcal{L}} \sum_{\{i,j\} \in E} f_{ij}^{w\ell} d_{ij} + \sum_{\ell \in \mathcal{L}} \sum_{\ell' : \ell' \neq \ell} \sum_{i \in \ell \cap \ell'} f_i^{w\ell\ell'} (uc_i + \frac{60}{2\psi_{\ell'}}) \\ &+ \sum_{\ell \in \mathcal{L}} \sum_{j \in N(w_s)} \frac{60 f_{w_s j}^{w\ell}}{2\psi_\ell} \} / \bar{f}_w^{RTS}, \ w = (w_s, w_t) \in W. \end{aligned} \tag{2.2}$$

  The first term in (2.2) is the in-vehicle time. The second one represents the time spent in transfers, which is defined as half the time between services in the line to transfer, plus the average time uc_i between platforms at each station i. Last term denotes the waiting time at the origin station, which is defined as half of time between services of this line.

  With these considerations, the average travel time for the passengers who use the RTS system can be defined as

$$z_{RTT} = \sum_{w \in W} g_w u_w \bar{f}_w^{RTS}. \tag{2.3}$$

The travel time for all passengers who use the alternative transport is expressed as

$$z_{ATT} = \sum_{w \in W} d'_w g_w (1 - \bar{f}_w^{RTS}). \tag{2.4}$$

Therefore, the total travel time z_{STT} in the system is computed as the sum of (2.3) and (2.4).

- System cost

The total cost z_{SC} can be defined by means of the building cost z_{BC}, the operating cost z_{OC} and the fleet acquisition cost z_{FAC} (Figure 2.3.3). The construction cost z_{BC} represents costs related to the infrastructure costs and is expressed in monetary units

$$z_{BC} = \sum_{\{i,j\} \in E} c_{ij} x_{ij} + \sum_{i \in N} c_i y_i. \tag{2.5}$$

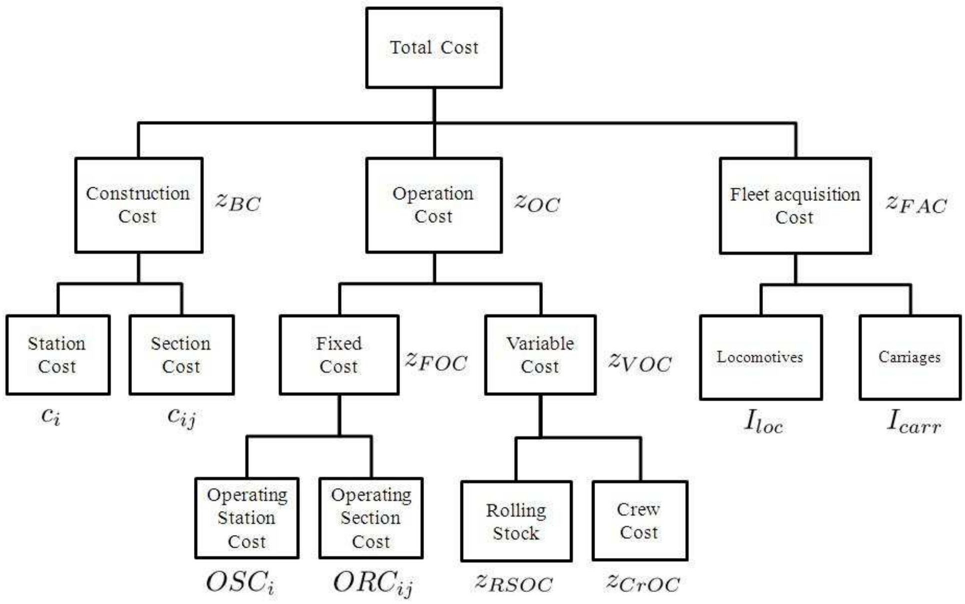


Figure 2.1.: Cost structure.

The operating cost z_{OC} of a RTS is the sum of a fixed cost z_{FOC} and a variable cost z_{VOC}. In accordance to (Claessens et al. (1998), Goossens et al. (2006)) fixed operating cost includes maintenance costs and overheads. More precisely, the fixed operating cost z_{FOC} over the planning horizon is the sum of the operating station cost OSC_i for each station i and operating rail cost ORC_{ij} for each edge $\{i,j\}$. It can be expressed as

$$z_{FOC} = \hat{\rho} \sum_{\{i,j\}\in E} ORC_{ij} \cdot x_{ij} + \hat{\rho} \sum_{i\in N} OSC_i \cdot y_i. \tag{2.6}$$

The variable operating cost z_{VOC} over the planning horizon is the sum of the crew operating cost z_{CrOC} and of the rolling stock cost z_{RSOC}. The crew operating cost z_{CrOC} is the personnel costs due to the operation of all trains in the time horizon $\hat{\rho}$. Thus, this cost is affected by the required fleet size. At this point, the length of each line ℓ, length(ℓ), is unknown and can be expressed as a function of the length of segments that will finally be included to define ℓ, taking into account variables x^{ℓ}_{ij}. We denote by ν_ℓ the cycle time of line ℓ, that is, the time necessary for a train to go from the initial station to the final station of line ℓ and returning back (Goossens et al. (2004)). Denoting by $\text{length}(\ell) = \sum_{\{i,j\}\in\ell} d_{ij}x^{\ell}_{ij}$ the length of line ℓ, $\nu_\ell = 2 \cdot \text{length}(\ell)/\lambda$. Without of generality, we do not consider additional reserve trains and, therefore, the required fleet for each line ℓ is the product of its frequency times its cycle time ν_ℓ as

$$B_\ell = \lceil \psi_\ell \nu_\ell \rceil = \lceil \psi_\ell \cdot 2|\ell|/\lambda \rceil.$$

The crew operating cost in the stated time horizon is

$$z_{CrOC} = \hat{\rho} \cdot c_{crew} \sum_{\ell\in\mathcal{L}} B_\ell. \tag{2.7}$$

The rolling stock operating cost of a train in one hour is a linear function of the number of carriages (García and Martín (2012)). Thus, the rolling stock operating cost z_{RSOC} in the whole planning horizon is

$$z_{RSOC} = \rho\hat{\rho} \sum_{\ell\in\mathcal{L}} B_\ell \cdot \lambda(c_{loc} + \delta_\ell \cdot c_{carr}). \tag{2.8}$$

Finally, the variable operating cost is

$$z_{OC} = z_{FOC} + z_{VOC} = z_{FOC} + z_{CrOC} + z_{RSOC}.$$

The fleet acquisition cost for each train is the cost of purchasing the locomotives and carriages as follows:

$$z_{FAC} = \sum_{\ell \in \mathcal{L}} B_\ell (I_{loc} + I_{carr} \cdot \delta_\ell). \tag{2.9}$$

We can formulate the perspective's community (z_{PC}) by combining the objectives above:

$$z_{PC} = \xi_1^c \cdot z_{TrC} + \xi_2^c \cdot z_{STT} + \xi_3^c \cdot z_{SC}, \tag{2.10}$$

where $\xi_1^c, \xi_2^c, \xi_3^c \in \mathbb{R}$, which allows to the objective function to be homogenized. Concretely, these parameters allow us express all terms of (2.10) at same unit. For instance, Gallo et al. (2011) introduced several parameters representing the perceived value of time in order to transform the objective function into monetary terms.

Secondly, we will consider the point of view of passengers by considering the following terms:

- The total travel time

  (see z_{STT})

- Direct trip

  The number of direct trips is measured by the following expression:

  $$z_{DT} = \sum_{\ell \in L} \sum_{w \in W} b_w^\ell, \tag{2.11}$$

  where b_w^ℓ is a variable explicitly defined as

  $$b_w^\ell = \sum_{w_s \in \ell} \sum_{w_t \in \ell, w_s \neq w_t} \bar{f}_w^{RTS} g_w.$$

  A drawback of this objective is that it may yield a network with few transfers but with long travel times when it is used alone.

If we can compute other aspects z_{others} such as security, comfort, fares, etc, the objective function that generalizes the preferences of the users is the following:

$$z_{PP} = \xi_1^p z_{STT} + \xi_2^p z_{DT} + \xi_3^p z_{others}, \tag{2.12}$$

where $\xi_1^p, \xi_2^p, \xi_3^p \in \mathbb{R}$, which allows to the objective function to be homogenized.

Finally, we consider the perspectives's operator. The main objective for the operator is the net profit of the public network (Li et al. (2011b)). This profit is expressed as the difference between revenue and total cost in terms of monetary units over a planning horizon. From the point of view of operators, the total revenue for the $\hat{\rho}$ years is computed as the number of passengers who use the PTN during the planning horizon, times the passenger fare η, plus the passenger subsidy, which is the same for all passengers independently of the length of their trips. So, the revenue is mathematically expressed as

$$z_{REV} = (\eta + \tau)\rho\hat{\rho} \sum_{w \in W} g_w \bar{f}_w^{RTS}. \tag{2.13}$$

Specifically, the net profit is computed as the difference between the revenue and the system cost.

$$z_{NET} = z_{REV} - z_{SC}. \tag{2.14}$$

We are now able to describe a general objective function which generalize all perspectives.

Table 2.2 shows the different terms which appears at each perspective definition and Table 2.1 presents some of possible combination of terms that can appear at each perspective.

objective	perspective	z_{TrC}	z_{BC}	z_{OC}	z_{FAC}	z_{RTT}	z_{ATT}	z_{DT}	z_{others}	z_{REV}
	Community	✓								
z_1	Passenger					✓				
	Operator		✓	✓	✓					✓
	Community	✓								
z_2	Passenger							✓		
	Operator		✓	✓	✓					✓
	Community		✓	✓	✓					
z_3	Passenger							✓		
	Operator									✓
	Community		✓							
z_4	Passenger							✓		
	Operator			✓	✓					✓
	Community					✓	✓			
z_5	Passenger							✓		
	Operator			✓	✓					✓

Table 2.1.: Possible combinations of terms to define objective functions.

	z_{TrC}	z_{BC}	z_{OC}	z_{FAC}	z_{STT}	z_{DT}	z_{others}	z_{REV}
Community	✓	✓	✓	✓	✓			
Passenger					✓	✓	✓	
Operator		✓	✓	✓				✓

Table 2.2.: Terms at each perspective.

2.3.4. Constraints

We will group the constraints according to their aims as follows.

- Budget constraints

$$\sum_{\{i,j\}\in E} c_{ij}x_{ij} + \sum_{i\in N} c_i y_i \leq C_{max}. \tag{2.15}$$

- Design constraints

$$x^\ell_{ij} \leq y^\ell_i,\ \{i,j\} \in E, i<j,\ \ell \in \mathcal{L} \tag{2.16}$$

$$x^\ell_{ij} \leq y^\ell_j,\ \{i,j\} \in E, i<j,\ \ell \in \mathcal{L} \tag{2.17}$$

$$x^\ell_{ij} = x^\ell_{ji},\ \{i,j\} \in E, i<j,\ \ell \in \mathcal{L} \tag{2.18}$$

$$y_i \leq \sum_{\ell\in\mathcal{L}} y^\ell_i,\ i \in N \tag{2.19}$$

$$y^\ell_i \leq y_i,\ i \in N,\ \ell \in \mathcal{L} \tag{2.20}$$

$$x_{ij} \leq \sum_{\ell\in\mathcal{L}} x^\ell_{ij},\ \{i,j\} \in E, i<j \tag{2.21}$$

$$\sum_{\ell\in\mathcal{L}} x^\ell_{ij} \leq \gamma x_{ij},\ \{i,j\} \in E, i<j \tag{2.22}$$

$$\sum_{j\in N(i)} x^\ell_{ij} \leq 2,\ i \in N,\ \ell \in \mathcal{L} \tag{2.23}$$

$$h_\ell + \sum_{\{i,j\}\in E} x^\ell_{ij} = \sum_{i\in N} y^\ell_i,\ \ell \in \mathcal{L} \tag{2.24}$$

$$\sum_{i\in B}\sum_{j\in B} x^\ell_{ij} \leq |B|-1,\ B \subseteq N,\ |B| \geq 2,\ \ell \in \mathcal{L} \tag{2.25}$$

$$\ell_{\min} h_\ell \leq length(\ell) \leq \ell_{\max} h_\ell,\ \ell \in \mathcal{L}. \tag{2.26}$$

- Demand conservation constraints

$$\sum_{\ell \in \mathcal{L}} \sum_{j \in N(w_s)} f^{w\ell}_{w_s j} = \bar{f}^{RTS}_w, \; w = (w_s, w_t) \in W \tag{2.27}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{i \in N(w_t)} f^{w\ell}_{i w_t} = \bar{f}^{RTS}_w, \; w = (w_s, w_t) \in W \tag{2.28}$$

$$\sum_{i \in N(k)} f^{w\ell}_{ik} - \sum_{\ell' \in \mathcal{L}\setminus\{\ell\}} f^{w\ell\ell'}_k + \sum_{\ell' \in \mathcal{L}\setminus\{\ell\}} f^{w\ell'\ell}_k - \sum_{j \in N(k)} f^{w\ell}_{kj} = 0, \tag{2.29}$$

$$\ell \in \mathcal{L}, \; w = (w_s, w_t) \in W, \; k \neq \{w_s, w_t\}, \; k \in N.$$

- Location-allocation constraints

$$f^{w\ell}_{ij} \leq x^{\ell}_{ij}, \; w \in W, \ell \in \mathcal{L}, \; \{i,j\} \in E, i < j \tag{2.30}$$

$$f^{w\ell\ell'}_i \leq y^{\ell}_i, \; w \in W, \ell, \ell' \in \mathcal{L}, \ell \neq \ell', i \in N \tag{2.31}$$

$$f^{w\ell\ell'}_i \leq y^{\ell'}_i, \; w \in W, \ell, \ell' \in \mathcal{L}, \ell \neq \ell', i \in N. \tag{2.32}$$

- Splitting demand constraints

Many approaches assign passengers to the different modes according to binary variables (see Laporte et al. (2010, 2012)). Concretely, the demand is split into the railway system and the alternative mode according to the generalized costs of each mode and binary variables. Another approach, more realistic but computationally less efficient, is the assignment by logit type functions (Ortúzar and Willumsem (1990)). Some works such as Marín and García-Ródenas (2009) and Perea et al. (2014) use logit model to simulate the modal split. This approach estimates the proportion of users assigned to each mode for each origin-destination pair in a continuous way. So, the number of passengers who use a transport system varies depending on the service offered. For this function, two positive real parameters α and β for each transport mode are needed. The parameter α simulates the market share for each mode and β weights the importance of each mode (Marín and García-Ródenas (2009)). We consider, α^{RAIL} for the railway mode and α^{ALT} in the alternative mode. If we want to give the same importance to both modes, we fix the parameter β independent of the modes as in García-Ródenas et al. (2006). Let us denote $\alpha = \alpha^{ALT} - \alpha^{RAIL}$. Therefore, the proportion of OD pair w using the railway mode is

$$f_w^{RTS} = \frac{1}{1 + e^{(\alpha - \beta(u_w^{ALT} - u_w^{RTS}))}},\ w \in W \tag{2.33}$$

$$\bar{f}_w^{RTS} \leq f_w^{RTS},\ w \in W. \tag{2.34}$$

- Capacity constraints

$$\sum_{w \in W} f_{ij}^{w\ell} g_w \leq \psi_\ell \cdot \delta_\ell \cdot \Theta,\ \ell \in \mathcal{L}, \{i,j\} \in E. \tag{2.35}$$

- Binary constraints

$$x_{ij}, y_i, x_{ij}^\ell, y_i^\ell, h_\ell, b_w^\ell \in \{0,1\},\ i \in N, \{i,j\} \in E, \ell \in \mathcal{L}, w \in W.$$

- Integrality constraints

$$\delta_\ell \in \{\delta^{min}, \ldots, \delta^{max}\},\ \ell \in \mathcal{L}$$
$$\psi_\ell \in \{\psi^{min}, \ldots, \psi^{max}\},\ \ell \in \mathcal{L}.$$

- Bounding constraints

$$f_{ij}^{w\ell}, f_i^{w\ell}, f_i^{w\ell\ell'}, \bar{f}_w^{RTS}, f_w^{RTS}, f_w^{ALT} \in [0,1],$$
$$i \in N, \{i,j\} \in E, \ell, \ell' \in \mathcal{L}, w \in W.$$

An upper bound on the total cost of the overall network is imposed in Constraint (2.15). Constraints (2.16) and (2.17) guarantee that an edge is selected to be built in the RTS only if its incident nodes are also selected. In order to allow edges to be used in both directions Constraint (2.18) must be considered. Constraints (2.19) and (2.20) force that a station i is built if and only if it is already included in the itinerary of a line. Constraints (2.21) impose that if an edge is built, then there exists a line that uses it. Constraints (2.22) define an upper bound on the number of lines that can traverse an edge. Constraints (2.23) force each node to have at most two associated edges of each line. Constraints (2.24) and (2.25) guarantee that lines are connected and do not contain cyclic subgraphs. Constraint (2.26) imposes that each line has at least ℓ_{min} length units and at most ℓ_{max} length units. Constraints (2.27)–(2.29) is flow conservation for each OD pair including transfers between lines. In order to allow the flow corresponding of each OD pair to use an edge of a line ℓ only if this edge belongs to ℓ, we impose constraints (2.30). In order to ensure that a transfer between two lines is made at node i only if i is built for both lines, we include Constraints (2.31) and (2.32). The modal split is described by means of Constraints (2.33) and (2.34). Constraints (2.35) impose a limits on the maximum number of passengers that each line can transport per hour. The character of the variables as imposed in the remaining constraints.

2.4. Special cases

We describe the main papers that deal with problems related to rapid transit network design problem and line planning as well as differences between the general model defined in Section 2.3 and each paper. Our survey is structured as follows. We classify papers into two categories: network design and network design with robustness aspect. In the network design category, we start with the first paper that describes the transit network design as an optimization problem (Laporte et al. (2007)), followed by an extension of the previous paper taking into account transfers between lines (see García-Ródenas et al. (2006)). Later, Guan et al. (2006) described a model combining the line planning problem and the passenger line assignment. In Section 2.4.4 a model which includes the incorporation of a variable number of lines is introduced. The main contributions of Marín and García-Ródenas (2009) are the inclusion of location constraints based on minimizing the number of routing intersections and the consideration of a logit model, which improves the simulation of the demand behavior. Marín and Jaramillo (2008) presented a model for the multi-period capacity expansion problem. The next model considered is that by Escudero and Muñoz (2009) which introduces a modification in the Marín (2007) model in order to allow circular lines. However, the model does not guarantee that lines are connected paths. Marín and Jaramillo (2009) proposes a Benders decomposition algorithm in order to improve the computational time of the model proposed for the rapid transit network design problem. In Section 2.4.9 a model based on transit routes is described by Kermanshahi et al. (2010). A bi-objective model which simultaneously minimizes the network design cost and maximizes origin-destination traffic capture is presented in Gutiérrez-Jarpa et al. (2013). In order to model with robustness aspects, Laporte et al. (2010) presented a model using game theory. Later, Laporte et al. (2012) proposed a model for the design of a robust rapid transit network regarding to by introducing redundant capacity into the system. In García-Archilla et al. (2013) an integer linear programming model for the design of railway network infrastructure as well as their corresponding robust and probabilistic version are introduced. Finally, Cadarso and Marín (2012) introduce recoverable robustness in the model proposed in Marín and Jaramillo (2009).

The rest of this section is dedicated to the preparation of a summary table which contains the main characteristics of the network design models above mentioned.

2.4.1. An integrated methodology for the rapid transit network design problem

The main aim of Laporte et al. (2007) is to integrate the steps of network design and line planning into an optimization process. The authors introduce these steps by considering three stages. The first stage consists of *selecting key nodes*, i.e., main sites are considered as potential stations in rapid transit network which is being designed. The second stage is to design *the core network*, where the selected stations in the first stage are connected with a small number of alignments maximizing the trip coverage and the third named *locating secondary stations*, consists of determining the location of the rest of stations. So, an integer programming model is formulated according to the mode and route user decisions. Roth et al. (2012) have later studied the temporal evolution of the structure of the world's subway networks. They showed that this structure converges to the same shape as that proposed in the work of Laporte et al. (2007): a core with branches radiating from it.

We will now show the objective function used in this model and will comment differences with respect to the general model introduced in Section 2.3.

Objective function and constraints

The objective is the trip coverage.

As in the general model, constraints are classified by groups taking into accounts their aims.

- Budget constraints

  In this model, budget constraints impose a lower and upper bound on the cost of each line and on the overall network.

- Design forcing

  The terminal stations for each line are selected from a set of potential stations. So, constraints (2.16)–(2.25) are modified by constrains where origins and destinations are fixed.

- Routing demand conservation constraints

  In the model, the flow variables are binaries. The authors do not take into account transfers between trains of different lines and the flow is expressed without considering lines (that is, variables such as $f_i^{w\ell\ell'}$ and $f_{ij}^{w\ell}$ do not appear in the model).

- Location-allocation constraints

  Due to the fact that flow variables are binary variables, constraints (2.30)–(2.32) are modified. In order to assign demand on an edge if it is previously built, flow and construction variables appear in constraints.

- Splitting demand constraints

  These constraints guarantee that all demand of an OD pair is assigned to RTS mode if the RTS cost is less than the corresponding to the alternative mode.

2.4.2. Analysis of the parameters of transfers in rapid transit network design

Although the paper García-Ródenas et al. (2006) was published on-line before than Laporte et al. (2007), the chronological order must be the contrary since García-Ródenas et al. (2006) is a continuation of Laporte et al. (2007).

García-Ródenas et al. (2006) propose a new design model which includes transfers between lines. The objective function considered is the trip coverage taking into account a different transport mode competing with the public mode. The potential location for the stations, the distance matrix between pairs of nodes and the travel patterns are known. In the model, users choose the most convenient route (and mode) in order to carry out their trips. The problem they are concerned with is to choose a number of lines covering as much as possible the travel demand between potential stations, subject to different constraints. The authors have studied different values of the parameters such as transfer costs and line frequency. The proposed model has been tested on a network with six nodes.

We now outline the differences between the proposed model in this paper and the general model.

Objective function and constraints

The objective function is the trip coverage and differences with respect to general model are the followings

- Budget constraints

  The construction cost associated to stations is not considered. A lower and upper bound on the line cost as well as on the overall network are imposed.

- Design forcing

  The terminal stations for each line are selected in a set of potential stations. So, the constraints (2.16)–(2.25) are modified by constrains where a set of origins and destinations are fixed.

- Routing demand conservation constraints

  The considered flow variables are binary. The authors do not take into account transfers between trains of different lines and the flow is expressed without considering lines (that is, variables such as $f_i^{w\ell\ell'}$ and $f_{ij}^{w\ell}$ do not appear in the model).

- Location-allocation constraints

  Due to the fact that the variables are binary, constraints (2.30)–(2.32) are modified at the same manner as Laporte et al. (2007).

- Splitting demand constraints

  These constraints guarantee that all demand of an OD pair is assigned to RTS mode if its corresponding cost is less than the corresponding to the alternative mode.

- Transfer constraints

  The authors consider specific constraints to include transfers between trains of different lines. However, in the general model, these constraints are introduced according to the flow conservation constraints. To be more precise, variables in regard to transfer appear when balances between flows are done.

2.4.3. Simultaneous optimization of transit line configuration and passenger line assignment

Guan et al. (2006) presented a model in which the line planning problem and the passenger line assignment are simultaneously considered. The model was formulated as a linear binary integer program. The authors first described both problems separately and then they formulated a model combining both problems. The line planning problem defined consists of finding a set of lines (of a given line pool) that connects all stations of a given infrastructure network, minimizing the total length of all lines. The passenger line assignment was described by means of paths (sequences of edges and nodes that connects origin-destination pairs). A set of feasible paths for each origin-destination pair is predefined and determined by the k-shortest path method according to in-vehicle travel time. Later, a penalty based on the expected time to transfer is added, for each transfer, to the in-vehicle time previously computed. To integrate both problems, the objective function is defined as a convex combination of the proposed at each problem: the total length of all transit length (for the line planning problem) and total passenger in-vehicle travel time and total number of passenger transfers (for the passenger line assignment).

Constraints

The constraints are as follows:

- Budget constraints

  This group of constraints are not considered since that costs are included on the objective function.

- Design forcing

  Due to the fact that the infrastructure network is already given, constraints such as (2.16)–(2.26) are not presented.

- Routing demand conservation constraints

  This group of constraints is not defined.

- Capacity constraints

  The authors assume the capacity on each arc as well as the frequency of each line are given.

- Splitting demand constraints

  In this case, passengers are assigned on paths according to travel times.

2.4.4. An extension to rapid transit network design problem

The originality of Marín (2007) lies on the incorporation of a variable number of lines as well as the consideration of lines without fixing origins and destinations, in the model proposed by Laporte et al. (2007). Thus, this paper may be considered an extension of the Laporte et al. (2007), where lines have a certain degree of freedom. In the following, we will analyze the objective function and constraints used in this model.

Objective function and constraints

The author considers a linear combination of two objectives: trip coverage and routing cost upper bound. The differences with respect to the general model are as follows.

- Budget constraints

  They are analogous to those defined in Section 2.4.1.

- Design forcing

  These constraints are similar to those the general model. However, they accept lines with an only edge and they allow a certain degree of freedom on the number of lines that can traverse an edge at the same time.

- Routing demand conservation constraints

  In this group, the origin and destination of each line are not fixed and binary variables are used in the model. The transfer variables are not taken into account.

- Location-allocation constraints

  Similarly to previous papers, variables are binaries and constraints (2.30)–(2.32) are modified at the same manner as Laporte et al. (2007).

- Splitting demand constraints

  The demand is not elastic and this group of constraints is the same as the model described in Section 2.4.2.

2.4.5. Location of infrastructure in urban railway networks

In the paper of Marín and García-Ródenas (2009), different models for the railway network design problem are proposed. Two aspects are taken into account: demand model and transit supply model. The main contributions of this paper are the inclusion of location constraints (in order to minimize the number of routing intersections) and the consideration of the logit model. This model expresses the proportion of users which are assigned to each mode for each OD pair. The authors assume that each transportation mode depends exclusively on the associated transportation costs. A strategy to approximate the non-linear logit function by a polygonal curve is developed.

As before, we will compare this model with the model proposed in Section 2.3.

Objective function and constraints

The objective function includes two criteria: trip coverage and private congestion.

- Budget constraints

  In this case, the authors consider the same constraint as in Section (2.3).

- Design forcing

  The authors distinguish between two classes of variables: related to infrastructure and those related to line design. According to infrastructure, relationships between flow variables and infrastructure variables are stated. In regard to lines, they introduce two different concepts: covering by lines and crossing constraints. In the covering by lines, similar constraints to previous models are described. The crossing concept is introduced by means of variables that indicate if a station is a terminal station (the start or end) or if such station belongs to several lines at the same time. An upper bound on the number of intersection station is imposed.

- Splitting demand constraints

  As previously commented, the proportion of users in each mode of transportation is obtained according to the logit model. A strategy to approximate the logit function by a polygonal curve is developed.

- Routing demand conservation constraints

  The proposed constraints in this model are similar to previous model but taking into account the logit function.

2.4.6. Urban rapid transit network capacity expansion

Marín and Jaramillo (2008) presented a model for the multi-period capacity expansion problem. This model is an extension of Laporte et al. (2007) and Marín (2007).

Concretely, a set of key stations, connections between these stations, pattern demands, construction costs related to stations and edges as well as an alternative mode are given. The problem deals with the determination of the infrastructure network and the set of lines maximizing the trip coverage taking into account costs related to routing and location. The problem consists of solving the rapid transit network design over different interval times. Due to the large scale of the problem a heuristic algorithm is proposed. At each time interval, the model is solved taking into account the constructed network on the previous time period and that the estimation of costs is based on the time evolution of demand, prices, congestion and available resources.

Objective function

The objective function considered in this work is the trip coverage.

Constraints

The constraints are as follows:

- Budget constraints

  At each time period, this group of constraints are similar to (2.15).

- Design forcing

  At each time period, this group of constraints are similar to those defined at the general model.

- Routing demand conservation constraints

  This group of constraints is similar to that in Marín (2007) but considering different time periods.

- Transfer constraints

  The authors do not consider transfer between lines.

- Splitting demand constraints

  At each period the demand distribution is defined by means of generalized costs of each transportation system and binary variables.

2.4.7. An approach for solving a modification of the extended rapid transit network design problem

In order to allow circular lines, a modification of the extended rapid transit network design problem (see Marín (2007)) is introduced. Thus, a two-stage approach for solving this problem is presented. In the first stage, an integer model is solved in order to select stations and links between them, without exceeding the available budget and maximizing the number of users. The resulting model may yield an undesirable line set formed by non-connected lines consisting of one non-circular sub-line and one or various circular sub-lines. To avoid such lines, the authors propose to define each sub-line as a line. This model also allows the possibility of more than one line linking two locations. In the second stage, the authors present a procedure for solving the above problem by assigning each selected link to exactly one line in order to minimize the number of lines.

As pointed before, we will describe the objective function and constraints as follows.

Objective function and constraints

The objective function is the trip coverage.

As in the general model, constraints are classified into groups taking into account their main characteristics.

- Budget constraints

  The same constraint than the general model is proposed.

- Design forcing

  Constraint (2.24) is changed by other constraint in order to allow circular lines. Also constraints (2.26)–(2.25) are eliminated. The rest of the constraints are similar to the general model but using binary variables.

- Routing demand conservation constraints

  Constraints (2.27)–(2.29) are replaced by constraints with binary variables.

- Location-allocation constraints

  In order to guarantee that a demand is routed on an edge only if this edge belongs to the rapid transit network, the authors consider similar constraints to previous models.

- Splitting demand constraints

  Constraints in the general model are modified by constraints which force the demand to be assigned to the RTS mode if the associated RTS cost does not exceed to corresponding cost of the alternative mode.

2.4.8. Urban rapid transit network design: accelerated Benders decomposition

Marín and Jaramillo (2009) proposed a model for the rapid transit network design problem. The node set is formed by key stations and centroids (representing transportation areas) and the edge set by fictitious arcs between stations and centroid as well as edges representing alignments in the rapid transit network. Demand patterns, infrastructure building costs as well as generalized costs associated to the alternative and rapid transit mode are given. In order to improve the computational time an extension of Bender decomposition is applied.

Objective function and constraints

The objective function is defined as a combination of several dimensionless terms related to covered demand, routing cost and location cost. The introduced constraints are classified into different constraint groups.

- Budget constraints

  A lower and upper bound on the total line cost is considered.

- Design forcing

  This group of constraints is similar to the general model. However, they accept lines with an only edge and they allow a certain degree of freedom on the number of lines that can traverse an edge at the same time.

- Routing demand conservation constraints

  Constrains (2.27) and (2.29) are modified by using binary variables. They do not consider transfers between lines.

- Location-allocation constraints

  The model assigns flow on each built edge in a binary manner. So, constraints (2.30)–(2.32) are modified.

- Splitting demand constraints

  The demand is assigned on each transportation mode according to generalized costs in a binary way.

2.4.9. Rapid transit network design using simulated annealing

In Kermanshahi et al. (2010) a meta-heuristic algorithm is adapted for the rapid transit network design problem. The proposed model is based on transit route network design model. The authors assume that the start and the end of all routes are predetermined. The main input data of this model is a graph whose set of nodes represents the main transit stations and a set of arcs denoting the potential links. So, the routes are formed by sequences of links, and all together constitute a rapid transit network. To tackle the problem the authors select a set of feasible routes and extract on the optimal combination from them. In the model, the construction costs associated to links and stations, an origin-destination matrix as well as the travel time on the alternative and public transport are known.

Next, we will comment on the objective function and on the constraints of the model.

Objective function and constraints

The objective is the trip coverage. The constraints are as follows:

- Budget constraints

  Since the considered problem is based on transit route network design model, this group of constraints is defined by means of variables related to lines such as h_ℓ.

- Design forcing

  Due to the fact that the authors select a set of feasible routes, in this group only appears variables related to flows and routes.

- Routing demand conservation constraints

  This group of constraints is similar to previous models.

- Transfer constraints

  The authors assume that transfer variables are binaries in accordance with a OD pair transfers or not at each node.

- Splitting demand constraints

  By means of binary variables, passengers are assigned to each transport mode according to travel times and transfers.

2.4.10. Rapid transit network design for optimal cost and origin destination demand capture

Gutiérrez-Jarpa et al. (2013) present a model for the rapid transit network design according to two objectives: travel cost and captured traffic. Specifically, the authors describes a bi-objective model which simultaneously minimizes the network design cost taking into account distances between census tracts and stations, and maximizes origin-destination traffic capture. To this end, the authors propose a methodology for designing a metro network considering a predefined shape. So, the model locates optimally stops on the topology selected, captures maximum origin-destination traffic and minimizes costs.

Constraints of this model can be found in Appendix B.

2.4.11. A game theoretic framework for the Robust Railway Network Design problem

In Laporte et al. (2010) a game theoretic framework is used in order to solve the problem of designing a railway transit network in presence of failures. First the authors describe a deterministic model for the problem of designing a railway transit network and then it is extended as follows. They consider only two agents acting in the problem: the planner and the demon. The planner wants to minimize trip coverage or total travel time whereas the demon makes the system works as bad as possible. In the paper two versions of this problem are formulated: Probabilistic Railway Network Design (PRND) and the Stochastic Railway Network Design (SRND) problem. These models are based on a integer lineal model. Later model, consists of deciding a set of potential stations and their connections as well as a line plan covering as many trips as possible in presence of an alternative mode.

Objective function and constraints

We will focus on the deterministic model introduced in this paper. The objective function is the trip coverage. The constraints that appear in this model are the following:

- Budget constraints

In this case, the constraints are similar to (2.15).

- Design forcing

  This group of constraints is the same as that introduced in Section 2.4.4.

- Routing demand conservation constraints

  Constraints (2.27)–(2.29) are replaced with constraints with binary variables.

- Location-allocation constraints

  This group of constraints are expressed by means of binary variables.

2.4.12. Designing robust rapid transit networks with alternative routes

The aim of Laporte et al. (2012) is to propose a model for the design of a robust rapid transit network. The authors consider that a network is robust when in the event of arc failures, the total trip coverage does not decrease too. Firstly, they deal with the deterministic model for the rapid transit network design problem. The mobility patterns in a metropolitan area as well as a different transportation mode competing with the RTS are known. The demand assignment is based on generalized costs. Secondly, they introduce robustness constraints in the model by means of new variables which provides alternative routes if one arc fails and permits avoid congestion on a restricted set of arcs. Finally, computational experiments are presented.

Objective function and constraints

We will focus on the deterministic model introduced in this paper. The objective function is the trip coverage.

The constraints that appear in this model are the following:

- Budget constraints

  In this case, the considered constraints are similar to (2.15).

- Design forcing

  This group of constraints is the same that introduced in Section 2.4.4.

- Routing demand conservation constraints

  Constraints (2.27)–(2.29) are replaced by constraints with binary variables.

- Location-allocation constraints

  This group of constraints are expressed by means of binary variables.

- Splitting demand constraints

  These constraints force to demand to be assigned to the RTS mode if the generalized cost associated to public mode is not more than the corresponding cost of alternative mode.

2.4.13. GRASP algorithms for the robust railway network design

In García-Archilla et al. (2013), an integer linear programming model for the design of railway network infrastructure (stations and links between stations) as well as their corresponding robust version are introduced. The paper is structured in two phases: firstly, the deterministic problem for the rapid transit network is considered and secondly, the robustness property of the rapid transit network is addressed. The authors describe a railway network design problem in presence of a competing mode. So, the demand uses the faster mode to go from its origin to its destination. The mobility patterns, the potential stations and construction costs are known. The authors propose to design lines in the line planning phase in order to avoid too complex models such as the ones described Laporte et al. (2010, 2012) (which consider lines in the design phase). The results obtained in a computational experience indicate the Greedy Randomized Adaptive Search Procedure (GRASP) algorithm is an excellent tool for the resolution of this problem.

In a second phase, the probabilistic version for the railway network design is considered. This problem was originally introduced in Laporte et al. (2012), in which any link can fail but no more than one link can fail at the same time. In the event of link interruption, an alternative transportation mode is provided to passengers, which generates extra costs.

Finally, a computational experiments illustrate the validity of GRASP algorithm. As mentioned at the beginning, we will check the differences with the general model.

Objective function and constraints

The objective function is the trip coverage. The introduced constraints are classified into different constraint groups.

- Budget constraints

  These constraints are the same as in the general model.

- Design forcing

  In this paper, the authors only consider the infrastructure network, and, therefore constraint (2.16) to (2.25) are changed according to x_{ij} and y_i.

- Routing demand conservation constraints

  Constrains (2.27) and (2.29) are also modified by using binary variables. They do not consider transfers between lines.

- Location-allocation constraints

  The authors do not consider elastic demand and this group of constraints are modified.

- Splitting demand constraints

  It is expressed in the same terms as previous models.

2.4.14. Recoverable robustness in rapid transit network design

Cadarso and Marín (2012) present an extension of the model in Marín and Jaramillo (2009) by introducing robustness in the model. Concretely, the authors consider recoverable robustness in the rapid transit network design problem. They suppose interruption on edges and define a recovery algorithm: the demand is distributed on the network without the affected edge. The problem can be formulated as a two stage optimization problem. At each scenario an edge is eliminated. In the first stage the network is built and the second stage, recovery actions are carried out as a consequence of the considered scenario.

2.4.15. Summary table

In this section we present a summary table which shows the main characteristics of all models that appear in the above sections. It will be useful in order to understand the differences in the revised models in the Sections (2.4.1)–(2.4.14). First of all, we will provide in Table (2.3) a number associated with each model in order to facilitate readability.

We have compared these models in Table (2.4) under ten entries. The main characteristics are the objective functions, the set of feasible lines considered in the numerical examples, the constraints and the considered algorithms to obtain a solution. In Table (2.4) “BC” stands for “Budget Constraint”, “DFC” for “Design Forcing Constraint”,

Literature Review		
Item	**Authors**	**Paper**
1	Laporte et al. (2007)	An integrated Methodology for the Rapid Transit Network Design Problem
2	García-Ródenas et al. (2006)	Analysis of the Parameters of Transfers in Rapid Transit Network Design
3	Guan et al. (2006)	Simultaneous Optimization of Transit Line Configuration and Passenger Line Assignment
4	Marín (2007)	An extension to Rapid Transit Network Design Problem
5	Marín and García-Ródenas (2009)	Location of Infrastructure in Urban Railway Networks
6	Marín and Jaramillo (2008)	Urban Rapid Transit Network Capacity Expansion
7	Escudero and Muñoz (2009)	An Approach for Solving a Modification of the Extended Rapid Transit Network Design Problem
8	Marín and Jaramillo (2009)	Urban Rapid Transit Network Design: Accelerated Benders Decomposition
9	Kermanshahi et al. (2010)	Rapid Transit Network Design Using Simulated Annealing
10	Gutiérrez-Jarpa et al. (2013)	Rapid Transit Network Design for Optimal Cost and Origin Destination Demand Capture
11	Laporte et al. (2010)	A Game Theoretic Framework for the Robust Railway Network Design Problem
12	Laporte et al. (2012)	Designing Robust Rapid Transit Networks with Alternatives Routes
13	García-Archilla et al. (2013)	GRASP Algorithms for the Robust Railway Network Design
14	Cadarso and Marín (2012)	Recoverable Robustness in Rapid Transit Network Design

Table 2.3.: Classification of main papers on network design

“FCC” for “Flow Conservation Constraint”, “LAC” for “Location Allocation Constraint”, “SDC” for “Splitting Demand Constraint”, “TC” for ”Transfer Constraint” and “CC” for “Capacity Constraint”.

Summary table											
				Constraints							
Model	Year	Objective functions	lines	BC	DFC	FCC	LAC	SDC	TC	CC	Models/Algorithms
1	2004	Trip coverage	2	✓	✓	✓	✓	✓			ILP
2	2005	Trip coverage	3	✓	✓	✓	✓	✓	✓		ILP
3	2006	Costs and travel time	4		✓			✓		✓	ILP
4	2007	Trip coverage routing cost	3	✓	✓	✓	✓	✓			ILP Branch and bound
5	2008	Trip coverage, cost in the alternative mode	2	✓	✓	✓	✓	✓			ILP
6	2008	Covered demand and costs	4	✓	✓	✓	✓	✓			ILP/Branch and bound
7	2008	Trip coverage	3	✓	✓	✓	✓	✓			ILP/Branch and bound
8	2008	Covered demand and costs	4	✓	✓	✓	✓	✓			ILP/Branch and bound Benders decomposition
9	2010	Trip coverage	–	✓	✓	✓	✓	✓			(SA) algorithm
10	2013	Traffic demand and costs	–		✓						ILP
11	2009	Trip coverage	3	✓	✓	✓	✓	✓			ILP
12	2010	Trip coverage	4	✓	✓	✓	✓	✓			ILP/Branch and bound
13	2011	Trip coverage	–	✓	✓	✓	✓	✓			GRASP algorithm
14	2012	Covered demand and costs	3	✓	✓	✓	✓	✓			ILP

Table 2.4.: Characteristics of main papers addressing the network design problem

2.5. Conclusions

In this chapter we have reviewed the existing literature on the rapid transit network design problem. We have proposed a general model which contains all characteristics of the different models considered in the domain of transit network design. We have dealt with the problem of determining the infrastructure network, the set of lines, the frequency of each line as well as the capacity of services for a rapid transit network. A key factor in our problem is the inclusion of a long-term public economic support for the operating and acquisition rolling stock. So, a mixed integer non-linear program model integrating network design, line planning and fleet investment has been proposed. Considering both problems is beneficial to achieve meaningful results. The trip assignment can be done more efficiently when frequencies are known. An analysis on the different perspective in the objectives when a network is designed has been done. Specifically, each objective is mathematically defined and different general objective is expressed. Moreover, long-term public economic support for a network profitable operation is considered as a key factor in the network design problem.

In order to compare the different models, we have introduced a notation valid to all papers considered in Appendix A. Also we have provided a summary table which contains the main characteristics of all models.

In *Annals of Operations Research: D. Canca, A. De Los Santos, G. Laporte, J.A. Mesa. A General Rapid Network Design, Line Planning and Fleet Investment Integrated Model*, conditionally accepted, that discount rate is incorporated to our problem adding new parameters. In this work we have done several experiments with our general model and the net profit on small networks. The model was solved using a branch-and-bound and a relaxed nonlinear programming problem (NLP).

Appendix A

Data, notation and variables

A.1. Data and notation

In this appendix we state notations and variables needed to describe all models considered as special case of the general model. So, taking into account this appendix and the data, notation and variables described in Chapter 2, we can describe all the reviewed models. We introduce the following input data:

- A set N_c of nodes representing centroid points.
- A set E_f representing fictitious links between nodes of N_c and N.
- For each node i, $\bar{N}(i) = \{j \in N : \{i,j\} \in E'\}$ denotes the set of adjacent nodes to i on the graph related to the alternative mode.
- Nodes o_ℓ and d_ℓ denoting the origin and destination station in the itinerary of line ℓ, respectively.
- An lower and upper bound, c^ℓ_{min} and c^ℓ_{max} on the construction cost of each line.
- There exists a lower bound C_{min} on the total construction cost of the RTS.
- c^{RTS}_{ij} is the generalized cost to traverse the edge $\{i,j\}$ by the RTS mode.
- A penalty TP on the rapid transit travel time for each transfer.

- uc_i^ℓ is the user cost for transferring line ℓ at station i.
- A sufficiently large real number M and a small tolerance $\varepsilon > 0$.
- Λ is a congestion factor.
- c^k is the cost construction of path k.
- $|k|$ number of elements of path k.
- $\Gamma_{ij}^{wk} = 1$ is the path-link incident matrix whose value is 1 if path k for OD pair w, uses arc (i, j), 0 otherwise.
- Let Γ_w be the pre-identified feasible paths from OD pair w.
- $\bar{M}$ is the maximum number of transfers allowed at each trip.
- fr_ℓ is the frequency of line ℓ.
- Θ_{ij} is the capacity of arc $(i, j) \in A$.
- $M_1 \geq 1 + \max_{w \in W} \wedge u_w^{ALT}$ is a sufficiently large real number.
- $M_2 \geq |E|/2$ is a sufficiently large real number.
- Γ is the maximum number of paths.
- Let S be the set of corridors.
- N_s is the set of candidate nodes to be stations of the corridor $s \in S$.
- $|S|$ number of elements of S.
- N_t is the terminal node set that are candidates to be end nodes.
- l_{min} is the minimum length between two adjacent nodes.

A.2. Variables

The problems we are dealing with require the following variables:

- $\tilde{f}_{ij}^w = 1$ if demand of the OD pair w traverses arc $(i, j) \in A$, 0 otherwise.

- $\varphi_{ij}^{w} \in [0,1]$ denotes the proportion of demand of w that uses arc $(i,j) \in E'$ on the alternative mode.
- $\tilde{\varphi}_{ij}^{w} = 1$ if demand of w traverses arc $(i,j) \in E'$ on the alternative mode, 0 otherwise.
- $\tilde{f}_{ij}^{w\ell} = 1$ if demand of w going through arc $(i,j) \in A$ uses line ℓ, 0 otherwise.
- $\tilde{f}_{i}^{w\ell} = 1$ if demand of w transfers in station i to line ℓ, 0 otherwise.
- $p_w = 1$ if demand of w is allocated to the RTS, 0 otherwise.
- $\tilde{x}_{ij}^{\ell} = 1$ if line ℓ traverse arc $(i,j) \in A$.
- $\tilde{x}_{ij}^{s} = 1$ if the arc (i,j) belongs to the corridor $s \in S$.
- $\tilde{y}_i = 1$ if and only if the node i is selected to be a end node.
- $\tilde{y}_i^{s} = 1$ if and only if the node i belongs to the corridor $s \in S$.
- $\hat{h}_w^{k} = 1$ if the path k is selected for OD pair w.
- $\tilde{f}^{w\ell} = 1$ if demand of the OD w pair uses line ℓ.
- $\tilde{f}_{ij}^{wk} = 1$ if demand of w going through arc $(i,j) \in A$ uses path k, 0 otherwise.
- $\tilde{f}_i^{w} = 1$ if demand of w transfers at node i.

Appendix B

Tables

In this subsection we propose several tables which show constraints that appear in each model in order to helpful understanding. Moreover, we will analyze the number of constraints and variables for each model. We denote by $m = |E|$, $n = |V|$, $q = |L|$ and $r = |W|$.

Modelling of the rapid transit network with lines

	Budget constraints	Design	Flow conservation	Location-allocation	Splitting demand	♯Constraints	♯Variables
Model 1	$\sum_{\{i,j\}\in E} c_{ij}x_{ij}^{\ell} + \sum_{i\in N} c_i y_i^{\ell} \in [c_{min}^{\ell}, c_{max}^{\ell}]$ $\sum_{\ell\in\mathcal{L}} \sum_{\{i,j\}\in E} c_{ij}x_{ij}^{\ell} +$ $+\sum_{i\in N} c_i \sum_{\ell\in\mathcal{L}} y_i^{\ell} \in [c_{min}, c_{max}]$	$\sum_{j\in N(o_\ell)} x_{o_\ell j}^{\ell} = 1$ $\sum_{i\in N(d_\ell)} x_{id_\ell}^{\ell} = 1$ $x_{ij}^{\ell} = x_{ji}^{\ell}$ $y_{o_\ell}^{\ell} = y_{d_\ell}^{\ell} = 1$ $\sum_{j\in N(i)} x_{ij}^{\ell} = 2y_i^{\ell}$	$\sum_{j\in N(w_s)} \tilde{f}_{w_s j}^{w} = 1$ $\sum_{i\in N(w_s)} \tilde{f}_{iw_s}^{w} = 0$ $\sum_{i\in N(w_t)} u_{iw_t}^{w} = 1$ $\sum_{j\in N(w_t)} \tilde{f}_{w_t j}^{w} = 0$ $\sum_{i\in N(j)} \tilde{f}_{ij}^{w} - \sum_{k\in N(j)} \tilde{f}_{jk}^{w} = 0$	$\tilde{f}_{ij}^{w} + p_w - 1 \leq \sum_{\ell\in\mathcal{L}} x_{ij}^{\ell}$	$\sum_{\{i,j\}\in E} c_{ij}^{RTS} \tilde{f}_{ij}^{w} - u_w^{ALT} - M(1-p_w) \leq 0$	$1 + q(3+n+m) + r(5+n+m)$	$q(n+m) + r(m+1)$
Model 2	$\sum_{\{i,j\}\in E} d_{ij}x_{ij}^{\ell} \in [c_{min}^{\ell}, c_{max}^{\ell}]$ $\sum_{\ell\in\mathcal{L}} \sum_{\{i,j\}\in E} d_{ij}x_{ij}^{\ell} \in [C_{min}, C_{max}]$ <u>TRANSFERS CONSTRAINTS:</u> $\tilde{f}_{ij}^{w\ell} \leq x_{ij}^{\ell}$ $\tilde{f}_{ij}^{w} + p_w - 1 \leq \sum_{\ell\in\mathcal{L}} \tilde{f}_{ij}^{w\ell}$ $\tilde{f}_{ij}^{w} - p_w + 1 \geq \sum_{\ell\in\mathcal{L}} \tilde{f}_{ij}^{w\ell}$ $\sum_{\{i,j\}\in E} \sum_{\ell\in\mathcal{L}} \tilde{f}_{ij}^{w\ell} \leq Mp_w$ $\sum_{j\in N(i)} \tilde{f}_{ij}^{w\ell} - \sum_{j\in N(i)} \tilde{f}_{ji}^{w\ell} \geq 2\tilde{f}_i^{w\ell} - 1$ $\sum_{j\in N(i)} \tilde{f}_{ij}^{w\ell} - \sum_{j\in N(i)} \tilde{f}_{ji}^{w\ell} \leq 2\tilde{f}_i^{w\ell}$	$\sum_{j\in N(o_\ell)} x_{o_\ell j}^{\ell} = 1$ $\sum_{i\in N(d_\ell)} x_{id_\ell}^{\ell} = 1$ $y_{o_\ell}^{\ell} = y_{d_\ell}^{\ell} = 1$ $\sum_{j\in N(i)} x_{ij}^{\ell} = 2y_i^{\ell}$ $x_{ij}^{\ell} = x_{ji}^{\ell}$	$\sum_{j\in N(w_s)} \tilde{f}_{w_s j}^{w} = 1$ $\sum_{i\in N(w_s)} \tilde{f}_{iw_s}^{w} = 0$ $\sum_{i\in N(w_t)} \tilde{f}_{iw_t}^{w} = 1$ $\sum_{j\in N(w_t)} \tilde{f}_{w_t j}^{w} = 0$ $\sum_{i\in N(j)} \tilde{f}_{ij}^{w} - \sum_{k\in N(j)} \tilde{f}_{jk}^{w} = 0$	$\tilde{f}_{ij}^{w} + p_w - 1 \leq \sum_{\ell\in\mathcal{L}} x_{ij}^{\ell}$	$u_w^{RTS} - u_w^{ALT} - M(1-p_w) \leq 0$	$1 + q(2+n+m) + r(3+n+m)$	$q(n+m+mr) + r(1+m+nq)$ $+r(1+m+nq)$
Model 3	<u>CAPACITY CONSTRAINTS:</u> $\sum_{\ell\in\mathcal{L}} \tilde{x}_{ij}^{\ell} \cdot fr_\ell \cdot h_\ell \leq \Theta_{ij}$	$\sum_{\ell\in\mathcal{L}} \tilde{x}_{ij}^{\ell} \cdot h_\ell \geq 1$ $h_\ell(\sum_{\{i,j\}\in E} \tilde{x}_{ij}^{\ell} d_{ij} - \ell_{\max}) \leq 0$ $h_\ell(\sum_{\{i,j\}\in E} \tilde{x}_{ij}^{\ell} d_{ij} - \ell_{\min}) \geq 0$			$\sum_{\ell\in\mathcal{L}} \tilde{f}_{ij}^{w\ell} \geq \Gamma_{ij}^{wk} \cdot \hat{h}_w^k$ $h_\ell - \tilde{f}^{w\ell} \geq 0$ $\sum_{k\in\Gamma_w} \hat{h}_w^k = 1$ $\sum_{\ell\in\mathcal{L}} \tilde{f}^{w\ell} \leq \bar{M}$	$2m + 2q + 2r + mr\|k\| + qr$	$m + q(2 + rm + m + r) + r\|k\|(m+1)$

Modelling of the rapid transit network with lines

	Budget constraints	Design	Flow conservation	Location-allocation	Splitting demand	♯Constraints	♯Variables
Model 4	$\sum_{\{i,j\}\in E} c_{ij}x^{\ell}_{ij} + \sum_{i\in N} c_i y^{\ell}_i \in [c^{\ell}_{min}, c^{\ell}_{max}]$ $\sum_{\ell\in\mathcal{L}}\{\sum_{\{i,j\}\in E} c_{ij}x^{\ell}_{ij}+$ $+\sum_{i\in N} c_i y^{\ell}_i\} \in [C_{min}, C_{max}]$	$x^{\ell}_{ij} \leq y^{\ell}_i$ $x^{\ell}_{ij} \leq y^{\ell}_j$ $x^{\ell}_{ij} = x^{\ell}_{ji}$ $\sum_{j\in N(i)} x^{\ell}_{ij} \leq 2$ $\frac{1}{2} - \sum_{\{i,j\}\in E} x^{\ell}_{ij} + M(h_\ell - 1) \leq 0$ $\frac{1}{2} - \sum_{\{i,j\}\in E} x^{\ell}_{ij} + Mh_\ell \geq 0$ $1 + \sum_{\{i,j\}\in E} x^{\ell}_{ij} = \sum_{i\in N} y^{\ell}_i$ $\sum_{i\in B}\sum_{j\in N} x^{\ell}_{ij} \leq \vert B\vert - 1$	$\sum_{i\in N(w_s)} \tilde{f}^w_{iw_s} - \sum_{j\in N(w_s)} \tilde{f}^w_{w_sj} = -1$ $\sum_{i\in N(w_t)} \tilde{f}^w_{iw_t} - \sum_{j\in N(w_t)} \tilde{f}^w_{w_tj} = 1$ $\sum_{i\in N(k)} \tilde{f}^w_{ik} - \sum_{j\in N(k)} \tilde{f}^w_{kj} = 0$	$\tilde{f}^w_{ij} + p_w - 1 \leq \sum_{\ell\in\mathcal{L}} x^{\ell}_{ij}$	$u^{RTS}_w - \Lambda u^{ALT}_w - M(1-p_w) \leq 0$ $u^{RTS}_w = \sum_{\{i,j\}\in E} d_{ij}\tilde{f}^w_{ij}$	$1 + q(4+n+3m) + r(3+n+m)$ subtour elimination constraints: $\Theta(2^n)$	$q(n+m) + r(1+m)$
Model 5	$\sum_{\{i,j\}\in E} c_{ij}x_{ij} + \sum_{i\in N} c_i y_i \leq C_{max}$	$x_{ij} \leq y_i$ $x_{ij} \leq y_j$ $x^{\ell}_{ij} = x^{\ell}_{ji}$ $\sum_{j\in N(i)} x^{\ell}_{ij} \leq 2$ $h_\ell \leq \sum_{\{i,j\}\in E} x^{\ell}_{ij} \leq M\cdot h_\ell$ $h_\ell \leq \sum_{\{i,j\}\in E} x^{\ell}_{ij}$ $\sum_{i\in B}\sum_{j\in N} x^{\ell}_{ij} \leq \vert B\vert - 1$	$\sum_{j\in N(w_s)} \tilde{f}^w_{jw_s} - \sum_{j\in N(w_s)} \tilde{f}^w_{w_sj} = -1$ $\sum_{j\in N(w_t)} \tilde{f}^w_{jw_t} - \sum_{j\in N(w_t)} \tilde{f}^w_{w_tj} = 1$ $\sum_{j\in N(i)} \tilde{f}^w_{ji} - \sum_{j\in N(i)} \tilde{f}^w_{ij} = 0$	$\tilde{f}^w_{ij} \leq y_i$ $\tilde{f}^w_{ij} + \tilde{f}^w_{ji} \leq \sum_{\ell\in\mathcal{L}} x^{\ell}_{ij}$ $\tilde{f}^w_{ij} \leq \sum_{\ell\in\mathcal{L}} y^{\ell}_j$ $\tilde{f}^w_{ij} \leq \sum_{\ell\in\mathcal{L}} y^{\ell}_i$	$u_w = u^{(RTS)}_w - u^{(ALT)}_w$ $p^{(PUB)}_w(u_w) = G(u) = \frac{1}{1+\exp\{-(\alpha+\beta u)\}}$ $p^{(PUB)}_w = p_0 + \sum_{i=1}^{n} a_i u^w_i$ $\Delta u_1 z^w_1 \leq u^w_1 \leq \Delta u_1$ $\Delta u_i z^w_i \leq u^w_i \leq \Delta u_i z^w_{i-1}$ $i = 2, ..., n-1$ $0 \leq u^w_n \leq \Delta u_n z^w_{n-1}$ $z^w_{i-1} \leq z^w_i, i = 2, ..., n$ $z^w_i \in \{0,1\}$	$3 + 2m + q(3+n+m) + r(n+4m)$ subtour elimination constraints: $\Theta(2^n)$	$q(1+m) + r(2n+m) + n + m$
Model 7	$\sum_{\ell\in\mathcal{L}}\sum_{\{i,j\}\in E} c_{ij}x^{\ell}_{ij}+$ $+\sum_{\ell\in\mathcal{L}}\sum_{i\in N} c_i y^{\ell}_i \leq C_{max}$	$x^{\ell}_{ij} \leq y^{\ell}_i$ $x^{\ell}_{ij} \leq y^{\ell}_j$ $x^{\ell}_{ij} = x^{\ell}_{ji}$ $\sum_{j\in N(i)} x^{\ell}_{ij} \leq 2$ $1 + \sum_{\{i,j\}\in E} x^{\ell}_{ij} \geq \sum_{i\in N} y^{\ell}_i$	$\sum_{j\in N(w_s)} \tilde{f}^w_{jw_s} - \sum_{j\in N(w_s)} \tilde{f}^w_{w_sj} = -1$ $\sum_{j\in N(w_t)} \tilde{f}^w_{jw_t} - \sum_{j\in N(w_t)} \tilde{f}^w_{w_tj} = 1$ $\sum_{j\in N(i)} \tilde{f}^w_{ji} - \sum_{j\in N(i)} \tilde{f}^w_{ij} = 0$ if $i \notin \{w_s, w_t\}$	$\tilde{f}^w_{ij} + p_w - 1 \leq \sum_{\ell\in\mathcal{L}} x^{\ell}_{ij}$	$\sum_{\{i,j\}\in E} d_{ij}\tilde{f}^w_{ij} - \Lambda u^{ALT}_w - M(1-p_w) \leq 0$	$1 + q(1+n+3m) + r(1+n+m)$	$q(n+m) + r(1+m)$
Model 8	$\sum_{\ell\in\mathcal{L}}(\sum_{\{i,j\}\in E} c_{ij}x^{\ell}_{ij} + \sum_{i\in N} c_i y^{\ell}_i) \in [C_{\min}, C_{\max}]$	$x^{\ell}_{ij} \leq y^{\ell}_i$ $x^{\ell}_{ij} \leq y^{\ell}_j$ $x^{\ell}_{ij} = x^{\ell}_{ji}$ $\sum_{j\in N(i)} x^{\ell}_{ij} \leq 2$ $h_\ell + \sum_{\{i,j\}\in E} x^{\ell}_{ij} = \sum_{i\in N} y^{\ell}_i$ $h_\ell \leq \sum_{\{i,j\}\in E} x^{\ell}_{ij} \leq M_2 h_\ell$ $\sum_{i\in B}\sum_{j\in N} x^{\ell}_{ij} \leq \vert B\vert - 1$	$\sum_{j\in N(w_s)} \tilde{f}^w_{jw_s} - \sum_{j\in N(w_s)} \tilde{f}^w_{w_sj} = -1$ $\sum_{j\in N(w_t)} \tilde{f}^w_{jw_t} - \sum_{j\in N(w_t)} \tilde{f}^w_{w_tj} = 1$ $\sum_{j\in N(i)} \tilde{f}^w_{ji} - \sum_{j\in N(i)} \tilde{f}^w_{ij} = 0$	$\tilde{f}^w_{ij} \leq \sum_{\ell\in\mathcal{L}} x^{\ell}_{ij}$ $f^w_{w_sj} \leq \sum_{\ell\in\mathcal{L}} y^{\ell}_j$ $f^w_{iw_t} \leq \sum_{\ell\in\mathcal{L}} y^{\ell}_i$	$(1/\lambda)\sum_{\{i,j\}\in E} d_{ij}\cdot\tilde{f}^w_{ij} + \sum_{\{i,j\}\in E_f} M_1\cdot\tilde{f}^w_{ij}-$ $-\Lambda u^{ALT}_w - M_1(1-p_w) \leq 0$	$1 + q(2+n+3m) + r(4+3m)$ subtour elimination constraints: $\Theta(2^n)$	$q(1+n+m) + r(1+m)$

Modelling of the Rapid Transit Network with lines							
	Budget constraints	Design	Flow conservation	Location-allocation	Splitting demand	♯Constraints	♯Variables
Model 9	$\sum_{w\in W}\sum_{k\in P_w}\hat{h}_w^k \leq \Gamma$ $\sum_{w\in W}\sum_{k\in P_w}\hat{h}_w^k c^k \leq C_{max}$ <u>TRANSFERS CONSTRAINTS:</u> $\sum_{j\in N(i)}\tilde{f}_{ji}^{wk} + \tilde{f}_i^w \geq \sum_{j\in N(i)}\tilde{f}_{ij}^{wk}$	$\tilde{f}_{ij}^{wk} \leq \Gamma_{ij}^{wk}\hat{h}_w^k$ $\tilde{f}_{ij}^w \leq \sum_{k\in P_w}\tilde{f}_{ij}^{wk}$	$\sum_{i\in N(w_s)} f_{iw_s}^w - \sum_{j\in N(w_s)} f_{w_sj}^w = -1$ $\sum_{i\in N(w_t)}\tilde{f}_{iw_t}^w - \sum_{j\in N(w_t)}\tilde{f}_{w_tj}^w = 1$ $\sum_{i\in N(k)}\tilde{f}_{ik}^w - \sum_{j\in N(k)}\tilde{f}_{kj}^w = 0$		$(1/\lambda)\sum_{\{i,j\}\in E} d_{ij}\sum_{k\in P_w}\tilde{f}_{ij}^{wk}+$ $+TP\sum_{i\in N}\tilde{f}_i^w-$ $-\Lambda u_w^{ALT} - M(1-p_w) \leq 0$	$2 + r(1+\|k\|(n+m)+n+m)$	$r(1+\|k\|+2\|k\|m+n)$
Model 10		$\sum_{i\in N_t}\sum_{j\in N_s\setminus N_t}(\tilde{x}_{ij}^s + \tilde{x}_{ji}^s) \geq 1$ $\sum_{i\in N_t}\tilde{y}_i = 1$ $\tilde{y}_i^s + \tilde{y}_j^s \leq 1$ $\sum_{i,j\in Q\subseteq N_s}\tilde{x}_{ij}^s \leq \sum_{k\in Q\setminus\{q\}:q\in Q,\|q\|>2}\tilde{y}_k^s$		$p_{ij} \leq \sum_{s\in S}\tilde{y}_i^s$ $p_{ij} \leq \sum_{s\in S}\tilde{y}_j^s$		$1+2m+\|S\|+n\|S\|$ subtour elimination constraints: $\Theta(2^n)$	$n+n\|S\|+m$
Model 11	$\sum_{\{i,j\}\in E} c_{ij}x_{ij} + \sum_{\ell\in\mathcal{L}}\sum_{i\in N} c_i y_i^\ell \leq C_{max}$	$x_{ij}^\ell \leq y_i^\ell$ $x_{ij}^\ell \leq y_j^\ell$ $x_{ij}^\ell \leq x_{ij}$ $x_{ij} \leq \sum_{\ell\in\mathcal{L}} x_{ij}^\ell$ $\sum_{j\in N(i)} x_{ij}^\ell + \sum_{j\in N(i)} x_{ji}^\ell \leq 2$ $h_\ell + \sum_{\{i,j\}\in E} x_{ij}^\ell = \sum_{i\in N} y_i^\ell$ $\frac{1}{2} - \sum_{\{i,j\}\in E} x_{ij}^\ell + M(h_\ell - 1) \leq 0$ $+M(h_\ell - 1) \leq 0$ $\frac{1}{2} - \sum_{\{i,j\}\in E} x_{ij}^\ell + Mh_\ell \geq 0$ $\sum_{\{i,j\}\in B:i,j\in B} x_{ij}^\ell \leq \|B\|-1$	$\sum_{(i,w_s)\in E}\tilde{f}_{iw_s}^w + \sum_{(i,w_s)\in E'}\tilde{\varphi}_{iw_s}^w = 0$ $\sum_{w_s\in N(j)}\tilde{f}_{w_sj}^w + \sum_{w_s\in\bar{N}(j)}\tilde{\varphi}_{w_sj}^w = 1$ $\sum_{i\in N(w_t)}\tilde{f}_{iw_t}^w + \sum_{i\in\bar{N}(w_t)}\tilde{\varphi}_{iw_t}^w = 1$ $\sum_{w_t\in N(j)}\tilde{f}_{w_tj}^w + \sum_{w_t\in\bar{N}(j)}\tilde{\varphi}_{w_tj}^w = 0$ $\sum_{i\in\bar{N}(k)}\tilde{\varphi}_{ik}^w - \sum_{k\in\bar{N}(j)}\tilde{\varphi}_{kj}^w = 0$ $\sum_{i\in N(k)}\tilde{f}_{ik}^w - \sum_{k\in\bar{N}(j)}\tilde{f}_{kj}^w = 0$ $\tilde{f}_{ij}^w + \tilde{\varphi}_{ij}^w \leq 1$	$\tilde{f}_{ij}^w + p_w - 1 \leq \sum_{\ell\in\mathcal{L}} x_{ij}^\ell$	$\epsilon + u_w - u_w^{RTS} - M(1-p_w) \leq 0$ $u_w = \sum_{\{i,j\}\in E} d_{ij}\tilde{f}_{ij}^w + \sum_{\{i,j\}\in E'} d'_{ij}\tilde{\varphi}_{ij}^w$	$1+n+3q(m+1)+$ $+2r(1+n+m)$ subtour elimination constraints: $\Theta(2^n)$	$q(1+n+m)+2r(1+m)+m$

Modelling of the physical railway network

	Budget constraints	Design	Flow conservation	Location-allocation	Splitting demand	♯Constraints	♯Variables
Model 12	$\sum_{\{i,j\}\in E} c_{ij}x_{ij}+$ $+\sum_{\ell\in\mathcal{L}}\sum_{i\in N} c_i y_i^\ell \leq C_{max}$	$x_{ij}^\ell \leq y_i^\ell$ $x_{ij}^\ell \leq y_j^\ell$ $x_{ij}^\ell = x_{ji}^\ell$ $x_{ij}^\ell \leq x_{ij}$ $x_{ij} \leq \sum_{\ell\in\mathcal{L}} x_{ij}^\ell$ $h_\ell + \sum_{\{i,j\}\in E} x_{ij}^\ell = \sum_{i\in N} y_i^\ell$ $\frac{1}{2} - \sum_{\{i,j\}\in E} x_{ij}^\ell + M(h_\ell - 1) \leq 0$ $\frac{1}{2} - \sum_{\{i,j\}\in E} x_{ij}^\ell + Mh_\ell \geq 0$ $\sum_{i\in B}\sum_{j\in B} x_{ij}^\ell \leq \|B\| - 1$ $\sum_{j\in N(i)} x_{ij}^\ell \leq 2$	$\sum_{i\in N(w_s)} \tilde{f}_{iw_s}^w + \sum_{i\in\bar{N}(w_s)} \tilde{\varphi}_{iw_s}^w = 0$ $\sum_{w_s\in N(j)} \tilde{f}_{w_sj}^w + \sum_{w_s\in\bar{N}(j)} \tilde{\varphi}_{w_sj}^w = 1$ $\sum_{i\in N(w_t)} \tilde{f}_{iw_t}^w + \sum_{i\in\bar{N}(w_t)} \tilde{\varphi}_{iw_t}^w = 1$ $\sum_{w_t\in N(j)} \tilde{f}_{w_tj}^w + \sum_{w_t\in\bar{N}(j)} \tilde{\varphi}_{w_tj}^w = 0$ $\sum_{i\in N(k)} \tilde{f}_{ik}^w - \sum_{k\in N(j)} \tilde{f}_{kj}^w = 0$ $\sum_{i\in\bar{N}(k)} \tilde{\varphi}_{ik}^w - \sum_{k\in\bar{N}(j)} \tilde{\varphi}_{kj}^w = 0$ $\tilde{f}_{ij}^w + \tilde{\varphi}_{ij}^w \leq 1$	$\tilde{f}_{ij}^w + p_w - 1 \leq \sum_{\ell\in\mathcal{L}} x_{ij}^\ell$	$\epsilon + u_w - u_w^{ALT} - M(1-p_w) \leq 0$ $u_w = \sum_{\{i,j\}\in E} d_{ij}\tilde{f}_{ij}^w + \sum_{\{i,j\}\in E'} d'_{ij}\tilde{\varphi}_{ij}^w$	$1 + m + q(1+n+4m)+$ $+2r(1+n+m)$ subtour elimination constraints: $\Theta(2^n)$	$q(1+n+m)+$
Model 13	$\sum_{\{i,j\}\in E} c_{ij}x_{ij} + \sum_{i\in N} c_i y_i \leq C_{\max}$	$x_{ij} \leq y_i$ $x_{ij} = x_{ji}$	$\sum_{\{i,j\}\in E: i\in i\in N(w_s)} \tilde{f}_{iw_s}^w = 0$ $\sum_{j\in N(w_s)} \tilde{f}_{w_sj}^w = p_w$ $\sum_{i\in N(w_t)} \tilde{f}_{iw_t}^w = p_w$ $\sum_{j\in N(w_t)} \tilde{f}_{w_tj}^w = 0$ $\sum_{i\in N(k)} \tilde{f}_{ik}^w - \sum_{j\in N(k)} \tilde{f}_{kj}^w = 0$	$\tilde{f}_{ij}^w + p_w - 1 \leq x_{ij}$	$\epsilon + u_w - u_w^{RTS} - M(1-p_w) \leq 0$ where $u_w = \sum_{\{i,j\}\in E} d_{ij}\tilde{f}_{ij}^w + u_w^{RTS}(1-p_w)$ if the objective function varies have to add the constraint $-u_w + u_w^{RTS} - Mp_w \leq 0$	$1 + 2m + r(4+n+m)$	$r(2+m)+n+m$
Model 14	$\sum_{\ell\in\mathcal{L}}(\sum_{\{i,j\}\in E} c_{ij}x_{ij}^\ell + \sum_{i\in N} c_i y_i^\ell) \leq C_{\max}$	$x_{ij}^\ell \leq y_i^\ell$ $x_{ij}^\ell \leq y_j^\ell$ $x_{ij}^\ell = x_{ji}^\ell$ $\sum_{j\in N(i)} x_{ij}^\ell \leq 2$ $h_\ell + \sum_{\{i,j\}\in E} x_{ij}^\ell = \sum_{i\in N} y_i^\ell$ $h_\ell \leq \sum_{\{i,j\}\in E} x_{ij}^\ell \leq M_2 h_\ell$ $\sum_{i\in B}\sum_{j\in N} x_{ij}^\ell \leq \|B\| - 1$	$\sum_{j\in N(w_s)} \tilde{f}_{jw_s}^w - \sum_{j\in N(w_s)} \tilde{f}_{w_sj}^w = -1$ $\sum_{j\in N(w_t)} \tilde{f}_{jw_t}^w - \sum_{j\in N(w_t)} \tilde{f}_{w_tj}^w = 1$ $\sum_{j\in N(i)} \tilde{f}_{ji}^w - \sum_{j\in N(i)} \tilde{f}_{ij}^w = 0$	$\tilde{f}_{ij}^w \leq \sum_{\ell\in\mathcal{L}} x_{ij}^\ell$ $f_{w_sj}^w \leq \sum_{\ell\in\mathcal{L}} y_j^\ell$ $f_{iw_t}^w \leq \sum_{\ell\in\mathcal{L}} y_i^\ell$	$\sum_{\{i,j\}\in E\cup E_f} d_{ij}\cdot\tilde{f}_{ij}^w \leq \Lambda u_w^{RTS}(1-p_w)$	$1 + q(n+3m)+$ $+r(1+3n+m)$ subtour elimination constraints: $\Theta(2^n)$	k $q(1+n+m)+r(1+m)$

Chapter 3

Simultaneous frequency and capacity problem

3.1. Introduction

In this chapter, we will focus on the line planning process. In the strategic planning process of a rapid transit network, decisions about a line plan, the size of trains and the number of crews are necessary. We assume the infrastructure (tracks and stations) as well as its associated lines are already given.

In railway terminology, a line is characterized by several aspects: two different terminal stations, a sequence of intermediate stops, its frequency, and the vehicle capacity. The traditional line planning problem consists of finding a set of lines (a line plan) and their frequencies providing a good service according to a certain objective, which is usually oriented towards the passengers or the operator. A review of different objective functions is presented in Schöbel (2011). As in the rapid transit network design problems, the models can be classified into several categories depending on the point of view that is considered. A classification of these models is presented in Schöbel (2011). The author distinguishes between models oriented to passenger and models oriented to costs.

Usually, the goal of models oriented to costs is to minimize the train operating costs. The problem presented in Claessens et al. (1998) consists of determining a set of lines from a line pool, the frequency and type of train for each line as well as the number of carriages for each train, minimizing costs related to train operation. The cost structure is defined according to fixed costs per carriage and hour, variable costs per carriages and kilometer and variable costs per train and kilometer. By means of a modal split procedure, the passengers are assigned a priori in the system. In order to simplify the problem, the authors define binary variables representing if a line ℓ is served by trains of type t with c carriages. However, the model is a nonlinear programming program and some techniques to make the problem manageable are considered. In Goossens et al. (2004) a branch-and-cut approach based on the models of Claessens et al. (1998) is described. More recently, Goossens et al. (2006) extend this model to the multi-type case in which not all trains need to stop at all stations. First they describe the same problem than Claessens et al. (1998) and then extend the model by considering different types of trains (regional, intercity and interregional). The passengers are assigned a priori to the different train types. This problem is described as a multi-commodity flow problem.

With respect to passenger oriented, one of most common objectives in the literature is to maximize the number of direct trips, see Bussieck (1998) and Bussieck et al. (1997). A major drawback of this objective is that it does not take travel times into account, and therefore it may yield a network with few transfers but with long travel times. Another paper such as Schöbel and Scholl (2006) considered as objective function the total travel time of all passengers. In order to compute the travel time in the system, a penalty for each transfer representing the inconvenience for the passengers is introduced. In order to model the line planning problem, the authors define a graph structure named *Change&Go*. The aim is to find a set of lines, a path for each origin-destination pair, respecting a budget on the operating costs.

The problem we are treating with consists of determining the frequency and the train size of each line, maximizing the net profit of line plan. Furthermore, we have assumed that all passengers willing to travel in the RTS can be transported. The passengers choose their routes and their transport mode according to traveling times, which are defined by means of the considered frequencies. Concretely, in this chapter we simultaneously select the frequency and the number of carriages for each line of the RTS maximizing the net profit.

The following example shows how both the frequency and the number of carriages are decisive factors to be considered when planning rapid transit lines.

Example 3.1.1 *Consider a simple case in which* $S = \{1, 2, 3\}, \mathcal{L} = \{\ell_1 = \{1, 2\}, \ell_2 = \{2, 3\}\}$. *A trip from* 1 *to* 3 *includes a waiting time at the origin station and a transfer time at station* 2. *Note that therefore the expected travel time of this OD pair depends on the frequencies of the lines, and thus passengers choose the RTS mode or the competing mode* *depending on this. Note as well that if the number of potential passengers is large enough, then we also need to consider including more carriages in the trains, as the maximum allowed frequency might not be enough to transport them all.*

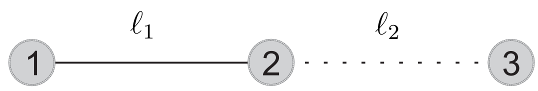


Figure 3.1.: Example with two lines.

The main differences between our model and those defined in Claessens et al. (1998) and Goossens et al. (2006) are the following: they do not take into account an alternative mode competing with the rapid transit transport, they consider different train types and we present a model integrating the traffic assignment procedure in the optimization process. Furthermore, we present a different objective function which includes income derived from the passengers as well as costs related to the rolling stocks acquisition and personnel costs due to operation of trains. Thanks to the incorporation of a logit function, the level of demand will depend on the quality of the services offered. Moreover, we consider two different situations: the problem with an unlimited number of carriages and the problem with a maximum number of carriages. The first problem assumes the maximum number of possible carriages is a sufficiently large number in order to transport all people traveling on each line in the RTS. In other words, the RTS is a non-crowding network. In contrast, the capacitated problem has a limitation on the number of carriages and the RTS can become a congested network. For the last case, a congestion function measuring the level of in-vehicle crowding is introduced in the model.

The remainder of the chapter is structured as follows. In Section 3.2 we describe the problem, the needed data and notations as well as the objective function. In Section 3.3 we formally define the problem for the case without capacity limitation, which is modeled in Section 3.3.1 as a mathematical programming program. A heuristic algorithm as well as efficient approaches are presented in Section 3.3.2. The problem with a limited number of carriages is introduced in Section 3.4. Computational experiments are carried out on Appendix C. Computational comments are presented in Section 3.5.2. The chapter ends with some conclusions.

3.2. The problem

As commented, the problem we are dealing with consists of maximizing the net profit of a line plan by selecting the frequency and the train size of each line, assuming that all passengers willing to travel in the RTS can be transported. We distinguish two different versions of this problem. At the first problem an unlimited number of possible carriages is considered. This yields solutions where the number of carriages per train is the minimum number of carriages so that all passengers can be transported. However, this is not true in the second version of this problem. In this problem a limited number of carriages is taken into account and the network can be a congested network. We introduce the crowding effect by means of a congestion function which is depending on the load on each arc. This function assigns a time penalty on each congested arc, therefore modifying, the problem instance. The crowding effect is assumed to be the in-vehicle crowding. In other words, we only assume congestion inside train and not at the platforms. We also want to remark that solutions in which platform crowding appears, are not taken into account.

In both problems presented, we define a parameter σ in order to allow solutions that exceed the capacity in a small number of passengers. Apart from the number of carriages, an important difference between the uncapacitated and capacitated problem is the congestion effect which influences on the passenger's behavior and, therefore, on the profit.

3.2.1. Data and notation

We now formally describe the Rapid Transit System Simultaneous Frequency and Capacity Problem (RTSSFCP), which takes the following input data. We assume the existence of a set of stations, $N = \{i_1, \ldots, i_n\}$ and a set of lines $\mathcal{L} = \{\ell_1, \ldots, \ell_{|\mathcal{L}|}\}$ in the RTS. For the sake of readability we will identify a station with its subindex whenever this creates no confusion. Let denote len_ℓ and n_ℓ be the length and the number of stations of line ℓ, respectively. Each line $\ell \in \mathcal{L}$ is defined as $\ell = \{(i_1, i_2), (i_2, i_3), \ldots, (i_{n_\ell - 1}, i_{n_\ell})\}$, where arcs form two paths; $\{i_1, i_2, i_3, \ldots, i_{n_\ell}\}$ and $\{i_{n_\ell}, i_{n_\ell - 1}, \ldots, i_1\}$. Each couple of arcs (i_{j_1}, i_{j_2}) and (i_{j_2}, i_{j_1}) represents an edge $\{i_{j_1}, i_{j_2}\}$.

In order to compute traffic flows we need the set of (directed) arcs associated with E. We therefore define A as the set of (directed) arcs of the network. Note that $E = \{\{i, j\} : (i, j) \in A, i < j\}$. Let $((N, E), \mathcal{L})$ be a RTS line network describing the RTS system. Let $d_{ij} = d_{ji}$ be the length of edge $\{i, j\} \in E$. The parameter d_{ij} can also represent the time needed to traverse edge $\{i, j\}$, transforming distances in times by means of the parameter

λ, which represents the average distance traveled by a train in a hour (commercial speed). We consider the same value of λ for all trains. Let ν_ℓ be the cycle time of line ℓ, that is, the time necessary for a train of line ℓ to go from the initial station to the final station and returning back. Note that $\nu_\ell = 2 \cdot len_\ell/\lambda$. The competing mode (private car, bus, etc) is represented by an undirected graph $G_{E'} = G(N, E')$. The nodes are assumed to be coincident with those of the rapid transit mode: they could represent origin or destination of the aggregated demands; however, edges are possibly different. For each edge $\{i, j\} \in E'$, let d'_{ij} be the traversing time of such link by the competing mode. Let $W = \{w_1, \ldots, w_{|W|}\} \subseteq N \times N$ be a set of ordered origin-destination (OD) pairs, $w = (w_s, w_t)$. For each OD pair $w \in W$, g_w is the expected number of passengers per hour for an average day and u_w^{ALT} is the travel time using the alternative mode of OD pair w, respectively.

With respect to costs, we distinguish three types: related to the operation, the personnel and the investment. Concerning rolling stock, we define a cost for operating one locomotive per unit of length c_{loc} as well as a cost representing operating cost of one carriage c_{carr} per length unit. Both parameters include running costs such as fuel or energy consumption. These terms can be easily adapted to another type of rolling stock. Related to the personnel costs, a cost c_{crew} per train and year is given. For the rolling stock acquisition, we consider two costs: the purchase price of the necessary locomotives I_{loc} per train and the purchase price of one carriage I_{carr}. Concerning capacity, let Θ be the carriage capacity measured in number of passengers seating and standing. We consider a minimum number $\delta^{\min}$ of carriages and a sufficiently large number Δ of carriages that can be included in a train. We define the capacity associated to a train as the maximum number of passengers that it can transport at any given time. More precisely, we define the capacity of a train of a line ℓ by means of two factors: the capacity of a carriage (Θ) and the number of carriages forming the train (δ_ℓ). We consider a fixed finite set of possible frequencies $\mathcal{F}$ for lines of the RTS. We assume the frequency of each line takes values between a minimum and maximum frequency in order to guarantee a certain level of service in the network. To be more precise, not all feasible frequency values between this minimum and maximum can be considered. Note that in real systems the frequencies have to produce a regular timetable. To take this requirement into account, we describe the set of ordered possible frequencies as $\mathcal{F} = \{\phi^1, \phi^2, \ldots, \phi^{|\mathcal{F}|}\}$, where each $\phi^q \in \mathbb{N}$, $1 \leq q \leq |\mathcal{F}|$ and $|\mathcal{F}| \geq 2$. Let ρ be the total number of hours that a train is operating per year and η be the fare per trip which is the same for all trips regardless of their length/duration. A parameter needed to compute the transfer time is uc_i, which represents the time spent in changing platforms at station i.

3.3. Uncapacitated problem

We now describe the Uncapacitated Rapid Transit System Simultaneous Frequency and Capacity Setting Problem (URTSSFCP) in more detail. Section 3.3.1 presents the mathematical programming program. In Section 3.3.2 several techniques for improving the efficiency of the mathematical programming model are presented.

3.3.1. A mathematical programming program

We first introduce a mathematical programming program to solve the URTSSFCP, which uses the following sets of variables:

- $\psi_\ell \in \mathcal{F}$ is the frequency of line ℓ (number of services per hour).
- $\delta_\ell \in \{\delta^{min}, \dots, \Delta\}$ is the number of carriages used by trains of line ℓ.
- $u_w^{RTS} > 0$ represents the travel time of pair w using the RTS network.
- $f_w^{RTS} \in [0, 1]$ is the proportion of OD pair w using the RTS network.
- $\tilde{f}_{ij}^{w\ell} = \begin{cases} 1, & \text{if the OD pair } w \text{ traverses arc } (i,j) \in A \text{ using line } \ell \\ 0, & \text{otherwise.} \end{cases}$
- $\tilde{f}_i^{w\ell\ell'} = \begin{cases} 1, & \text{if demand of pair } w \text{ transfers in station } i \text{ from line } \ell \text{ to line } \ell' \\ 0, & \text{otherwise.} \end{cases}$

The average travel time associated to OD pair $w = (w_s, w_t) \in W$ using the RTS network can be explicitly defined as follows:

$$\begin{aligned} u_w^{RTS} &= \sum_{\ell \in \mathcal{L}} \sum_{j:\{w_s,j\} \in \ell} \frac{60 \tilde{f}_{w_s j}^{w\ell}}{2\psi_\ell} + (60/\lambda) \sum_{\ell \in \mathcal{L}} \Big(\sum_{\{i,j\} \in \ell} \tilde{f}_{ij}^{w\ell} d_{ij} \Big) \\ &+ \sum_{\ell \in \mathcal{L}} \sum_{\ell' : \ell' \neq \ell} \sum_{i \in \ell \cap \ell'} \tilde{f}_i^{w\ell\ell'} \Big(\frac{60}{2\psi_{\ell'}} + uc_i \Big), \; w = (w_s, w_t) \in W. \end{aligned} \tag{3.1}$$

The first term in (3.1) is the average waiting time at the origin station. Since we are dealing with high frequencies systems, we assume passengers go to stations without knowing the exact departures time of trains and, therefore, it is reasonable assuming that the average waiting time is half the headway (time between consecutive services). The second term in (3.1) is the in-vehicle time. The third one is the time spent in transfers. For the reason previously commented, the average waiting time is assumed to be half the headway. In this case, we have to add the required time to change platforms.

Another variable that can be explicitly defined is the assignment f_w^{RTS} of demand to the RTS system. We assume the number of passengers who use a transport system varies depending on the provided service. More specifically, the proportion of an OD pair using each mode may be different depending on the characteristics of the RTS to be designed and on the competing transport mode. Therefore, the demand is split between the RTS and the alternative mode according to the generalized cost of each mode. The modal split is modeled by using logit type functions (Ortúzar and Willumsem (1990)) as opposed to binary variables which are used in very complex problems. Two real positive parameters are usually required: α, representing the market share of each mode, and β, representing the importance of each transportation mode (Marín and García-Ródenas (2009)). Let α^{RTS} and α^{ALT} be the market share of RTS and alternative mode, respectively. Let us denote $\alpha = \alpha^{ALT} - \alpha^{RTS}$. As in García-Ródenas et al. (2006), β is supposed to be independent of the modes. Therefore, the proportion of OD pair w using the RTS mode is

$$f_w^{RTS} = \frac{1}{1 + e^{(\alpha - \beta(u_w^{ALT} - u_w^{RTS}))}}, \; w \in W. \tag{3.2}$$

The logit model estimates the proportion of users assigned to each mode for each origin-destination pair in a continuous way. Note that this proportion depends on the travel time in each transport mode.

In Schmidt and Schöbel (2010) the route decisions are integrated in the line planning problem. To this end, the authors consider a change and go graph. On this graph, a modified Dijkstra algorithm is applied and adapted to obtain the shortest path of an origin-destination pair.

Objective function

As mentioned, we consider the existence of public economic support for the operation of the RTS during certain planning horizon. This assumption is very common in the development of rapid transit networks around the world. Usually, governments provide subsidies on the basis of the number of passengers or passenger-kilometer in order to guarantee certain positive margin to companies exploiting the transportation system. For instance (see newspaper http://www.20minutos.es/noticia/2028399/0/madrid/empresas-privadas/metro-ligero/).

The objective function considered is the net profit z_{NET} of the rapid transit network (Li et al. (2011b), Feifei and Haicheng (2012), Feifei (2014)). This profit is expressed as the difference between revenue and total operation cost in terms of monetary units over a planning horizon. In this chapter, we will not consider costs related to the construction of the RTS since the infrastructure network is already built. For the sake of readability, we will repeat some terms presented in Chapter 2.

The total revenue for the $\hat{\rho}$ years is computed as the number of passengers who use the RTS during the planning horizon, times the passenger fare μ plus the passenger subsidy, η, which is the same for all passengers independently of the length of their trips. So, the revenue is mathematically expressed as

$$z_{REV} = (\eta + \mu)\rho\hat{\rho} \sum_{w \in W} g_w f_w^{RTS}. \tag{3.3}$$

The operation cost of a network is expressed by means of a fixed cost z_{FOC} and a variable cost z_{VOC}. The fixed operating cost includes maintenance costs and overheads. The fixed operating cost depends on the infrastructure. This term does not affect the objective function and is not considered, see Feifei and Haicheng (2012), Feifei (2014). The variable operating cost z_{VOC} over the planning horizon is defined as the sum of the crew operating cost z_{CrOC} and the rolling stock cost z_{RSOC}.

The crew operating cost z_{CrOC} includes the crew cost induced by the operation of all trains in the time horizon $\hat{\rho}$. This cost is affected by the required fleet size B_ℓ. The required fleet for each line ℓ can be defined by means of the product of its frequency and its cycle time ν_ℓ as follows:

$$B_\ell = \lceil \psi_\ell \nu_\ell \rceil = \lceil 2\psi_\ell \cdot len_\ell / \lambda \rceil.$$

Thus, the crew operating cost in the planning horizon is

$$z_{CrOC} = \hat{\rho} \cdot c_{crew} \sum_{\ell \in \mathcal{L}} B_\ell. \tag{3.4}$$

The rolling stock operation cost of a train in one hour is defined as the distance λ traveled by the train, times the cost of moving the train with δ_ℓ carriages and is approximated by $c_{loc} + c_{carr}\delta_\ell$ (García and Martín (2012)). Therefore, the rolling stock operation cost in the whole planning horizon z_{RSOC} is

$$z_{RSOC} = \hat{\rho}\rho \sum_{\ell \in \mathcal{L}} B_\ell \lambda (c_{loc} + c_{carr}\delta_\ell), \tag{3.5}$$

and the variable operating cost in the planning horizon is $z_{VOC} = z_{RSOC} + z_{CrOC}$.

The fleet investment cost for each train is the cost of purchasing the locomotives and the carriages. Therefore, the fleet acquisition cost of all trains z_{FAC} is computed as

$$z_{FAC} = \sum_{\ell \in \mathcal{L}} B_\ell (I_{loc} + I_{carr} \cdot \delta_\ell). \tag{3.6}$$

So, we define the net profit z_{NET} associated to the rapid transit network as

$$z_{NET} = z_{REV} - (z_{VOC} + z_{FAC}). \tag{3.7}$$

Constraints

The constraints of the problem are formulated as follows.

- Flow conservation constraints

$$\sum_{\ell \in \mathcal{L}} \sum_{j:(w_s,j) \in \ell} \tilde{f}^{w\ell}_{w_s j} = 1,\ w = (w_s, w_t) \in W \tag{3.8}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{i:(i,w_t) \in \ell} \tilde{f}^{w\ell}_{i w_t} = 1,\ w = (w_s, w_t) \in W \tag{3.9}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{j:(j,w_s) \in \ell} \tilde{f}^{w\ell}_{j w_s} = 0,\ w = (w_s, w_t) \in W \tag{3.10}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{j:(w_t,j) \in \ell} \tilde{f}^{w\ell}_{w_t j} = 0,\ w = (w_s, w_t) \in W \tag{3.11}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{i:(i,k) \in \ell} \tilde{f}^{w\ell}_{ik} - \sum_{\ell \in \mathcal{L}} \sum_{j:(k,j) \in \ell} \tilde{f}^{w\ell}_{kj} = 0,\ w = (w_s, w_t) \in W, k \in N \setminus \{w_s, w_t\}. \tag{3.12}$$

- Transfers

$$2\tilde{f}_i^{w\ell\ell'} \geq \sum_{k:(k,i)\in\ell} \tilde{f}_{ki}^{w\ell} + \sum_{k:(i,k)\in\ell'} \tilde{f}_{ik}^{w\ell'} - 1 \tag{3.13}$$

$$2\tilde{f}_i^{w\ell\ell'} \leq \sum_{k:(k,i)\in\ell} \tilde{f}_{ki}^{w\ell} + \sum_{k:(i,k)\in\ell'} \tilde{f}_{ik}^{w\ell'}, \tag{3.14}$$

$$\ell, \ell' \in \mathcal{L}, \ell \neq \ell', i \in \ell \cap \ell', i \neq w_s, i \neq w_t, w = (w_s, w_t) \in W.$$

- Capacity constraints

$$\sum_{w\in W} g_w f_w^{RTS} \tilde{f}_{ij}^{w\ell} \leq \Theta\delta_\ell\psi_\ell, \ \ \ell \in \mathcal{L}, \{i,j\} \in \ell. \tag{3.15}$$

- Binary constraints

$$\tilde{f}_{ij}^{w\ell}, \tilde{f}_k^{w\ell\ell'} \in \{0,1\}, \ k \in N, (i,j) \in A, \ell \in \mathcal{L}, w \in W.$$

- Integer constraints

$$\delta_\ell \in \{\delta^{min}, \ldots, \Delta\}, \ell \in \mathcal{L}$$
$$\psi_\ell \in \mathcal{F}, \ell \in \mathcal{L}.$$

- Other constraints

$$f_w^{RTS} = \frac{1}{1 + e^{(\alpha - \beta(u_w^{ALT} - u_w^{RTS}))}} \in [0,1], w \in W.$$

Recall that the model includes constraints related to the proportion of passengers using the RTS (Equation (3.2)) as well as the travel time for each OD pair (Equation (3.1)). Constraints (3.8) to (3.12) are flow conservation constraints for the f variables. Constraints (3.13) and (3.14) ensure that if an OD pair w enters station $k \in N$ using one line and exits this station using another line, then a transfer is done.

Constraints (3.15) indicates the total capacity per hour of such line is a sufficiently large number in order to transport all passengers preferring to travel in the RTS. The URTSSFCP consists of maximizing z_{NET} subject to constraints (3.1), (3.2) and (3.8)–(3.15).

Note that this model has some nonlinearities: the definition of f_w^{RTS} and u_w^{RTS}, as well as constraints (3.15). Some of them can be easily removed, some others cannot. Therefore our mathematical programming program is non-linear, which makes it difficult for realistic instances.

3.3.2. Efficient approaches

In this section we show different techniques for improving the efficiency of the model presented in the previous section.

Constraint linearization

The terms in (3.15) expressed as a product of a binary variable and a real variable, that is, $\tilde{f}_{ij}^{w\ell}$ and f_w^{RTS}, are transformed into linear constraints by means of a new variable $\xi_{ij}^{w\ell} = f_w^{RTS} \tilde{f}_{ij}^{w\ell} \geq 0$. Constraints (3.15) are substituted by

$$\sum_{w \in W} g_w \xi_{ij}^{w\ell} \leq \Theta\, \delta_\ell\, \psi_\ell,\ \ell \in \mathcal{L}, \{i,j\} \in \ell \tag{3.16}$$

$$\xi_{ij}^{w\ell} \leq \tilde{f}_{ij}^{w\ell}, \ell \in \mathcal{L}, \{i,j\} \in \ell, w \in W \tag{3.17}$$

$$f_w^{RTS} - (1 - \tilde{f}_{ij}^{w\ell}) \leq \xi_{ij}^{w\ell},\ \ell \in \mathcal{L}, \{i,j\} \in \ell, w \in W \tag{3.18}$$

$$\xi_{ij}^{w\ell} \leq f_w^{RTS},\ \ell \in \mathcal{L}, \{i,j\} \in \ell, w \in W. \tag{3.19}$$

A naive algorithm

In this section a naive algorithm to solve our model is presented. We construct all possible combinations of frequencies. So, for each element of this set, the frequency of each line $\psi_\ell \in \mathcal{F}$ is fixed. For each combination, we solve a mathematical model. We select the solution with maximum profit z_{NET}. Note that constraint (3.16) is now a linear. To fix ideas, let $\upsilon \in \mathbb{N}^{|\mathcal{L}|}$ be a vector representing a possible frequency combination. Each component $\upsilon_q \in \mathcal{F}$ is the assigned frequency to line ℓ_q. We define the restricted problem taking into account these parameters and we denote it by URTSSFCP(υ). The solution procedure is shown in Algorithm 1.

Note that, thanks to the linearization defined in Section 3.3.2 and this naive algorithm, all constraints but the assignment demand constraint (3.2), which could also be linearizated,

Data: All possible frequencies combinations
for *each combination* v **do**
  solve $URTSSFCP(v)$;
end
Result: $\arg\max_{v} URTSSFCP(v)$.

Algorithm 1: Pseudocode for the naive algorithm.

are linear constraints. Observe that we have to solve $(|\mathcal{F}|)^{|\mathcal{L}|}$ times model URTSSFCP(v) (once for each possible combination of frequencies).

An exact algorithm

In this section we introduce an algorithm that solves our problem to optimality, that is, it provides a configuration of frequencies and capacities (number of carriages per train of each line) that maximizes the net profit of the network. The idea is to iteratively check all possible combinations of frequencies. Once the frequencies have been set, the shortest path that takes into account transfer and waiting times on the rapid transit network for each OD pair can easily be calculated. From these shortest paths we compute the number of passengers traveling on each line and arc. For each line, the arc that has the highest number of passengers is the one defining the minimum capacity that such line should have. Once these minimum required number of carriages have been calculated for each line, we can easily compute the profit of the network. Note that this value is the maximum profit for a fixed configuration of frequencies. Algorithm 2 shows a pseudocode of this algorithm.

We would like to emphasize that, once the frequencies are known, the problem reduces to finding the minimum number of carriages per train and line so that all passengers can be transported. It is trivial to prove that such a combination of number of carriages yields the solution that transports all passengers at minimum cost.

A heuristic algorithm

Due to the complexity of the mathematical program that models the URTSSFCP, a heuristic technique is proposed to obtain good solutions in a reasonable amount of time. A heuristic algorithm is a procedure which obtains "good" solutions in a reasonable amount of time, although there is no guarantee that such solutions are optimal. For solving the

Data: A line network $((N,E),\mathcal{L})$, a set of possible frequencies $\mathcal{F}$ and a minimum capacity δ^{min}
for *each possible combination of frequencies* **do**
  $z = \{\}$;
  for *each line* ℓ **do**
    Find the arc e_ℓ of ℓ with maximum load;
    Find the minimum number of carriages needed to transport all passengers traversing e_ℓ;
    Compute the profit z_{NET};
    $z = z \cup \{z_{NET}\}$;
  end
end
return $\max\{z\}$;
Result: The frequencies and capacities configuration with the maximum profit.

Algorithm 2: The exact algorithm for the rapid transit network frequency and capacity setting problem.

problem we propose a new method inspired on the Heuristic Local Search Algorithm (HLSA) defined in Gallo et al. (2011).

We begin by describing certain local moves we will apply in order to obtain a variety of solutions. Local search is a successful general approach for finding high quality solutions to hard combinatorial optimization problems in a reasonable amount of time (see Stützle (1999)). In general, a local search algorithm considers an initial solution and iteratively modifies it in order to obtain a better solution. The modifications are done by means of an appropriated neighborhood structure for each solution. In other words, a local change (defined by a neighborhood structure) is applied to a solution in each iteration in order to improve it.

Definition 3.1 *Moves on the current solution*

For each line ℓ_p, ψ_{ℓ_p} takes a value in $\mathcal{F}$ which we will denote by ϕ^{q_p}, $1 \leq q_p \leq |\mathcal{F}|$. For this frequency ϕ^{q_p} we define two possible moves.

- *$Move^+$: The selected frequency value is changed by the following value in $\mathcal{F}$ if this is possible. That is,*

$$\begin{aligned} Move^+ :\ & \mathbb{N}^+ \to \mathbb{N}^+ \\ & \phi^{q_p} \mapsto Move^+(\phi^{q_p}) = \phi^{q_p+1}(mod\ |\mathcal{F}|). \end{aligned}$$

- *$Move^-$: The selected frequency value is changed by the previous value in $\mathcal{F}$ if this* *is possible.*

$$\begin{aligned} Move^- :\ & \mathbb{N}^+ \rightarrow \mathbb{N}^+ \\ & \phi^{q_p} \mapsto Move^-(\phi^{q_p}) = \phi^{q_p-1}(mod\ |\mathcal{F}|). \end{aligned}$$

Note that if $q_p = |\mathcal{F}|$, $\phi^{q_p+1} = \phi^1$ and if $q_p = 1$, $\phi^{q_p-1} = \phi^{|\mathcal{F}|}$. In our problem, we define the neighborhood structure by means of these movements as follows.

Definition 3.2 *Neighborhood structure*

Given a frequency configuration $C_{\mathcal{F}} = (\phi^{q_1}, \ldots, \phi^{q_{|\mathcal{L}|}})$*, where each* $\phi^{q_p} \in \mathcal{F}$ *is the assigned frequency to line* ℓ_p*, another frequency configuration is a neighbor if the latter can be obtained by applying either* $Move^+$ *or* $Move^-$ *to only one of the frequencies* ϕ^{q_p}*. For the neighborhood structure, we assume that if* $q_p = 1$ *for a line* ℓ_p*, we can only apply* $Move^+$ *on* ϕ^{q_p} *and if* $q_p = |\mathcal{F}|$*, we can only apply* $Move^-$ *on* ϕ^{q_p}*. We denote this neighborhood as* $\mathcal{N}(C_{\mathcal{F}})$*.*

Our method consists of four phases.

- Phase 1 (Initial solution)

  In a first phase, we find the optimal configuration of line frequencies (the one that maximizes the net profit) in which all lines have the same frequency, and we keep such solution.

- Phase 2 (Neighborhood search)

  In the second phase, the neighborhood of the solution obtained in the first one is explored. We find the neighbor with maximum profit. We compare this profit with the obtained for the initial solution. We keep the best solution as the current solution.

- Phase 3 (Movement search)

  In the third phase we consider a local search different from the one in phase 2. For each line of current solution, we increase its frequency using the operation defined as $Move^+$, while we improve the solution (no need to analyze frequency configurations in which all frequencies are the same, as these were computed in phase 1). In case of not improving the solution after applying the first $Move^+$, we decrease its frequency using the operation defined as $Move^-$, while we improve the solution. Finally, this solution is stored and it is not used at following iterations. We repeat the same process for all lines starting with the solution obtained in phase 2, and we keep the best solution.

- Phase 4

  The fourth phase is defined as follows. From the stored solution in phase 3, we first try to improve the objective function by applying the movements described in the third phase ($Move^+$ or $Move^-$) only to the first line. We keep the best solution and, over such a solution, apply the same to the second line. And so on. We keep the best solution found and the algorithm stops.

We remark that third and fourth phases are different in the following sense. In the third phase, we compute the best solution at each iteration applying $Move^+$ or $Move^-$. Then this solution is stored and it is not used at the following iterations. However, in the fourth phase, we apply $Move^+$ or $Move^-$ at each iteration, starting with the best stored solution from previous iterations in this phase. The following algorithms show the pseudocode for the heuristic.

Data: A line network $((N, E), \mathcal{L})$.
Phase 1: Initial solution construction.
$z = \{\}$.
for *each possible value of frequency* $\phi^q \in \mathcal{F}$ **do**
    Set all frequencies to ϕ^q;
    Compute z_{NET} according to Loop I;
    $z = z \cup \{z_{NET}\}$
end
return $\arg\max z$, the optimal frequency configuration $C_{\mathcal{F}}^{ini} = (\phi^{q^*}, \phi^{q^*}, \dots, \phi^{q^*})$ and carriages $(\delta^*_{\ell_1}, \dots, \delta^*_{\ell_{|\mathcal{L}|}})$.
Phase 2: The neighborhood $\mathcal{N}(C_{\mathcal{F}}^{ini})$ of $C_{\mathcal{F}}^{ini} = (\phi^{q^*}, \phi^{q^*}, \dots, \phi^{q^*})$ is generated. For each neighbor, Loop I is applied and the number of carriages for each line and z_{NET} is computed. The profit of all neighbors and the initial solution are compared. Let $C_{\mathcal{F}}^{neig}$ be the frequency configuration with maximum z_{NET}.
Phase 3: From $C_{\mathcal{F}}^{neig}$, a local search by means of movements is considered (see Algorithm 6).
Phase 4: From solution phase 3, a different local search procedure by means of movements is considered (see Algorithm 7).
Result: The frequency configuration $(\psi^*_{\ell_1}, \psi^*_{\ell_2}, \dots, \psi^*_{\ell_{|\mathcal{L}|}})$ and capacity configuration $(\delta^*_{\ell_1}, \delta^*_{\ell_2}, \dots, \delta^*_{\ell_{|\mathcal{L}|}})$ with maximum profit.

Algorithm 3: HLSA heuristic.

Data: The frequency $\phi^{q_p} \in \mathcal{F}$ assigned to line ℓ_p on which we will apply the movements and, the current frequency configuration in the way: $C^{cur} = (\phi^{q_1}, \dots, \phi^{q_{|\mathcal{L}|}})$.

Loop II;
$z = z_{NET}(\phi^{q_1}, \dots, \phi^{q_{|\mathcal{L}|}})$;
$\hat{\phi}^{q_p} = Move^{+}(\phi^{q_p})$;
$\hat{z} = z_{NET}(\phi^{q_1}, \dots, \hat{\phi}^{q_p}, \dots, \phi^{q_{|\mathcal{L}|}})$;
if $z < \hat{z}$ **then**
    while $z \leq \hat{z}$ **do**
        $z = \hat{z}$;
        $\hat{\phi}^{q_p} = Move^{+}(\hat{\phi}^{q_p})$;
        $\hat{z} = z_{NET}(\phi^{q_1}, \dots, \hat{\phi}^{q_p}, \dots, \phi^{q_{|\mathcal{L}|}})$;
    end
else
    $\hat{\phi}^{q_p} = Move^{-}(\phi^{q_p})$;
    $\hat{z} = z_{NET}(\phi^{q_1}, \dots, \hat{\phi}^{q_p}, \dots, \phi^{q_{|\mathcal{L}|}})$;
    while $z \leq \hat{z}$ **do**
        $z = \hat{z}$;
        $\hat{\phi}^{q_p} = Move^{-}(\hat{\phi}^{q_p})$;
        $\hat{z} = z_{NET}(\phi^{q_1}, \dots, \hat{\phi}^{q_p}, \dots, \phi^{q_{|\mathcal{L}|}})$;
    end
end
$z_p = \hat{z}$;
return the best profit z_p and the best frequency for ℓ_p denoted by $\hat{\phi}^{q_p}$.

Result: The frequency for line ℓ_p with maximum profit as well as the configuration of carriages and such profit. Note that, now the configuration is $C_\phi = (\phi^{q_1}, \dots, \hat{\phi}^{q_p}, \dots, \phi^{q_{|\mathcal{L}|}})$

Algorithm 4: Pseudocode for Loop II which is used in phase 3 and 4.

Data: A frequency configuration in the way $C_{\mathcal{F}}^{\phi^q} = (\phi^q, \phi^q, \ldots, \phi^q)$.
Loop I;
for *each line* ℓ **do**
  Find the arc e_ℓ of ℓ with maximum load;
  Find the minimum number of carriages δ_ℓ needed to transport all passengers traversing arc e_ℓ;
  Keep δ_ℓ.
end
Compute the profit z_{NET}.
Result: z_{NET} associated to the frequency configuration $C_{\mathcal{F}}^{\phi^q} = (\phi^q, \phi^q, \ldots, \phi^q)$ and carriages $(\delta_{\ell_1}, \ldots, \delta_{\ell_{|\mathcal{L}|}})$

Algorithm 5: Pseudocode for Loop I which is used in phase 1 and 2.

Data: A configuration of frequencies $(\phi^{q_1}, \ldots, \phi^{q_{|\mathcal{L}|}})$.
Set $z = \{\}$;
for *each line* ℓ_p **do**
  Obtain the best profit z_p applying $Move^+$ or $Move^-$ on the frequency ϕ^{q_p} associated to ℓ_p and according to $(\phi^{q_1}, \ldots, \phi^{q_{|\mathcal{L}|}})$, by means of Loop II;
  $z = z \cup \{z_p\}$.
end
return $\arg\max z$.
Result: The frequency and carriages configuration of maximum profit.

Algorithm 6: Pseudocode for the phase 3.

Data: A configuration of frequencies $C_{\mathcal{F}}^{cur} = (\phi^{q_1}, \ldots, \phi^{q_{|\mathcal{L}|}})$.
Set $z = 0$;
for *each line* ℓ_p **do**
  Obtain the best profit z_p applying $Move^+$ or $Move^-$ on the frequency ϕ^{q_p} associated to ℓ_p and according to $C_{\mathcal{F}}^{cur}$, by means of Loop II.
  Do $\phi^{q_p} = \hat{\phi}^{q_p}$ and $C_{\mathcal{F}}^{cur} = (\phi^{q_1}, \ldots, \hat{\phi}^{q_p}, \ldots, \phi^{q_{|\mathcal{L}|}})$;
  $z = z_p$.
end
return z.
Result: The frequency and carriages configuration of maximum profit.

Algorithm 7: Pseudocode for the phase 4.

Example 3.3.1 *Consider two lines ℓ_1, ℓ_2 and five possible frequencies for each line, namely, $\mathcal{F} = \{1, 2, 3, 4, 5\}$.*

- *Phase 1: calculate the net profit of the following frequencies: [1,1], [2,2], [3,3], [4,4], [5,5]. Assume that the best profit is given by configuration [3,3].*

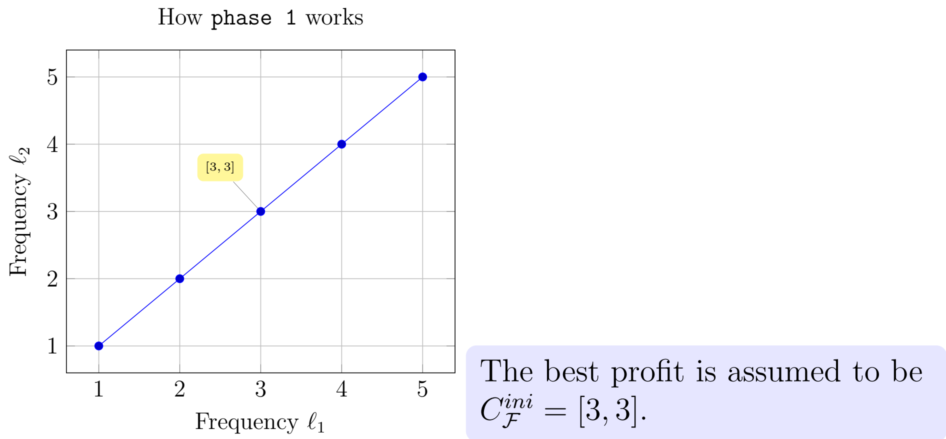


Figure 3.2.: Phase 1. Initialization.

- *Phase 2: in this phase, the neighborhood of [3,3] is $\mathcal{N}(C_{\mathcal{F}}^{ini}) = \{[4, 3], [2, 3], [3, 4], [3, 2]\}$. Assume that configuration [3,4] yields the highest profit, so we keep this solution.*

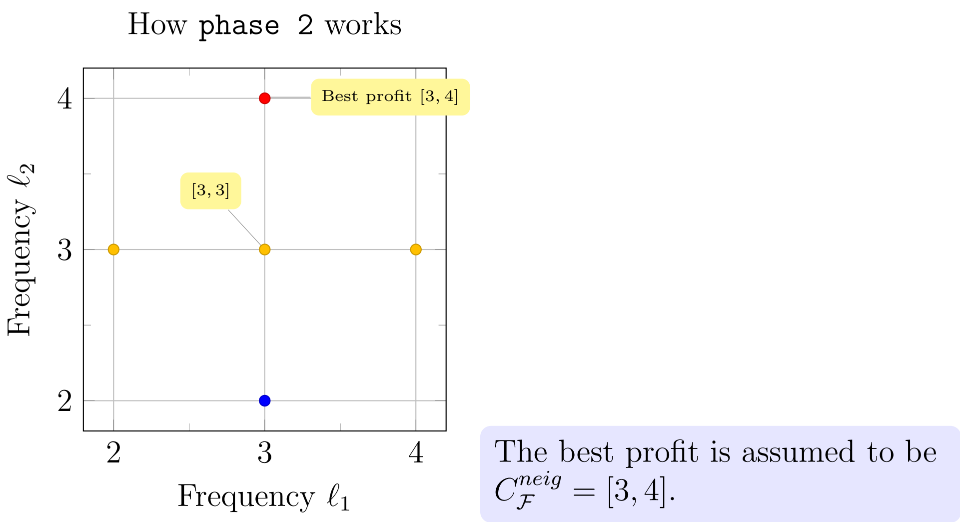


Figure 3.3.: Phase 2. Neighborhood.

- *Phase 3: from solution [3,4], we start by increasing the frequency of the first line (note that [4,4] is not here analyzed). So we compute the next, [5,4], which let us* *assume yields a better profit. We then analyze [1,4] (as frequency 6 is not feasible), then [2,4],.. until we stop improving the profit. If at the first iteration of* $Move^+$ *([5,4]), a better profit is not obtained, we decrease the frequency of the first line in the same way as* $Move^+$*. We do the same for the second line. Assume that the best solution in this phase is [3,1].*

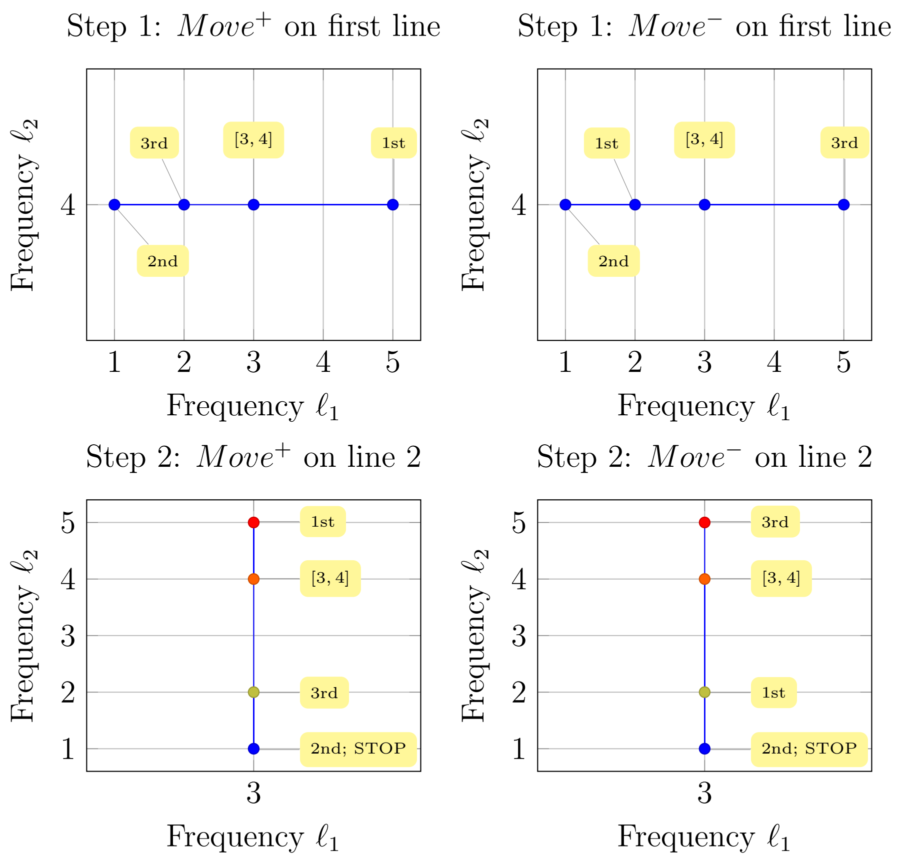


Figure 3.4.: Phase 3. A local search by movements.

- *Phase 4: We now apply* $Move^+$ *or* $Move^-$ *on the frequency of each line in solution [3,1] as phase 3. Assume that the best solution found for first line is [4,1]. We now apply* $Move^+$ *or* $Move^-$ *to the second line in solution [4,1] while we improve the profit. The solution obtained after these steps is the final solution of the algorithm.*

3.4. Capacitated problem

As mentioned, in this section we will define the Capacitated Rapid Transit System Simultaneous Frequency and Capacity Setting Problem (CRTSSFCP) under assumption

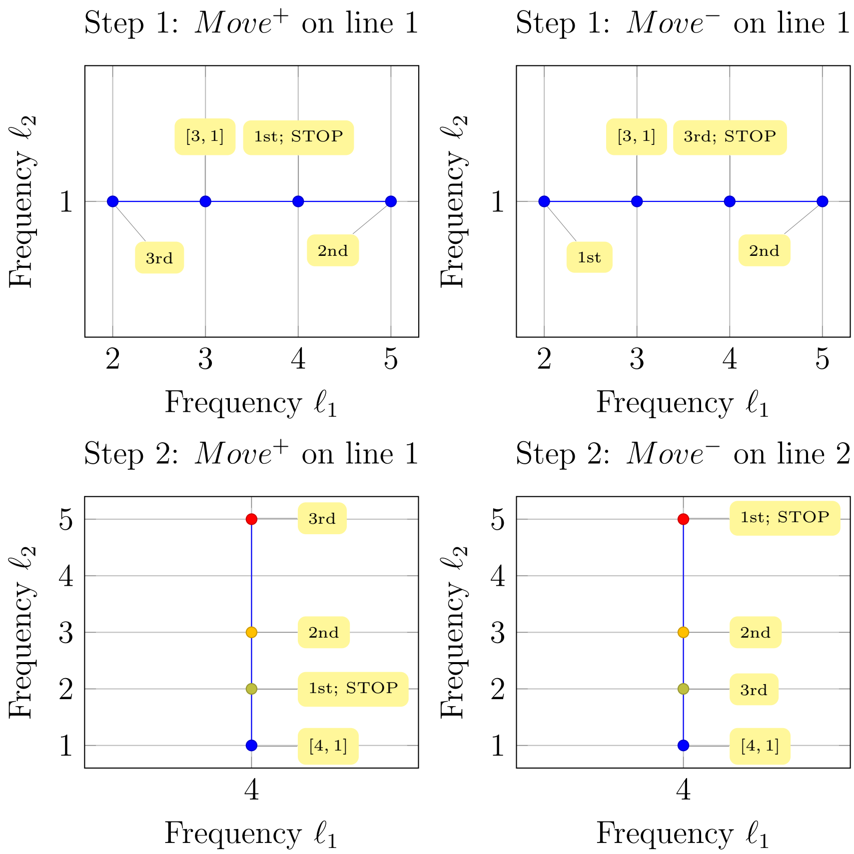


Figure 3.5.: Phase 4: a local search by movements.

of a limited number of carriages. An interesting aspect to take into account in this problem are the crowding levels as a consequence of taking a limited capacity. In overcrowding situations, many passengers choose an alternative path or a different transportation mode. So, congestion not only causes an increase in the traveler's disutility, but also a benefit loss to operators. For the sake of clarify, we introduce the following terms. Let κ_{ij}^{ℓ} be the number of passengers traversing arc (i, j) of ℓ by hour. The train capacity of a line ℓ is the carriage capacity Θ times the number of carriages associated to one train of ℓ. The carriage capacity have been defined as the nominal capacity or crush capacity (Oldfield and Bly (1988), Jara-Díaz and Gschwender (2003)) which includes both seating and standing. The maximum number of passenger Nb_{ℓ} who can travel on line ℓ by hour can be computed as its frequency times its train capacity. By means of these terms, the load factor ϱ_{ij}^{ℓ} is defined as the ratio $\kappa_{ij}^{\ell}/Nb_{\ell}$. Observe that if $\varrho_{ij}^{\ell} \leq 1$, the arc $(i, j) \in \ell$ is not affected by congestion. Therefore, if the train capacity of a line ℓ is not enough to transport all passengers traveling inside ℓ, the rapid transit network can become a congested network. In recent research, the load factor is introduced to stipulate the crowding levels. There exists four crowding types: in-vehicle crowding, platform crowding, excessive waiting time and increased dwell time. We will analyze the in-vehicle crowding effects, which can be defined by means of crowding penalties. There are three possible ways to describe this term: time multiplier, the monetary value per time unit and the monetary value per trip. We will consider the time multiplier in our problem. Due to each transport mode is different, it is not possible to define a general crowding function valid for all transport modes, as stated Feifei and Haicheng (2012), Feifei (2014). De Palma et al. (2010) propose an exponential function for the crowding penalty in the context of railway system using a load factor. This crowding function is expressed as

$$CF(\varrho_{ij}^{\ell}) = 1 + \frac{\varsigma_1}{1 + \exp(\varsigma_2(1 - \varrho_{ij}^{\ell}))} + \varsigma_3 \exp(\varsigma_4(\varrho_{ij}^{\ell} - \varsigma_5)), \qquad (3.20)$$

in which the parameters are positive and ς_1 and ς_3 should be carefully calibrated (see Feifei and Haicheng (2012)). The parameter $\varsigma_5 > 1$ represents the threshold from which the passengers starts to perceive overcrowding. It can be observed that this function reflects the inconvenience associated with in-vehicle crowding. The important point to note here is the form of this function. If the load factor $\varrho_{ij}^{\ell} \leq 1$, $CF(\varrho_{ij}^{\ell})$ becomes approximately one, the second term in the Equation (3.20) is close to zero for a proper value of parameter ς_2 and the last term $\varsigma_3 \exp(\varsigma_4(\varrho_{ij}^{\ell} - \varsigma_5))$ is approximately zero (recall $\varrho_{ij}^{\ell} < \varsigma_5$). Analogously, when the load factor $1 \leq \varrho_{ij}^{\ell} \leq \varsigma_5$, in-vehicle crowding starts to working on the traveling time of arc (i,j). The penalty impact will be very depending on the ς_2 parameter.

Our next goal is to introduce congestion in the model presented in Section 3.3 by means of the crowding function defined previously. To this end, we introduce the following variables and parameters. Let ϱ_{ij}^{ℓ} be the load factor on the arc $(i, j) \in \ell$ defined above. Due to the fact that we are only including in-vehicle crowding effects, solutions whose load factor is greater than the value of the parameter σ are not allowed. Observe that if $\varrho_{ij}^{\ell} > \sigma$, penalties according to the excess waiting time, platform crowding and increased dwell time have to be included in the model. In order to introduce the in-vehicle congestion effects on the model, we include constraint $\varrho_{ij}^{\ell} \leq \sigma$ in the mathematical programming program defined in Section 3.3.1.

The main difference between this model and that defined in Section 3.3 lies in the introduction of an upper bound $\delta^{\max}$ on the number of possible carriages. As a consequence, lines in specific sections of the RTS can be congested. We consider $\bar{d}_{ij}^{\ell} = CF(\varrho_{ij}^{\ell}) \cdot d_{ij}$ as the perceived time to traverse arc (i,j) of ℓ using the rapid transit system. As commented, if the arc $(i,j) \in \ell$ is not congested, $\bar{d}_{ij}^{\ell} \simeq d_{ij}$. So, variable u_w^{RTS} representing the average travel time must be redefined. The average travel time associated to OD pair w using the rapid transit network under crowding can be explicitly defined as follows:

$$\begin{aligned} u_w^{RTS} \quad &= \sum_{\ell \in \mathcal{L}} \sum_{j:\{w_s,j\} \in \ell} \frac{60 \tilde{f}_{w_s j}^{w\ell}}{2\psi_\ell} + (60/\lambda) \sum_{\ell \in \mathcal{L}} \Big(\sum_{\{i,j\} \in \ell} \tilde{f}_{ij}^{w\ell} \bar{d}_{ij}^{\ell} \Big) \\ &+ \sum_{\ell \in \mathcal{L}} \sum_{\ell' : \ell' \neq \ell} \sum_{i \in \ell \cap \ell'} \tilde{f}_i^{w\ell\ell'} \Big(\frac{60}{2\psi_{\ell'}} + uc_i \Big), \; w = (w_s, w_t) \in W. \end{aligned} \tag{3.21}$$

The first term in (3.21) is the waiting time at the origin station, which is also assumed to be half of time between services of this line. The second term is the in-vehicle time which can be affected by congestion. Finally, the last one constitutes the required time by the transfers. The rest of variables, parameters, constraints as well as objective function are equals to those defined in Section 3.3.

This model has several limitations. The first one, is the time needed to find good solutions. The second and most important, is the problem complexity. Note that if the congestion function is activated, data related to distance change. So, the instance become to be different when the penalty is applied on the distances. Concretely, the congestion effect influences on the travel time of each path, and, therefore, on the number of passengers in the RTS. The passengers' behavior is different in congestion presence and, as a consequence, it is different for each instance modification. It can be observed that the penalization process stops when the network is not congested or a fixed point is searched. In other words, passengers take a different path or mode and an equilibrium is searched (all passengers can be transported). The solution reflects not only the number of carriages and frequencies, but also a medium-term analysis of the passenger's behavior under congestion.

The excess waiting time effects can be incorporated on our problem as follows. The passengers affected by this aspect are who waiting for next train to the fact that the first train was full and they were left behind, increasing waiting time and discomfort to travel. In the context of bus transport, Oldfield and Bly (1988) presented a formal definition of this type of crowding. They expressed the waiting time according to headway and crowding level. However, the inclusion of excess waiting time effects in our model is not trivial. For the purpose, the travel time of all passengers waiting for next train is increased according to an additional time which depends on the frequency of the congested line. The rerouting passengers process is very difficult. This is due to the fact that the passengers affected by the excessive waiting time have different travel time from the rest of passengers and, as a consequence, a different instance associated. So, the initial instance is split into two different instances: one associated to excessive waiting time and the other one, related to in-vehicle crowding. Analogously, the origin-destination matrix is divided into two matrixes: one containing the passenger associated to the in-vehicle crowding and other one, according to the excessive waiting time. The crowding phenomenon is also defined as the congestion effect at train stations; the access/egress to/from the station, on platforms (see Douglas and Karpouzis (2005)) and on the increased dwell times as Lin and Wilson (1992).

The following section is devoted to introduce two algorithms to solve our problem.

3.4.1. Two algorithms for our problem

In this section we introduce two different algorithms that solve our problem: one with the nominal capacity and other one with number of seats. These algorithms check each possible frequency and each number of carriages per train of each line. The idea is to iteratively test all possible combinations of frequencies and carriages. Once the frequencies and carriages have been set, the shortest path on the rapid transit network for each OD pair can easily be computed by a modified Dijkstra algorithm. From these shortest paths the number of passengers traveling on each line and arc is calculated and the capacity constraint is tested on the arc with maximum load. If there is a congested arc, the penalization process is activated. Depending on the considered definition of carriage capacity, the congestion is perceived before or later. The perceived travel time to traverse each congested arc is defined as the travel time to traverse each arc times its corresponding penalty. When the penalization process is finished, the rerouting passenger process is activated. For the purpose, the shortest path taking into account transfer and waiting times on the RTS for each OD pair is recalculated and the capacity constraint is rechecked and so on. Due to the fact that travel time increase, the number of passengers on congested arc is smaller than in the previous iteration. Some passengers will take an alternative path or an alternative transport mode. This procedure breaks when the congestion ends or when a fixed point is reached. Algorithm 10 shows the pseudocode to solve the CRTSSCFP with nominal capacity and Algorithm 13 is the pseudocode to solve the CRTSSCFP with the number of seats at each carriage. For the congestion with the seat capacity, the in-vehicle crowding is activated when the load factor reaches 140% or standing density is over four passengers per square meter (see Feifei and Haicheng (2012), Feifei (2014)).

Data: A combination of frequencies and carriages
Let $n_{iter} = 0$ be the number of iterations;
Loop III(a): Check the capacity constraint
for *each line* ℓ **do**
  Find the arc $(i, j) \in \ell$ with maximum load ϱ^{ℓ}_{ij};
  if $1 < \varrho^{\ell}_{ij} \leq \sigma$ **then**
    penalize the traverse time of each arc by means of CF-function;
    $n_{iter} = n_{iter} + 1$;
    go Loop IV(a);
  end
end
Result: A rapid transit system.

Algorithm 8: Checking the capacity constraint with nominal capacity.

Data: A line network $(S, \mathcal{L})$
Loop IV(a): Check fixed point
if n_{iter} *is equal to one* **then**
  We define $(S^{pre}, \mathcal{L}^{pre})$ as $(S, \mathcal{L})$;
  go Loop III(a);
else
  if *the network* $(S, \mathcal{L})$ *is the same than* $(S^{pre}, \mathcal{L}^{pre})$ **then**
    break;
  else
    We define $(S^{pre}, \mathcal{L}^{pre})$ as $(S, \mathcal{L})$;
    go Loop III(a);
  end
end
Result: A rapid transit system.

Algorithm 9: Testing the fixed point with nominal capacity.

Data: A line network $(S, \mathcal{L})$, a set of possible frequencies and a minimum and maximum capacity
Set $z = \{\}$;
for *each possible combination of frequencies and carriages* **do**
  go Loop III(a);
  Compute the profit z_{NET};
  $z = z \cup \{z_{NET}\}$
end
return $\arg\max\{z\}$;
Result: The frequency and capacity configuration with the maximum profit.

Algorithm 10: The algorithm for the rapid transit system frequency and capacity setting problem under congestion with nominal capacity.

Data: A combination of frequencies and carriages
Let $n_{iter} = 0$ be the number of iterations;
Loop III(b): Check the capacity constraint
for *each line* ℓ **do**
  Find the arc $(i, j) \in \ell$ with maximum load ϱ_{ij}^{ℓ};
  Let $\hat{\varrho}_{ij}^{\ell}$ be the load with the nominal capacity;
  if $\varrho_{ij}^{\ell} > 1.4$ *and* $\hat{\varrho}_{ij}^{\ell} \leq \sigma$ **then**
    penalize the traverse time of each arc by means of CF-function;
    $n_{iter} = n_{iter} + 1$;
    go Loop IV(b);
  end
end
Result: A rapid transit system.

Algorithm 11: Checking the capacity constraint with seat capacity.

Data: A line network $(S, \mathcal{L})$
Loop IV(b): Check fixed point
if n_{iter} *is equal to one* **then**
  We define $(S^{pre}, \mathcal{L}^{pre})$ as $(S, \mathcal{L})$;
  go Loop III(b);
else
  if *the network* $(S, \mathcal{L})$ *is the same than* $(S^{pre}, \mathcal{L}^{pre})$ **then**
    break;
  else
    We define $(S^{pre}, \mathcal{L}^{pre})$ as $(S, \mathcal{L})$;
    go Loop III(b);
  end
end
Result: A rapid transit system.

Algorithm 12: Testing the fixed point with seat capacity.

Data: A line network $(S, \mathcal{L})$, a set of possible frequencies and a minimum and maximum capacity
Set $z = \{\}$;
for *each possible combination of frequencies and carriages* **do**
  go Loop III(b);
  Compute the profit z_{NET};
  $z = z \cup \{z_{NET}\}$
end
return $\arg\max\{z\}$;
Result: The frequency and capacity configuration with the maximum profit.

Algorithm 13: An algorithm for the rapid transit system frequency and capacity setting problem under congestion with seat capacity.

3.5. Computational experiments

All the calculations in this section were performed with a Java code in a computer with 8 Gb of RAM memory and 2.8 Ghz CPU. For purpose of evaluating the performance of our algorithms, we have used several instances of networks (see 3.5.1). Specifically, we have selected instances with 2,3,5 and 6 lines.

There are no previously reported solutions for the proposed problem as far as we are aware. We have compared our HLSA heuristic algorithm against the optimal solution obtained with the exact algorithm described in Section 3.4. The comparison of these results are presented in Tables C.2, C.4, C.6, C.8 and C.10. In Tables C.1, C.3, C.5, C.7 and C.9 we have reported the optimal solutions obtained by the algorithm, the solution values obtained by the heuristic procedure as well as some relevant characteristic at each case.

Out of the 170 instances tested in the uncapacitated case, our algorithm was able to provide optimal solutions for most instances: it found the optimal solution in 155 instances (91.17%). From this analysis, it can be seen that our heuristic was so able to provide good results in a very small CPU time. For instance, for networks with 5 lines our heuristic algorithm took on average 5.57 seconds compared to 5077.36 of the exact algorithm.

Finally, we also wanted to analyze the capacitated problem defined in section 3.4. We have performed tests to asses the impact of the congestion on the networks 6×2, 7×3 and 8×3. To this end, we have gradually increased the number of carriages and we have solved the problem with our algorithm (see Algorithm 10). The results of these experiments are presented in Tables C.11, C.12, C.15 and C.16. It can be seen that when the maximum number of carriages is small, the solution has high frequencies in order to transport all passengers. This is due to the problem definition: we have imposed that all passengers willing to travel in the RTS have to be transported.

A key factor to solve this problem was the introduction of the congested function defined in Section 3.4, which is based on in-vehicle crowding. A total of 200 experiments were carried out in our analysis.

The description of our computational experiments is as follows. First, we introduce all parameters needed to carry out the experiments as well as the considered networks. Secondly, we will comment conclusions in Section 3.5.2 from the computational experiments performed in Appendix C.

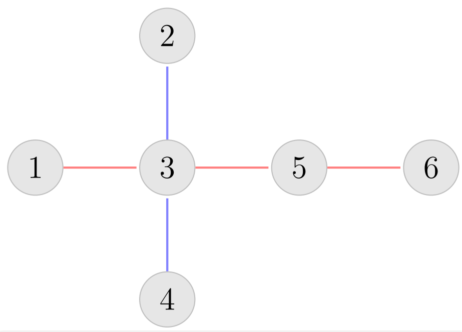
3.5.1. Parameter setting

In Table 4.4 we report the parameters that govern our algorithms. The data reported in this table are based on the specific train model *Civia*, usually used for regional railway passengers transportation in Spain by the National Spanish Railways Service Operator (RENFE). One important characteristic of *Civia* trains is that the number of carriages can be adapted to the demand. Each *Civia* train constains two electric automotives (one at each end) and a variable number of passenger carriages. Each automotive or carriage has a maximum capacity of 200 passengers. In our experimentation, we will assume that the train is composed by only one electric locomotive (for traction purposes and null capacity) and several passengers carriages (which cannot move without a locomotive) as in Cordeau et al. (2000) and Alfieri et al. (2006). The purchase price of rolling stock used in this experimentation is also based on the real data of *Civia* trains. The price of ticket and subvention considered in our experimentation, have been taken from the newspaper (http://www.20minutos.es/noticia/2028399/0/madrid/empresas-privadas/metro-ligero/).

Parameters		
Name	**Description**	**Value**
$\hat{\rho}$	years to recover the purchase	20
ρ	number of operative hours per year	6935
c_{loc}	costs for operating one locomotive per kilometer [€/km]	34
c_{carr}	operating cost of a carriage per kilometer [€/km]	2
c_{crew}	per crew and year for each train [€/ year]	$75 \cdot 10^3$
I_{loc}	purchase cost of one locomotive in €	$2.5 \cdot 10^6$
I_{carr}	purchase cost of one carriage in €	$0.9 \cdot 10^6$
Θ	capacity of each carriage (number of passengers)	$2 \cdot 10^2$
λ	average commercial speed in [km /h]	30
γ	maximum number of lines traversing an edge	4
ψ^{min}	minimum frequency of each line	3
ψ^{max}	maximum frequency of each line	20
ψ_ℓ	possible values	{3,4,5,6,10,12,15,20}

Table 3.1.: Model parameters for RNFCSP.

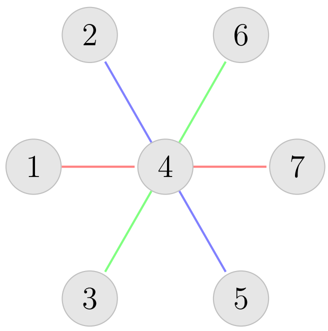
In the experiments we have considered five networks. The first one is defined by six nodes, five edges and two lines as follows



The lines are defined as:
red line $\ell_1 = \{1, 3, 5, 6\}$ and
blue line $\ell_2 = \{2, 3, 4\}$.

Figure 3.6.: Representation of 6×2-configuration.

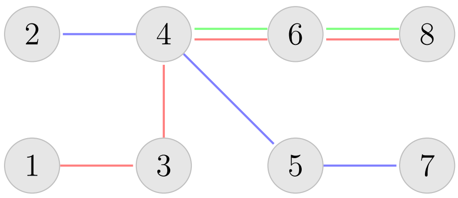
The second one is a star network with six nodes and three lines.



The lines are defined as: blue line $\ell_1 = \{2, 4, 5\}$, red line $\ell_2 = \{1, 4, 7\}$ and green line $\ell_3 = \{3, 4, 6\}$.

Figure 3.7.: Representation of 7×3-configuration.

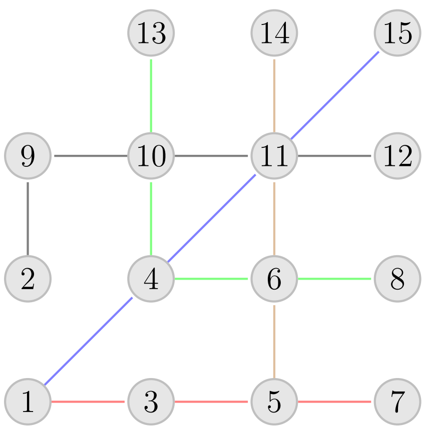
The following network is defined by eight nodes, seven edges and three lines.



The lines are defined as: red line $\ell_1 = \{1, 3, 4, 6, 8\}$, blue line $\ell_2 = \{2, 4, 5, 7\}$ and green line $\ell_3 = \{4, 6, 8\}$.

Figure 3.8.: Representation of 8×3-configuration.

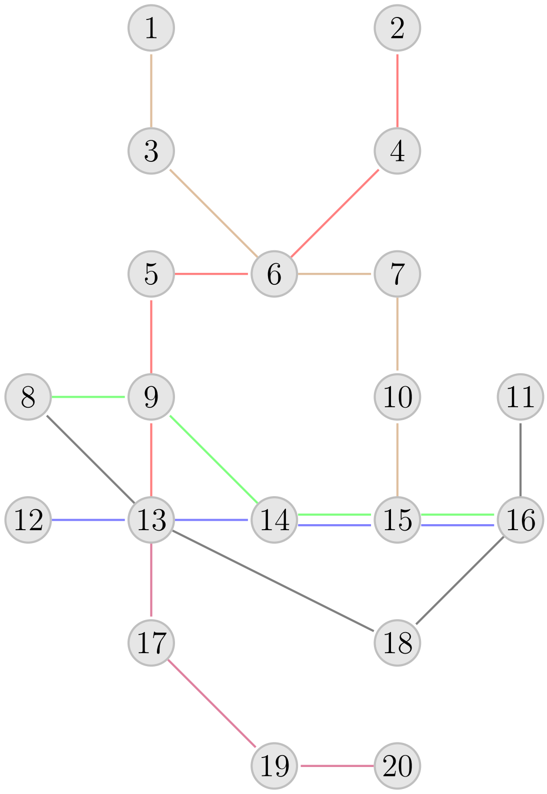
The following network is a grid configuration formed by fifteen nodes, seventeen edges and fives lines.



The lines are defined as: red line $\ell_1 = \{1, 3, 5, 7\}$, blue line $\ell_2 = \{1, 4, 11, 15\}$, green line $\ell_3 = \{13, 10, 4, 6, 8\}$, gray line $\ell_4 = \{2, 9, 10, 11, 12\}$ and brown line $\ell_5 = \{5, 6, 11, 14\}$.

Figure 3.9.: Representation of 15×5-configuration.

Next configuration is defined by twenty nodes, twenty three edges and six lines.



The lines are defined as:
red line $\ell_1 = \{2, 4, 6, 5, 9, 13\}$,
brown $\ell_2 = \{1, 3, 6, 7, 10, 15\}$,
blue line $\ell_3 = \{12, 13, 14, 15, 16\}$,
purple line $\ell_4 = \{13, 17, 19, 20\}$,
gray line $\ell_5 = \{8, 13, 18, 16, 11\}$
and green line
$\ell_6 = \{8, 9, 14, 15, 16\}$.

Figure 3.10.: Representation of 20×6-configuration.

For each configuration, we have randomly generated 10 different instances for the OD-matrix and length data. To this end, the number of passengers of each OD pair w, was obtained according to the product of two parameters. The first one was randomly set in the interval [5,15] by using a uniform distribution, whereas the other one was set in a different interval for each configuration. Concretely, for the 6×2-network, the interval considered was set as [65,77], generating around 20.000 passengers at each instance of such configuration. For 7×3 and 8×3-networks, the number of passengers was approximately 30.000 passengers at each case and the parameters were defined in the intervals [68, 80] and [51, 59], respectively. The parameter for the 20×6-configuration was set to 16 for all instances and [23, 25] for the 15×5-configuration. The following Table 3.2 reportes the number of passengers considered for each instance.

	Instance									
Configuration	seed1	seed2	seed3	seed4	seed5	seed6	seed7	seed8	seed9	seed10
example6 × 2	20236	20523	21862	20898	20073	20554	21275	20224	20898	20802
example7 × 3	30294	30505	30071	30843	30886	28095	30356	30927	30000	30366
example8 × 3	30360	30041	30555	30779	30151	30497	30298	30465	30995	30260
example15 × 5	47922	50454	51121	51400	50021	51157	50112	50370	50450	50973
example20 × 6	61359	60384	62124	61319	60188	60956	60457	61561	61233	61680

Table 3.2.: Number of possible trips at each instance.

To define each arc length, the coordinates of each station were set randomly by means of an uniform distribution. So, the arc length at each instance is different since each arc connects to different stations.

For the experiments, the travel times u_w^{alt} by the alternative mode, were obtained by means of the Euclidean distance and a speed of 20 km/h, whereas, the travel times in the RTS were obtained according to in-vehicle travel time, waiting and transfer times. The waiting time was supposed to be half of the corresponding time between services of lines at the origin station, whereas, the transfer time was assumed to be half time between two consecutive services at the line to transfer. We assume two possible values for the σ parameter: 1.1 and 1.2. So, for $\sigma = 1.1$, if the number of passenger traveling inside each line is higher than capacity of line over 10%, the solution is taken into account.

3.5.2. Computational comments

To conclude this section, we will show some conclusions on the results in the computational experiments reported in Appendix C. First, we have performed experiments for the uncapacitated problem and secondly for the capacitated version. In the uncapacitated problem we have compared the heuristic algorithm defined in Section 3.3.2 against the exact algorithm. In Table 3.3 we report the gaps, the optimality ratio and the ratio time between exact and heuristic time.

	gap %	ratio opt	ratio time
6×2	0.00179079	97.5	3.30321746
7×3	0.93977468	80	20.8142911
8×3	0.04214467	95	20.8872632
15×5	0.08713446	90	913.972632
20×6	0	100	6712.15308

Table 3.3.: Results for the uncapacitated problem.

For 6×2-configuration the average of all CPU times at the heuristic was 0.063375 seconds while if we consider the exact, its average was 0.19335 seconds. The data reported in Table C.2 showed that at the 97.5% the heuristic found the optimal solution and the average improve of the CPU time was 65.6%. At the 7×3-configuration the average CPU time for the heuristic was 0.16995 and for the exact was 3.48045 seconds. In Table C.4 it can be seen that at the 80% the heuristic found the optimal solution and the average improve of the CPU time was 95.11%. From the obtained results for this configuration, it can be observed that the optimal solution for each instance was reached for small frequencies and high capacities. The average CPU time in the heuristic for 8×3-configuration was 0.2836 and the exact was 5.89 seconds, which represents an improvement of 95.18% in the time. For the 15×5-configuration it can be seen that the heuristic improved the average CPU time in 99.9%. Due to the spent time in the exact algorithm, we have only solved ten instances for the 20×6-configuration. At these cases our heuristic was able to find the optimal solution for all instances (100%) and the average improve of the CPU time was $99,98\%$.

In a second stream of experiments, we have studied the impact of congestion on the 6×2, 7×3 and 8×3-configurations. To this end, we have gradually increased the maximum number of carriages and we have fixed σ to 1.1 in our experimentation. Detailed information on these solutions are shown in Tables C.11, C.12, C.15 and C.16. From these results it can be seen that the number of trains decreases according as the maximum number of carriages increases. It can be observed that for the most instances the solutions do not correspond to congested networks. This fact indicates that in-vehicle crowding has a significant effect on the solutions. The solutions obtained at the uncapacitated case can be analyzed with the congestion effect. It is important to note that the passengers' behavior changes when the congestion is introduced in the problem and that it is more economically interesting to add carriages than to lose passengers.

3.6. Conclusions

We have described a problem in the line planning context, which consists of selecting, for each line, the number of services per hour and the number of train carriages in presence of a competing transportation mode. In this problem, all passengers preferring to use the RTS have a service and a certain net benefit is maximized. This problem requires to incorporate a long term public economic support for the operating and acquisition rolling stock. We have classified our problem into two categories: uncapacitated and capacitated problem. The first problem assumes the maximum number of possible carriages is a sufficiently natural number in order to allow all people preferring to travel in the RTS can be transported. The second one can lead to congested networks since the number of possible carriages is a limited value. We have proposed two different algorithms to solve the first problem: an exact and heuristic algorithm. The heuristic technique is a procedure based on a Local Search Algorithm. The modifications are done by means of an appropriated neighborhood structure and movements. The input data in the computational experiments has been based on real data in order to calibrate all parameters that appear in our problem. Moreover, we have randomly generated instances for different types of networks. Comparative tests on a large set of instances have shown that our heuristic in the uncapacitated problem can provide high quality solutions within reasonable computing times. Out of the 170 instances tested in the uncapacitated case, our algorithm was able to provide optimal solutions for most instances: it found the optimal solution in 155 instances (91.17%). On the other hand, the algorithm defined in Section 3.4 have been tested on small networks showing the effect of the congestion on the solutions. The congestion impact have been studied by means of a congestion function which measures the level of in-vehicle crowding. A total of 200 experiments were carried out in our analysis. From the obtained results, it can be observed that the profit is more economically interesting when the network is not a congested network. In other words, the demand is sensitive to congestion and it is more profitable to add carriages or trains than to lose passengers.

This problem can easily be extended to the case of a set of possible lines (a line pool) analyzing iteratively all combinations of lines. For each possible set of lines, the problem is reduced to our problem.

Appendix C

Computational results

In the next section, we will report the results for each network, considering a sufficiently large number of carriages. Section C.0.2 is devoted to the congested case.

C.0.1. Computational experiments for the uncapacitated problem

We assume two possible values for the σ parameter: 1.1 and 1.2. So, for $\sigma = 1.1$, if the number of passenger traveling inside each line is higher than capacity of line over 10%, the solution is taken into account. The δ^{max} parameter was adjusted in the way all passengers willing to travel in the RTS can be transported. In fact, we consider $\delta^{max} = 8$ for the 6×2, 7×3, 8×3 and 15×5 configurations and $\delta^{max} = 10$ for the 20×6-configuration.

A total of 170 experiments were carried out, as showed in Tables C.1, C.3, C.5, C.7 and C.9. In the most cases, the heuristic and the exact procedure lead to the same results. In the following, we will analyze each network separately.

6×2-configuration

It can be observed that the average of all CPU times at the heuristic is 0.063375 seconds while if we consider the exact, its average is 0.19335 seconds. So, the heuristic improves the average CPU time in 67.22% respect to the exact. This indicates that the heuristic procedure is promising and that for instances of large size the heuristic is expected to be much faster than the exact procedure. Note that the maximum net profit is obtained for the Seed1-instance at all scenarios. However, the revenue at this instance is not the highest in the first, second and fourth scenario. This is due to the high costs for the operation and investment trains. The profit and revenue are graphically represented in Figure C.1 and C.2 for the first and fourth scenario from Table C.1.

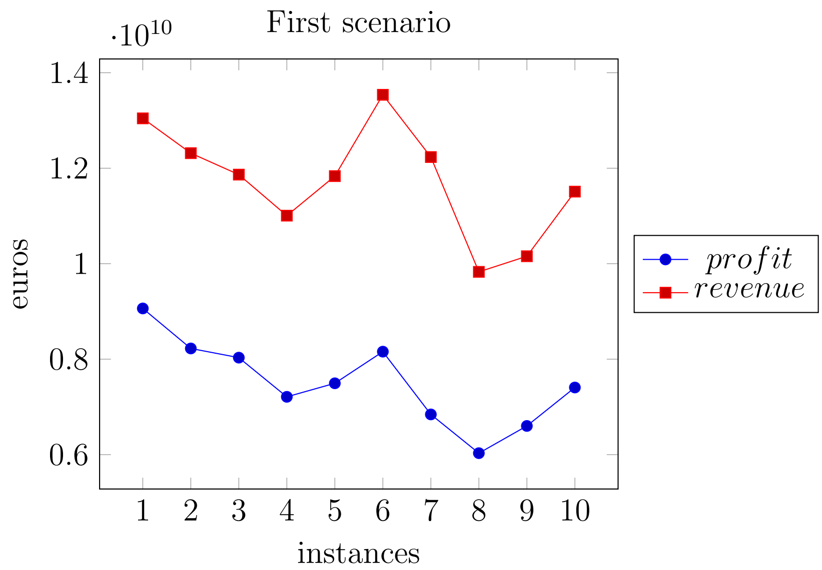


Figure C.1.: Revenue and profit for first scenario of 6 × 2-configuration.

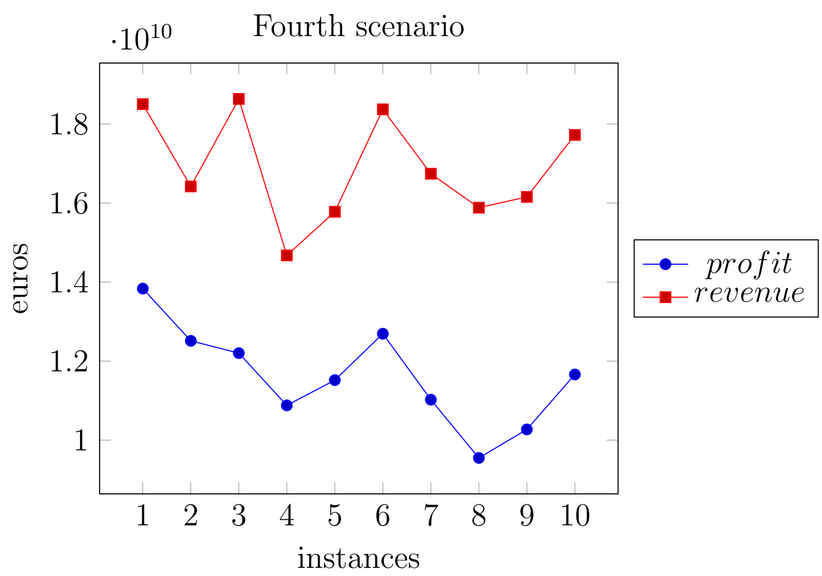


Figure C.2.: Revenue and profit for fourth scenario of 6 × 2-configuration.

				Heuristic							Exact						
Configuration	instance	fare	σ	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains
example6 × 2	seed1	6	1.1	9.06E+09	1.3E+10	[3, 3]	[8,5]	0.094	15675	6	9.06E+09	1.3E+10	[3, 3]	[8,5]	0.109	15675	6
example6 × 2	seed2	6	1.1	8.22E+09	1.23E+10	[3, 3]	[7,5]	0.031	14799	6	8.22E+09	1.23E+10	[3, 3]	[7,5]	0.125	14799	6
example6 × 2	seed3	6	1.1	8.03E+09	1.19E+10	[3, 3]	[5,5]	0.031	14259	6	8.03E+09	1.19E+10	[3, 3]	[5,5]	0.125	14259	6
example6 × 2	seed4	6	1.1	7.21E+09	1.1E+10	[3, 3]	[5,3]	0.031	13226	6	7.21E+09	1.1E+10	[3, 3]	[5,3]	0.125	13226	6
example6 × 2	seed5	6	1.1	7.49E+09	1.18E+10	[4, 4]	[5,4]	0.047	14221	8	7.49E+09	1.18E+10	[4, 4]	[5,4]	0.109	14221	8
example6 × 2	seed6	6	1.1	8.16E+09	1.35E+10	[4, 5]	[5,3]	0.047	16268	9	8.16E+09	1.35E+10	[4, 5]	[5,3]	0.109	16268	9
example6 × 2	seed7	6	1.1	6.84E+09	1.22E+10	[4, 5]	[4,3]	0.031	14701	9	6.84E+09	1.22E+10	[4, 5]	[4,3]	0.125	14701	9
example6 × 2	seed8	6	1.1	6.03E+09	9.83E+09	[3, 3]	[5,3]	0.047	11811	6	6.03E+09	9.83E+09	[3, 3]	[5,3]	0.125	11811	6
example6 × 2	seed9	6	1.1	6.6E+09	1.02E+10	[3, 3]	[5,3]	0.031	12204	6	6.6E+09	1.02E+10	[3, 3]	[5,3]	0.125	12204	6
example6 × 2	seed10	6	1.1	7.41E+09	1.15E+10	[3, 3]	[5,4]	0.031	13831	6	7.41E+09	1.15E+10	[3, 3]	[5,4]	0.125	13831	6
example6 × 2	seed1	6	1.2	9.24E+09	1.3E+10	[3, 3]	[7,4]	0.422	15675	6	9.24E+09	1.3E+10	[3, 3]	[7,4]	0.5	15675	6
example6 × 2	seed2	6	1.2	8.41E+09	1.23E+10	[3, 3]	[6,4]	0.11	14799	6	8.41E+09	1.23E+10	[3, 3]	[6,4]	0.39	14799	6
example6 × 2	seed3	6	1.2	8.11E+09	1.19E+10	[3, 3]	[5,4]	0.109	14259	6	8.11E+09	1.19E+10	[3, 3]	[5,4]	0.391	14259	6
example6 × 2	seed4	6	1.2	7.21E+09	1.1E+10	[3, 3]	[5,3]	0.125	13226	6	7.21E+09	1.1E+10	[3, 3]	[5,3]	0.407	13226	6
example6 × 2	seed5	6	1.2	7.57E+09	1.18E+10	[4, 4]	[5,3]	0.125	14221	8	7.57E+09	1.18E+10	[4, 4]	[5,3]	0.391	14221	8
example6 × 2	seed6	6	1.2	8.16E+09	1.35E+10	[4, 5]	[5,3]	0.187	16268	9	8.16E+09	1.35E+10	[4, 5]	[5,3]	0.406	16268	9
example6 × 2	seed7	6	1.2	6.84E+09	1.22E+10	[4, 5]	[4,3]	0.156	14701	9	6.84E+09	1.22E+10	[4, 5]	[4,3]	0.422	14701	9
example6 × 2	seed8	6	1.2	6.03E+09	9.83E+09	[3, 3]	[5,3]	0.125	11811	6	6.03E+09	9.83E+09	[3, 3]	[5,3]	0.39	11811	6
example6 × 2	seed9	6	1.2	6.6E+09	1.02E+10	[3, 3]	[5,3]	0.11	12204	6	6.6E+09	1.02E+10	[3, 3]	[5,3]	0.407	12204	6
example6 × 2	seed10	6	1.2	7.52E+09	1.21E+10	[3, 4]	[5,3]	0.125	14487	7	7.52E+09	1.21E+10	[3, 4]	[5,3]	0.406	14487	7
example6 × 2	seed1	8	1.1	1.37E+10	1.87E+10	[4, 4]	[7,4]	0.094	16880	8	1.37E+10	1.87E+10	[4, 4]	[7,4]	0.125	16880	8
example6 × 2	seed2	8	1.1	1.23E+10	1.64E+10	[3, 3]	[7,5]	0.031	14799	6	1.23E+10	1.64E+10	[3, 3]	[7,5]	0.125	14799	6
example6 × 2	seed3	8	1.1	1.22E+10	1.86E+10	[5, 6]	[4,3]	0.032	16795	11	1.22E+10	1.86E+10	[5, 6]	[4,3]	0.125	16795	11
example6 × 2	seed4	8	1.1	1.09E+10	1.47E+10	[3, 3]	[5,3]	0.032	13226	6	1.09E+10	1.47E+10	[3, 3]	[5,3]	0.125	13226	6
example6 × 2	seed5	8	1.1	1.14E+10	1.61E+10	[4, 5]	[5,3]	0.046	14493	9	1.14E+10	1.61E+10	[4, 5]	[5,3]	0.125	14493	9
example6 × 2	seed6	8	1.1	1.27E+10	1.81E+10	[4, 5]	[5,3]	0.047	16268	9	1.27E+10	1.81E+10	[4, 5]	[5,3]	0.125	16268	9
example6 × 2	seed7	8	1.1	1.09E+10	1.63E+10	[4, 5]	[4,3]	0.047	14701	9	1.09E+10	1.63E+10	[4, 5]	[4,3]	0.125	14701	9
example6 × 2	seed8	8	1.1	9.47E+09	1.65E+10	[6, 6]	[3,2]	0.047	14852	12	9.47E+09	1.65E+10	[6. 6]	[3,2]	0.125	14852	12
example6 × 2	seed9	8	1.1	1.01E+10	1.68E+10	[6, 6]	[3,3]	0.031	15151	12	1.01E+10	1.68E+10	[6, 6]	[3,3]	0.125	15151	12
example6 × 2	seed10	8	1.1	1.14E+10	1.79E+10	[5, 5]	[4,3]	0.031	16155	10	1.14E+10	1.61E+10	[3, 4]	[6,3]	0.125	14487	7
example6 × 2	seed1	8	1.2	1.38E+10	1.85E+10	[3, 5]	[8,3]	0.094	16677	8	1.38E+10	1.85E+10	[3, 5]	[8,3]	0.125	16677	8
example6 × 2	seed2	8	1.2	1.25E+10	1.64E+10	[3, 3]	[6,4]	0.031	14799	6	1.25E+10	1.64E+10	[3, 3]	[6,4]	0.141	14799	6
example6 × 2	seed3	8	1.2	1.22E+10	1.86E+10	[6, 5]	[3,3]	0.047	16795	11	1.22E+10	1.86E+10	[6, 5]	[3,3]	0.109	16795	11
example6 × 2	seed4	8	1.2	1.09E+10	1.47E+10	[3, 3]	[5,3]	0.031	13226	6	1.09E+10	1.47E+10	[3, 3]	[5,3]	0.125	13226	6
example6 × 2	seed5	8	1.2	1.15E+10	1.58E+10	[4, 4]	[5,3]	0.047	14221	8	1.15E+10	1.58E+10	[4, 4]	[5,3]	0.125	14221	8
example6 × 2	seed6	8	1.2	1.27E+10	1.84E+10	[4, 6]	[5,2]	0.031	16558	10	1.27E+10	1.84E+10	[4, 6]	[5,2]	0.11	16558	10
example6 × 2	seed7	8	1.2	1.1E+10	1.67E+10	[4, 6]	[4,2]	0.032	15089	10	1.1E+10	1.67E+10	[4, 6]	[4,2]	0.125	15089	10
example6 × 2	seed8	8	1.2	9.55E+09	1.59E+10	[5, 6]	[3,2]	0.047	14314	11	9.55E+09	1.59E+10	[5, 6]	[3,2]	0.14	14314	11
example6 × 2	seed9	8	1.2	1.03E+10	1.62E+10	[5, 6]	[3,2]	0.047	14560	11	1.03E+10	1.62E+10	[5, 6]	[3,2]	0.125	14560	11
example6 × 2	seed10	8	1.2	1.17E+10	1.77E+10	[4, 6]	[4,2]	0.047	15974	10	1.17E+10	1.77E+10	[4, 6]	[4, 2]	0.125	15974	10

Table C.1.: Results for 6 × 2-configuration at uncapacitated case.

The following Table C.2 presents a comparison between the exact and heuristic algorithm. The data here deported show that at the 97.5% the heuristic finds the optimal solution and average improve of the CPU time is 65.6%.

Comparison heuristic and exact							
Configuration	instance	fare	σ	Dif z_{NET}	Dif z_{NET} %	Dif time	Dif time %
example6 × 2	seed1	6	1.1	0	0	0.015	13.76147
example6 × 2	seed2	6	1.1	0	0	0.094	75.2
example6 × 2	seed3	6	1.1	0	0	0.094	75.2
example6 × 2	seed4	6	1.1	0	0	0.094	75.2
example6 × 2	seed5	6	1.1	0	0	0.062	56.88073
example6 × 2	seed6	6	1.1	0	0	0.062	56.88073
example6 × 2	seed7	6	1.1	0	0	0.094	75.2
example6 × 2	seed8	6	1.1	0	0	0.078	62.4
example6 × 2	seed9	6	1.1	0	0	0.094	75.2
example6 × 2	seed10	6	1.1	0	0	0.094	75.2
example6 × 2	seed1	6	1.2	0	0	0.078	18.4834
example6 × 2	seed2	6	1.2	0	0	0.28	71.79487
example6 × 2	seed3	6	1.2	0	0	0.282	72.12276
example6 × 2	seed4	6	1.2	0	0	0.282	69.28747
example6 × 2	seed5	6	1.2	0	0	0.266	68.03069
example6 × 2	seed6	6	1.2	0	0	0.219	53.94089
example6 × 2	seed7	6	1.2	0	0	0.266	63.03318
example6 × 2	seed8	6	1.2	0	0	0.265	67.94872
example6 × 2	seed9	6	1.2	0	0	0.297	72.97297
example6 × 2	seed10	6	1.2	0	0	0.281	69.21182
example6 × 2	seed1	8	1.1	0	0	0.031	24.8
example6 × 2	seed2	8	1.1	0	0	0.094	75.2
example6 × 2	seed3	8	1.1	0	0	0.093	74.4
example6 × 2	seed4	8	1.1	0	0	0.093	74.4
example6 × 2	seed5	8	1.1	0	0	0.079	63.2
example6 × 2	seed6	8	1.1	0	0	0.078	62.4
example6 × 2	seed7	8	1.1	0	0	0.078	62.4
example6 × 2	seed8	8	1.1	0	0	0.078	62.4
example6 × 2	seed9	8	1.1	0	0	0.094	75.2
example6 × 2	seed10	8	1.1	8183830	0.071632	0.094	75.2
example6 × 2	seed1	8	1.2	0	0	0.031	24.8
example6 × 2	seed2	8	1.2	0	0	0.11	78.01418
example6 × 2	seed3	8	1.2	0	0	0.062	56.88073
example6 × 2	seed4	8	1.2	0	0	0.094	75.2
example6 × 2	seed5	8	1.2	0	0	0.078	62.4
example6 × 2	seed6	8	1.2	0	0	0.079	71.81818
example6 × 2	seed7	8	1.2	0	0	0.093	74.4
example6 × 2	seed8	8	1.2	0	0	0.093	66.42857
example6 × 2	seed9	8	1.2	0	0	0.078	62.4
example6 × 2	seed10	8	1.2	0	0	0.078	62.4

Table C.2.: Comparison exact and heuristic for 6×2-configuration at uncapacitated case.

7 × 3-configuration

At this configuration the average CPU time for the heuristic is 0.16995 and for the exact is 3.48045 seconds. That is, the heuristic improves the average CPU time in 95.12% respect to the exact. Detailed information on these solutions are shown in Table C.3. Note that due to the way in which we estimate the demand attraction (logit function) the higher the frequencies are, the higher the number of trips are. However, the maximum net profit and revenue is obtained for the Seed6-instance with minimum frequencies at all scenarios. The interpretation of this result is the following. As demand captured with minimum frequencies is high (81.57%), the number of carriages must be high since the number of trains is only 9 (see Table C.3). In this case, it does not compensate to attract more passengers since this fact would imply more trains and, therefore, more costs. Observe that it is cheaper trains with many carriages than many trains with few carriages.

				Heuristic							Exact						
Configuration	instance	fare	σ	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains
example7 × 3	seed1	6	1.1	7.89E+09	1.54E+10	[3, 3, 3]	[6,5,4]	0.219	18537	9	8.62E+09	1.8E+10	[3, 5, 4]	[6,3,4]	3.562	21664	12
example7 × 3	seed2	6	1.1	9.99E+09	1.77E+10	[3, 3, 3]	[6,6,5]	0.172	21302	9	9.99E+09	1.77E+10	[3, 3, 3]	[6,6,5]	3.516	21302	9
example7 × 3	seed3	6	1.1	8.76E+09	1.57E+10	[3, 3, 3]	[6,4,4]	0.172	18898	9	8.76E+09	1.57E+10	[3, 3, 3]	[6,4,4]	3.484	18898	9
example7 × 3	seed4	6	1.1	8.35E+09	1.54E+10	[3, 3, 3]	[6,5,4]	0.156	18547	9	8.57E+09	1.75E+10	[3, 4, 5]	[6,5,3]	3.484	21011	12
example7 × 3	seed5	6	1.1	8.7E+09	1.61E+10	[3, 3, 3]	[5,5,5]	0.156	19287	9	8.7E+09	1.61E+10	[3, 3, 3]	[5,5,5]	3.485	19287	9
example7 × 3	seed6	6	1.1	1.11E+10	1.91E+10	[3, 3, 3]	[6,6,6]	0.157	22918	9	1.11E+10	1.91E+10	[3, 3, 3]	[6,6,6]	3.484	22918	9
example7 × 3	seed7	6	1.1	8.05E+09	1.54E+10	[3, 3, 3]	[5,5,4]	0.172	18537	9	8.05E+09	1.54E+10	[3, 3, 3]	[5,5,4]	3.484	18537	9
example7 × 3	seed8	6	1.1	8.91E+09	1.62E+10	[3, 3, 3]	[5,4,5]	0.156	19520	9	8.91E+09	1.62E+10	[3, 3, 3]	[5,4,5]	3.485	19520	9
example7 × 3	seed9	6	1.1	8.36E+09	1.58E+10	[3, 3, 3]	[5,5,5]	0.157	18972	9	8.36E+09	1.58E+10	[3, 3, 3]	[5,5,5]	3.515	18972	9
example7 × 3	seed10	6	1.1	9.4E+09	1.72E+10	[3, 3, 3]	[6,4,5]	0.156	20686	9	9.4E+09	1.72E+10	[3, 3, 3]	[6,4,5]	3.516	20686	9
example7 × 3	seed1	6	1.2	8.14E+09	1.54E+10	[3, 3, 3]	[5,4,4]	0.203	18537	9	8.85E+09	1.88E+10	[3, 5, 5]	[5,3,4]	3.5	22606	13
example7 × 3	seed2	6	1.2	1.01E+10	1.77E+10	[3, 3, 3]	[6,5,5]	0.172	21302	9	1.01E+10	1.77E+10	[3, 3, 3]	[6,5,5]	3.438	21302	9
example7 × 3	seed3	6	1.2	8.89E+09	1.57E+10	[3, 3, 3]	[5,4,4]	0.157	18898	9	8.89E+09	1.57E+10	[3, 3, 3]	[5,4,4]	3.453	18898	9
example7 × 3	seed4	6	1.2	8.49E+09	1.54E+10	[3, 3, 3]	[5,5,4]	0.157	18547	9	8.71E+09	1.75E+10	[3, 4, 5]	[5,5,3]	3.484	21011	12
example7 × 3	seed5	6	1.2	8.7E+09	1.61E+10	[3, 3, 3]	[5,5,5]	0.156	19287	9	8.7E+09	1.61E+10	[3, 3, 3]	[5,5,5]	3.484	19287	9
example7 × 3	seed6	6	1.2	1.12E+10	1.91E+10	[3, 3, 3]	[6,5,6]	0.172	22918	9	1.12E+10	1.91E+10	[3, 3, 3]	[6,5,6]	3.469	22918	9
example7 × 3	seed7	6	1.2	8.15E+09	1.54E+10	[3, 3, 3]	[5,4,4]	0.156	18537	9	8.15E+09	1.54E+10	[3, 3, 3]	[5,4,4]	3.469	18537	9
example7 × 3	seed8	6	1.2	8.91E+09	1.62E+10	[3, 3, 3]	[5,4,5]	0.172	19520	9	8.91E+09	1.62E+10	[3, 3, 3]	[5,4,5]	3.484	19520	9
example7 × 3	seed9	6	1.2	8.36E+09	1.58E+10	[3, 3, 3]	[5,5,5]	0.156	18972	9	8.36E+09	1.58E+10	[3, 3, 3]	[5,5,5]	3.485	18972	9
example7 × 3	seed10	6	1.2	9.4E+09	1.72E+10	[3, 3, 3]	[6,4,5]	0.172	20686	9	9.4E+09	1.72E+10	[3, 3, 3]	[6,4,5]	3.468	20686	9
example7 × 3	seed1	8	1.1	1.48E+10	2.51E+10	[3, 5, 5]	[6,4,4]	0.281	22606	13	1.48E+10	2.51E+10	[3, 5, 5]	[6,4,4]	3.5	22606	13
example7 × 3	seed2	8	1.1	1.59E+10	2.36E+10	[3, 3, 3]	[6,6,5]	0.156	21302	9	1.59E+10	2.36E+10	[3, 3, 3]	[6,6,5]	3.453	21302	9
example7 × 3	seed3	8	1.1	1.4E+10	2.1E+10	[3, 3, 3]	[6,4,4]	0.172	18898	9	1.4E+10	2.1E+10	[3, 3, 3]	[6,4,4]	3.469	18898	9
example7 × 3	seed4	8	1.1	1.35E+10	2.06E+10	[3, 3, 3]	[6,5,4]	0.156	18547	9	1.44E+10	2.33E+10	[3, 4, 5]	[6,5,3]	3.469	21011	12
example7 × 3	seed5	8	1.1	1.4E+10	2.14E+10	[3, 3, 3]	[5,5,5]	0.157	19287	9	1.4E+10	2.14E+10	[3, 3, 3]	[5,5,5]	3.484	19287	9
example7 × 3	seed6	8	1.1	1.75E+10	2.54E+10	[3, 3, 3]	[6,6,6]	0.156	22918	9	1.75E+10	2.54E+10	[3, 3, 3]	[6,6,6]	3.469	22918	9
example7 × 3	seed7	8	1.1	1.32E+10	2.06E+10	[3, 3, 3]	[5,5,4]	0.156	18537	9	1.32E+10	2.06E+10	[3, 3, 3]	[5,5,4]	3.485	18537	9
example7 × 3	seed8	8	1.1	1.43E+10	2.17E+10	[3, 3, 3]	[5,4,5]	0.156	19520	9	1.45E+10	2.43E+10	[3, 5, 5]	[5,3,4]	3.484	21923	13
example7 × 3	seed9	8	1.1	1.37E+10	2.18E+10	[3, 3, 4]	[5,6,4]	0.172	19685	10	1.37E+10	2.18E+10	[3, 3, 4]	[5,6,4]	3.469	19685	10
example7 × 3	seed10	8	1.1	1.53E+10	2.37E+10	[3, 4, 3]	[6,3,5]	0.172	21329	10	1.53E+10	2.37E+10	[3. 4. 3]	[6,3,5]	3,515	21329	10
example7 × 3	seed1	8	1.2	1.51E+10	2.51E+10	[3, 5, 5]	[5,3,4]	0.25	22606	13	1.51E+10	2.51E+10	[3, 5, 5]	[5,3,4]	3.468	22606	13
example7 × 3	seed2	8	1.2	1.6E+10	2.36E+10	[3, 3, 3]	[6,5,5]	0.156	21302	9	1.6E+10	2.36E+10	[3, 3, 3]	[6,5,5]	3.438	21302	9
example7 × 3	seed3	8	1.2	1.41E+10	2.1E+10	[3, 3, 3]	[5,4,4]	0.157	18898	9	1.41E+10	2.1E+10	[3, 3, 3]	[5,4,4]	3.484	18898	9
example7 × 3	seed4	8	1.2	1.36E+10	2.06E+10	[3, 3, 3]	[5,5,4]	0.156	18547	9	1.45E+10	2.33E+10	[3, 4, 5]	[5,5,3]	3.5	21011	12
example7 × 3	seed5	8	1.2	1.4E+10	2.14E+10	[3, 3, 3]	[5,5,5]	0.172	19287	9	1.4E+10	2.14E+10	[3, 3, 3]	[5,5,5]	3.469	19287	9
example7 × 3	seed6	8	1.2	1.76E+10	2.54E+10	[3, 3, 3]	[6,5,6]	0.156	22918	9	1.76E+10	2.54E+10	[3, 3, 3]	[6,5,6]	3.453	22918	9
example7 × 3	seed7	8	1.2	1.33E+10	2.06E+10	[3, 3, 3]	[5,4,4]	0.172	18537	9	1.33E+10	2.06E+10	[3, 3, 3]	[5,4,4]	3.438	18537	9
example7 × 3	seed8	8	1.2	1.43E+10	2.17E+10	[3, 3, 3]	[5,4,5]	0.156	19520	9	1,47E+10	2.43E+10	[3, 5, 5]	[5,3,3]	3.469	21923	13
example7 × 3	seed9	8	1.2	1.38E+10	2.18E+10	[3, 3, 4]	[5,5,4]	0.171	19685	10	1.38E+10	2.18E+10	[3, 3, 4]	[5,5,4]	3,469	19685	10
example7 × 3	seed10	8	1.2	1.53E+10	2.37E+10	[3, 4, 3]	[6,3,5]	0.172	21329	10	1.53E+10	2.37E+10	[3, 4, 3]	[6,3,5]	3.484	21329	10

Table C.3.: Results for 7×3 configuration at uncapacitated case.

The following Table C.4 presents a comparison between the exact and heuristic algorithm. The data here deported show that at the 80% the heuristic finds the optimal solution and average improve of the CPU time is 95.11%. The average error at these cases is 4.7%.

Comparison heuristic and exact							
Configuration	instance	fare	σ	Dif z_{NET}	Dif z_{NET} %	Dif time	Dif time %
example7 × 3	seed1	6	1.1	7.25E+08	8.414799	3.343	93.85177
example7 × 3	seed2	6	1.1	0	0	3.344	95.10808
example7 × 3	seed3	6	1.1	0	0	3.312	95.06315
example7 × 3	seed4	6	1.1	2.22E+08	2.594155	3.328	95.52239
example7 × 3	seed5	6	1.1	0	0	3.329	95.52367
example7 × 3	seed6	6	1.1	0	0	3.327	95.49369
example7 × 3	seed7	6	1.1	0	0	3.312	95.06315
example7 × 3	seed8	6	1.1	0	0	3.329	95.52367
example7 × 3	seed9	6	1.1	0	0	3.358	95.53343
example7 × 3	seed10	6	1.1	0	0	3.36	95.56314
example7 × 3	seed1	6	1.2	7.02E+08	7.935962	3.297	94.2
example7 × 3	seed2	6	1.2	0	0	3.266	94.99709
example7 × 3	seed3	6	1.2	0	0	3.296	95.45323
example7 × 3	seed4	6	1.2	2.22E+08	2.552337	3.327	95.49369
example7 × 3	seed5	6	1.2	0	0	3.328	95.52239
example7 × 3	seed6	6	1.2	0	0	3.297	95.0418
example7 × 3	seed7	6	1.2	0	0	3.313	95.50303
example7 × 3	seed8	6	1.2	0	0	3.312	95.06315
example7 × 3	seed9	6	1.2	0	0	3.329	95.52367
example7 × 3	seed10	6	1.2	0	0	3.296	95.04037
example7 × 3	seed1	8	1.1	0	0	3.219	91.97143
example7 × 3	seed2	8	1.1	0	0	3.297	95.48219
example7 × 3	seed3	8	1.1	0	0	3.297	95.0418
example7 × 3	seed4	8	1.1	9.06E+08	6.291492	3.313	95.50303
example7 × 3	seed5	8	1.1	0	0	3.327	95.49369
example7 × 3	seed6	8	1.1	0	0	3.313	95.50303
example7 × 3	seed7	8	1.1	0	0	3.329	95.52367
example7 × 3	seed8	8	1.1	1.74E+08	1.199384	3.328	95.52239
example7 × 3	seed9	8	1.1	0	0	3.297	95.0418
example7 × 3	seed10	8	1.1	0	0	3.343	95.10669
example7 × 3	seed1	8	1.2	0	0	3.218	92.79123
example7 × 3	seed2	8	1.2	0	0	3.282	95.46248
example7 × 3	seed3	8	1.2	0	0	3.327	95.49369
example7 × 3	seed4	8	1.2	9.06E+08	6.230734	3.344	95.54286
example7 × 3	seed5	8	1.2	0	0	3.297	95.0418
example7 × 3	seed6	8	1.2	0	0	3.297	95.48219
example7 × 3	seed7	8	1.2	0	0	3.266	94.99709
example7 × 3	seed8	8	1.2	3.48E+08	2.372124	3.313	95.50303
example7 × 3	seed9	8	1.2	0	0	3.298	95.07063
example7 × 3	seed10	8	1.2	0	0	3.312	95.06315

Table C.4.: Comparison exact and heuristic for 7×3-configuration at uncapacitated case.

8×3-configuration

The data related to this configuration are shown in the Table C.5. The maximum net profit and revenue are obtained for the Seed9-instance in the third first scenarios. The captured demand represents at these cases the 61.67% on the total trips number. In the last scenario, the maximum revenue is searched for Seed3-instance (63.85% demand) and the maximum net profit for the Seed9-instance. The average CPU time in the heuristic is 0.2836 and the exact is 5.89 seconds, which represents an improvement of 95.18% in the time.

				Heuristic							Exact						
Configuration	instance	fare	σ	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains
example8 × 3	seed1	6	1.1	5.09E+09	1.14E+10	[3, 3, 3]	[2,4,3]	0.359	13724	9	5.09E+09	1.14E+10	[3, 3, 3]	[2,4,3]	5.75	13724	9
example8 × 3	seed2	6	1.1	4.78E+09	1.16E+10	[3, 3, 3]	[3,5,3]	0.266	13922	9	4.78E+09	1.16E+10	[3, 3, 3]	[3,5,3]	5.906	13922	9
example8 × 3	seed3	6	1.1	7.99E+09	1.46E+10	[4, 3, 3]	[3,5,5]	0.281	17521	10	7.99E+09	1.46E+10	[4, 3, 3]	[3,5,5]	5.828	17521	10
example8 × 3	seed4	6	1.1	8.26E+09	1.48E+10	[4, 3, 3]	[3,5,5]	0.282	17756	10	8.26E+09	1.48E+10	[4, 3, 3]	[3,5,5]	5.984	17756	10
example8 × 3	seed5	6	1.1	6.96E+09	1.27E+10	[3, 3, 3]	[3,4,3]	0.281	15229	9	6.96E+09	1.27E+10	[3, 3, 3]	[3,4,3]	6.015	15229	9
example8 × 3	seed6	6	1.1	7.34E+09	1.39E+10	[3, 3, 3]	[3,6,3]	0.266	16749	9	7.34E+09	1.39E+10	[3, 3, 3]	[3,6,3]	5.781	16749	9
example8 × 3	seed7	6	1.1	6.14E+09	1.27E+10	[3, 3, 3]	[2,5,3]	0.265	15257	9	6.14E+09	1.27E+10	[3, 3, 3]	[2,5,3]	5.782	15257	9
example8 × 3	seed8	6	1.1	5.85E+09	1.31E+10	[4, 3, 3]	[3,5,4]	0.266	15742	10	5.85E+09	1.31E+10	[4, 3, 3]	[3,5,4]	5.781	15742	10
example8 × 3	seed9	6	1.1	9.95E+09	1.59E+10	[3, 3, 3]	[5,5,4]	0.266	19115	9	9.95E+09	1.59E+10	[3, 3, 3]	[5,5,4]	5.859	19115	9
example8 × 3	seed10	6	1.1	7.58E+09	1.45E+10	[3, 3, 3]	[5,6,4]	0.281	17388	9	7.58E+09	1.45E+10	[3, 3, 3]	[5,6,4]	6.078	17388	9
example8 × 3	seed1	6	1.2	5.09E+09	1.14E+10	[3, 3, 3]	[2,4,3]	0.328	13724	9	5.09E+09	1.14E+10	[3, 3, 3]	[2,4,3]	5.859	13724	9
example8 × 3	seed2	6	1.2	4.91E+09	1.16E+10	[3, 3, 3]	[2,5,3]	0.266	13922	9	4.91E+09	1.16E+10	[3, 3, 3]	[2,5,3]	5.922	13922	9
example8 × 3	seed3	6	1.2	8.07E+09	1.46E+10	[4, 3, 3]	[3,5,4]	0.281	17521	10	8.07E+09	1.46E+10	[4, 3, 3]	[3,5,4]	5.828	17521	10
example8 × 3	seed4	6	1.2	8.38E+09	1.42E+10	[3, 3, 3]	[3,5,4]	0.266	17103	9	8.38E+09	1.42E+10	[3, 3, 3]	[3,5,4]	5.922	17103	9
example8 × 3	seed5	6	1.2	7.02E+09	1.27E+10	[3, 3, 3]	[3,4,2]	0.281	15229	9	7.02E+09	1.27E+10	[3, 3, 3]	[3,4,2]	6.141	15229	9
example8 × 3	seed6	6	1.2	7.34E+09	1.39E+10	[3, 3, 3]	[3,6,3]	0.265	16749	9	7.34E+09	1.39E+10	[3, 3, 3]	[3,6,3]	5.719	16749	9
example8 × 3	seed7	6	1.2	6.14E+09	1.27E+10	[3, 3, 3]	[2,5,3]	0.265	15257	9	6.14E+09	1.27E+10	[3, 3, 3]	[2,5,3]	5.766	15257	9
example8 × 3	seed8	6	1.2	5.99E+09	1.31E+10	[4, 3, 3]	[3,4,4]	0.266	15742	10	6E+09	1.42E+10	[5, 3, 4]	[3,4,3]	5.781	17105	12
example8 × 3	seed9	6	1.2	1E+10	1.59E+10	[3, 3, 3]	[4,5,4]	0.265	19115	9	1E+10	1.59E+10	[3, 3, 3]	[4,5,4]	5.813	19115	9
example8 × 3	seed10	6	1.2	7.83E+09	1.45E+10	[3, 3, 3]	[4,5,4]	0.281	17388	9	7.83E+09	1.45E+10	[3, 3, 3]	[4,5,4]	6.063	17388	9
example8 × 3	seed1	8	1.1	8.89E+09	1.52E+10	[3, 3, 3]	[2,4,3]	0.312	13724	9	8.89E+09	1.52E+10	[3, 3, 3]	[2,4,3]	5.719	13724	9
example8 × 3	seed2	8	1.1	8.65E+09	1.54E+10	[3, 3, 3]	[3,5,3]	0.265	13922	9	8.65E+09	1.54E+10	[3, 3, 3]	[3,5,3]	5.875	13922	9
example8 × 3	seed3	8	1.1	1.29E+10	2.01E+10	[5, 3, 3]	[3,5,5]	0.281	18153	11	1.29E+10	2.01E+10	[5, 3, 3]	[3,5,5]	5.781	18153	11
example8 × 3	seed4	8	1.1	1.35E+10	2.1E+10	[5, 3, 4]	[3,5,4]	0.297	18944	12	1.37E+10	2.18E+10	[6, 3, 4]	[3,5,4]	5.875	19651	13
example8 × 3	seed5	8	1.1	1.12E+10	1.69E+10	[3, 3, 3]	[3,4,3]	0.266	15229	9	1.12E+10	1.69E+10	[3, 3, 3]	[3,4,3]	6.047	15229	9
example8 × 3	seed6	8	1.1	1.2E+10	1.86E+10	[3, 3, 3]	[3,6,3]	0.266	16749	9	1.2E+10	1.86E+10	[3, 3, 3]	[3,6,3]	5.703	16749	9
example8 × 3	seed7	8	1.1	1.04E+10	1.69E+10	[3, 3, 3]	[2,5,3]	0.265	15257	9	1.04E+10	1.69E+10	[3, 3, 3]	[2,5,3]	5.828	15257	9
example8 × 3	seed8	8	1.1	1.05E+10	1.9E+10	[5, 3, 4]	[3,5,4]	0.313	17105	12	1.05E+10	1.9E+10	[5, 3, 4]	[3,5,4]	5.734	17105	12
example8 × 3	seed9	8	1.1	1.53E+10	2.12E+10	[3, 3, 3]	[5,5,4]	0.266	19115	9	1.53E+10	2.12E+10	[3, 3, 3]	[5,5,4]	5.859	19115	9
example8 × 3	seed10	8	1.1	1.24E+10	1.93E+10	[3, 3, 3]	[5,6,4]	0.266	17388	9	1.24E+10	1.93E+10	[3, 3, 3]	[5,6,4]	6.125	17388	9
example8 × 3	seed1	8	1.2	8.89E+09	1.52E+10	[3, 3, 3]	[2, 4, 3]	0.328	13724	9	8.89E+09	1.52E+10	[3, 3, 3]	[2,4,3]	5.813	13724	9
example8 × 3	seed2	8	1.2	8.77E+09	1.54E+10	[3, 3, 3]	[2,5,3]	0.266	13922	9	8.77E+09	1.54E+10	[3, 3, 3]	[2,5,3]	5.875	13922	9
example8 × 3	seed3	8	1.2	1.3E+10	2.01E+10	[5, 3, 3]	[3,5,4]	0.329	18153	11	1.3E+10	2.01E+10	[5, 3, 3]	[3,5,4]	5.89	18153	11
example8 × 3	seed4	8	1.2	1.37E+10	2.18E+10	[6, 3, 4]	[3,5,4]	0.328	19651	13	1.37E+10	2.18E+10	[6, 3, 4]	[3,5,4]	5.938	19651	13
example8 × 3	seed5	8	1.2	1.13E+10	1.76E+10	[4, 3, 3]	[2,5,2]	0.281	15846	10	1.13E+10	1.76E+10	[4, 3, 3]	[2,5,2]	6.172	15846	10
example8 × 3	seed6	8	1.2	1.2E+10	1.86E+10	[3, 3, 3]	[3,6,3]	0.266	16749	9	1.2E+10	1.86E+10	[3, 3, 3]	[3,6,3]	5.843	16749	9
example8 × 3	seed7	8	1.2	1.04E+10	1.69E+10	[3, 3, 3]	[2,5,3]	0.297	15257	9	1.04E+10	1.69E+10	[3, 3, 3]	[2,5,3]	5.984	15257	9
example8 × 3	seed8	8	1.2	1.07E+10	1.9E+10	[5, 3, 4]	[3,4,3]	0.313	17105	12	1.07E+10	1.9E+10	[5, 3, 4]	[3,4,3]	5.812	17105	12
example8 × 3	seed9	8	1.2	1.53E+10	2.12E+10	[3, 3, 3]	[4,5,4]	0.281	19115	9	1.53E+10	2.12E+10	[3, 3, 3]	[4,5,4]	5.844	19115	9
example8 × 3	seed10	8	1.2	1.27E+10	1.93E+10	[3, 3, 3]	[4,5,4]	0.281	17388	9	1.27E+10	1.93E+10	[3, 3, 3]	[4,5,4]	6.11	17388	9

Table C.5.: Results for 8×3-configuration at uncapacitated case.

Table C.6 shows relevant details in order to compare the exact and heuristic algorithm. It can be observed that the optimal solution is found in 95% of the carried out experiments and the average improve of the CPU time is 95.18%.

Comparison heuristic and exact							
Configuration	instance	fare	σ	Dif z_{NET}	Dif z_{NET} %	Dif time	Dif time %
example8 × 3	seed1	6	1.1	0	0	5.391	93.75652
example8 × 3	seed2	6	1.1	0	0	5.64	95.49611
example8 × 3	seed3	6	1.1	0	0	5.547	95.17845
example8 × 3	seed4	6	1.1	0	0	5.702	95.28743
example8 × 3	seed5	6	1.1	0	0	5.734	95.32835
example8 × 3	seed6	6	1.1	0	0	5.515	95.39872
example8 × 3	seed7	6	1.1	0	0	5.517	95.41681
example8 × 3	seed8	6	1.1	0	0	5.515	95.39872
example8 × 3	seed9	6	1.1	0	0	5.593	95.45998
example8 × 3	seed10	6	1.1	0	0	5.797	95.37677
example8 × 3	seed1	6	1.2	0	0	5.531	94.40178
example8 × 3	seed2	6	1.2	0	0	5.656	95.50827
example8 × 3	seed3	6	1.2	0	0	5.547	95.17845
example8 × 3	seed4	6	1.2	0	0	5.656	95.50827
example8 × 3	seed5	6	1.2	0	0	5.86	95.4242
example8 × 3	seed6	6	1.2	0	0	5.454	95.36632
example8 × 3	seed7	6	1.2	0	0	5.501	95.40409
example8 × 3	seed8	6	1.2	16537852	0.275527	5.515	95.39872
example8 × 3	seed9	6	1.2	0	0	5.548	95.44125
example8 × 3	seed10	6	1.2	0	0	5.782	95.36533
example8 × 3	seed1	8	1.1	0	0	5.407	94.5445
example8 × 3	seed2	8	1.1	0	0	5.61	95.48936
example8 × 3	seed3	8	1.1	0	0	5.5	95.13925
example8 × 3	seed4	8	1.1	1.93E+08	1.41026	5.578	94.94468
example8 × 3	seed5	8	1.1	0	0	5.781	95.60112
example8 × 3	seed6	8	1.1	0	0	5.437	95.33579
example8 × 3	seed7	8	1.1	0	0	5.563	95.45299
example8 × 3	seed8	8	1.1	0	0	5.421	94.54133
example8 × 3	seed9	8	1.1	0	0	5.593	95.45998
example8 × 3	seed10	8	1.1	0	0	5.859	95.65714
example8 × 3	seed1	8	1.2	0	0	5.485	94.35747
example8 × 3	seed2	8	1.2	0	0	5.609	95.47234
example8 × 3	seed3	8	1.2	0	0	5.561	94.41426
example8 × 3	seed4	8	1.2	0	0	5.61	94.47625
example8 × 3	seed5	8	1.2	0	0	5.891	95.44718
example8 × 3	seed6	8	1.2	0	0	5.577	95.44754
example8 × 3	seed7	8	1.2	0	0	5.687	95.03676
example8 × 3	seed8	8	1.2	0	0	5.499	94.61459
example8 × 3	seed9	8	1.2	0	0	5.563	95.19165
example8 × 3	seed10	8	1.2	0	0	5.829	95.40098

Table C.6.: Comparison exact and heuristic for 8×3-configuration at uncapacitated case.

15 × 5-configuration

In this section, we have tested our algorithms on networks with 15 nodes and 5 lines. The results of these experiments are shown in Table C.7. It can be observed that our heuristic was able to find the optimal solution for most of instances (90%). The maximum profit is searched for the Seed-4 instance at all scenarios. However, the maximum revenue is obtained for Seed2-instance in the first, second and third scenario and Seed8-instance for the last scenario. In Figure C.3 and C.4 the revenue and profit are graphically represented for the first and fourth scenario, respectively.

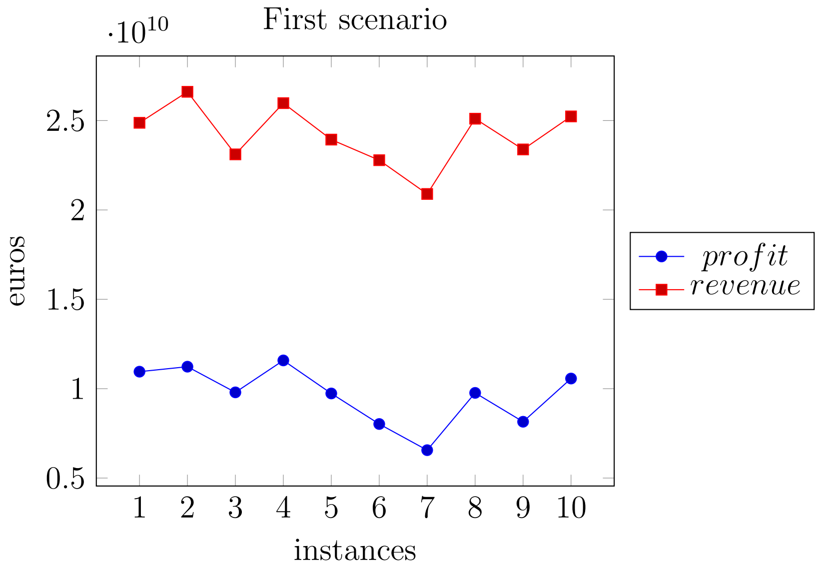


Figure C.3.: Revenue and profit of each instance for first scenario of 15×5-configuration.

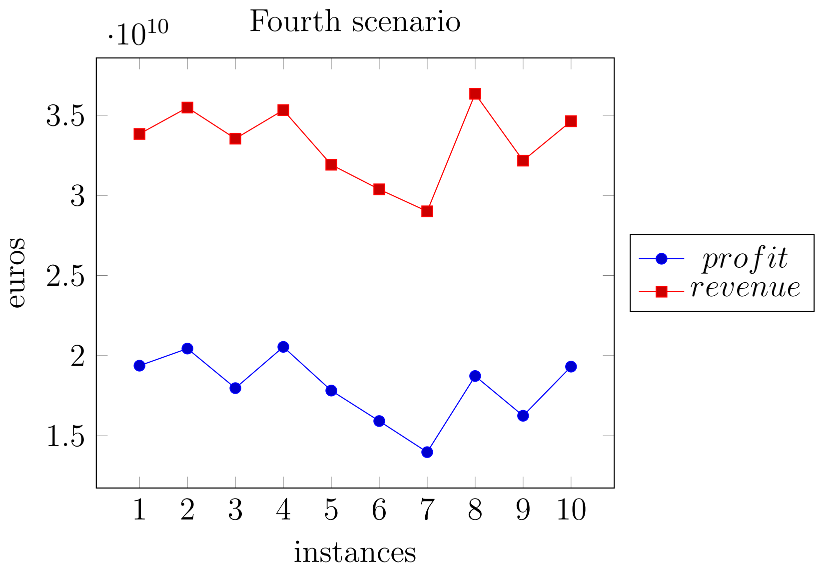


Figure C.4.: Revenue and profit of each instance for fourth scenario of 15×5-configuration.

				Heuristic							Exact						
Configuration	instance	fare	σ	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains
example15 × 5	seed1	6	1.1	1.10E+10	2.49E+10	[3, 3, 3, 3, 4]	[4,4,3,5,5]	5.757	29890	16	1.095E+10	2.49E+10	[3, 3, 3, 3, 4]	[4,4,3,5,5]	5347.519	29890	16
example15 × 5	seed2	6	1.1	1.12E+10	2.66E+10	[3, 3, 4, 3, 4]	[3,4,5,5,4]	5.571	31974	17	1.123E+10	2.66E+10	[3, 3, 4, 3, 4]	[3,4,5,5,4]	5082.537	31974	17
example15 × 5	seed3	6	1.1	9.79E+09	2.31E+10	[3, 3, 3, 3, 3]	[4,4,5,3,5]	5.151	27768	15	9.794E+09	2.31E+10	[3, 3, 3, 3, 3]	[4,4,5,3,5]	5169.428	27768	15
example15 × 5	seed4	6	1.1	1.16E+10	2.6E+10	[3, 3, 3, 3, 3]	[3,4,7,5,7]	5.199	31212	15	1.158E+10	2.6E+10	[3, 3, 3, 3, 3]	[3,4,7,5,7]	5201.598	31212	15
example15 × 5	seed5	6	1.1	9.73E+09	2.39E+10	[3, 3, 3, 3, 4]	[3,4,4,6,4]	5.524	28761	16	9.733E+09	2.39E+10	[3, 3, 3, 3, 4]	[3,4,4,6,4]	5372.934	28761	16
example15 × 5	seed6	6	1.1	8.02E+09	2.28E+10	[3, 3, 3, 3, 4]	[4,4,3,5,5]	5.308	27377	16	8.024E+09	2.28E+10	[3, 3, 3, 3, 4]	[4,4,3,5,5]	5122.384	27377	16
example15 × 5	seed7	6	1.1	6.56E+09	2.09E+10	[3, 3, 3, 3, 3]	[2,4,3,6,4]	5.306	25106	15	6.56E+09	2.09E+10	[3, 3, 3, 3, 3]	[2,4,3,6,4]	5297.319	25106	15
example15 × 5	seed8	6	1.1	9.76E+09	2.51E+10	[3, 3, 3, 4, 4]	[4,4,4,4,4]	5.728	30166	17	9.765E+09	2.51E+10	[3, 3, 3, 4, 4]	[4,4,4,4,4]	5232.515	30166	17
example15 × 5	seed9	6	1.1	8.15E+09	2.34E+10	[3, 3, 3, 5, 3]	[3,4,3,3,6]	5.573	28100	17	8.15E+09	2.34E+10	[3, 3, 3, 5, 3]	[3,4,3,3,6]	5253.981	28100	17
example15 × 5	seed10	6	1.1	1.06E+10	2.52E+10	[3, 3, 3, 3, 3]	[3,4,6,6,5]	5.181	30320	15	1.057E+10	2.52E+10	[3, 3, 3, 3, 3]	[3,4,6,6,5]	5191.442	30320	15
example15 × 5	seed1	6	1.2	1.11E+10	2.41E+10	[3, 3, 3, 3, 3]	[3,4,3,4,5]	5.649	28977	15	1.114E+10	2.41E+10	[3, 3, 3, 3, 3]	[3,4,3,4,5]	5367.26	28977	15
example15 × 5	seed2	6	1.2	1.16E+10	2.66E+10	[3, 3, 4, 3, 4]	[3,3,4,5,4]	5.587	31974	17	1.157E+10	2.66E+10	[3, 3, 4, 3, 4]	[3,3,4,5,4]	5107.919	31974	17
example15 × 5	seed3	6	1.2	1.00E+10	2.31E+10	[3, 3, 3, 3, 3]	[3,3,5,3,5]	5.18	27768	15	1.005E+10	2.31E+10	[3, 3, 3, 3, 3]	[3,3,5,3,5]	5192.534	27768	15
example15 × 5	seed4	6	1.2	1.18E+10	2.6E+10	[3, 3, 3, 3, 3]	[3,4,7,4,6]	5.211	31212	15	1.183E+10	2.6E+10	[3, 3, 3, 3, 3]	[3,4,7,4,6]	5206.448	31212	15
example15 × 5	seed5	6	1.2	9.84E+09	2.39E+10	[3, 3, 3, 3, 4]	[2,4,4,6,4]	5.555	28761	16	9.842E+09	2.39E+10	[3, 3, 3, 3, 4]	[2,4,4,6,4]	5392.681	28761	16
example15 × 5	seed6	6	1.2	8.33E+09	2.28E+10	[3, 3, 3, 3, 4]	[4,4,3,4,4]	5.323	27377	16	8.325E+09	2.28E+10	[3, 3, 3, 3, 4]	[4,4,3,4,4]	5134.968	27377	16
example15 × 5	seed7	6	1.2	6.73E+09	2.18E+10	[3, 3, 3, 3, 4]	[2,4,3,6,3]	5,493	26140	16	6.728E+09	2.18E+10	[3, 3, 3, 3, 4]	[2,4,3,6,3]	5323.229	26140	16
example15 × 5	seed8	6	1.2	9.89E+09	2.43E+10	[3, 3, 3, 3, 4]	[4,4,4,4,4]	5.42	29153	16	9.89E+09	2.43E+10	[3, 3, 3, 3, 4]	[4,4,4,4,4]	5246.836	29153	16
example15 × 5	seed9	6	1.2	8.23E+09	2.24E+10	[3, 3, 3, 4, 3]	[3,4,3,3,5]	5.414	26865	16	8.234E+09	2.24E+10	[3, 3, 3, 4, 3]	[3,4,3,3,5]	5268.521	26865	16
example15 × 5	seed10	6	1.2	1.07E+10	2.52E+10	[3, 3, 3, 3, 3]	[3,4,6,5,5]	5.196	30320	15	1.072E+10	2.52E+10	[3, 3, 3, 3, 3]	[3,4,6,5,5]	5229.379	30320	15
example15 × 5	seed1	8	1.1	1.94E+10	3.38E+10	[3, 3, 3, 3, 5]	[4,4,3,5,4]	5.944	30492	17	1.937E+10	3.38E+10	[3, 3, 3, 3, 5]	[4,4,3,5,4]	5393.586	30492	17
example15 × 5	seed2	8	1.1	2.01E+10	3.62E+10	[3, 3, 5, 3, 4]	[3,4,4,5,4]	5.633	32643	18	2.011E+10	3.62E+10	[3, 3, 5, 3, 4]	[3,4,4,5,4]	5134.516	32643	18
example15 × 5	seed3	8	1.1	1.76E+10	3.16E+10	[3, 3, 3, 3, 4]	[4,4,5,3,4]	5.384	28453	16	1.778E+10	3.37E+10	[3, 3, 4, 3, 6]	[4,4,4,3,3]	5190.443	30389	19
example15 × 5	seed4	8	1.1	2.03E+10	3.53E+10	[3, 3, 3, 3, 4]	[4,4,7,5,5]	5.368	31832	16	2.029E+10	3.53E+10	[3, 3, 3, 3, 4]	[4,4,7,5,5]	5204.513	31832	16
example15 × 5	seed5	8	1.1	1.77E+10	3.19E+10	[3, 3, 3, 3, 4]	[3,4,4,6,4]	5.572	28761	16	1.771E+10	3.19E+10	[3, 3, 3, 3, 4]	[3,4,4,6,4]	5383.466	28761	16
example15 × 5	seed6	8	1.1	1.58E+10	3.14E+10	[3, 3, 3, 4, 4]	[4,4,3,4,5]	5.774	28329	17	1.582E+10	3.14E+10	[3, 3, 3, 4, 4]	[4,4,3,4,5]	5122.021	28329	17
example15 × 5	seed7	8	1.1	1.38E+10	2.9E+10	[3, 3, 3, 3, 4]	[2,4,3,6,4]	5.477	26140	16	1.381E+10	2.9E+10	[3, 3, 3, 3, 4]	[2,4,3,6,4]	5300.952	26140	16
example15 × 5	seed8	8	1.1	1.85E+10	3.56E+10	[4, 3, 4, 4, 4]	[3,4,4,4,5]	6.691	32061	19	1.845E+10	3.56E+10	[4, 3, 4, 4, 4]	[3,4,4,4,5]	5235.464	32061	19
example15 × 5	seed9	8	1.1	1.61E+10	3.23E+10	[3, 3, 4, 5, 3]	[3,4,3,3,6]	6.024	29113	18	1.615E+10	3.23E+10	[3, 3, 4, 5, 3]	[3,4,3,3,6]	5247.679	29113	18
example15 × 5	seed10	8	1.1	1.92E+10	3.46E+10	[3, 3, 3, 3, 4]	[3,4,7,6,4]	5.353	31207	16	1.925E+10	3.62E+10	[3, 3, 4, 3, 5]	[3,4,5,6,4]	5196.776	32625	18
example15 × 5	seed1	8	1.2	1.94E+10	3.38E+10	[3, 3, 3, 3, 5]	[4,4,3,5,4]	5.929	30492	17	1.937E+10	3.38E+10	[3, 3, 3, 3, 5]	[4,4,3,5,4]	5430.932	30492	17
example15 × 5	seed2	8	1.2	2.04E+10	3.55E+10	[3, 3, 4, 3, 4]	[3,3,4,5,4]	5.679	31974	17	2.044E+10	3.55E+10	[3, 3, 4, 3, 4]	[3,3,4,5,4]	5157.745	31974	17
example15 × 5	seed3	8	1.2	1.79E+10	3.23E+10	[3, 3, 3, 3, 5]	[4,3,5,3,3]	5.554	29080	17	1.797E+10	3.35E+10	[4, 3, 3, 3, 6]	[3,3,5,3,3]	5199.319	30226	19
example15 × 5	seed4	8	1.2	2.05E+10	3.53E+10	[3, 3, 3, 3, 4]	[3,4,7,4,5]	5.415	31832	16	2.055E+10	3.53E+10	[3, 3, 3, 3, 4]	[3,4,7,4,5]	5228.194	31832	16
example15 × 5	seed5	8	1.2	1.78E+10	3.19E+10	[3, 3, 3, 3, 4]	[2,4,4,6,4]	5.602	28761	16	1.782E+10	3.19E+10	[3, 3, 3, 3, 4]	[2,4,4,6,4]	5410.262	28761	16
example15 × 5	seed6	8	1.2	1.59E+10	3.04E+10	[3, 3, 3, 3, 4]	[4,4,3,4,4]	5.353	27377	16	1.592E+10	3.04E+10	[3, 3, 3, 3, 4]	[4,4,3,4,4]	5155.644	27377	16
example15 × 5	seed7	8	1.2	1.40E+10	2.9E+10	[3, 3, 3, 3, 4]	[2,4,3,6,3]	5.523	26140	16	1.398E+10	2.9E+10	[3, 3, 3, 3, 4]	[2,4,3,6,3]	5329.518	26140	16
example15 × 5	seed8	8	1.2	1.87E+10	3.63E+10	[4, 3, 4, 4, 5]	[4,4,3,4,4]	6.709	32750	20	1.873E+10	3.63E+10	[4, 3, 4, 4, 5]	[4,4,3,4,4]	5106.545	32750	20
example15 × 5	seed9	8	1.2	1.59E+10	3.23E+10	[3, 3, 5, 4, 4]	[3,4,2,3,4]	6.537	29105	19	1.625E+10	3.22E+10	[3, 3, 3, 6, 3]	[3,4,3,2,6]	5119.634	28990	18
example15 × 5	seed10	8	1.2	1.93E+10	3.46E+10	[3, 3, 3, 3, 4]	[3,4,6,6,4]	5.243	31207	16	1.931E+10	3.46E+10	[3, 3, 3, 3, 4]	[3,4,6,6,4]	5065.33	31207	16

Table C.7.: Results for 15x5-configuration at uncapacitated case

The following table presents a comparison between the exact and heuristic procedures. A relevant aspect in the results showed at this table is the CPU time. The data here reported show that at the 90% the heuristic finds the optimal solution and the average improve of the CPU time is 99.9%. This indicates that the heuristic procedure is promising in the sense that in real instances the heuristic is expected to be much faster than the exact algorithm as well as to find the optimal solution for the most cases.

Comparison heuristic and exact							
Configuration	instance	fare	σ	Dif z_{NET}	Dif z_{NET} %	Dif time	Dif time %
example15 × 5	seed1	6	1.1	0	0	5341.762	99.8923426
example15 × 5	seed2	6	1.1	0	0	5076.966	99.89038939
example15 × 5	seed3	6	1.1	0	0	5164.277	99.90035648
example15 × 5	seed4	6	1.1	0	0	5196.399	99.90004995
example15 × 5	seed5	6	1.1	0	0	5367.41	99.89718839
example15 × 5	seed6	6	1.1	0	0	5117.076	99.89637637
example15 × 5	seed7	6	1.1	0	0	5292.013	99.89983612
example15 × 5	seed8	6	1.1	0	0	5226.787	99.89053065
example15 × 5	seed9	6	1.1	0	0	5248.408	99.89392805
example15 × 5	seed10	6	1.1	0	0	5186.261	99.90020114
example15 × 5	seed1	6	1.2	0	0	5361.611	99.89475077
example15 × 5	seed2	6	1.2	0	0	5102.332	99.89062082
example15 × 5	seed3	6	1.2	0	0	5187.354	99.90024139
example15 × 5	seed4	6	1.2	0	0	5201.237	99.89991257
example15 × 5	seed5	6	1.2	0	0	5387.126	99.89699001
example15 × 5	seed6	6	1.2	0	0	5129.645	99.89633821
example15 × 5	seed7	6	1.2	0	0	5317.736	99.89681075
example15 × 5	seed8	6	1.2	0	0	5241.416	99.89669965
example15 × 5	seed9	6	1.2	0	0	5263.107	99.89723871
example15 × 5	seed10	6	1.2	0	0	5224.183	99.9006383
example15 × 5	seed1	8	1.1	0	0	5387.642	99.88979503
example15 × 5	seed2	8	1.1	0	0	5128.883	99.89029151
example15 × 5	seed3	8	1.1	141392624	0.795379	5185.059	99.8962709
example15 × 5	seed4	8	1.1	0	0	5199.145	99.89685875
example15 × 5	seed5	8	1.1	0	0	5377.894	99.89649791
example15 × 5	seed6	8	1.1	0	0	5116.247	99.88727106
example15 × 5	seed7	8	1.1	0	0	5295.475	99.89667894
example15 × 5	seed8	8	1.1	0	0	5228.773	99.87219853
example15 × 5	seed9	8	1.1	0	0	5241.655	99.88520639
example15 × 5	seed10	8	1.1	81521042	0.423464	5191.423	99.89699383
example15 × 5	seed1	8	1.2	0	0	5425.003	99.89082905
example15 × 5	seed2	8	1.2	0	0	5152.066	99.88989374
example15 × 5	seed3	8	1.2	37340298	0.207763	5193.765	99.89317832
example15 × 5	seed4	8	1.2	0	0	5222.779	99.89642695
example15 × 5	seed5	8	1.2	0	0	5404.66	99.89645603
example15 × 5	seed6	8	1.2	0	0	5150.291	99.89617204
example15 × 5	seed7	8	1.2	0	0	5323.995	99.89636962
example15 × 5	seed8	8	1.2	0	0	5099.836	99.86861959
example15 × 5	seed9	8	1.2	334532946	2.058772	5113.097	99.87231509
example15 × 5	seed10	8	1.2	0	0	5060.087	99.89649243

Table C.8.: Comparison exact and heuristic for 15 × 5-configuration at uncapacitated case

20 × 6-configuration

In this section, we have tested our algorithms on networks with 20 nodes and 6 lines. The results of these experiments are shown in Table C.9. Due to the spent time in the exact algorithm, we have only solved ten instances for this configuration. It can be observed that our heuristic was able to find the optimal solution for all instances (100%)

and the average improve of the CPU time is $99,98\%$. The maximum profit is searched for the Seed-2 instance as well as the maximum revenue.

				Heuristic							Exact						
Configuration	instance	fare	σ	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains	z_{NET}	z_{REV}	ψ_ℓ	δ_ℓ	CPU time	nb trips	nb Trains
example20 × 6	seed1	6	1.1	1.37E+10	3.58E+10	[3, 3, 3, 3, 3, 3]	[8,4,3,6,2,3]	23.572	32241	18	1.37E+10	3.58E+10	[3, 3, 3, 3, 3, 3]	[8,4,3,6,2,3]	155319.5	32241	18
example20 × 6	seed2	6	1.1	1.65E+10	3.93E+10	[3, 4, 4, 3, 3, 3]	[8,3,3,7,2,2]	23.683	35456	20	1.65E+10	3.93E+10	[3, 4, 4, 3, 3, 3]	[8,3,3,7,2,2]	152070	35456	20
example20 × 6	seed3	6	1.1	1.62E+10	3.74E+10	[3, 3, 3, 3, 3, 3]	[7,5,3,6,2,2]	22.061	33742	18	1.62E+10	3.74E+10	[3, 3, 3, 3, 3, 3]	[7,5,3,6,2,2]	152330.7	33742	18
example20 × 6	seed4	6	1.1	1.67E+10	3.85E+10	[3, 3, 3, 3, 3, 3]	[9,5,4,6,2,3]	22.185	34662	18	1.67E+10	3.85E+10	[3, 3, 3, 3, 3, 3]	[9,5,4,6,2,3]	141784	34662	18
example20 × 6	seed5	6	1.1	1.24E+10	3.42E+10	[3, 3, 3, 3, 3, 3]	[7,4,4,6,2,3]	22.247	30843	18	1.24E+10	3.42E+10	[3, 3, 3, 3, 3, 3]	[7,4,4,6,2,3]	154176.8	30843	18
example20 × 6	seed6	6	1.1	1.5E+10	3.87E+10	[3, 4, 5, 3, 3, 3]	[7,4,2,5,2,3]	24.65	34868	21	1.5E+10	3.87E+10	[3, 4, 5, 3, 3, 3]	[7,4,2,5,2,3]	154160.7	34868	21
example20 × 6	seed7	6	1.1	1.5E+10	3.68E+10	[3, 3, 3, 3, 3, 3]	[8,4,3,6,2,2]	21.966	33186	18	1.5E+10	3.68E+10	[3, 3, 3, 3, 3, 3]	[8,4,3,6,2,2]	152161	33186	18
example20 × 6	seed8	6	1.1	1.35E+10	3.39E+10	[3, 3, 3, 3, 3, 3]	[6,3,3,5,2,3]	22.482	30540	18	1.35E+10	3.39E+10	[3, 3, 3, 3, 3, 3]	[6,3,3,5,2,3]	156696.6	30540	18
example20 × 6	seed9	6	1.1	8.39E+09	3.06E+10	[3, 3, 4, 3, 3, 3]	[5,2,2,4,2,3]	23.168	27536	19	8.39E+09	3.06E+10	[3, 3, 4, 3, 3, 3]	[5,2,2,4,2,3]	157478.7	27536	19
example20 × 6	seed10	6	1.1	1.58E+10	3.75E+10	[3, 3, 3, 3, 3, 3]	[8,4,4,6,2,2]	22.232	33811	18	1.58E+10	3.75E+10	[3, 3, 3, 3, 3, 3]	[8,4,4,6,2,2]	154221.4	33811	18

Table C.9.: Results for 20×6-configuration at uncapacitated case.

Comparison heuristic and exact						
Configuration	instance	fare	σ	Dif z_{NET}	Dif time	Dif time %
example20 × 6	seed1	6	1,1	0,00E+00	155295,9	99,98482
example20 × 6	seed2	6	1,1	0,00E+00	152046,3	99,98443
example20 × 6	seed3	6	1,1	0,00E+00	152308,6	99,98552
example20 × 6	seed4	6	1,1	0,00E+00	141761,8	99,98435
example20 × 6	seed5	6	1,1	0,00E+00	154154,5	99,98557
example20 × 6	seed6	6	1,1	0,00E+00	154136	99,98401
example20 × 6	seed7	6	1,1	0,00E+00	152139,1	99,98556
example20 × 6	seed8	6	1,1	0,00E+00	156674,1	99,98565
example20 × 6	seed9	6	1,1	0,00E+00	157455,6	99,98529
example20 × 6	seed10	6	1,1	0,00E+00	154199,2	99,98558

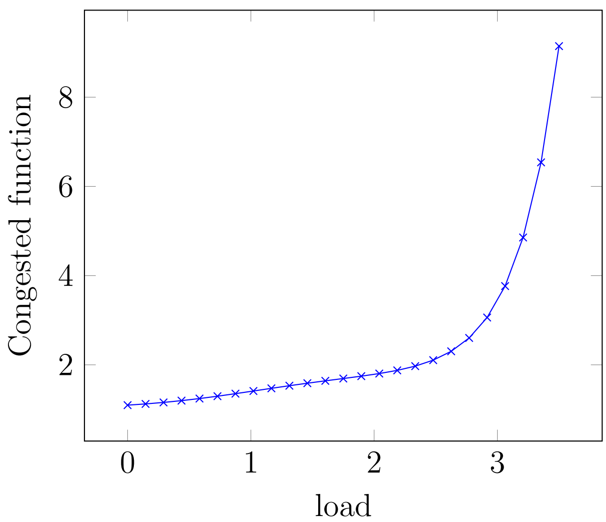
Table C.10.: Comparison exact and heuristic for 20 × 6-configuration at uncapacitated case.

C.0.2. Computational experiments for the capacitated problem

In this section we will analyze the capacitated version of our problem. To evaluate the performance of our algorithm, we have adapted the crowding function defined in Section 3.4 to our problem. Concretely, the crowding penalty was mathematically defined for the nominal capacity as

$$CF(x) = 1 + \frac{0.8}{1 + \exp(2 * (1 - x))} + 0.01 \exp(3 * (x - 1.3)). \tag{C.1}$$

The following figures show a representation of the crowding functions above defined.



load	penalty
1.08	1.44
1.05	1.42
1.01	1.41

Figure C.5.: In-vehicle crowding function.

In order to evaluate the impact of in-vehicle crowding on the solution of our problem, we have gradually increased the maximum number of carriages in our experimentation. Detailed information on these solutions are shown in Table C.11, C.12, C.15 and C.16. The data here reported corresponds to the optimal solution for each value of δ^{max} in which the problem is feasible. Moreover, we have analyzed the solutions obtained at the uncapacitated case when the in-vehicle crowding is introduced. At this way, we have measured the in-vehicle crowding effects on these solutions. The parameter σ considered here was fixed to 1.1, which implies that if the number of passengers of each line is higher than capacity line over 10%, the solution is taken into account. The seventh column in Table C.11, C.12, C.15 and C.16 represents the number of iterations in the Loop III. A total of 200 experiments of the 6×2, 7×3 and 8×3-configuration were tested. For 6×2 configuration our algorithm was able to obtain the optimal solution in a very small CPU time. However, on instances with 15 nodes and 5 lines the algorithm took too time. It is difficult if not imposible to make experiments with real instances, which indicates the necessity of using heuristic algorithm to solve the problem.

6×2-configuration

The data reported in Table C.11 represent the solutions obtained in the capacitated case. We have analyzed the solutions when the parameter δ^{max} is less than or equal to 8. It can be seen that for $\delta^{max} \leq 4$, the problem is always infeasible. For $\delta^{max} > 4$, the most cases are feasible and the optimal solutions are not affected by congestion (see the seventh column). This fact indicates the maximum number of carriages is a sufficient number in order to transport all passengers willing to use the RTS. It can be observed that the average of all CPU time is 6.15 seconds when $\delta^{max} = 8$.

7×3-configuration

In Table C.12 it can be seen that for $\delta^{max} \leq 2$, the optimal solutions have high frequencies in order to transport all passengers. From these results it can be observed that the number of trains decreases when the maximum number of carriages increases. The profit starts to be economically interesting when the number of carriages is greater than two for seed1, seed3, seed4, seed6 and seed9-instances and it is greater than three for rest of instances. The most cases, the optimal solution corresponds to a non-congested network. It is interesting to observe that for $\delta^{max} = 6$ only two solutions (seed-8 and seed9-instance) are the same than unlimited capacity. This fact indicates the in-vehicle crowding directly affects to the solutions. Indeed, the optimal solutions for uncapacitated case are affected

Configuration 6 × 2, $\sigma = 1.1$, fare=6										
instance	$\delta^{\max}$	profit	revenue	ψ_ℓ	δ_ℓ	n° iterations	CPU time	nb trips	% demand	nb trains
seed1	8	9.06E+09	1.3E+10	[3, 3]	[8, 5]	0	5.344	15675	77.46	6
seed2	7	8.22E+09	1.23E+10	[3, 3]	[7, 5]	0	4.156	14799	72.11	6
seed2	8	8.22E+09	1.23E+10	[3, 3]	[7, 5]	0	5.593	14799	72.11	6
seed3	5	8.03E+09	1.19E+10	[3, 3]	[5, 5]	0	2.078	14259	65.22	6
seed3	6	8.03E+09	1.19E+10	[3, 3]	[5, 5]	0	3.25	14259	65.22	6
seed3	7	8.03E+09	1.19E+10	[3, 3]	[5, 5]	0	4.625	14259	65.22	6
seed3	8	8.03E+09	1.19E+10	[3, 3]	[5, 5]	0	6.157	14259	65.22	6
seed4	5	7.21E+09	1.1E+10	[3, 3]	[5, 3]	0	2.187	13226	63.29	6
seed4	6	7.21E+09	1.1E+10	[3, 3]	[5, 3]	0	3.469	13226	63.29	6
seed4	7	7.21E+09	1.1E+10	[3, 3]	[5, 3]	0	4.86	13226	63.29	6
seed4	8	7.21E+09	1.1E+10	[3, 3]	[5, 3]	0	6.625	13226	63.29	6
seed5	6	7.49E+09	1.18E+10	[4, 4]	[5, 4]	0	3.125	14221	70.85	8
seed5	7	7.49E+09	1.18E+10	[4, 4]	[5, 4]	0	4.468	14221	70.85	8
seed5	8	7.49E+09	1.18E+10	[4, 4]	[5, 4]	0	6.093	14221	70.85	8
seed6	6	8.16E+09	1.35E+10	[4, 5]	[5, 3]	0	3.172	16268	79.15	9
seed6	7	8.16E+09	1.35E+10	[4, 5]	[5, 3]	0	4.5	16268	79.15	9
seed6	8	8.16E+09	1.35E+10	[4, 5]	[5, 3]	0	6.157	16268	79.15	9
seed7	5	6.84E+09	1.22E+10	[4, 5]	[4, 3]	0	2.14	14701	69.1	9
seed7	6	6.84E+09	1.22E+10	[4, 5]	[4, 3]	0	3.328	14701	69.1	9
seed7	7	6.84E+09	1.22E+10	[4, 5]	[4, 3]	0	4.735	14701	69.1	9
seed7	8	6.84E+09	1.22E+10	[4, 5]	[4, 3]	0	6.328	14701	69.1	9
seed8	5	6.03E+09	9.83E+09	[3, 3]	[5, 3]	0	2.172	11811	58.4	6
seed8	6	6.03E+09	9.83E+09	[3, 3]	[5, 3]	0	3.375	11811	58.4	6
seed8	7	6.03E+09	9.83E+09	[3, 3]	[5, 3]	0	4.765	11811	58.4	6
seed8	8	6.03E+09	9.83E+09	[3, 3]	[5, 3]	0	6.406	11811	58.4	6
seed9	5	6.6E+09	1.02E+10	[3, 3]	[5, 3]	0	2.14	12204	58.4	6
seed9	6	6.6E+09	1.02E+10	[3, 3]	[5, 3]	0	3.297	12204	58.4	6
seed9	7	6.6E+09	1.02E+10	[3, 3]	[5, 3]	0	4.703	12204	58.4	6
seed9	8	6.6E+09	1.02E+10	[3, 3]	[5, 3]	0	6.391	12204	58.4	6
seed10	5	7.41E+09	1.15E+10	[3, 3]	[5, 4]	0	2.11	13831	66.5	6
seed10	6	7.41E+09	1.15E+10	[3, 3]	[5, 4]	0	3.281	13831	66.5	6
seed10	7	7.41E+09	1.15E+10	[3, 3]	[5, 4]	0	4.688	13831	66.5	6
seed10	8	7.41E+09	1.15E+10	[3, 3]	[5, 4]	0	6.453	13831	66.5	6

Table C.11.: Results for 6 × 2-configuration at capacitated case.

by the in-vehicle crowding when the congestion is introduced in our problem as show Tables C.13 and C.14. For instance, it can be observed that the optimal solution for seed1-configuration at the capacitated case has one more carriage than the uncapacitated case. In other words, when the congestion is taken into account, the passenger's behavior changes, and it is more economically interesting to add a carriage than to lose passengers.

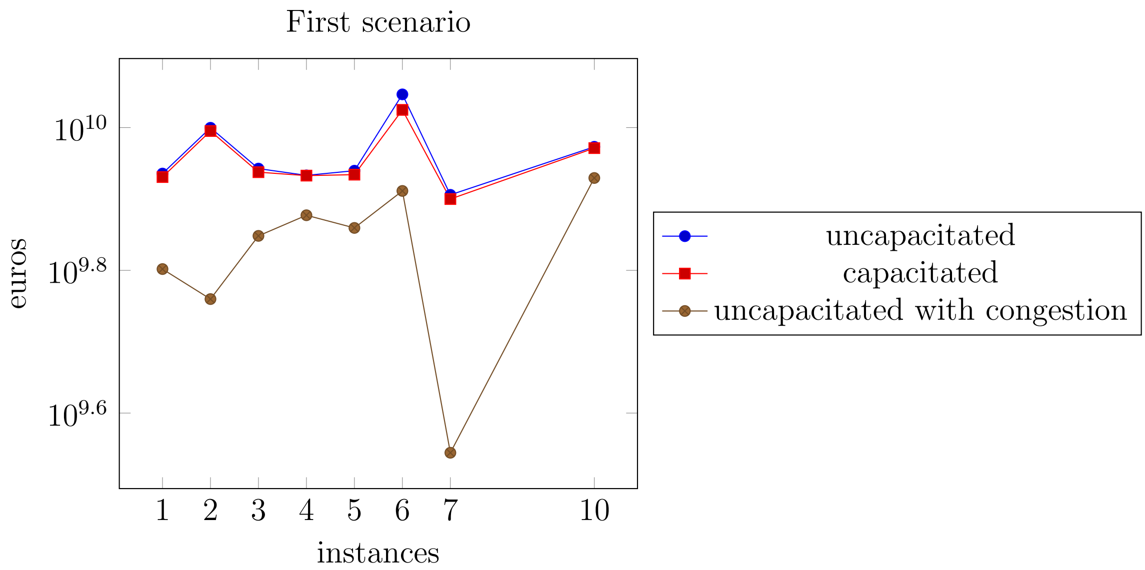


Figure C.6.: Profit for 7×3-configuration. Optimal solution for uncapacitated problem, capacitated problem and the uncongestion optimal solution with the congestion effect.

deltaMax=5				
	profit	revenue	passengers	nb iterations
seed5	7.23E+09	1.46E+10	17525	1
seed7	3.50E+09	1.09E+10	13075	2

Table C.13.: Results for uncapacitated optimal solutions of 7×3-configuration with crowding and $\delta^{max} = 5$.

Configuration 7×3, sigma 1.1, fare=6									
instance	$\delta^{\max}$	profit	revenue	ψ_ℓ	δ_ℓ	nº iterations	CPU time	nb trips	nb trains
seed1	2	-2.11E+09	1.93E+10	[10, 10, 10]	[2, 2, 2]	0	10.688	23221	30
seed1	3	3.44E+09	1.93E+10	[6, 6, 10]	[3, 3, 2]	0	41.562	23208	22
seed1	4	7.86E+09	1.88E+10	[4, 5, 5]	[4, 4, 4]	0	124.797	22606	14
seed1	5	7.86E+09	1.88E+10	[4, 5, 5]	[4, 4, 4]	0	242.328	22606	14
seed1	6	8.52E+09	1.88E+10	[3, 5, 5]	[6, 4, 4]	0	289.381	22606	
seed2	1	-2.05E+10	1.69E+10	[20, 20, 15]	[1, 1, 1]	2	1.047	20348	55
seed2	2	-3.68E+09	1.77E+10	[10, 10, 10]	[2, 2, 2]	0	10.906	21304	30
seed2	3	4.18E+09	1.77E+10	[6, 6, 6]	[3, 3, 3]	0	43.265	21302	18
seed2	4	6.55E+09	1.77E+10	[5, 4, 5]	[4, 4, 4]	0	123.516	21302	14
seed2	5	7.89E+09	1.77E+10	[4, 4, 4]	[5, 4, 5]	0	259.11	21302	12
seed2	6	9.89E+09	1.77E+10	[3, 3, 3]	[6, 6, 6]	0	300.52	21302	
seed3	2	-2.43E+09	1.77E+10	[10, 10, 10]	[2, 2, 2]	0	10.984	21258	30
seed3	3	4.90E+09	1.64E+10	[6, 5, 5]	[3, 3, 3]	0	43.25	19656	16
seed3	4	7.37E+09	1.63E+10	[4, 4, 4]	[4, 4, 4]	0	125.781	19590	12
seed3	5	8.03E+09	1.57E+10	[4, 3, 3]	[4, 4, 5]	0	263.515	18898	10
seed3	6	8.66E+09	1.57E+10	[3, 3, 3]	[6, 4, 5]	0	299.444	18898	
seed4	2	-3.14E+09	1.81E+10	[10, 12, 10]	[2, 2, 2]	0	10.36	21803	32
seed4	3	2.83E+09	1.80E+10	[6, 10, 6]	[3, 3, 3]	0	40.359	21622	22
seed4	4	7.90E+09	1.75E+10	[4, 5, 4]	[4, 4, 4]	0	111.406	21011	14
seed4	5	7.90E+09	1.75E+10	[4, 5, 4]	[4, 4, 4]	0	239.375	21011	14
seed4	6	8.56E+09	1.75E+10	[3, 5, 4]	[6, 4, 4]	0	271.957	21011	
seed5	1	-1.75E+10	1.81E+10	[15, 20, 20]	[1, 1, 1]	1	1.187	21697	55
seed5	2	-2.91E+09	1.81E+10	[10, 10, 10]	[2, 2, 2]	0	11.734	21767	30
seed5	3	4.61E+09	1.64E+10	[5, 5, 6]	[3, 3, 3]	0	47.063	19719	16
seed5	4	6.72E+09	1.61E+10	[4, 4, 4]	[4, 4, 4]	0	127.765	19299	12
seed5	5	8.07E+09	1.61E+10	[3, 3, 4]	[5, 5, 4]	0	284.297	19291	10
seed5	6	8.59E+09	1.61E+10	[3, 3, 3]	[5, 5, 6]	0	313.92	19287	
seed6	2	-2.50E+09	1.92E+10	[10, 10, 10]	[2, 2, 2]	0	10.547	23056	30
seed6	3	3.07E+09	1.92E+10	[6, 6, 10]	[3, 3, 2]	0	40.922	23056	22
seed6	4	7.92E+09	1.92E+10	[5, 4, 5]	[4, 4, 4]	0	110.953	23056	14
seed6	5	9.23E+09	1.92E+10	[4, 4, 4]	[5, 4, 5]	0	237.844	23056	12
seed6	6	1.06E+10	1.92E+10	[3, 3, 4]	[6, 6, 5]	0	270.085	23046	
seed7	1	-1.96E+10	1.69E+10	[15, 20, 20]	[1, 1, 1]	1	1.172	20255	55
seed7	2	-4.11E+09	1.73E+10	[10, 10, 10]	[2, 2, 2]	0	11.578	20786	30
seed7	3	4.47E+09	1.71E+10	[5, 6, 6]	[3, 3, 3]	0	44.907	20582	17
seed7	4	6.52E+09	1.60E+10	[4, 4, 4]	[4, 4, 4]	0	123.547	19267	12
seed7	5	7.94E+09	1.54E+10	[3, 3, 3]	[5, 5, 5]	0	275.75	18537	9
seed7	6	7.94E+09	1.54E+10	[3, 3, 3]	[5, 5, 5]	0	299.318	18537	
seed8	1	-1.46E+10	1.85E+10	[15, 15, 20]	[1, 1, 1]	0	1.203	22286	50
seed8	2	-2.77E+09	1.85E+10	[10, 10, 10]	[2, 2, 2]	0	11.625	22241	30
seed8	3	5.27E+09	1.78E+10	[5, 6, 6]	[3, 3, 3]	1	46.296	21399	17
seed8	4	7.58E+09	1.82E+10	[4, 5, 5]	[4, 3, 4]	0	126.765	21923	14
seed8	5	8.91E+09	1.62E+10	[3, 3, 3]	[5, 4, 5]	0	281.672	19520	9
seed8	6	8.91E+09	1.62E+10	[3, 3, 3]	[5, 4, 5]	0	307.259	19520	
seed9	2	-4.64E+09	1.79E+10	[10, 12, 10]	[2, 2, 2]	0	11.172	21563	32
seed9	3	2.50E+09	1.79E+10	[5, 10, 6]	[3, 3, 3]	0	42.766	21532	21
seed9	4	6.73E+09	1.76E+10	[4, 5, 5]	[4, 4, 4]	0	116.094	21170	14
seed9	5	8.36E+09	1.58E+10	[3, 3, 3]	[5, 5, 5]	0	260.547	18972	9
seed9	6	8.36E+09	1.58E+10	[3, 3, 3]	[5, 5, 5]	0	282.689	18972	
seed10	1	-2.08E+10	1.81E+10	[20, 15, 20]	[1, 1, 1]	1	1.078	21741	55
seed10	2	-3.96E+09	1.82E+10	[10, 10, 10]	[2, 2, 2]	1	11.218	21927	30
seed10	3	4.27E+09	1.77E+10	[6, 5, 6]	[3, 3, 3]	1	43.266	21210	17
seed10	4	7.53E+09	1.77E+10	[5, 3, 4]	[4, 4, 4]	0	118.922	21329	12
seed10	5	8.39E+09	1.77E+10	[4, 3, 4]	[5, 4, 4]	0	264.172	21329	11
seed10	6	9.36E+09	1.77E+10	[3, 3, 4]	[6, 4, 4]	0	297.072	21329	

Table C.12.: Results for 7×3-configuration at capacitated case.

deltaMax=6				
	profit	revenue	passengers	nb iterations
seed1	6.33E+09	1.57E+10	18918	1
seed2	5.75E+09	1.35E+10	16205	2
seed3	7.05E+09	1.42E+10	17086	2
seed4	7.54E+09	1.65E+10	19769	1
seed5	7.23E+09	1.46E+10	17525	1
seed6	8.15E+09	1.61E+10	19339	1
seed7	3.50E+09	1.09E+10	13075	2
seed10	8.50E+09	1.63E+10	19605	1

Table C.14.: Results for uncapacitated optimal solutions of 7×3-configuration with crowding and $\delta^{max} = 6$.

8×3-configuration

Detailed information on the results for 8×3-configuration is reported in Tables C.15 and C.16. The results provided in this table reveal, for most cases, the system becomes productive from three carriages. As observed in 7×3-configuration, the frequencies are highs when the capacities are small, in order to transport all passenger willing to travel on the RTS. The average CPU time for $\delta^{max} = 1$ is 1.36 seconds whereas for $\delta^{max} = 6$ is 823. The optimal solutions for uncapacitated case are affected by the in-vehicle crowding at the capacitated case as show Tables C.17 and C.18.

Configuration 8 × 3, sigma 1.1, fare 6									
instance	$\delta^{\max}$	profit	revenue	ψ_ℓ	δ_ℓ	n° iterations	CPU time	nb trips	nb Trains
seed1	1	-8.74E+09	1.40E+10	[10, 15, 10]	[1, 1, 1]	0	1.69	16788	25
seed1	2	1.64E+09	1.14E+10	[3, 6, 6]	[2, 2, 2]	0	16.38	13735	15
seed1	3	3.89E+09	1.14E+10	[3, 4, 4]	[2, 3, 3]	0	63.96	13734	11
seed1	4	5.01E+09	1.14E+10	[3, 3, 3]	[2, 4, 4]	0	171.90	13724	9
seed1	5	5.01E+09	1.14E+10	[3, 3, 3]	[2, 4, 4]	0	564.78	13724	9
seed1	6	5.01E+09	1.14E+10	[3, 3, 3]	[2, 4, 4]	0	1011.83	13724	9
seed2	1	-7.65E+09	1.16E+10	[3, 15, 12]	[1, 1, 1]	0	1.90	13936	30
seed2	2	-1.72E+09	1.16E+10	[3, 10, 6]	[1, 2, 2]	0	17.82	13927	19
seed2	3	3.29E+09	1.16E+10	[3, 5, 3]	[3, 3, 3]	0	66.50	13922	11
seed2	4	3.99E+09	1.16E+10	[3, 4, 3]	[3, 4, 3]	0	173.29	13922	10
seed2	5	4.78E+09	1.16E+10	[3, 3, 3]	[3, 5, 3]	0	570.36	13922	9
seed2	6	4.78E+09	1.16E+10	[3, 3, 3]	[3, 5, 3]	0	1018.70	13922	9
seed3	1	-1.06E+10	1.64E+10	[15, 15, 20]	[1, 1, 1]	0	1.25	19661	50
seed3	2	-1.91E+09	1.56E+10	[10, 10, 10]	[2, 2, 2]	0	11.41	18758	30
seed3	3	6.04E+09	1.46E+10	[4, 5, 5]	[3, 3, 3]	1	48.61	17553	14
seed3	4	6.96E+09	1.46E+10	[4, 4, 4]	[3, 4, 4]	0	137.19	17552	12
seed3	5	7.99E+09	1.46E+10	[4, 3, 3]	[3, 5, 5]	0	468.77	17521	10
seed3	6	7.99E+09	1.46E+10	[4, 3, 3]	[3, 5, 5]	0	863.56	17521	10
seed4	1	-1.17E+10	1.79E+10	[20, 15, 20]	[1, 1, 1]	0	1.08	21458	55
seed4	2	1.26E+09	1.36E+10	[5, 10, 4]	[2, 2, 2]	0	12.97	16312	119
seed4	3	6.37E+09	1.67E+10	[6, 5, 6]	[3, 3, 3]	0	42.84	20037	17
seed4	4	7.53E+09	1.58E+10	[5, 4, 4]	[3, 4, 4]	0	125.29	18944	14
seed4	5	8.26E+09	1.48E+10	[4, 3, 3]	[3, 5, 5]	0	439.73	17756	10
seed4	6	8.26E+09	1.48E+10	[4, 3, 3]	[3, 5, 5]	0	820.11	17756	10
seed5	1	-8.35E+09	1.56E+10	[15, 15, 12]	[1, 1, 1]	0	1.31	18688	42
seed5	2	-2.10E+08	1.73E+10	[10, 10, 10]	[2, 2, 2]	0	10.77	20751	30
seed5	3	5.90E+09	1.31E+10	[3, 5, 3]	[3, 3, 3]	0	54.10	15778	11
seed5	4	6.96E+09	1.27E+10	[3, 3, 3]	[3, 4, 3]	0	151.52	15229	9
seed5	5	6.96E+09	1.27E+10	[3, 3, 3]	[3, 4, 3]	0	513.94	15229	9
seed5	6	6.96E+09	1.27E+10	[3, 3, 3]	[3, 4, 3]	0	946.48	15229	9

Table C.15.: Results for 8 × 3-configuration at capacitated case for seed1 to seed5-instances.

Configuration 8 × 3, sigma 1.1, fare 6									
instance	$\delta^{\max}$	profit	revenue	ψ_ℓ	δ_ℓ	n^o iterations	CPU time	nb trips	nb Trains
seed6	1	-1.57E+10	1.56E+10	[15, 20, 15]	[1, 1, 1]	0	1.171	18785	50
seed6	2	-2.80E+09	1.49E+10	[10, 10, 6]	[2, 2, 2]	1	11.374	17850	26
seed6	3	4.66E+09	1.45E+10	[4, 6, 4]	[3, 3, 3]	0	47.051	17459	14
seed6	4	5.91E+09	1.40E+10	[3, 5, 3]	[3, 4, 4]	0	130.277	16798	11
seed6	5	6.56E+09	1.40E+10	[3, 4, 3]	[3, 5, 4]	0	449.984	16795	10
seed6	6	7.25E+09	1.39E+10	[3, 3, 3]	[3, 6, 4]	0	841.11	16749	9
seed7	1	-8.16E+09	1.28E+10	[4, 15, 15]	[1, 1, 1]	0	1.919	15361	34
seed7	2	-1.29E+09	1.28E+10	[5, 10, 5]	[2, 2, 2]	0	17.8	15333	20
seed7	3	4.01E+09	1.21E+10	[3, 5, 3]	[3, 3, 3]	1	66.675	14514	11
seed7	4	5.32E+09	1.27E+10	[3, 4, 3]	[2, 4, 4]	0	172.147	15258	10
seed7	5	6.05E+09	1.27E+10	[3, 3, 3]	[2, 5, 4]	0	573.578	15257	9
seed7	6	6.05E+09	1.27E+10	[3, 3, 3]	[2, 5, 4]	0	1045.719	15257	9
seed8	1	-1.34E+10	1.49E+10	[12, 15, 20]	[1, 1, 1]	0	1.125	17940	47
seed8	2	-2.32E+09	1.48E+10	[6, 10, 10]	[2, 2, 2]	0	11.311	17725	26
seed8	3	4.09E+09	1.44E+10	[5, 5, 5]	[3, 3, 3]	0	47.457	17362	15
seed8	4	4.97E+09	1.42E+10	[5, 4, 4]	[3, 4, 4]	0	133.303	17105	13
seed8	5	5.77E+09	1.31E+10	[4, 3, 3]	[3, 5, 5]	0	466.813	15742	10
seed8	6	5.77E+09	1.31E+10	[4, 3, 3]	[3, 5, 5]	0	862.828	15742	10
seed9	1	-1.29E+10	1.77E+10	[20, 20, 15]	[1, 1, 1]	0	1.077	21248	55
seed9	2	-1.60E+08	1.71E+10	[10, 10, 10]	[2, 2, 2]	0	10.75	20498	30
seed9	3	7.73E+09	1.68E+10	[5, 5, 5]	[3, 3, 3]	0	42.355	20202	15
seed9	4	8.80E+09	1.60E+10	[4, 4, 3]	[4, 4, 4]	0	120.277	19229	11
seed9	5	9.95E+09	1.59E+10	[3, 3, 3]	[5, 5, 4]	0	430.297	19115	9
seed9	6	9.95E+09	1.59E+10	[3, 3, 3]	[5, 5, 4]	0	815	19115	9
seed10	1	-1.77E+10	1.72E+10	[20, 20, 15]	[1, 1, 1]	0	1.109	20612	55
seed10	2	-2.43E+09	1.71E+10	[10, 10, 10]	[2, 2, 2]	0	10.921	20566	30
seed10	3	5.16E+09	1.58E+10	[5, 6, 4]	[3, 3, 3]	0	43.197	19033	15
seed10	4	6.61E+09	1.48E+10	[4, 4, 3]	[4, 4, 4]	0	124.801	17776	11
seed10	5	6.97E+09	1.45E+10	[3, 4, 3]	[5, 4, 4]	0	438.688	17396	10
seed10	6	7.58E+09	1.45E+10	[3, 3, 3]	[5, 6, 4]	0	866.312	17388	9

Table C.16.: Results for 8 × 3-configuration at capacitated case for seed6 to seed10-instances.

deltaMax=5				
	profit	revenue	passengers	nb iterations
seed1(delta=4 also)	4.06E+09	1.04E+10	12497	1
seed8	5.17E+09	1.24E+10	14921	1

Table C.17.: Results for 8 × 3-configuration at capacitated case.

deltaMax=6				
	profit	revenue	passengers	nb iterations
seed1	4.06E+09	1.04E+10	12497	1
seed6	5.31E+09	1.19E+10	14317	1
seed8	5.17E+09	1.24E+10	14921	1

Table C.18.: Results for 8×3-configuration at capacitated case.

Chapter 4

An adaptive neighborhood search heuristic for metro network design

4.1. Introduction

In this chapter we will focus on the network design and line planning taking into account aspects related to rolling stock and personnel costs. Concretely, we will analyze a particular case from the general model proposed in Chapter 2, which integrates the stages in the railway process above commented. The problem consists of maximizing the net profit of a RTS by selecting the location of stations and their connections, a set of lines, each characterized by two different terminal stations, a sequence of intermediate stations (an itinerary), frequencies of each line and the size of trains, assuming that all passengers willing to travel in the RTS can be transported. We assume the existence of an alternative transportation system (e.g. private car, bus, bicycle) competing with the RTS as well as passengers choose their routes and their transport mode according to traveling times. As mentioned in Chapter 2, the travel time is composed of several terms: waiting time, in vehicle time and transfer time. Each term is depending of the system's characteristics such as the topology, the line configuration as well as the considered frequencies. The demand is supposed to be elastic and changes accordingly to the characteristics. This aspect is included in the model by means of a logit function. Moreover, we will concentrate on the effective resolution of this problem for small and medium instances. Due to the complexity of the problem that we are proposing, a heuristic procedure is needed to solve the problem for instances of large size. Specifically, we propose a new method based on the Adaptive Large Neighborhood Search Heuristic (ALNS) which provides a powerful algorithmic framework capable of simultaneously handling the network design, line planning, rolling stock and personnel planning. This algorithm is an iterative procedure that combines the network design problem and the line planning problem described in Chapter 3. Concretely, at each iteration it defines a possible infrastructure and line configuration, that is, a Rapid Transit Line (RTL), and the HLSA heuristic is applied on this network in order to evaluate the RTS built. Figure 4.1 shows a flow chart for this procedure.

The main contributions of this chapter are the introduction of a mathematical programming programm to solve small instances of our problem as well as the development of a powerful ALNS heuristic to solve real instances.

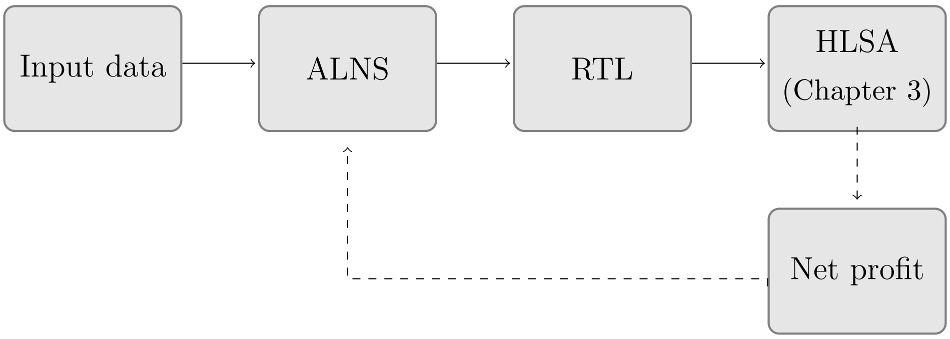


Figure 4.1.: Flow chart for the ALNS and the HLSA algorithms.

The remainder of this chapter is organized as follows. In next section we will review the different techniques used for solving the rapid transit network design problem. In Section 4.2 we propose a mathematical programming program for the problem that integrates the rapid transit network design problem and line planing problem. Some techniques for improving the efficiency of our model are presented in Section 4.2.4. Our ALNS algorithm is presented in Section 4.3. Computational experiments are carried out on Section 4.4. The chapter ends with some conclusions.

4.1.1. Literature review on algorithms for RTND

Last decades, mixed-integer models for the rapid transit network design (RTND) have been proposed. First studies are concentrated on the problem of designing a single rapid transit alignment. Gendreau et al. (1995) described the main criteria used to design rapid transit alignments, Dufourd et al. (1996) proposed a tabu search algorithm for solving this problem, Bruno et al. (2002) presented a heuristic for solving the problem of designing a rapid transit line maximizing the population coverage and Laporte et al. (2005) presented several heuristics to solve the same problem according to the trip coverage.

In recent years, most of the papers have been devoted to the inclusion of heuristic techniques for solving the transit network design problem. This fact is motivated by the complexity of this problem. Moreover, the exact algorithms can only deal with small networks as well as simplified demand data and there are not exists a technology available to solve real instances in an exact manner, as stated García-Archilla et al. (2013). A review of different mathematical models and heuristics for the rapid transit network is presented in Laporte et al. (2011).

Obviously, when several alignments are considered at the same time (i.e. a network), enormous difficulty is added to the previous problem. This is the reason for organizing the literature review on algorithms into two parts. First, we discuss the problem associated with a single alignment whereas the second part is concentrated on network design problem for rapid transit. Finally, this section ends with a summary table containing the main characteristics of each analyzed paper.

1. Locating of a single rapid transit alignment

   Gendreau et al. (1995) described the main criteria used to design rapid transit alignments. The authors considered the problem of locating a single line maximizing the population coverage. To solve the problem, they applied several techniques such as a greedy criterion and a tabu search algorithm. The results show that these tools can help the design process. Dufourd et al. (1996) proposed a tabu search algorithm for solving the problem of locating a metro or a rapid transit line maximizing the total population covered. The authors represent a city as a grid where each node is a potential station with an associated demand. An alignment is a sequence of stations or equivalently a sequence of positions in the plane. The problem consists of selecting a set of stations from this grid (an alignment) respecting an inter-station spacing, and maximizing the population coverage. Tabu search starts with an initial solution and iteratively modifies it in order to analyze its neighborhood. The initial solution is obtained by locating stations over one of two main diagonals of a square grid. The modifications are done by means of movements on the current solution. The movements are applied on the coordinates of each station. So, given an alignment, another alignment is its neighbor if the latter can be obtained by applying one movement to only one station. For the sake of clarify, let $s_i = (x, y)$ be the cartesian coordinates of a station. The possible movement of this solution is formed by $(x-1, y), (x+1, y), (x, y-1)$ and $(x, y+1)$.

Bruno et al. (1998) proposed a bicriterion model for the location of a rapid transit line. The objectives considered are construction cost and passenger travel time. In order to describe the model in a realistic way, the authors consider several transportation systems: pedestrian, public, private and bi-modal pedestrian-public system, which are described as follows. In fact, they assume each trip starts and ends with centroid nodes. The public system describes links between potential stations. They assume associated a public node, there exists an only pedestrian node. The private network contains links between nodes which can be accessed by the private mode. They introduce boarding arcs that connect centroid of transportation areas with stations and alighting arcs which connects transit stops to centroid nodes. From these systems, they define an alignment as a path connecting two public nodes on the public network and a bi-modal path associated to a demand as a passenger route on the bi-modal system between two nodes. They assume the demand patterns are known and there exist a private transportation competing with the hybrid pedestrian-public transportation system. The passengers select the option with least travel time between their origin and destination stations. The authors formulated a bicriterion integer linear model. In order to estimate the set of non-inferior solutions, they developed an algorithm inspired on the defined by Current et al. (1985). The procedure is divided into several steps. First k shortest paths are defined regarding cost criterion. From each path obtained, a bimodal network is built adding on the pedestrian system the arcs and nodes that this path contains as well as the corresponding boarding and alighting arcs. Once time the shortest path on the bi-modal system is defined and the associated travel time is computed, this is compared to its corresponding path on the private transportation. Finally, the dominated paths are deleted. They tested the example introduced by Current et al. (1985). This network is formed by 21 nodes and 39 arcs. A total of 250 paths were considered.

Bruno et al. (2002) developed a two-phase heuristic for the problem of constructing an alignment in an urban context maximizing the population coverage. To this end, a discretized grid network was defined. Each integer coordinate has an associated population. Moreover, each station captures the closest population to the considered station. A minimum and maximum space between each consecutive stations were imposed. So, the problem consists of locating a given number of stations n taking into account the inter-station spacing and maximizing the population coverage. The heuristic proposed is described into two phases: initial solution construction and an improvement phase. The procedure developed in the first phase is as follows. At each iteration, a partial alignment (initially formed by two stations) is gradually extended by adding a new station, maximizing the coverage captured. To this end, a subset of stations linking the partial alignment and the new station is selected. If the inter-station spacing is satisfied, the new alignment is considered. So, this algorithm extends iteratively the partial alignment until an alignment with n-stations. In the second phase, an algorithm is applied in order to generate and analyze the neighborhood of the current solution. The neighborhood is constructed by selecting from the current solution, a set of consecutive stations with a determined size, and then it is extended as in the initial phase. The procedure stops when all possible sets of consecutive stations are analyzed. This heuristic was tested on instances randomly generated as well as on real data from the city of Milan.

Laporte et al. (2005) presented several heuristics for the construction of a rapid transit alignment maximizing trip coverage. Based on the model of Laporte et al. (2002) for the station catchment area, the authors were able to define the trip coverage for each origin-destination pair. The problem here dealt with consists of determining a subset of stations from an underlying network, maximizing the trip coverage and respecting a maximum length. For the purpose, a mathematical model as well as two different heuristics are presented. Concretely, the first heuristic starts with an edge and it is iteratively extended by adding a new edge. This edge is inserted at the beginning or at the end in the alignment, whenever the maximum length is not exceeded. This procedure stops when the current alignment cannot be extended. The second heuristic starts with a feasible initial solution and then a node is added in the current solution according to several criterion. Finally, a post-optimization procedure is applied on this solution. Computational experiments are carried out on the city of Seville as well as data randomly generated.

In Laporte et al. (2009) a Voronoi diagram for solving the problem of locating a metro line maintaining a minimum distance between the alignment which is being designed and historical buildings is proposed. The methodology developed is composed by several steps. First, a set of nodes representing historical buildings as well as two fixed nodes describing the origin and destination for the metro line in a planar region are defined. The Voronoi diagram induced by these nodes is constructed. Secondly, from this Voronoi diagram a graph is extracted by considering a safe circle around each node and eliminating all edges inside of each circle. In the third step, the shortest path between the fixed nodes is obtained. Finally, a procedure to improve this shortest path is considered. This methodology was applied to a line in the city of Seville.

Recently, Li et al. (2011b) presented analytical models for the problem of designing a linear transit line maximizing the associated profit. The authors described two different pricing structures for computing the revenue: a same fare for all passengers independently of the length of their trips and a distance-dependent fare. The cost structure includes the train operating cost and costs related to line and stations. The problem consists of deciding the spacing between stations from a fixed main station, the headway and fare. The passenger demand for each station is obtained by means of a function which depends on the distance to the fixed station. The problem is solved by deriving partially the objective function. A heuristic algorithm was used to solve this problem. The algorithm is divided into several steps. In the first step, an initial solution is conveniently chosen and the corresponding passenger demand for each station is obtained. The following steps are concentrated on modifying sequentially the decision variables. Finally, a stop criterion is checked. An analysis on the effect of population density and the rail capital cost on the profit as well as the effect of the population distribution and the corridor length are carried out.

2. Rapid transit network design

   Depending on the characteristic, features, constraints and objectives considered, there exists numerous variants of the rapid transit network design. We will distinguish between the design of infrastructure, design of line network on a given infrastructure network as well as network design taking into account aspects related to robustness.

   a) Infrastructure design

      In Blanco et al. (2011) a model for the expansion of transportation networks is proposed. Specifically, given an infrastructure network, the problem consists of deciding what stations and edges to construct at each time period minimizing the total construction and the operating cost over the planning horizon. The construction cost includes costs related to build stations and links between stations whereas the operating cost is the operating cost due to the operation of passengers. The purpose of this work is connecting a set of countries given with the current railway network. A mathematical programming model as well as a heuristic procedure were described. The algorithm is composed by a construction and an improvement phase. In the construction phase, a feasible partial solution for each period is generated. The idea is to construct new stations (and links) in the network until reaching the available budget for the considered period. To this end, at each iteration the station with the highest flow is built and it is linked to the closest built station. In the improvement phase, a scatter search heuristic on the previous solution is applied. In this phase two classes of movements are defined. In the first movement an edge is randomly selected whereas in the second movement a 2-edge path (a path with two edges) is selected. Concretely, the first movement consists of changing one end node of an edge by other station and the second one, replaces a path by an edge or by a new path, linking the same origin and final node than in the initial path. The authors defined four type of movements by combining the movements above. A case study on the Spanish PEIT (Strategic Planning of Infrastructure and Transport) is considered.

The goal of García-Archilla et al. (2013) is to design the infrastructure network maximizing the trip coverage in presence of a competing mode. They presented a mathematical model for this problem as well as for the robust version of this problem. The authors used a Greedy Randomized Adaptive Search Procedure (GRASP) heuristic to solve the infrastructure railway network design problem. The GRASP algorithm consists of two phases: a construction and an improvement phase. In the construction phase a feasible solution is generated as follows. This phase starts with a set E formed by an edge randomly selected. At each iteration, an edge is added to E. To this end, $k \geq 2$ edges maximizing the trip coverage and satisfying the budget constraint when they are individually added to E, are selected. From this k-edges, an edge randomly selected is added to E. The construction phase ends when the budget constraint is not satisfied. In the improvement phase an edge is randomly removed of E and it is replaced by a new edge (or several if it is possible). These two phases are repeated a given number times. The authors tested 70 instances randomly generated with at most 18 nodes, and a real size instance with 49 nodes, obtaining good and fasters solutions.

b) Line and infrastructure design

Bruno and Laporte (2002) described a visual interactive decision system which solves the rapid transit network design problem by means of heuristics. The authors focus on extending the algorithm developed in Bruno et al. (2002) for a multi-alignment. In the application, the users select the shape of the network, the inter-station spacing as well as the number of lines and stations for each line. The localization of each line is iteratively determined using the algorithm defined in Bruno et al. (2002). Moreover, several effectiveness measures are presented in the application. Experiments for the city of Milan are considered.

Laporte et al. (2007) presented a model for the rapid transit network design problem which integrates network design and line planning problems. This model is formulated as a linear binary integer program according to key nodes. A small network with 6 nodes and 9 edges is tested and it is implemented in GAMS 2.0.27.7 and CPLEX 9.0. Different congestion coefficients contributing to the private costs are taken into account. In the experimentation, they solve first the problem by considering length constraints and then they add a cost constraint in the model. The optimal solution obtained is defined by one or two lines depending on the imposed requirement. A similar model taking into account transfers between lines is proposed in García-Ródenas et al. (2006). This aspect is defined as the half of time between two consecutive services at the line to transfer plus time spent between platforms. In the experimentation, the network defined in Laporte et al. (2007) is tested by using CPLEX 8.0. The authors studied different values of the parameters such as transfer costs and line frequency and analyzed the effect of these parameters on the solution.

Marín (2007) used a branch-and-bound algorithm in order to solve the rapid transit network design. This work may be considered an extension of Laporte et al. (2007), where lines have a certain degree of freedom. Specifically, the author assumes a variable number of lines as well as lines have not fixed terminal stations. The objective function is a linear combination of the trip coverage and the routing cost. The computational experiments are concentrated on two small networks: the network presented in Laporte et al. (2007) and a network with 9 nodes and 16 edges. The model was implemented in GAMS 21.6 by using CPLEX 9.0. The author considers a maximum number of lines and different congestion factors.

The same algorithm is used in Guan et al. (2006). The authors simultaneously determine the transit line configuration and the passenger line assignment. The line planning problem defined consists on finding a set of lines (of a given line pool) that connects all stations of a given infrastructure network, minimizing the total length of all lines. The passenger line assignment was described by means of paths (sequences of edges and nodes that connects origin-destination pairs). To integrate both problems, the objective function is defined as a convex combination of those proposed at each problem: the total length of all transit length (for the line planning problem) and total passenger in-vehicle travel time and total number of passenger transfers (for the passenger line assignment). The authors analyzed several types of networks: two spanning tree networks with 6 and 9 nodes and a simplified real network. For the purpose, the effect of demand between origin-destination pairs, minimum and maximum line length as well as maximum link capacity and maximum number of transfers, on the spanning tree networks is analyzed. A numerical case of study on the Hong Kong mass transit railway (MTR) network is carried out. The MTR network is formed by 6 transit lines and 49 stations. Due to the complexity of the problem, a simplified version of this network is presented. Concretely, this network is reduced to 9 stations and 10 edges as follows. The node set is defined by the terminal stations of each line and nodes representing intersection of lines. The number of possible lines in the line pool is also reduced. An analysis of sensibility of the factors that appear in the convex combination of the objective function is presented.

In order to allow circular lines, a modification of the extended rapid transit network design problem (see Marín (2007)) is introduced by Escudero and Muñoz (2009). Thus, a two-stage approach for solving this problem is presented. In the first stage, an integer model is solved in order to select stations and links between them, without exceeding the available budget and maximizing the number of users. The resulting model may yield an undesirable line set formed by non-connected lines consisting of one non-circular sub-line and various circular sub-lines. To avoid such lines, the authors propose to define each sub-line as a line. This model also allows the possibility of more than one line linking two locations. In the second stage, the authors present a procedure for solving the above problem by assigning each selected link to exactly one line in order to minimize the number of lines. The same networks that proposed in Marín (2007) were examined. In the experimentation, the available budget as well as the congestion factor were tested. The models were implemented in Microsoft Visual C++ 2005 and CPLEX 11.0. In the branching process, the priorities for the variables were changed in order to improve the CPU time.

The main contributions of Marín and García-Ródenas (2009) are the inclusion of location constraints (in order to minimize the number of routing intersections) and the consideration of the logit model (instead of the previous all or nothing models). This model expresses the proportion of users which are assigned to each mode for each OD pair. The authors assume that each transportation mode depends exclusively on the associated transportation costs. A strategy to approximate the non-linear logit function by a polygonal curve is developed. In the experiments, a small and a median size network have been used. Concretely, the small network is the defined in Laporte et al. (2007) and the median network represents a simplification of the Seville's metro used in Laporte et al. (2007). Comparative tests on the different described models have been considered and implemented in GAMS with CPLEX 10.0. The small network is defined by 6 nodes and 9 edges whereas the network representing the Seville's metro has 24 nodes and 276 edges. In the analysis a total of 30 demands and subsets of 552 demands for the small and Seville's network are considered, respectively.

The difficulty for solving the rapid transit network design problem for medium-size networks, was one of the reasons to motivate the search of new techniques and to develop faster methods. For instance, Marín and Jaramillo (2009) use algorithms based on Benders decomposition to find optimal solutions for the RTND problem. The authors present a general objective taking into account the trip coverage, the routing and location cost. The problem is divided into a *Master Model* (MM), which defines a feasible network, and a *Sub Model* (SM), which assigns demand to this network. The Benders decomposition simultaneously considers (MM) and (SM). At each iteration, the dual variables of the (SM) define Benders Cuts, which are added to the constraints of the master problem. In order to improve the computational time, several extensions of Benders decomposition are proposed. In their experimentation three network are tested: two small networks (the defined in Marín (2007)) as well as a network representing the city of Seville (see Laporte et al. (2007)). The computational experience is done with GAMS 21.7 and CPLEX 9.0. A comparison between Branch and Bound and the proposed extensions of Benders decomposition is developed. The results indicate that these extensions reduce the computational time to obtain solutions.

Based on the Simulated Annealing heuristic (SA), Kermanshahi et al. (2010) solved the RTND problem by maximizing the trip coverage. The problem is defined according to transit routes. The authors assume the start and the end of all routes are predetermined. The feasible routes are generated taking into account two criterion: an acceptable length and not entailing tours. The heuristic starts with an initial solution obtained from this set of routes. Concretely, the initial solution is formed by a subset of routes randomly extracted on the feasible routes set. The current solution is iteratively modified by means of movements. In this work only a type of movement is defined. Specifically, it consists of changing a route randomly selected from the current solution, by a generated new route according to two criterion. So, the difference between two neighbors is only a route. The probability of choosing each criterion is controlled by an input parameter. The first criterion proposes a new route connecting stations before non-connected. The second one selects the route with the highest demand. Therefore, at each iteration, a candidate solution is selected and an acceptance criterion based on Simulated Annealing is applied. If the candidate solution is better than the current solution, the solution is accepted. Otherwise, the acceptance depends on a temperature parameter and a cool factor. The computational experiments on the network defined in Marín (2007) shows that this algorithm gives good results in a reasonable amount of time.

c) Robustness

Using a branch-and-cut algorithm in GAMS 22.2 and CPLEX 10.0, Laporte et al. (2012) have solved the problem of designing a robust rapid transit network. To this end, the authors first deal with the deterministic version of the problem and then with the effects of possible failures in the network. The objective function is a convex combination of trip coverage with possible failures and total routing maximal. Three ways of introducing robustness in the deterministic model by capacity constraints is developed. In order to to illustrate the feasibility of integrating robustness considerations in a planning model, a small network with 9 nodes is tested.

In Laporte et al. (2010) a game theoretic framework is used in order to solve the problem of designing a railway transit network in the presence of failures. The authors consider only two agents acting in the problem: the planner and the demon. The planner wants to minimize trip coverage or total travel time whereas the demon makes the system works as bad as possible. In the paper two versions of this problem are formulated: Probabilistic Railway Network Design (PRND) and the Stochastic Railway Network Design (SRND) problem. The same network as in Laporte et al. (2007) is considered in the experimentation. First, they focus on the trip coverage and then on the total travel time. The PRND is based on the assumption of each arc failures with the same probability whereas in the SRND the failure probability of each arc is unknown. For both version of the problem, the objective function is a linear combination of trip coverage or total travel time with and without failures.

An attempt to integrate robust network design and line planning is carried out in Marín et al. (2009). The authors first described the rapid network design (RND) and the line planning (LP) problem separately and then they integrated both problems. They presented a model for solving the RND where the only decision variables are nodes, arcs and flows. Once the RND is solved, the infrastructure network defines an input data in the line planning problem. Two different definitions of robustness are presented: one from the user's point of view and the other from the operator's point of view. The robustness concept for the user is defined by means of a measure which gives information on the travel time in failures presence. For the operator, the robustness is introduced by an index expressing the effect of failures on the fleet of vehicles. The main contribution of this paper is the introduction of an iterative procedure that combines the robust rapid network design and the robust line planning problem. For the user robustness concept the algorithm is defined as follows. First, the RND problem is solved considering the total travel time as utility function. From the topological network obtained, the line planning problem is solved. This solution defines the initial solution in the iterative process. In the following iterations, this network is extended covering the same OD pairs but in a more robust way than the initial. Similarly this algorithm is adapted to the operator's preferences. In order to show the applicability of this algorithm, a network representing the capitals of provinces in Andalucía's regions is tested. This network is formed by 10 nodes and 16 edges.

In a related field, Mauttone and Urquhart (2009b) used the Route Generation Algorithm inspired in the structure of Baaj and Mahmassani (1995) to determine a set of bus routes for a public transportation system. They inserted pairs of vertices with high values of demand on existing routes. The shortest path between these vertices defines a new route in the system. Some of meta-heuristic algorithms used in this field are Genetic Algorithm (Tom and Mohan (2003), Ngamchai and Lovell (2003)), Tabu Search (Fan and Machemehl (2004)), Simulated Annealing (Fan and Machemehl (2006)), GRASP (Mauttone and Urquhart (2009a)) and Scatter Search (Gendreau and Potvin (2010)). In Guihaire and Hao (2008) a non-exhaustive classification of the strategic and tactical steps of transit planning is presented. They concentrated the literature review on the design and scheduling of networks in the context of urban buses and railways. More concretely, a classification of 69 approaches related to the design, frequencies setting, timetabling of transit lines in the field of railways and urban buses are proposed.

Literature reviewed				
Authors	Problem	Objectives	Solution procedure	Application
Gendreau et al. (1995)	design rapid transit alignment	population coverage	Greedy	
Dufourd et al. (1996)	locating a rapid transit line	population coverage	tabu search	
Bruno et al. (1998)	locating a rapid transit line	construction cost and travel cost	heuristic	median network: 21 nodes and 39 arcs
Bruno et al. (2002)	constructing an alignment	population coverage	several heuristics	median examples
Laporte et al. (2005)	constructing an alignment	trip coverage	several heuristic	median example
García-Ródenas et al. (2006)	rapid transit network design	trip coverage	exact	small network with 6 nodes and 9 edges
Guan et al. (2006)	line configuration and passenger line assignment	costs, travel time and number of transfers	exact	median network: Hong Kong railway network
Laporte et al. (2007)	rapid transit network design	trip coverage	exact	small network with 6 nodes and 9 edges
Marín (2007)	rapid transit network design	trip coverage and routing cost	exact	small networks
Laporte et al. (2009)	localizing a metro line	distance between buildings	heuristic based on Voronoi diagram	median example
Escudero and Muñoz (2009)	rapid transit network design	trip coverage	exact	small networks
Marín and García-Ródenas (2009)	railway network design	trip coverage and private cost	exact	small and median networks
Marín and Jaramillo (2009)	rapid transit network design	trip coverage, routing and location cost	exact. Bender decomposition	small and median networks
Marín et al. (2009)	robust rapid transit network design and line planning	travel time and costs	heuristic	small network.
Kermanshahi et al. (2010)	rapid transit network design	trip coverage	Simulated annealing	small networks
Laporte et al. (2011)	rapid transit network design	trip coverage	Exact. Branch and bound	small network
Blanco et al. (2011)	expansion of transportation networks	construction and operation cost	several heuristics	median network on Spanish transportation
Laporte et al. (2012)	robust rapid transit network design	trip coverage and total travel time		small network
García-Archilla et al. (2013)	railway network design	trip coverage	GRASP algorithm	median example: 18 nodes

Table 4.1.: Main solution methods found in the literature of rapid transit network design.

4.2. The mathematical programming program

As mentioned, the main innovative point of our model with respect to current literature is the simultaneous treatment of network design and line planning problems. Moreover, line frequency and capacity are also considered, as well as several aspects related to rolling stock and personnel planning. Our network design model also includes passenger transfers between the lines, flow conservation, as well as location and allocation constraints, and a competing mode. The main input data are the underlying network, that is, the potential location for the stations and their connections, the distance matrix between pairs of stations of the underlying network, the travel patterns as well as train capacities, building costs and operational costs. We assume that passengers choose theirs routes and transport according to travel time which is affected by the frequency. The objective function is the net profit (see Chapter 2). In our approach, we also consider the existence of public economic support for network deployment as a key factor in the network design analysis.

The main differences between the model here presented and the general model in Chapter 2 are the followings. The flow variables $f_{ij}^{w\ell}$ as well as variables representing transfers $f_i^{w\ell\ell'}$ between lines are binaries. This fact yields to introduce new variables and constraints. In the following, we formally describe our problem.

4.2.1. Data and notation

We assume the same input data as in Chapter 2. However, in order to simplify several constraints in our model, we will assume that the number of services of a line is given by means of its headway. Note that the frequency and headway are inversely proportional (frequency is 60/headway). So, we consider a fixed finite set of possible headway $\widehat{\mathcal{F}}$ for lines of the RTS. We assume the headway of each line takes values between a minimum and maximum headway in order to guarantee a certain level of service in the network. We describe the set of ordered possible headway as $\widehat{\mathcal{F}} = \{\zeta^1, \zeta^2, \ldots, \zeta^{|\widehat{\mathcal{F}}|}\}$, where each $\zeta^q \in \mathbb{N}$, $1 \leq q \leq |\widehat{\mathcal{F}}|$ and $|\widehat{\mathcal{F}}| \geq 2$.

4.2.2. Variable

The set of variables in our model is the same as in Chapter 2, with the exception of frequency that now is headway; flow variables that now are binaries, and others new variables introduced in this section.

For the sake of readability, we will repeat some variables and constraints already presented in Chapter 2.

- $y_i^\ell = 1$ if node i is selected to be a station of line $\ell \in \mathcal{L}$, 0 otherwise.
- $y_i = 1$ if node i is selected to be a station in the RTS, 0 otherwise.
- $x_{ij}^\ell = 1$ if edge $\{i,j\} \in E$ belongs to line $\ell \in \mathcal{L}$, 0 otherwise.
- $x_{ij} = 1$ if edge $\{i,j\} \in E$ is included in the RTS, 0 otherwise.
- $h_\ell = 1$ if line $\ell \in \mathcal{L}$ is included, 0 otherwise.
- $\zeta_\ell \in \widehat{\mathcal{F}}$ describing the headway of line ℓ (time between services expressed in minutes).
- $\delta_\ell \in \{\delta^{min}, \dots, \Delta\}$ representing the number of carriages used by trains of line ℓ.
- $u_w^{RTS} > 0$ is the travel time of pair w using the RTS network.
- $f_w^{RTS} \in [0,1]$ is the proportion of OD pair w using the RTS network.
- $\tilde{f}_{ij}^{w\ell} = 1$ if the OD pair w traverses arc $(i,j) \in A$ using line ℓ, 0 otherwise.
- $\tilde{f}_i^{w\ell\ell'} = 1$ if demand of pair w transfers in station i from line ℓ to line ℓ', 0 otherwise.
- $p_w = 1$ if demand of pair w is allocated to the railway network, 0 otherwise.

The average travel time associated to OD pair $w = (w_s, w_t) \in W$ using the RTS network can be explicitly defined as follows:

$$\begin{aligned} u_w^{RTS} &= \sum_{\ell \in \mathcal{L}} \sum_{j:\{w_s,j\} \in E} \frac{\zeta_\ell \tilde{f}_{w_s j}^{w\ell}}{2} + (60/\lambda) \sum_{\ell \in \mathcal{L}} \Big(\sum_{\{i,j\} \in E} \tilde{f}_{ij}^{w\ell} d_{ij} \Big) \\ &+ \sum_{\ell \in \mathcal{L}} \sum_{\ell':\ell' \neq \ell} \sum_{i \in N} \tilde{f}_i^{w\ell\ell'} \Big(\frac{\zeta_{\ell'}}{2} + uc_i \Big), \; w = (w_s, w_t) \in W. \end{aligned} \tag{4.1}$$

The first term in (4.1) is the waiting time at the origin station, which is also assumed to be half of time between services of this line. The second term in (4.1) is the in-vehicle time. The third one is the time spent in transfers, which is assumed to be half the time between two consecutive services in the line to transfer, plus the necessary time to walk from the platform of one line to the platform of the other.

As in Chapter 2, we define the proportion of OD pair w using the RTS mode as

$$f_w^{RTS} = \frac{1}{1 + e^{(\alpha - \beta(u_w^{ALT} - u_w^{RTS}))}}, \; w \in W. \tag{4.2}$$

4.2.3. Constraints

In this section we will comment the difference between this model and the general model, and we will describe some new constraints. In this model, we do not consider the budget constraints group, but this aspect could be incorporated to our model. Due to the fact that flow variables are binary variables, we have defined a new group of constraints for describing transfers. The constraints of the problem are formulated as follows.

- Design forcing

  These constraints are the same to those defined in the general model in Chapter 2, with the exception of Constraints (2.26), which are modified by (4.4). Moreover, we have included the following constraints.

$$x_{ij}^{\ell} \leq x_{ij},\ \{i,j\} \in E, i < j \tag{4.3}$$

$$N_{\min} h_{\ell} \leq \sum_{i \in N} y_i^{\ell} \leq N_{\max} h_{\ell},\ \ell \in \mathcal{L} \tag{4.4}$$

$$h_{\ell} \leq \sum_{\{i,j\} \in E} x_{ij}^{\ell},\ \ell \in \mathcal{L}. \tag{4.5}$$

- Routing demand constraints

  Due to the fact that flow variables are binary variables, all constraints presented in the general model are modified.

$$\sum_{\ell \in \mathcal{L}} \sum_{j:(w_s,j) \in A} \tilde{f}_{w_s j}^{w\ell} = p_w,\ w = (w_s, w_t) \in W \tag{4.6}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{i:(i,w_t) \in A} \tilde{f}_{i w_t}^{w\ell} = \sum_{\ell \in \mathcal{L}} \sum_{j:(w_s,j) \in A} \tilde{f}_{w_s j}^{w\ell},\ w = (w_s, w_t) \in W \tag{4.7}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{i:(i,k) \in A} \tilde{f}_{ik}^{w\ell} - \sum_{\ell \in \mathcal{L}} \sum_{j:(k,j) \in A} \tilde{f}_{kj}^{w\ell} = 0,\ w = (w_s, w_t) \in W, k \neq \{w_s, w_t\}. \tag{4.8}$$

- Location-allocation constraints

$$f_w^{RTS} \leq p_w,\ w \in W \tag{4.9}$$

$$p_w \leq 100 f_w^{RTS},\ w \in W \tag{4.10}$$

$$\tilde{f}_{ij}^{w\ell} \leq p_w,\ w \in W, \{i,j\} \in E, \ell \in \mathcal{L} \tag{4.11}$$

$$\tilde{f}_{ij}^{w\ell} + \tilde{f}_{ji}^{w\ell} \leq x_{ij}^{\ell},\ w \in W \ell \in \mathcal{L},\ \{i,j\} \in E, i < j \tag{4.12}$$

$$\tilde{f}_i^{w\ell\ell'} \leq y_i^{\ell},\ w \in W, \ell, \ell' \in \mathcal{L}, \ell \neq \ell', i \in N \tag{4.13}$$

$$\tilde{f}_i^{w\ell\ell'} \leq y_i^{\ell'},\ w \in W, \ell, \ell' \in \mathcal{L}, \ell \neq \ell', i \in N \tag{4.14}$$

$$\tilde{f}_i^{w\ell\ell'} \leq 0.5 \cdot (h_\ell + h'_\ell),\ w \in W, \ell, \ell' \in \mathcal{L}, \ell \neq \ell', i \in N. \tag{4.15}$$

- Transfers

$$\tilde{f}_i^{w\ell\ell'} \geq \sum_{k:(k,i)\in A} \tilde{f}_{ki}^{w\ell} + \sum_{k:(i,k)\in A} \tilde{f}_{ik}^{w\ell'} - 1,\ w \in W, i \in N, \ell, \ell' \in \mathcal{L}, \ell \neq \ell' \tag{4.16}$$

$$2\tilde{f}_i^{w\ell\ell'} \leq \sum_{k:(k,i)\in A} \tilde{f}_{ki}^{w\ell} + \sum_{k:(i,k)\in A} \tilde{f}_{ik}^{w\ell'},\ w \in W, i \in N, \ell, \ell' \in \mathcal{L}, \ell \neq \ell' \tag{4.17}$$

$$\tilde{f}_{w_s}^{w\ell\ell'} = 0,\ w = (w_s, w_t) \in W, \ell, \ell' \in \mathcal{L}, \ell \neq \ell' \tag{4.18}$$

$$\tilde{f}_{w_t}^{w\ell\ell'} = 0,\ w = (w_s, w_t) \in W, \ell, \ell' \in \mathcal{L}, \ell \neq \ell'. \tag{4.19}$$

- Capacity constraints

$$\zeta_\ell \sum_{w\in W} g_w f_w^{RTS} \tilde{f}_{ij}^{w\ell} \leq 60 \cdot \Theta \cdot \delta_\ell,\ \ \ell \in \mathcal{L}, \{i,j\} \in E. \tag{4.20}$$

- Binary constraints

$$x_{ij}, y_i, x_{ij}^{\ell}, y_i^{\ell}, h_\ell, \tilde{f}_{ij}^{w\ell}, \tilde{f}_k^{w\ell\ell'} \in \{0,1\},\ k \in N, \{i,j\} \in E, i \in N, \ell \in \mathcal{L}, w \in W.$$

- Integer constraints

$$\delta_\ell \in \{\delta^{min}, \ldots, \Delta\}, \ell \in \mathcal{L}$$
$$\zeta_\ell \in \widehat{\mathcal{F}}, \ell \in \mathcal{L}.$$

- Other constraints

$$f_w^{RTS} \in [0,1], w \in W.$$

Recall that the model includes constraints related to the proportion of passengers using the RTS (Equation 4.2) as well as the travel time for each OD pair (Equation 4.1). Constraints (4.3) impose that an edge is built for a specific line only if the edge is included in the RTS. Constraint (4.4) forces that each line must have at least N_{min} nodes and at most N_{max} nodes. Constraints (4.5) guarantee a line is not built if it has not constructed arcs. Constraints (4.6)–(4.8) define flow conservation for each OD pair. Constraints (4.9) and (4.10) force demand pairs to be assigned to the RTS if the associated travel time using the RTS (taking the fastest route) does not exceed the corresponding time of the alternative mode. We impose constraints (4.11) in order to ensure that there not exists a route for an OD pair by the RTS mode if the demand is not assigned to the RTS. We impose constraints (4.12) in order to allow the flow corresponding to each OD pair to use an edge of a line ℓ only if this edge belongs to ℓ. For the sake of clarify, we repeat Constraints (4.13) and (4.14) of the general model which guarantee that if a transfer between two lines is made at node i, station i is built for both lines. Constraints (4.15) ensure that if a transfer between two lines is made, both lines are already built. Constraints (4.16) and (4.17) ensure that if an OD pair w enters station $k \in N$ using one line and exits this station using another line, then a transfer is done. Constraints (4.18) and (4.19) impose that it is not possible to make a transfer at the origin or destination station of the considered OD pair. Constraints (4.20) indicates the total capacity per hour of such line is a sufficiently large number in order to transport all passengers preferring to travel in the RTS.

4.2.4. Efficient approaches

In this section we show different techniques for improving the efficiency of the model presented in the previous section.

Capacity constraints

The terms in (4.20) expressed as a product of a binary variable and a real variable, that is, $\tilde{f}_{ij}^{w\ell}$ and f_w^{RTS}, are transformed into linear constraints by means of a new variable

$\xi_{ij}^{w\ell} = f_w^{RTS} \tilde{f}_{ij}^{w\ell} \geq 0$. Constraints (4.20) are substituted by

$$\zeta_\ell \sum_{w \in W} g_w \xi_{ij}^{w\ell} \leq 60 \cdot \Theta \cdot \delta_\ell,\ \ell \in \mathcal{L}, \{i,j\} \in E \tag{4.21}$$

$$\xi_{ij}^{w\ell} \leq \tilde{f}_{ij}^{w\ell}, \ell \in \mathcal{L}, \{i,j\} \in E, w \in W \tag{4.22}$$

$$f_w^{RTS} - (1 - \tilde{f}_{ij}^{w\ell}) \leq \xi_{ij}^{w\ell},\ \ell \in \mathcal{L}, \{i,j\} \in E, w \in W \tag{4.23}$$

$$\xi_{ij}^{w\ell} \leq f_w^{RTS},\ \ell \in \mathcal{L}, \{i,j\} \in E, w \in W. \tag{4.24}$$

Logit

In this section we will develop a strategy in which the non-linear logit function f_w^{RTS} defined in Equation (4.2) is approximated by a polygonal curve (piecewise linear function). This linear function is defined by considering three intervals on the abscissa axis. Note that the number of intervals is not binding and this procedure can be extended. The logit function f_w^{RTS} can be rewritten as a function $F(x) = 1/(1+\exp(\alpha - \beta(u_w^{ALT} - x))$, where x represents the travel time u_w^{RTS} on the RTS mode. First, we approximate the curve at the point $(u_w^{ALT}, 0.5)$ by a linear function with slope $-\beta/4$ obtained by evaluating the derivative of $F(x)$ at that point. Second, this function is projected on the horizontal line $F(x) = 1$ obtaining a point of coordinates $(u_w^{ALT} - 2/\beta, 1)$ and then on the horizontal line $F(x) = 0$ obtaining a point of coordinates $(u_w^{ALT} + 2/\beta, 0)$.

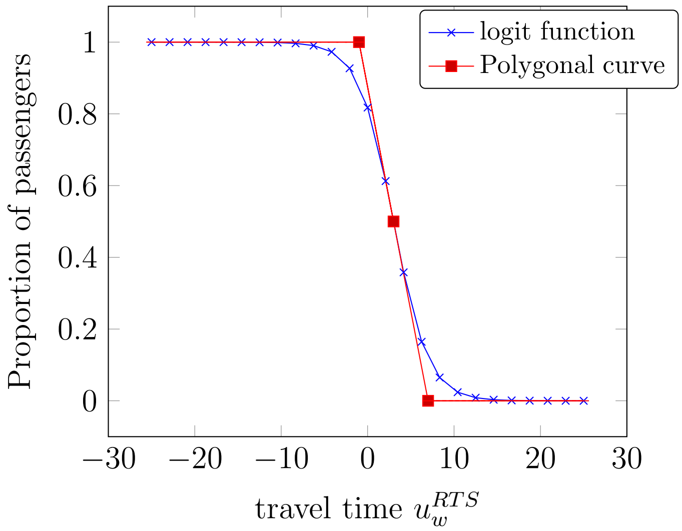


Figure 4.2.: Representation of a logit function and its corresponding polygonal curve for $u_w^{ALT} = 3$, $\alpha = 0$ and $\beta = 0.5$.

Therefore, the piecewise linear function is defined as

$$\mathcal{P}(x) := \begin{cases} 1, & x < u_w^{ALT} - 2/\beta \\ -\beta/4x + (2 + \beta u_w^{ALT})/4, & x \in [u_w^{ALT} - 2/\beta, u_w^{ALT} + 2/\beta] \\ 0, & x \geq u_w^{ALT} + 2/\beta. \end{cases}$$

This function can be modelled by means of the positive variables $u_{s_i w}^{RTS}$ and the binary variables $\chi_w^{s_i}$, $s_i = 1, \ldots, 3$ as follows:

$$\begin{aligned} & u_w^{RTS} = \sum_{s_i} u_{s_i w}^{RTS},\ w \in W \\ & \sum_{s_i} \chi_w^{s_i} = 1,\ w \in W \\ & -1000 \cdot \chi_w^{s_1} \leq u_{s_1 w}^{RTS} \leq \chi_w^{s_1} \cdot (\beta \cdot u_w^{ALT} - 2)/\beta,\ w \in W \\ & (\beta \cdot u_w^{ALT} - 2)/\beta \cdot \chi_w^{s_2} \leq u_{s_2 w}^{RTS} \leq \chi_w^{s_2} \cdot (\beta \cdot u_w^{ALT} + 2)/\beta,\ w \in W \\ & (\beta \cdot u_w^{ALT} + 2)/\beta \cdot \chi_w^{s_3} \leq u_{s_3 w}^{RTS} \leq 1000 \cdot \chi_w^{s_3},\ w \in W \\ & f_w^{RTS} = 1 - \chi_w^{s_3} + (-\beta/4 \cdot u_{s_2 w}^{RTS} + (2 + \beta \cdot u_w^{ALT})/4 - 1) \cdot \chi_w^{s_2},\ w \in W. \end{aligned}$$

Travel time

The terms in (4.1) expressed as a product of a binary variable and an integer variable, that is, $\tilde{f}_i^{w\ell\ell'}$ and $\zeta_{\ell'}$, are transformed into linear constraints by means of a new variable $\bar{\xi}_i^{w\ell\ell'} = \tilde{f}_i^{w\ell\ell'} \zeta_{\ell'} \geq 0$. Constraint (4.1) is substituted by

$$\bar{\xi}_i^{w\ell\ell'} \leq \zeta^{|\widehat{\mathcal{F}}|} \tilde{f}_i^{w\ell\ell'}, \ell \neq \ell' \in \mathcal{L}, i \in N, w \in W \tag{4.25}$$

$$\zeta_{\ell'} - \zeta^{|\widehat{\mathcal{F}}|}(1 - \tilde{f}_i^{w\ell\ell'}) \leq \bar{\xi}_i^{w\ell\ell'},\ \ell \neq \ell' \in \mathcal{L}, i \in N, w \in W \tag{4.26}$$

$$\bar{\xi}_i^{w\ell\ell'} \leq \zeta_{\ell'},\ \ell \neq \ell' \in \mathcal{L}, i \in N, w \in W. \tag{4.27}$$

Similarly, the product $\hat{\xi}_{ij}^{w\ell} = \tilde{f}_{ij}^{w\ell} \zeta_\ell \geq 0$ can be expressed as follows:

$$\hat{\xi}_{ij}^{w\ell} \leq \zeta^{|\widehat{\mathcal{F}}|} \tilde{f}_{ij}^{w\ell}, \ell \in \mathcal{L}, \{i,j\} \in E, w \in W \tag{4.28}$$

$$\zeta_\ell - \zeta^{|\widehat{\mathcal{F}}|}(1 - \tilde{f}_{ij}^{w\ell}) \leq \hat{\xi}_{ij}^{w\ell},\ \ell \in \mathcal{L}, \{i,j\} \in E, w \in W \tag{4.29}$$

$$\hat{\xi}_{ij}^{w\ell} \leq \zeta_\ell,\ \ell \in \mathcal{L}, \{i,j\} \in E, w \in W. \tag{4.30}$$

4.3. The heuristic. Adaptive large neighborhood search heuristic

Due to the NP-hard character of our problem, a powerful heuristic is required for solving instances of realistic size. Local Search heuristics are often applied for solving problems related to the rapid transit network design (see Section 4.1.1). This kind of heuristic builds a neighborhood by means of movements based on small changes of the current solution. A major drawback of these algorithms is the difficulty of exploring new promising search spaces. As a consequence, the solution can became a local optimal. For solving the problem described in previous section, we propose a new method based on the Adaptive Large Neighborhood Search Heuristic (ALNS). This heuristic was introduced by Ropke and Pisinger (2006). It is considered into the category of large scale neighborhood search defined in Ahuja et al. (2002) but only examines a relatively low number of solutions. The ALNS concept extends the large neighborhood search heuristic of Shaw (1997). The main difference between Ropke and Pisinger (2006) and Shaw (1997) is regarding to the probability of choosing each operator. Coelho et al. (2012) propose not to use the traditional destroy and repair but only to apply one operator at each iteration. In our ALNS, we define several destroy and repair operators which are independently applied as in Coelho et al. (2012). From these operators, we describe operators composed of one destroy and one repair method.

In this section we describe an Adaptive Large Neighborhood Search Heuristic (ALNS) for the rapid transit network design problem defined in Section 4.2. To this end, we assume a RTL is defined by a set of stations, a set of arcs linking these stations and a set of lines. Each line is characterized by two different terminal stations (initial and final stations), the intermediate stops, the frequency and the capacity of each train (number of carriages). The key ideas of our ALNS algorithm are the following. Initially, the RTL is formed by a set of lines randomly defined (see Section 4.3.1). At each iteration, a line is randomly modified by means of an operator. An operator is a heuristic method that modifies a candidate solution, in our case, modifies a line from a RTL. As mentioned, the operators are classified into two classes: destructor and repair methods. In our algorithm, there are two types of operators for each class and two types of destroy-repair operators combining both classes. The repair methods consist of inserting new lines or extending existing lines, whereas the destroy operators consist of removing partial or totally an existing line in the current RTL. The destroy-repair operators are defined as a combination of destroy and repair operators: one destroy-repair operator consists of eliminating an existing line and then, inserting a new line at the same iteration, whereas the other one, consists of removing partially a line and then, extending a line randomly selected, at the same iteration. The lines are randomly selected and the operators are chosen with a certain probability. The probability of selecting a determined operator depends on its efficiency in the past iterations. Thus, at each iteration, an operator is randomly selected with a certain probability and the current solution is modified. The goodness of the solution is evaluated by the HLSA heuristic defined in Chapter 3. Given a line network, this heuristic solves the problem of maximizing the profit of a line plan by selecting the frequency and the train size of each line, assuming that all passengers willing to travel in the RTS can be transported. We consider the uncapacitated version of this problem. The reason for assuming an unlimited number of possible carriages is to yield a non-congested network. The following figure shows a flow chart for our ALNS algorithm.

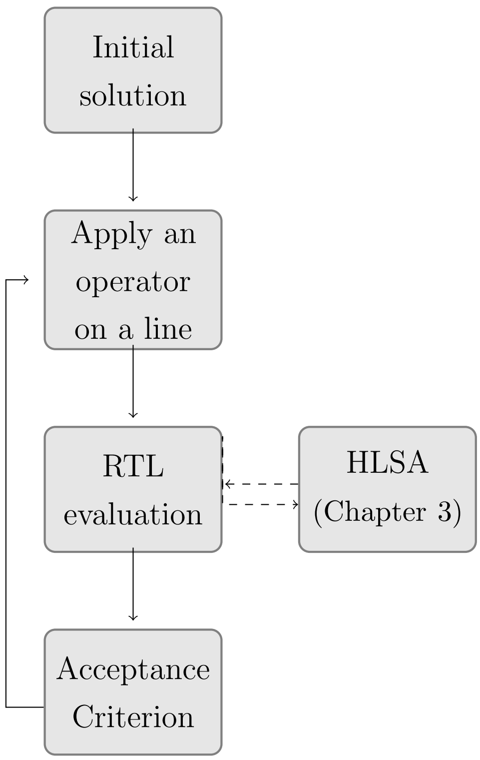


Figure 4.3.: Flow chart for our ALNS algorithm.

If the current solution is better than the previous solution, the search continues from this new solution and the probability of applying this operator is increased. Otherwise, this solution is accepted according to an acceptance criterion such as simulated annealing (SA). In the following, we will describe the main components of our ALNS heuristic.

1. Large neighborhood

   At each iteration, the current network is modified by means of an operator previously selected. If this operator is a repair method, data on the given underlying network are required. Specifically, the repair operator randomly selects two nodes belonging to the underlying network and computes the shortest path between these nodes. This path defines a new line in the current RTS (or extends an existing line), if several requirements are satisfied. If the selected operator is a destroy method, a line of the current solution is randomly selected and it is partial or totally eliminated. We define two possible destroy-repair method: one method removes randomly a line and then it adds a new line in the RTL, whereas the other one, removes partially a line and then it extend a line randomly selected.

   Thus, two RTS, G_{RTS} and $G_{RTS'}$ are neighbor if they have at most two different lines. It can be observed that the number of nodes and edges in the underlying network determine the size of the neighborhood.

2. Adaptive search engine

   The operators are selected according to a probability function. Each operator i has associated a weight $\widehat{\omega}_i$ which gives a measure of how well the operator has performed recently. Concretely, if there exists h operators with weights $\widehat{\omega}_j$, $j = 1, \ldots, h$, the probability of selecting the operator m is $\widehat{\omega}_m / \sum_{j=1}^{h} \widehat{\omega}_j$.

3. Adaptive weight adjustment

   In this section we describe how the weights $\widehat{\omega}_j$ are adjusted at each iteration. Let φ be the number of iterations considered in the implementation. For each block of s iterations ($s \leq \varphi$) the operator's behavior is observed. More precisely, φ can be defined as a multiple of s, that is, $\varphi = s \cdot k$, $k \in \mathbb{N}$. Therefore, the algorithm will observe k times the performance of the operators. At the beginning, all weights are fixed to one, that is, $\widehat{\omega}_i = 1, i = 1, \ldots, h$. After carrying out the first s-iterations, the weights are modified according to another parameters called *scores*. These last parameters show the performance of the operators during the s-iterations. At the beginning of each block of s-iterations, all scores are equal to zero. Once the RTL is evaluated by the HLSA, the score associated with the operator applied is taken into account. If the new network is improved or accepted by the considered acceptance criterion, the score is increased by means of three parameters σ_1, σ_2 and σ_3 as follows.

If this solution is better than the best global solution, its score is increased by σ_1; if it is better than the incumbent solution, its score is increased by σ_2, and if it is not better than the incumbent solution but is accepted, its score is increased by σ_3. The better the solution is, the higher the score is, that is $\sigma_1 \geq \sigma_2 \geq \sigma_3$. Another needed parameter to describe the weight $\widehat{\omega}_i$ is o_i. This parameter controls the number of times that operator i is used in the incumbent s-iterations. Let π_i be the score of the operator i and let $\nu_i \geq 1$ be a factor representing the computational effort that it requires as in Coelho et al. (2012). So, the weights are updated as follows:

$$\widehat{\omega}_i := \begin{cases} \widehat{\omega}_i & \text{if } o_i = 0 \\ (1-\varpi)\widehat{\omega}_i + \varpi\pi_i/\nu_i o_i & \text{if } o_i \neq 0 \end{cases}$$

where $\varpi \in [0,1]$ is a parameter called the *reaction factor*, controlling how quickly the weight adjustment algorithm reacts to changes in the effectiveness of the operator.

4. Objective function

   The objective function $z_{NET}(G_{RTS})$ considered in our algorithm is the net profit defined Chapter 2, which takes into account aspects related to construction, operation and personnel costs.

5. Acceptance and stopping criteria

   The acceptance criterion used in the ALNS is the same as in Simulated Annealing (SA). Two parameters are needed: the current temperature $\tau > 0$ and the cooling rate $0 < \tilde{\phi} < 1$. The temperature starts with τ_{start} and at determined iterations, it is cooled by the cooling rate $\tilde{\phi}$ ($\tau = \tau_{start} \cdot \tilde{\phi}$). The parameter τ_{start} may be set by inspecting the initial solution. In Ropke and Pisinger (2006) τ_{start} is set in the way that a solution 5% worse than the initial solution has 50% probability of being accepted. Given the current solution G_{RTS}, we accept a new neighbor solution G'_{RTS} with probability $\exp\{(z_{NET}(G'_{RTS}) - z_{NET}(G_{RTS}))/\tau\}$. Obviously, if $z_{NET}(G'_{RTS}) > z_{NET}(G_{RTS})$, the new solution is accepted. In our algorithm, if the difference between $z_{NET}(G'_{RTS}) - z_{NET}(G_{RTS})$ is less than $\bar{\varsigma}$% of $z_{NET}(G_{RTS})$, the acceptance probability is 0.5, i.e., $\exp\{(z_{NET}(G'_{RTS}) - z_{NET}(G_{RTS}))/\tau_{start}\} = 0.5$ or, equivalently, $\tau_{start} = (z_{NET}(G'_{RTS}) - z_{NET}(G_{RTS}))/\ln(0.5)$. This parameter $\bar{\varsigma}$ is adjusted by controller.

   At the traditional Simulated Annealing, the temperature is cooled with the cooling rate at each iteration. At this case, next lemma shows how the maximum number of iterations can be adjusted.

Lemma 4.1 *If the parameter τ is adjusted at each iteration, the maximum number of iterations* $\max_{iter}$ *can be calculated as* $\max_{iter} = \log(\tau_{final}/\tau_{start})/\log(\tilde{\phi})$*, where* τ_{final} *is the final temperature.*

Proof.-

It can be observed that if τ is multiplied by $\tilde{\phi}$ at each iteration, $\tau_{final} = \tau_{start} \cdot \tilde{\phi}^{iter}$, where *iter* represents the number of times that τ is cooled (which is the same than the number of iterations) until that τ_{final} is reached. Therefore, $\max_{iter} = \log(\tau_{final}/\tau_{start})/\log(\tilde{\phi})$.

With respect to the stopping criteria, we stop when a certain number of iterations have been performed, the final temperature τ_{final} is reached or when the running time exceeds a user-controlled threshold.

4.3.1. Initial solution

We have defined a set of initial solutions formed by a set of lines each of them. By means of the HLSA, the algorithm computes the profit and consequently, the number of carriages, frequencies and costs. Our experiments have shown that the initial solution does not influence on the solution of our problem.

4.3.2. Operators

In the following, we describe all operators used in our algorithm.

Insert-line operator

Independently of the number of iterations carried out by the algorithm, this operator is defined as follows. First, two nodes are randomly selected from the underlying network. These nodes will represent the terminal stations for the new line. Second, the shortest path connecting these stations is defined according to the connection given by the underlying network. This path defines the itinerary of the new line if it respects the lower and upper bound on the number of nodes of a line. Otherwise, two different nodes are selected and the procedure above defined is repeated. The itinerary also defines the costs related to the infrastructure construction and the fixed operating costs. As commented, the construction cost of a line depends on its stations and its connections (edges). If an edge (resp. a station) is already considered by another line, its cost does not compute in the line cost. A line is not inserted if:

- There exists an edge in the itinerary that it does not satisfies Constraint (2.22).
- Its itinerary is contained in another existing line in the RTS.
- The itinerary of an existing line is contained in the new line.
- It is not connected with the existing lines.
- The construction cost is reached.

Extend-line operator

This method randomly selects a line ℓ to be extended. A line can be extended at the beginning or at the end of its itinerary. The extend-line operator randomly selects the place for extending the line. Once the line as well as the place have been selected, a node (not belonging to the selected line) of the underlying network is chosen. The shortest path between this node and the terminal station of line is computed. The itinerary of ℓ is extended according to the obtained path. The upper bound on the new number of nodes of ℓ is examined. The construction cost associated to ℓ is computed, if it is extended. A line is not extended if:

- It reaches the maximum number of nodes permitted.
- There exists an edge in the itinerary that it does not satisfies Constraint (2.22).

Remove-line operator

The remove-line operator randomly selects a line in the current RTS in order to be eliminated. Under determined conditions this operator is not applied. Concretely, if the network becomes a disconnected network, this operator is not considered. If finally the line is eliminated, aspects related to the infrastructure network and the costs have to be updated.

Remove-part-line operator

Remove-part-line operator randomly chooses a line to be partially eliminated. Similar to Extend-line operator, a line ℓ is partially removed at the beginning or at the end of its itinerary. To this end, a node of ℓ is randomly considered. The corresponding subpath between this node and the terminal selected station is eliminated from the itinerary of ℓ. The lower bound on the number of nodes of ℓ is checked. The construction cost associated to ℓ is computed, if it is partially removed. A line is not partially removed if its itinerary is contained in an existing line or the network becomes a disconnected network.

Remove-part-line and Extend-line operator

The idea of this operator is to apply remove-part-line and extend-part-line operators at the same iteration. Note that extend-part-line has sense if remove-part-line can be applied. Another observation is that both operators work in an independent way, that is, the selected lines can be different for each operator. The motivation for this method is to allow solutions that are discarded when these operators are independently applied.

Remove-line and Insert-line operator

First, this method removes a line by means of remove-line operator and then a new line is added using the insert-line operator, if it is possible. Basically, this method changes a line by another line.

Parameters setting and ALNS algorithm

In this section we show the pseudocode for the initial solution (see Algorithm 14) of ALNS as well as the ALN heuristic (Algorithm 15).

4.4. Computational results

In this section we presente some computational experiments for the mathematical programming model as well as experiments for our ALNS heuristic. We have considered some techniques described in Section 4.2.4 for improving the efficiency of the model and, consequently, the algorithm. Concretely, we have defined the logit function as the piecewise linear function (see Section 4.2.4) and the capacity constraint linearization. However, our problem has more non-linearities and a nonlinear programming problem (NLP) solver is needed.

Our ALNS algorithm was coded in JAVA using a standard computer. To evaluate the performance of the algorithm, we have tested several types of networks: small networks

Data: A underlying network $G_E = G(N, E)$, data related to costs and distances, the expected number of passenger for each OD pair, a set of possible headway and a minimum number of carriages δ^{min}.
Initialization phase
Given an initial network G_{RTS};
Compute the profit $z_{NET}(G_{RTS})$ with the HLSA heuristic;
$z_{curr} = z_{NET}(G_{RTS})$;
$G_{curr} := G_{RTS}$;
$G_{best} := G_{curr}$;
$z_{best} = z_{NET}(G_{best})$;
All weights $\widehat{\omega}_i$ are set to 1 and all scores π_i and o_i to 0, $i = 1, \ldots, 6$;
The parameter controlling the time, $time = 0$;
Let $n_{iter} := 0$ be the number of iterations;
Let $adjusted := false$ be a parameter indicating that τ_{start} is not adjusted.;
Result: The initial RTS, its corresponding profit and the needed parameters to start the ALNS.

Algorithm 14: Pseudocode for the initial phase in the ALNS.

with 7 nodes and 12 edges and a medium sized network with 100 nodes and 275 edges. We have compared our ALNS algorithm for the small networks against the optimal solutions obtained in the mathematical model described in Section 4.2.

A set of possible initial solutions formed by one or two lines have been considered in our ALNS. We have executed the algorithm three times for each initial solution. So, each instance has been tested twelve times. The results show that the ALNS obtains good solutions independently of the considered initial solution. We report the best solution for each test as well as average statistics for these tests in Tables 4.8 and 4.9.

A total of 72 experiments were carried out with small networks. We have compared our heuristic algorithm against the optimal solution obtained with the mathematical programming described in Section 4.2. The comparison of these results are presented in Table 4.10. In this table, we show average and the best solution provided by the ALNS. It can be observed that our algorithm was able to provide and to improve the optimal solutions for most tests, in a very small CPU. Indeed, the best solution is better than the optimal solution at all cases.

Finally, we also performed several experiments for a network of medium size. Our ALNS provided high quality solutions within reasonable computing times. It is expected that the ALNS can be powerful tool to be applied to real networks. Table 4.13 collected data related to the best solutions obtained using the ALNS and Figure 4.9 represents the best profit for each instance at each time instant.

Following sections are devoted to describe the instance generation as well as experiments on the small and medium networks.

4.4.1. Instance generation

Now, we describe the generation procedure for the instances. The small network was randomly defined by selecting 7 nodes from a grid with 16 nodes whereas the medium size network with 100 nodes was randomly selected from a grid with 225 nodes. Once the set of nodes is set, we define the edge set taking into account the Voronoi diagram in order to avoid crossings.

The following figure shows the underlying network with 7 nodes and 12 edges.

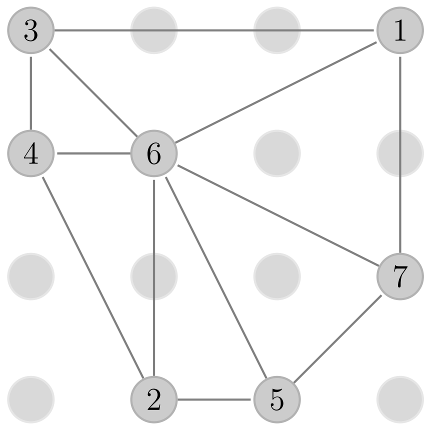


Figure 4.4.: Network with 7 nodes and 12 edges.

Figure 4.5 depicts the underlaying network with 100 nodes and 275 edges.

The distance associated to each arc is defined by means of the Euclidean distance. For the experiments, the travel times u_w^{alt} by the alternative mode, were obtained by means of the Euclidean distance and the speed of 20 km/h, whereas, the travel times into the RTS were obtained according to in-vehicle travel time, waiting and transfer times. The waiting time was supposed to be half of the corresponding time between services of lines at the origin station, whereas, the transfer time was assumed to be half time between two consecutive services at the line to transfer.

The passenger demand g_w is generated as a naturel random number following a discrete uniform distribution in an interval, which is different for each instance. So, for seed 1 with ten OD pairs, the interval is $[450, 1350]$ whereas for seed 2 with ten OD pairs is $[600, 1800]$. According to these intervals, a total of 9127 passengers was generated for seed1, 9750 for seed2. These values are provided in Table 4.2.

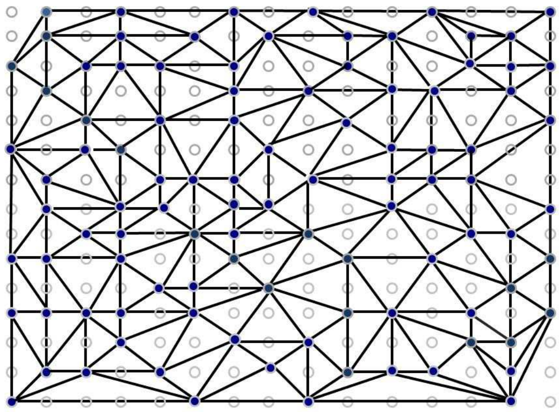


Figure 4.5.: Underlying network with 100 nodes and 275 edges.

		w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_{10}
	OD	(1,3)	(2,4)	(3,5)	(3,1)	(2,1)	(5,4)	(2,7)	(2,6)	(5,6)	(1,7)
seed1	g_w	691	523	1326	553	1237	1050	1131	1321	629	666
	u_w^{alt}	9.33	6.64	10.78	9.33	11.19	7.87	6.87	7.05	6.85	5.18
seed2	g_w	1035	695	811	1107	1598	904	915	640	964	1081
	u_w^{alt}	9.32	6.64	11.2	9.32	10.74	8.49	6.11	6.51	7.57	6.2

Table 4.2.: Demand and alternative travel time data for each OD pairs.

Similarly, the values of g_w for network with 7 nodes and 42 OD pairs were obtained according to the interval $[150, 450]$ as shows the matrices (4.31) and (4.32), for seed1 and seed2, respectively. A total of 12623 trips for the seed1 and 11442 for seed2 were generated. The passenger demand for the network with 100 nodes was randomly generated according to the interval $[1, 85]$. A total of 416498 trips were generated.

$$g_w = \begin{pmatrix} 0 & 219 & 230 & 178 & 182 & 189 & 222 \\ 412 & 0 & 421 & 174 & 281 & 440 & 377 \\ 184 & 368 & 0 & 224 & 442 & 159 & 323 \\ 429 & 258 & 305 & 0 & 363 & 374 & 361 \\ 374 & 447 & 206 & 350 & 0 & 210 & 216 \\ 431 & 395 & 170 & 392 & 352 & 0 & 289 \\ 323 & 228 & 400 & 177 & 201 & 347 & 0 \end{pmatrix} \tag{4.31}$$

$$g_w = \begin{pmatrix} 0 & 364 & 259 & 182 & 321 & 237 & 270 \\ 399 & 0 & 245 & 174 & 211 & 160 & 229 \\ 277 & 153 & 0 & 427 & 203 & 360 & 168 \\ 296 & 319 & 323 & 0 & 187 & 197 & 358 \\ 295 & 327 & 214 & 226 & 0 & 241 & 311 \\ 204 & 228 & 265 & 243 & 379 & 0 & 419 \\ 237 & 448 & 437 & 296 & 179 & 174 & 0 \end{pmatrix} \quad (4.32)$$

4.4.2. Parameters setting

In this section, we show the specific parameters for the ALNS heuristic and the input data for solving the problem.

As mentioned, there exists several parameters stated by the user: the cooling rate $\tilde{\phi}$, the final temperature τ_{final}, the block $\tilde{\varphi}$ of iterations for observing the performance of operators, the reaction factor ϖ and the parameters σ_i for increasing the scores. In our implementation, the parameters related to the temperature $\tilde{\phi}$ and τ_{final} were set to 0.9994 and 0.01, respectively. In order to consider a heterogeneous set of possible solutions for networks with 7 nodes, the parameter τ_{start} was set such that a solution 40% worse than the initial solution has 50% probability of being accepted. This assumption is motivated by the small set of possible solutions to be explored in a small network. For the medium network, this parameter was set to 33. With respect to weights, ν_i was assumed to be 1 and ϖ equal to 0.7. The parameter $\sigma_1, \sigma_2, \sigma_3$ for increasing the scores, were set to 10, 5 and 2, respectively. All these parameters were fixed independently of the size of the instance. We have tested different combinations for the parameters Max_{time}, Max_{iter} and $\tilde{\varphi}$. Depending on the size of instance, these parameters can be different. The following Table reports the different considered values for Max_{time}, Max_{iter} and $\tilde{\varphi}$.

# OD pairs	$Max_{time}(seg.)$	Max_{iter}	$\tilde{\varphi}$
10	500	10000	75
42	1000	20000	150
9900	28800	1000	20

Table 4.3.: Max_{time}, Max_{iter} and $\tilde{\varphi}$ in the ALNS.

For the network with 100 nodes and 9900 OD pairs, we have analyzed the parameters during a tuning phase. In this analysis, we have observed that the main parameter in the stop criterion is the time. This fact is due to the needed time for each iteration (there are many possible combinations of variables to be tested). Recall that the rest of parameters are adjusted in the algorithm.

The data reported in the Table 4.4 are based on the specific train model *Civia*, usually used for regional railway passengers transportation in Spain by the National Spanish Railways Service Operator (RENFE). One important characteristic of *Civia* trains is that the number of carriages can be adapted to the demand. Each *Civia* train constains two electric automotives (one at each end) and a variable number of passenger carriages. Each automotive or carriage has a maximum capacity of 200 passengers. In our experimentation, we will assume that the train is composed by only one electric locomotive (for traction purposes and null capacity) and several passengers carriages (which cannot move without a locomotive) as in Cordeau et al. (2000) and Alfieri et al. (2006). The purchase price of rolling stock used in this experimentation is also based on the real data of *Civia* trains. The price of ticket and subvention considered in our experimentation, have been taken from the newspaper (http://www.20minutos.es/noticia/2028399/0/madrid/empresas-privadas/metro-ligero/).

4.4.3. Experiments on a small network

The RTS model proposed in Section 4.2 was solved using a local non-linear optimization procedures. Concretely, we used the AlphaECP solver (see Westerlund and Lundqvist (2005)) which solves a sequence of mixed integer linear programming (MILP). This solver evaluates the non-linear constraints at each MILP solution and it adds linearizations to the MILP problem if a set of non-linear constraints do not hold. The performance of AlphaECP was compared with the GAMS solvers, Baron, Dicopt and SBB (see Lastusilta et al. (2009)). The algorithm starts from a given initial integer solution, which is depicted in Figure 4.6. Due to the non-linearities presented in our model, this kind of procedures can yield near-optimal solutions. This fact is reflected on the found optimal solutions (see Table 4.10). We provide in Table 4.5 the solutions obtained using the mathematical model. In this table the first column shows the name of the instance; the second presents the maximum number of lines; the third is the number of OD pairs; the fourth is the fare plus the subsidy; the fifth column is the name of the solution; the sixth is the CPU time and the last seventh columns show the profit and its corresponding costs.

In Table 4.6 the solutions obtained using the mathematical programming program is collected. The first column is the name of the network; the second column represent the itinerary, followed by number of nodes, number edges, headway and number of carriages.

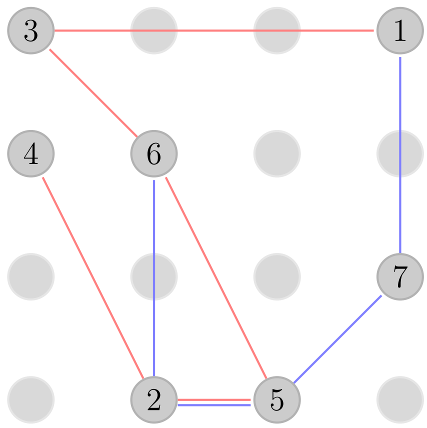


Figure 4.6.: Initial solution for the mathematical program.

Results (Mathematical programming program)						
solution	lines	itinerary	$\|N\|$	$\|E\|$	ζ_ℓ	δ_ℓ
r_1	ℓ_1	[1, 3, 6, 5, 2, 4]	6	5	4	1
	ℓ_2	[1, 7, 5, 2]	4	3	4	1
r_2	ℓ_1	[1, 7, 5, 2, 4, 6]	6	5	4	1
	ℓ_2	[1, 3, 6, 5]	4	3	4	1
r_3	ℓ_1	[1, 7, 5, 2, 6, 4]	6	5	4	1
	ℓ_2	[1, 3, 6]	3	2	4	1
r_4	ℓ_1	[5, 7, 1, 6, 4]	5	4	3	1
	ℓ_2	[7, 6, 3, 4, 2, 5]	6	5	3	1
r_5	ℓ_1	[1, 7, 5, 2, 6, 4]	6	5	3	1
	ℓ_2	[1, 3, 6]	3	2	3	1

Table 4.6.: Line configuration solutions with the mathematical program.

Now, we show the solutions obtained using the ALNS algorithm. In our experimentations, we have considered four different line configurations for the initial solution of our algorithm. The initial solutions are collected in Table 4.7. We have tested three times the algorithm for each initial solution and instance. So, each instance is tested twelve times. Average solutions for small networks as well as the best solution for each test are shown in Tables 4.8 and 4.9.

Initial solutions for the ALNS				
solution	lines	itinerary	$\|N\|$	$\|E\|$
sol_1	ℓ_1	[1, 7, 5, 2, 4]	6	5
	ℓ_2	[5, 6, 3]	4	3
sol_2	ℓ_1	[1, 7, 5, 2, 4]	6	5
	ℓ_2	[1, 3, 6, 5, 2]	5	4
sol_3	ℓ_1	[1, 3, 4, 2]	4	3
	ℓ_2	[6, 2, 5, 7]	4	3
sol_4	ℓ_1	[1, 7, 5, 2, 4]	5	4

Table 4.7.: Initial solutions for the ALNS.

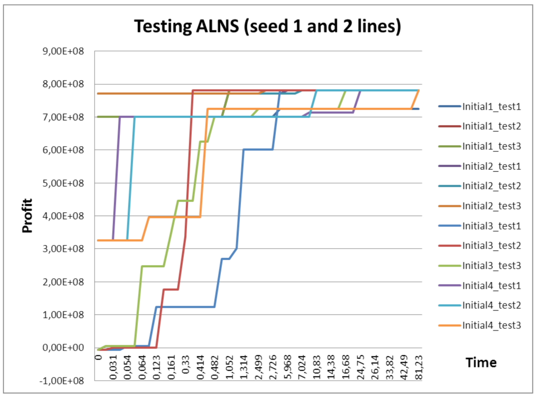
Results for network with 7 nodes and 10 OD pairs									
	instance	seed 1 with 2 lines		seed 1 with 3 lines		seed 2 with 2 lines		seed 2 with 3 lines	
	test	CPU time (seg.)	♯ best z_{NET}	CPU time (seg.)	♯ best z_{NET}	CPU time (seg.)	♯ best z_{NET}	CPU time (seg.)	z_{NET}
Initial solution 1	test1	133,226	7,8163E+08	2,2077E+01	7,8163E+08	2,8462E+01	1,1322E+09	4,9654E+02	1,0853E+09
	test2	38,35	7,8163E+08	2,7942E+01	7,8163E+08	5,4330E+00	9,9973E+08	4,3360E+00	9,9973E+08
	test3	22,218	7,8163E+08	2,2218E+01	7,8163E+08	1,1140E+00	9,9973E+08	3,0882E+02	1,1322E+09
Initial solution 2	test4	43,472	7,8163E+08	4,3472E+01	7,8163E+08	5,7143E+01	1,1322E+09	3,8631E+01	9,9973E+08
	test5	357,691	7,8163E+08	1,1462E+01	7,8163E+08	2,5349E+01	1,1322E+09	3,4124E+02	1,1219E+09
	test6	32,105	7,8163E+08	1,4282E+01	7,8163E+08	6,4691E+01	1,1322E+09	7,1153E+01	1,1322E+09
Initial Solution 3	test7	94,855	7,8163E+08	5,6969E+01	7,8163E+08	6,0629E+01	9,9973E+08	5,1324E+01	1,0853E+09
	test8	167,188	7,8163E+08	1,6719E+02	7,8163E+08	2,7180E+00	9,9973E+08	9,4430E+00	9,9973E+08
	test9	66,713	7,8163E+08	2,9627E+01	7,8163E+08	1,6063E+01	9,9973E+08	2,1230E+00	1,1322E+09
Initial Solution 4	test10	29,055	7,8163E+08	2,7621E+01	7,8163E+08	1,5010E+00	9,9973E+08	1,6860E+00	9,9973E+08
	test11	81,234	7,8163E+08	9,5964E+01	8,1924E+08	7,5670E+00	9,9973E+08	3,0788E+01	1,0853E+09
	test12	79,371	7,8038E+08	7,9371E+01	7,8038E+08	3,8700E-01	9,9973E+08	1,5965E+01	9,9973E+08
average		95,4565	7,8153E+08	4,9849E+01	7,8466E+08	2,2588E+01	1,0439E+09	1,1434E+02	1,0644E+09

Table 4.8.: Testing the ALNS with different initial solutions for small network with 10 OD pairs.

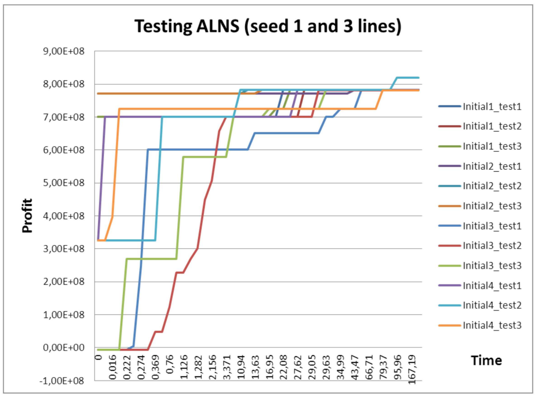
Results for network with 7 nodes and 42 OD pairs					
	instance	seed 1 with 2 lines		seed 2 with 2 lines	
	test	CPU time (seg.)	♯ best z_{NET}	CPU time (seg.)	♯ best z_{NET}
Initial solution 1	test1	5,20	1,0437E+09	10,35	8,5329E+08
	test2	71,07	1,0437E+09	2,91	7,7826E+08
	test3	38,19	1,0437E+09	20,74	8,5329E+08
Initial solution 2	test4	0,99	8,7765E+08	73,50	8,4601E+08
	test5	3,42	8,7765E+08	7,58	8,4601E+08
	test6	154,92	1,0437E+09	1,94	7,7826E+08
Initial Solution 3	test7	28,00	1,0437E+09	18,126	8,53E+08
	test8	1,26	1,0437E+09	19,583	8,53E+08
	test9	12,33	1,0437E+09	70,438	8,53E+08
Initial Solution 4	test10	64,70	1,0437E+09	134,20	8,5329E+08
	test11	61,94	1,0437E+09	1,12	8,4449E+08
	test12	64,14	1,0437E+09	46,50	8,5329E+08
average		42,18	1,0160E+09	33,92	8,3884E+08

Table 4.9.: Testing the ALNS with different initial solutions for small network with 42 OD pairs.

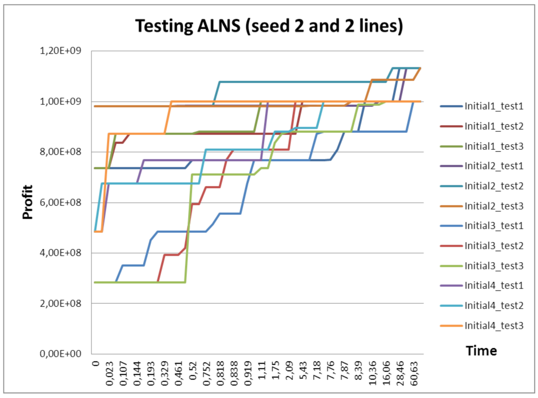
Figure 4.7 represents the different solutions provided by the ALNS for each instance at each time instant. Specifically, the horizontal axis represents the time instant whereas the vertical axis expresses the profit for each case. The horizontal axis start from 0, denoting the corresponding profit for the initial solution, to maximum time employed at each case. It can be observed that, independently of the start point, the behavior of each curve is the same.



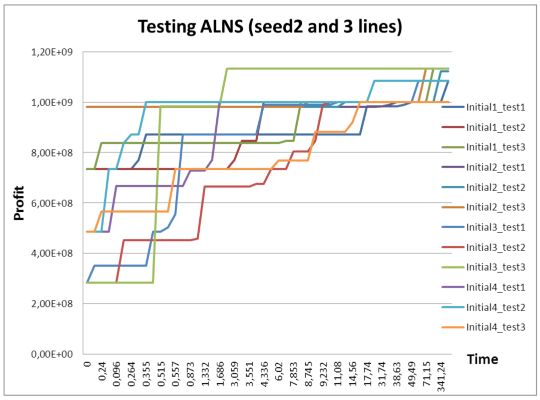
(a) Seed 1, 2 lines and 10 pairs



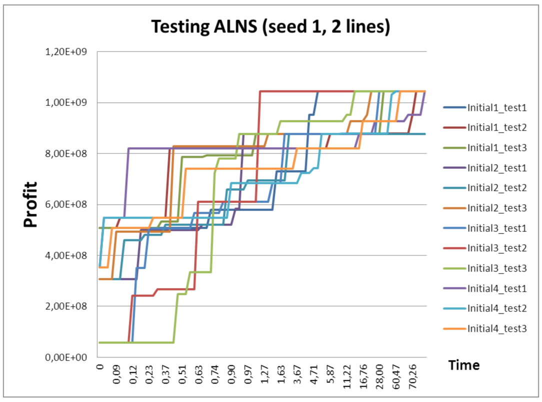
(b) Seed 1, 3 lines and 10 pairs



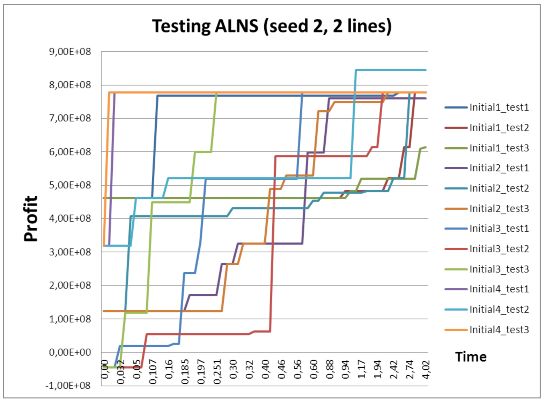
(c) Seed 2, 2 lines and 10 pairs



(d) Seed 2, 3 lines and 10 pairs



(e) Seed 1, 2 lines and 42 pairs



(f) Seed 2, 2 lines and 42 pairs

Figure 4.7.: Solutions ALNS for network with 7 nodes, 12 edges and 10 OD pairs.

We report in Table 4.10 a comparison between the optimal solutions as well as the best and the average solutions according to the ALNS heuristic. In this table the first column shows the name of the instance; the second presents the number of OD pairs; the third column is the maximum number of lines; the fourth and fifth column reports the profit and the CPU time for the optimal solution, respectively. The last six columns collect the profit, CPU time and gap for the average and best solution provided by the ALNS, respectively. It can be observed that our heuristic algorithm was able to improve the optimal solutions at all cases (see best solution columns) in a very small CPU time. The average results shows that the results provided by the ALNS are quite satisfactory. It can be note that the solution provided by ALNS were slightly worse than the exact solutions at two cases with a percent relative gaps lower than 0.07%, and ALNS took a very small time to obtain the solutions.

Detailed information on the best solution provided by the ALNS is reported in Tables 4.11 and 4.12.

Comparing ALNS and exact										
			Optimum		ALNS (average)			ALNS (best)		
instance	$\|W\|$	$\mathcal{L}_{max}$	z_{NET}	CPU time (seg.)	z_{NET}	CPU time (seg.)	gap (%)	z_{NET}	CPU time (seg.)	gap (%)
seed1	10	2	770099589.12	58.51	7,8153E+08	95.4565	-0.0186	7.8163E8	22,218	-0,0149
seed1	10	3	770099589,12	255.31	7.8467E+08	49.85	-0.0186	8,1924E+08	95,964	-0,0638
seed2	10	2	1,1204E+9	414.69	1,0439E+09	22.59	0.0683	1,1322E+09	25,349	-0,0105
seed2	10	3	1,1147E+9	1293.05	1,0644E+09	114.34	0.0451	1,1321E+09	2,123	-0,0156
seed1	42	2	921503794.43	48576.15	1,0160E+09	42,18	-0,1025	1,0437E+09	1,26	-0,1326
seed2	42	2	770150742.20	80004.69	8,3884E+08	33,92	-0,0891	8,5329E+08	10,35	-0,108

Table 4.10.: Comparing the ALNS and the optimal solution provided using the mathematical model.

Best results of ALNS											
$time_{max}$(seg.)	$\|W\|$	$\mu+\eta$	$\mathcal{L}_{max}$	instance	solution	z_{NET}	z_{REV}	$z_{CC}+z_{FOC}$	z_{FAC}	z_{RSOC}	z_{CrOC}
500	10	5	2	seed1	R_1	7.8163E8	2.589E9	2.9058E8	3.2703E7	1.4408E9	4.3284E7
500	10	5	3	seed1	R_2	8.1923E8	2.9546E9	3.5651E8	3.8354E7	1.6898E9	5.0763E7
500	10	5	2	seed2	R_3	1.1321E9	3.3728E9	3.6009E8	4.0546E7	1.7863E9	5.3664E7
500	10	5	3	seed2	R_3	1.1321E9	3.3728E9	3.6009E8	4.0546E7	1.7863E9	5.3664E7
1000	42	5	2	seed1	R_4	1.0437E9	3.106E9	3.135E8	3.770E7	1.6611E9	4.9902E7
1000	42	5	2	seed2	R_5	8.5329E8	2.8625E9	3.0722E8	3.6697E7	1.6168E9	4.857E7

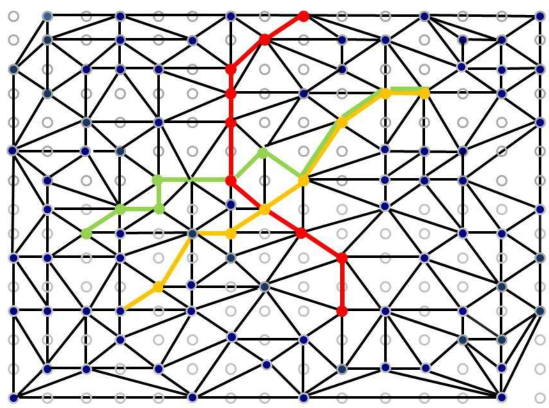
Table 4.11.: Best solutions provided from the ALNS algorithm.

Results ALNS						
solution	lines	itinerary	$\|N\|$	$\|E\|$	ζ_ℓ	δ_ℓ
R_1	ℓ_1	[3, 6, 5, 2, 4]	5	4	5	1
	ℓ_2	[2, 5, 7, 1]	4	3	4	1
R_2	ℓ_1	[3, 6, 5, 2, 4]	5	4	5	1
	ℓ_2	[2, 5, 7, 1, 3]	4	3	5	1
R_3	ℓ_1	[1, 7, 5, 2, 4]	5	4	4	1
	ℓ_2	[1, 3, 6, 5]	4	3	5	1
R_4	ℓ_1	[3, 6, 2, 5, 7, 1]	6	5	4	1
	ℓ_2	[7, 6, 4]	3	2	4	1
R_5	ℓ_1	[4, 6, 2, 5, 7, 1]	6	5	4	1
	ℓ_2	[7, 6, 3]	3	2	4	1

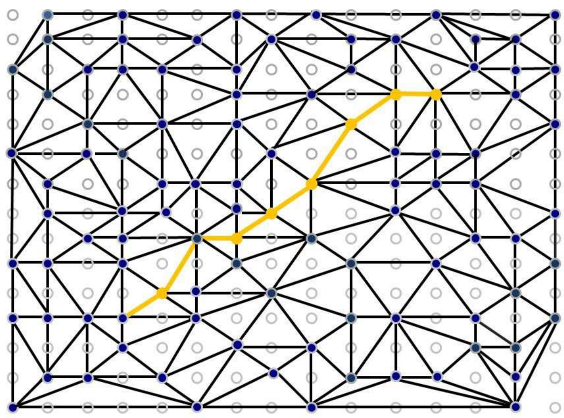
Table 4.12.: Best solution ALNS.

4.4.4. Experiments on a medium network

Due to the size of the network with 100 nodes, 275 edges and 9900 OD pairs, we allow the ALNS to iterate as times as needed in order to obtain good solutions. To this end, this network need much more time than the small networks. So, the maximum time is set to 28800 seconds (8 hours). We have tested the ALNS starting with two different initial solutions showed in Figure 4.8.



(a) Initial solution 1



(b) Initial solution 2

Figure 4.8.: Initial solutions considered in the experiments.

From these initial solutions, the ALNS provided two different configurations of lines with different profits. Detailed information on these solutions are collected in Table 4.13. In this table, first fourth columns represent the maximum time set in the ALNS, the number of OD pairs, the fare including subsidies and the maximum number of lines. The fifth columns denotes the name of each solution; the sixth column is the net profit; the seventh is the revenue; the octave column describes the construction cost plus the fixed operating cost; the ninth is the rolling stock operating cost and the last columns represent the crew operating cost.

Best results of ALNS										
$time_{max}(seg.)$	$\|W\|$	$\mu+\eta$	$\mathcal{L}_{max}$	solution	z_{NET}	z_{REV}	$z_{CC}+z_{FOC}$	z_{FAC}	z_{RSOC}	z_{CrOC}
12849,57	9900	5	2	N_1	9,7155E+09	2.5827E10	1.9135E9	3.4166E8	1.3465E10	3.9182E8
13460,25	9900	5	2	N_2	1,0597E+10	2.9798E10	2.6056E9	4.0789E8	1.5732E10	4.5475E8

Table 4.13.: Best solutions provided from the ALNS algorithm.

Figure 4.9 represents the different solutions obtained using the ALNS for each instance at each time instant. Similar to Figure 4.7, the horizontal axis represents the time instant whereas the vertical axis expresses the profit for each initial network. The horizontal axis start from 0, denoting the corresponding profit for the initial solution, to maximum time employed at each case.

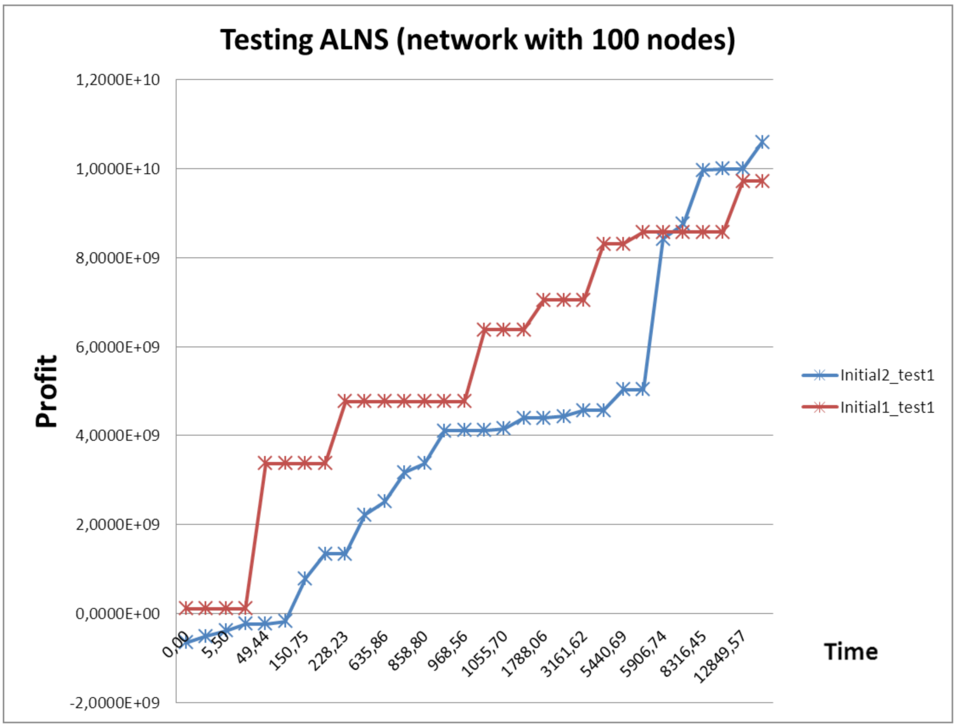
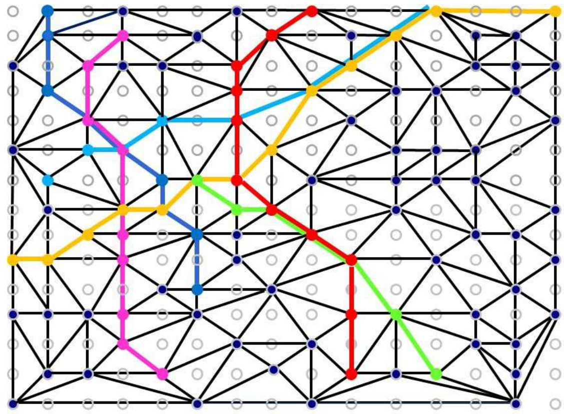


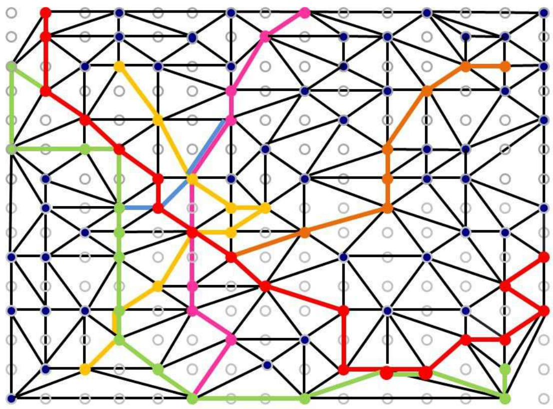
Figure 4.9.: Solutions ALNS.

It can be observed that the ALNS produces high quality solutions within a reasonable computing times for a network of medium size. Figure 4.10 depicts the solutions provided by our ALNS for the initial solution 1 and 2, respectively.



The headway and number of carriages of each line are:
red line $\zeta_\ell = 4$ and $\delta_\ell = 2$
yellow line $\zeta_\ell = 3$ and $\delta_\ell = 2$
pink line $\zeta_\ell = 4$ and $\delta_\ell = 2$
light blue line $\zeta_\ell = 4$ and $\delta_\ell = 1$
blue line $\zeta_\ell = 3$ and $\delta_\ell = 1$.

(a) Solution N1



The headway and number of carriages of each line are:
red line $\zeta_\ell = 3$ and $\delta_\ell = 2$
green line $\zeta_\ell = 5$ and $\delta_\ell = 2$
yellow line $\zeta_\ell = 4$ and $\delta_\ell = 1$
pink line $\zeta_\ell = 4$ and $\delta_\ell = 2$
blue line $\zeta_\ell = 4$ and $\delta_\ell = 1$
orange line $\zeta_\ell = 4$ and $\delta_\ell = 1$.

(b) Solution N2

Figure 4.10.: Solutions N1 and N2 provided by our ALNS.

4.5. Conclusions

In this chapter we have reviewed the existing literature on algorithms and resolutions method for the rapid transit network design problem. The main contributions of this chapter are the introduction of a mathematical programming program as well as the development of a powerful ALNS heuristic to solve real instances which integrates the algorithm described in Chapter 3. Concretely, we have also developed an algorithm composed of two heuristics. The first one is a global search heuristic called ALNS which constructs line networks by means of several methods which insert, remove, cut or extend lines of incumbent networks. The goodness of this solution is provided by the second heuristic HLSA, defined in Chapter 3, which solves the frequency and capacity setting problem. We have performed computational experiments on small and medium networks. Comparative tests on a large set of tests have shown that our heuristic can provide high quality solutions within reasonable computing times. Our ALNS is expected to be a powerful heuristic to solve real instances.

```
Data: The initial solution, a underlying network G_E = G(N, E), data related to costs and distances, the expected
      number of passenger for each pair, a set of possible headway and a minimum number of carriages δ^min.
      Data related to the ALNS algorithm such as τ_final, time, Max_time, ω̂_h, σ_i, ζ̃ and Max_iter.
while τ > τ_final & time < Max_time & n_iter < Max_iter do
    n_iter := n_iter + 1;
    G'_RTS := G_curr;
    Select an operator h according to ω̂_h, h = 1, ..., 6;
    while the selected operator cannot be applied do
        Select an operator h according to ω̂_h, h = 1, ..., 6;
        Attempts to apply the operator h;
    end
    G'_RTS is the new RTS;
    Compute z' := z_NET(G'_RTS) by using HLSA;
    Update the number of times the operator is used o_h := o_h + 1;
    if adjusted = false & z' < z_curr then
        if z_curr − z' is less than ζ̃% of z_curr then
            τ_start = (z_NET(G'_RTS) − z_NET(G_RTS))/ ln(0.5);
            adjusted = true;
        end
    end
    if z' > z_curr then
        G_curr := G'_RTS;
        z_curr := z';
        if z' > z_best then
            G_best := G'_RTS;
            z_best := z';
            Update the score: π_h = π_h + σ_1;
        else
            Update the score: π_h = π_h + σ_2;
        end
    else
        if G'_RTS is accepted by the SA criterion then
            G_curr := G'_RTS;
            z_curr := z';
            Update the score: π_h = π_h + σ_3;
        end
    end
    if n_iter is multiple of s then
        Update the weights of all operators and reset their scores;
    end
    if time > Max_time/3 & n_iter < Max_iter/3 then
        φ̃ := (τ_final/τ)^(1/(2·n_iter));
    end
    if time > Max_time/2 & n_iter < Max_iter/2 then
        φ̃ := (τ_final/τ)^(1/(n_iter/2));
    end
    if z' < z_curr & adjusted = true then
        τ := τ · φ̃;
    end
end
Result: The infrastructure network, set of lines (itinerary, frequency, number of carriages) and profit.
```

Algorithm 15: Pseudocode for the ALNS heuristic.

Parameters		
Name	**Description**	**Value**
$\hat{\rho}$	years to recover the purchase	20
ρ	number of operative hours per year	6935
ORC_{ij}	operating rail cost measured in € per year	$6 \cdot 10^4$
OSC_i	operating station cost expressed in € per year	$6 \cdot 10^4$
c_i	building cost of station at node i [€]	10^6
c_{ij}	building cost of link (i,j) [€]	$20^6 \cdot d_{ij}$
c_{loc}	costs for operating one locomotive per kilometer [€/km]	34
c_{carr}	operating cost of a carriage per kilometer [€/km]	2
c_{crew}	per crew and year for each train [€/ year]	$75 \cdot 10^3$
I_{loc}	purchase cost of one locomotive in €	$2.5 \cdot 10^6$
I_{carr}	purchase cost of one carriage in €	$0.9 \cdot 10^6$
Θ	capacity of each carriage (number of passengers)	$2 \cdot 10^2$
λ	average commercial speed in [km /h]	30
γ	maximum number of lines traversing an edge	3
N_{min}	lower bound on the number of nodes of each line	3
N_{max}	upper bound on the number of nodes of each line	6
ζ^{min}	minimum headway of each line (time between services in minutes)	3
ζ^{max}	maximum headway of each line (time between services in minutes)	20
ζ_ℓ	possible values	{3,4,5,6,10,12,15,20}
$\mu + \eta$	fare plus subsidy	5
v^{ALT}	speed in the alternative mode in [km /h]	20

Table 4.4.: Model parameters for RTNDP.

Results (Mathematical programming program)												
Instance	$\mathcal{L}_{max}$	$\|W\|$	$\mu+\eta$	solution	CPU time	z_{NET}	z_{REV}	z_{CC}	z_{FOC}	z_{FAC}	z_{RSOC}	z_{CrOC}
seed1	2	10	5	r_1	58.51	770099589.12	3.3502961E+9	332374198.94	24142451.94	47943583.88	2.1122815E+9	63454743.37
seed1	3	10	5	r_1	255.31	770099589.12	3.3502961E+9	332374198.94	24142451.94	47943583.88	2.1122815E+9	63454743.37
seed2	2	10	5	r_2	414.69	1.1203818E+9	3.6951904E+9	349090487.88	25145429.27	47445382.94	2.0903319E+9	62795359.77
seed2	3	10	5	r_3	1293.05	1.1147313E+9	3.2302753E+9	297746177.07	22064770.62	38716850.1	1.7057733E+9	51242889.84
seed1	2	42	5	r_4	48576.15	921503794.43	4.3609082E+9	360416054.51	25824963.27	65827639.02	2.9002109E+9	87124816.35
seed2	2	42	5	r_5	80004.69	770150742.20	3.3472504E+9	285906442.64	21354386.56	48938793.67	2.1561281E+9	64771932.79

Table 4.5.: Results for networks with 7 nodes and 12 edges.

Chapter 5

Robust network design

5.1. Introduction

The design of a Rapid Transit System (RTS) is a primary objective in many cities. Due to the high construction cost of a RTS, it is important to pay attention to the input data of the problem, since that, unfortunately, when a RTS is built, it is very difficult to change it. Indeed, in a realistic situation, several input data such as the origin-destination matrix, travel times by the alternative mode, costs can be uncertain. There are many RTS in which the real total cost is greater than the expected, as well as the demand estimations vary significantly after building the network. So, the uncertainty in the input data must be taken into account when we treat with real problems. This type of problem is addressed in the classic robustness problem. For instance, Bertsimas and Sim (2003, 2004) present models of robust optimization and robustness concepts that control the level of conservatism of the planner.

In this chapter we are interested in obtaining feasible solutions under uncertain circumstances. Concretely, we will analyze several robustness concepts for the rapid transit network design problem. For this purpose, we will focus on the infrastructure network design. The aim of this problem is to select from an underlying network, a set of stations and arcs connecting them such that the net profit would be maximized. In this case, since the operation of trains is not included, the net profit is defined as the difference between the revenue and the construction and fixed operating costs. Once this problem is raised, we will study the different approaches of robustness that can be applied on such problem.

The remainder is organized as follows. In Section 5.2 we presente the problem as well as the input data and variable needed to formulate the mathematical programming model. The

Section 5.3 is devoted to introduce the uncertainty sets in our problem. Robustness respect to several uncertainty set are presented in Sections 5.4 and 5.5. In Section 5.6 we treat with Light robustness. We will end with some conclusions and further work.

5.2. A mathematical programming model

In this section we present our problem as well as a mathematical programming model. The problem that we are dealing with consists of deciding where locating the stations and how connect them, taking into account a competing mode and maximizing the net profit. The demand is split according to travel times. We describe the infrastructure network by means of graphs, where stations are nodes and links between stations are edges. The model uses the notation introduced in Section 5.2.1 and the variables defined in Section 5.2.2.

5.2.1. Data and notation

The model uses the following data and notation.

- A set $N = \{1, \ldots, n\}$ of potential sites for locating stations.
- A set $A \subseteq N \times N$ of potential arcs. Each arc between two potential stations i and j will be represented by $a = (i, j)$.
- Let $E = \{\{i, j\} : (i, j) \in A,\ i < j\}$ be a set of feasible edges linking the elements of N (potential rail stretches or sections).
- An undirected graph $G_E = G(N, E)$, which represents the underlying network (from which sections and stations of lines are to be selected) and let m be the number of edges.
- For each node i, let $N(i) = \{j \in N : \{i, j\} \in E\}$ denote the set of nodes adjacent to node i.
- An undirected graph $G_{E'} = G(N, E')$, which represents the competing (private car, bus, etc.) mode network (nodes are assumed to be coincident with those of the public mode: they could represent origin or destination of the aggregated demands; however, edges are possibly different).
- For each edge $\{i, j\} \in E$, $d_{ij} = d_{ji}$ is the length of such link by the public system. This parameter can be interpreted as travel time or generalized costs needed to traverse edge $\{i, j\}$.
- A parameter uc_i representing the time spent between platforms at the station i.

- Let $W = \{w_1, \ldots, w_{|W|}\} \subseteq N \times N$ be a set of ordered origin-destination (OD) pairs, $w = (w_s, w_t)$ and r be the total number of OD pairs.
- Let $(g_w)_{w \in W}$ be the origin-destination matrix in which g_w be the expected number of trips from w_s to w_t.
- $\epsilon > 0$ is a small tolerance.
- M is a sufficiently large real number.
- A parameter η, expressing the fare paid by passenger to use the public transport, which is the same for all passengers independently of the length of their trips is introduced.
- τ denoting the passenger subsidy.
- Let c_{ij} and c_i denote the cost of building an edge $\{i, j\} \in E$ and the cost of building a node i, respectively.
- There exists an upper bound on the total construction of the RTS, denoted by C_{max}.
- Let u_w^{ALT} is the travel time using the alternative transport of OD pair w.
- A parameter ρ representing the total number of hours that a train is operating per year
- $\hat{\rho}$ denotes the horizon of years to recover the total building cost and the purchase cost of rolling stock.

5.2.2. Variables

We introduce the following variables.

- $p_w = 1$, if an OD pair w is allocated to the RTS, that is, if its fastest route in the public transport takes less time than the alternative transport u_w^{ALT}, zero otherwise.
- $y_i = 1$ if node i is selected to be a station in the RTS, zero otherwise.
- $x_{ij} = 1$ if edge $\{i, j\} \in E$ is included in the RTS, zero otherwise.
- $f_{ij}^w = 1$ if the OD pair w is assigned to the RTS and uses arc $(i, j) \in A$, zero otherwise.

5.2.3. Objective function

The objective function we consider is the net profit of the rapid transit network which is defined in Chapter 2. As mentioned, this profit is expressed as the difference between revenue and cost in terms of monetary units over a planning horizon. Recall that the total revenue is the monetary value of incomes obtained by means of the number of passengers who use the RTS during the planning horizon, times the passenger fare plus the passenger subsidy, $\eta + \mu$. Concretely, the revenue is expressed as

$$z_{REV} = (\eta + \mu)\rho\hat{\rho} \sum_{w \in W} g_w p_w. \tag{5.1}$$

In this chapter we will not consider costs related to the rolling stock operation, but only costs associated to the infrastructure network which is being built. So, in cost terms, we will take into account the operating z_{FOC} and the construction z_{BC} costs. Recall that in order to define these costs, we need several parameters already defined: OSC_i for each station i and ORC_{ij} for each edge $\{i, j\}$. So, we define the net profit associated to the rapid transit network as

$$z_{NET} = z_{REV} - (z_{FOC} + z_{BC}),$$

or equivalently,

$$z_{NET} = (\eta + \mu)\rho\hat{\rho} \sum_{w \in W} g_w p_w - \sum_{\{i,j\} \in E} \hat{c}_{ij} x_{ij} + \sum_{i \in N} \hat{c}_i \cdot y_i,$$

where, $\hat{c}_{ij} = c_{ij} + \hat{\rho} ORC_{ij}$ and $\hat{c}_i = c_i + \hat{\rho} OSC_i$.

5.2.4. Constraints

- Budget constraints

$$\sum_{\{i,j\} \in E} c_{ij} x_{ij} + \sum_{i \in N} c_i y_i \leq C_{max}. \tag{5.2}$$

- Design constraints

$$x_{ij} \leq y_i,\ i \in N, \{i, j\} \in E \tag{5.3}$$

$$x_{ij} \leq y_j,\ i \in N, \{i, j\} \in E \tag{5.4}$$

$$x_{ij} = x_{ji},\ \{i, j\} \in E. \tag{5.5}$$

- Demand conservation constraints

$$\sum_{j \in N(w_s)} f^w_{w_s j} = p_w,\ w = (w_s, w_t) \in W \tag{5.6}$$

$$\sum_{i \in N(w_t)} f^w_{i w_t} = p_w,\ w = (w_s, w_t) \in W \tag{5.7}$$

$$\sum_{i \in N(k)} f^w_{ik} - \sum_{j \in N(k)} f^w_{kj} = 0,\ w = (w_s, w_t) \in W,\ k \neq \{w_s, w_t\},\ k \in N. \tag{5.8}$$

- Location-allocation constraints

$$f^w_{ij} + p_w - 1 \leq x_{ij},\ w \in W, \{i,j\} \in E. \tag{5.9}$$

- Splitting demand constraints

$$\epsilon + \sum_{\{i,j\} \in E} d_{ij} f^{w\ell}_{ij} - u^{ALT}_w - M(1 - p_w) \leq 0,\ w \in W. \tag{5.10}$$

- Binary constraints

$$p_w, y_i, x_{ij}, f^w_{ij} \in \{0, 1\},\ i \in N, \{i,j\} \in E, w \in W. \tag{5.11}$$

Constraint (5.2) imposes an upper bound on the total cost of the network. Constraints (5.3) and (5.4) ensure that an edge is included in the RTS only if its incident nodes are also selected. In order to allow edges in both directions, Constraints (5.5) are imposed. Flow conservation for each OD pair is guaranteed by Constraints (5.6)–(5.8). In order to ensure that a demand is assigned on an edge only if it is already built, we introduce Constraints (5.9). The modal assignment is described by Constraints (5.10). The character of the variables are imposed in the remaining constraints.

5.2.5. Problem reformulation

The mathematical model can be expressed as follows.

$$(P) : \begin{cases} \max z_{NET}(x, \mathcal{D}) \\ s.t.\ \Upsilon(x, \mathcal{D}) \end{cases} \tag{5.12}$$

- $x \in \mathbb{R}^{\bar{n}}$, $\bar{n} = r + n + m + rm$ is the decision vector describing all variables in our problem. The first $r-$elements represent the variable $p_w,\ w \in W$, followed by $n-$entries describing variables $y_i,\ i \in N$, $m-$entries for the arcs $x_{ij},\ \{i,j\} \in E$ and $(mr)-$ elements corresponding to flow variables f_{ij}^w, $\{i,j\} \in E$, $w \in W$. Therefore, we can describe the decision vector as $x = (p_w, y_i, x_{ij}, f_{ij}^w) \in \mathbb{R}^{\bar{n}}$, where $w \in W, i \in N, \{i,j\} \in E$.

- $\mathcal{D} \in \mathbb{R}^{\bar{n}}$ is the input data on the model which can be described as a vector where the first $r-$elements describe $g_w(\eta+\mu)$, followed by $n-$elements representing costs associated to stations (construction and fixed operating costs), $m-$elements denoting the edge costs and $(rm)-$entries equal to zero.

- $z_{NET}(x,\mathcal{D})$ is the objective function (the net profit). The objective function is defined by means of two vectors: the decision vector x and a vector $\mathfrak{D}$ defined from $\mathcal{D}$. This last vector is expressed as the vector $\mathcal{D}$ where the components related to costs have negative signe. To fix ideas,

$$\begin{aligned} \mathfrak{D}^T &= ((\eta+\mu)g_{w_1}, \ldots, (\eta+\mu)g_{w_r}, -\hat{c}_1, \ldots, -\hat{c}_n, -\hat{c}_{e_1}, \ldots, \hat{c}_{e_m}, 0, \ldots, 0) \\ x^T &= (p_{w_1}, \ldots, p_{w_r}, y_1, \ldots, y_n, x_{e_1}, \ldots, x_{e_m}, f_{e_1}^{w_1}, \ldots, f_{e_1}^{w_r}, \ldots, f_{e_m}^{w_1}, \ldots, f_{e_m}^{w_r}), \end{aligned} \tag{5.13}$$

where $e_1, \ldots, e_m$ are the edges in E. So, the objective function can be expressed as $z_{NET}(x,\mathcal{D}) = \mathfrak{D}^T x \in \mathbb{R}$.

- $\Upsilon(x,\mathcal{D})$ represents the set of feasible solutions for the problem (5.15), i.e., $\Upsilon(x,\mathcal{D}) = \{x \in \mathbb{R}^{\bar{n}}$ satifying constraints (5.2)–(5.11)$\}$. It can be observed that $\Upsilon(x,\mathcal{D})$ describes $\bar{\bar{n}} = 1 + 2m(1+n) + r(2+n+2m) + n$ constraints.

This mathematical model can be reformulated in the way

$$(P): \begin{cases} \max z \\ s.t.\ z \leq z_{NET}(x,\mathcal{D}) \\ \Upsilon(x,\mathcal{D}). \end{cases} \tag{5.14}$$

5.3. Robustness in rapid transit network design

In this section we present the uncertainty model associated to our problem (5.15) as well as the possible uncertainty sets that can be taken into account in our problem. We consider the same type of optimization problems than Schöbel (2014). Concretely, this problem can be formulated as follows

$$P(\xi): \begin{cases} \max z_{NET}(x,\mathcal{D}(\xi)) \\ \text{s.t. } \Upsilon(x,\mathcal{D}(\xi)) \\ x \in \mathcal{X} \end{cases} \tag{5.15}$$

where $\mathcal{D}(\xi) \in \mathbb{R}^{\bar{n}}$ denotes the input parameters affected by uncertainty. We assume a given uncertainty set $\mathcal{U}$ indicating the possible values that the uncertainty can take. So, the uncertain problem corresponding to $P(\xi)$ is

$$P(\xi),\ \xi \in \mathcal{U}.$$

The underlying optimization problem is known robust counterpart associated to (5.15). The formulation of the optimization problem is depending on the selection of the uncertainty set $\mathcal{U}$. Some works analyze the mathematical programming associated to several uncertainty sets. The most common uncertainty sets in the literature (see Li et al. (2011a), Goerigk (2012), Ide (2014)) are: Interval uncertainty set, Ellipsoidal uncertainty set, Finite uncertainty, Polyhedral uncertainty set and uncertainty based on norms. We will concentrate on interval based uncertainty. Therefore, the uncertainty set can be expressed as $\mathcal{U} = \mathcal{U}_1 \times \mathcal{U}_2 \times \ldots \times \mathcal{U}_{\bar{n}}$, where each $\mathcal{U}_i$ is $[\tilde{\xi}_i - \hat{\xi}_i, \tilde{\xi}_i + \hat{\xi}_i]$, $\tilde{\xi}_i$ is the nominal scenario and $\hat{\xi}_i$ describes the maximum deviation from the nominal scenario. Depending on the type of parameters that we do not know, the uncertainty can affect to the objective function, constraints or both cases. Note that in real-world applications, the uncertainty may appear, at the same time, in both constraint and objective function.

Next section is devoted to discuss the different uncertainty sets that have sense in the rapid transit network design. For each uncertain parameter we distinguish two possible situations: all input are affected by the uncertainty or only some of them.

5.3.1. Possible uncertainty sets

The main input data in the network design are based on estimations or approximations, and as a consequence, presents uncertainty. The most interesting types of uncertainty in our problem are the demand, the alternative transport data and costs. In this chapter, we will concentrate on the uncertain demand. Note that a network is built to satisfy the demand and the demand estimation can be a bad approximation. We describe two types of uncertainty sets: one set is defined considering that all entries are affected by uncertainty whereas in the other one the uncertainty only affects a part of them.

First, we define the uncertainty set $\mathcal{U}_1$, in which the expected number of passengers g_w for each OD pair w is affected by uncertainty. Indeed, for each OD pair w, we assume that g_w varies in the interval $[\tilde{g}_w - \hat{g}_w, \tilde{g}_w + \hat{g}_w] = [g_w^{min}, g_w^{max}]$, where $g_w^{min}, g_w^{max} \in \mathbb{N}$ and $\hat{g}_w$ is the deviation from the nominal scenario $\tilde{g}$. The uncertainty set $\mathcal{U}_1$ is defined as

$$\mathcal{U}_1 = \{g \in \mathbb{N}^{|W|} : g^{min} \leq g \leq g^{max}\},$$

where g is a vector whose components are the corresponding entries of g_w, $w \in W$. The second set is defined under assumption of that only a subset of demand is affected by uncertainty. Let

$\Gamma \in \{0, \dots, |W|\}$ be a parameter denoting the number of entries affected by uncertainty and let J be an index set that contains information about the variables subject to uncertainty ($|J| \leq \Gamma$).

$\mathcal{U}_1^\Gamma = \{g \in \mathbb{N}^{|W|} : \exists J \subseteq W, |J| \leq \Gamma,$ where $\forall s \in J,\ g_s^{min} \leq g_s \leq g_s^{max},\ g_s = \tilde{g}_s, s \notin J\}$.

It can be observed that both uncertainty sets $\mathcal{U}_1$ and $\mathcal{U}_1^\Gamma$ affect to the objective function but not the constraints.

5.4. Robustness regarding uncertainty set $\mathcal{U}_1$

The robust counterpart of our problem (5.15) is

$$(RC) : \begin{cases} \max & \inf_{\xi \in \mathcal{U}} z_{NET}(x, \mathcal{D}(\xi)) \\ & \text{s.t. } \Upsilon(x, \mathcal{D}(\xi)), \xi \in \mathcal{U} \end{cases} \tag{5.16}$$

We now describe the robust counterpart associated to $\mathcal{U}_1$.

Using the formulation

$$(P) : \begin{cases} \max z \\ s.t.\ z \leq z_{NET}(x, \mathcal{D}) \\ \Upsilon(x, \mathcal{D}) \end{cases}$$

the robust counterpart of our problem taking into account uncertain demand is

$$(RC_1) : \begin{cases} \max & \inf_{g \in \mathcal{U}_1} z \\ (*) & \text{s.t. } z \leq z_{NET}(x, \mathcal{D}(g)) \\ & \Upsilon(x, \mathcal{D}(g)), g \in \mathcal{U}_1 \end{cases} \tag{5.17}$$

or equivalently

$$(RC(\mathcal{U}_1)) : \begin{cases} \max & \inf_{g \in \mathcal{U}_1} z \\ (*) & \text{s.t. } z \leq z_{NET}(x, \mathcal{D}(g)), g \in \mathcal{U}_1 \\ & \Upsilon(x, \mathcal{D}) \end{cases} \tag{5.18}$$

since that the demand only appears in the constraint $(*)$.

Due to the fact that all entries of G are positives and $p_w \geq 0$, $\forall w \in W$, the worst-case for the inequality $(*)$ is obtained by considering the lower bound g^{min} of the uncertainty set $\mathcal{U}_1$.

Lemma 5.1 *The strictly robust network design problem with uncertainty set $\mathcal{U}_1$ is equivalent to the following problem:*

$$(SR(\mathcal{U}_1)) : \begin{cases} \max z \\ z - \eta\rho\hat{\rho} \sum_{w \in W} \eta g_w^{min} \cdot p_w + \sum_{i \in N} c_i y_i + \sum_{\{i,j\} \in E} c_{ij} x_{ij} \leq 0 \\ \textit{(5.2)–(5.11)} \end{cases}$$

Proof.- Let $\mathcal{F}$ be the set of feasible solutions for $(RC(\mathcal{U}_1))$, i.e.,

$$\mathcal{F} = \{(x,z) \in \mathbb{R}^{\bar{n}+1} : z - \eta\rho\hat{\rho}\sum_{w\in W} \eta g_w^{min} p_w + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij} \leq 0,\ (5.2)\text{–}(5.11)\},$$

and let $\mathcal{F}'$ be the set of feasible solutions for (SR), i.e.,

$$\mathcal{F}' = \{(x,z) \in \mathbb{R}^{\bar{n}+1} : z - \eta\rho\hat{\rho}\sum_{w\in W} g_w p_w + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij} \leq 0,\ g_w \in \mathcal{U}_1,\ (5.2)\text{–}(5.11)\}.$$

$\subseteq$ Trivial.

$\supseteq$ Let $(x,z) \in \mathcal{F}'$ be, then, $z + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij} \leq \eta\rho\hat{\rho}\sum_{w\in W} g_w p_w$, $g_w \in \mathcal{U}_1$. On the other hand, as $g_w \in \mathcal{U}_1$, $g_w^{min} \leq g_w \leq g_w^{max}$ and $p_w \geq 0$, $0 \leq \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w \leq \eta\rho\hat{\rho}\sum_{w\in W} g_w p_w \leq \eta\rho\hat{\rho}\sum_{w\in W} g_w^{max} p_w$.

Thus, $0 \leq \max\{z + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij}, \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w\} \leq \eta\rho\hat{\rho}\sum_{w\in W} g_w p_w$ is a finite value.

- If $\max\{z + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij}, \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w\} = \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w$,

$$z + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij} - \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w \leq 0, \text{and, therefore } (x,z) \in \mathcal{F}.$$

- If $\max\{z + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij}, \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w\} = z + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij}$,

$$z + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij} x_{ij} - \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w > 0, \text{which is a contradiction}$$

with the constraint.

5.5. Robustness regarding uncertainty set $\mathcal{U}_1^\Gamma$

Using the formulation (5.15) and taking into account that the data affected by perturbations are the demands, the robust counterpart of our problem is

$$(RC(\mathcal{U}_1^\Gamma)):\begin{cases} \max & \inf_{g\in\mathcal{U}_1^\Gamma} z_{NET}(x,\mathcal{D}(g)) \\ (*) & \text{s.t. } \Upsilon(x,\mathcal{D}) \\ & g\in\mathcal{U}_1^\Gamma. \end{cases}$$

The approach by Bertsimas and Sim Bertsimas and Sim (2004) can be reinterpreted as a Strict robust in the sense of Ben-Tal and Nemirovski.

Due to the fact that the uncertainty affects to the objective function, we will consider two different sets from $\mathcal{U}_1^\Gamma$ (see Bertsimas and Sim (2003)).

1. We define the set $\mathcal{U}_{1,1}^\Gamma$ where at most Γ entries g_w takes values in $[\tilde{g}_w, \tilde{g}_w+\hat{g}_w]=[\tilde{g}_s, g_s^{max}]$, that is, at most Γ entries g_w can vary in some interval about their nominal value $\tilde{g}_w$. So, we define $\mathcal{U}_{1,1}^\Gamma=\{g\in\mathbb{N}^{|W|}:\exists S\subseteq W,\ |S|\leq\Gamma, \forall s\in S,\ g_s$ takes values in $[\tilde{g}_s,\tilde{g}_s+\hat{g}_s], g_s=\tilde{g}_s, s\notin S\}$. We are interested in finding an optimal solution valid for all scenarios, where Γ coefficients of the demand data g can change. Since that we want to maximize the objective function and each entries takes values in $[\tilde{g}_s,\tilde{g}_s+\hat{g}_s]$, the worst case is obtained on a subset of W (with at most Γ elements) which gives the worst possible deviation for the objective function. So, the worst case possible is considering $\tilde{g}$, that is, all elements are invariants. Similarly to Bertsimas and Sim (2003), the corresponding robust counterpart with uncertainty set $\mathcal{U}_{1,1}^\Gamma$ can be formulated as follows:

$$(RC(\mathcal{U}_{1,1}^\Gamma))$$

$$\max_x \min_{\{S/S\subseteq W,|S|\leq\Gamma\}}\{\eta\rho\hat{\rho}\sum_{w\in W} g_w p_w-\sum_{i\in N}c_i y_i-\sum_{\{i,j\}\in E}c_{ij}x_{ij}\}=$$
$$=\max_x\{\eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w-\sum_{i\in N}c_i y_i-\sum_{\{i,j\}\in E}c_{ij}x_{ij}\}$$
$$s.t.$$
$$(5.2)\text{–}(5.11).$$

2. We introduce the set $\widetilde{\mathcal{U}}_{1,1}^\Gamma$ where at most Γ entries g_w takes values in $(\tilde{g}_w,\tilde{g}_w+\hat{g}_w]=(\tilde{g}_s, g_s^{max}]$, that is, at most Γ entries g_w vary in some interval about their nominal value $\tilde{g}_w$. So, we define $\widetilde{\mathcal{U}}_{1,1}^\Gamma=\{g\in\mathbb{N}^{|W|}:\exists S\subseteq W,\ |S|\leq\Gamma,\forall s\in S,\ g_s\in(\tilde{g}_s,\tilde{g}_s+\hat{g}_s],\ g_s=\tilde{g}_s, s\notin S\}$. In this case, Γ coefficients of the demand data g must change. It can be noted that the

worst case is obtained on a subset of W (with at most Γ elements) which gives the worst possible deviation for the objective function. At the same way than Bertsimas and Sim (2003), the corresponding robust counterpart with uncertainty set $\widetilde{\mathcal{U}}_{1,1}^\Gamma$ can be formulated as follows:

$$(RC(\widetilde{\mathcal{U}}_{1,1}^\Gamma))$$

$$\max_x \{ \eta\rho\hat{\rho} \sum_{w\in W} \tilde{g}_w p_w - \sum_{i\in N} c_i y_i - \sum_{\{i,j\}\in E} c_{ij} x_{ij} + \eta\rho\hat{\rho} \min_{\{S/S\subseteq W, |S|\leq\Gamma\}} \{\sum_{w\in S} \hat{g}_w p_w\}\}$$

$s.t.$

(5.2)–(5.11).

Theorem 5.1 *For $\Gamma \geq 1$, the robust network design problem with uncertainty set $\mathcal{U}_{1,1}^\Gamma$, is equivalent to*

$$(SR(\mathcal{U}_{1,1}^\Gamma))$$

$$\max_x \{ \eta\rho\hat{\rho} \sum_{w\in W} \tilde{g}_w p_w - \sum_{i\in N} c_i y_i - \sum_{\{i,j\}\in E} c_{ij} x_{ij} + \theta\Gamma + \sum_{w\in W} \pi_w\}$$

$s.t.$

$\theta + \pi_w \geq -\eta\rho\hat{\rho}\hat{g}_w p_w,\ w \in W$

$\theta \geq 0$

$\pi_w \geq 0,\ w \in W$

(5.2)–(5.11).

Proof.- We convert the objective function of $(RC(\mathcal{U}_{1,1}^\Gamma))$ to a linear one as follows. Given a vector $x \in \mathbb{R}^{\bar{n}}$, we define:

$$\beta_{1,1}(x,\Gamma) = \min_{\{S/S\subseteq W, |S|\leq\Gamma\}} \{\eta\rho\hat{\rho} \sum_{w\in S} \hat{g}_w p_w\} =$$
$$\min\{\eta\rho\hat{\rho} \sum_{w\in W} \hat{g}_w p_w \alpha_w : \sum_{w\in W} \alpha_w \leq \Gamma,\ \alpha \in \{0,1\}^{|W|}\}, \tag{5.19}$$

representing the worst possible deviation for the objective function.

It can be observed that $\beta_{1,1}(x,\Gamma)$ is equal to:

$$\beta_{1,1}(x,\Gamma) = \min_{\{S/S\subseteq W,|S|\leq\Gamma\}}\{\sum_{w\in S}\hat{g}_w p_w\} = \tag{5.20}$$

$$\min\{\eta\rho\hat{\rho}\sum_{w\in W}\hat{g}_w p_w \alpha_w : \sum_{w\in W}\alpha_w \leq \Gamma,\, 0\leq\alpha_w\leq 1\}, \tag{5.21}$$

relaxing the variable α_w. So, the dual of (5.21) is:

$$\beta^*_{1,1}(x,\Gamma) = \max\{\theta\Gamma + \sum_{w\in W}\pi_w : \theta+\pi_w \geq -\eta\rho\hat{\rho}\hat{g}_w p_w;\, \pi_w,\, \theta\geq 0, w\in W\}. \tag{5.22}$$

By strong duality, since that (5.21) is feasible and bounded for all Γ, the dual (5.22) is also feasible and bounded and the solution in each case coincide.

Substituting this result on the objective function of $(RC(\widetilde{\mathcal{U}}^{\Gamma}_{1,1}))$ the proof can be concluded.

3. Now we define the set $\mathcal{U}^{\Gamma}_{1,2}$ where at most Γ entries g_w can vary in some interval about their nominal value $\tilde{g}_w$, as follows:

$$\{g\in\mathbb{N}^{|W|} : \exists S\subseteq W,\, |S|\leq\Gamma, \forall s\in S,\, g_s\in[g_s^{min},\tilde{g}_s] = [\tilde{g}_s-\hat{g}_s,\tilde{g}_s],\, g_s=\tilde{g}_s, s\notin S\}.$$

The worst case value in $[\tilde{g}_s-\hat{g}_s,\tilde{g}_s]$ is obtained when we take $\tilde{g}_s-\hat{g}_s$ on elements of $\mathcal{U}^{\Gamma}_{1,2}$. The corresponding robust counterpart can be expressed for $\mathcal{U}^{\Gamma}_{1,2}$ as follows:

$$\begin{aligned}
&(RC(\mathcal{U}^{\Gamma}_{1,2}))\\
&\max_x \min_{\{S/S\subseteq W,|S|\leq\Gamma\}}\{\sum_{w\in W}g_w p_w - \sum_{i\in N}c_i y_i - \sum_{\{i,j\}\in E}c_{ij}x_{ij}\} =\\
&\qquad\max_x\{\eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w - \sum_{i\in N}c_i y_i - \sum_{\{i,j\}\in E}c_{ij}x_{ij}\} - \qquad (5.23)\\
&\qquad - \max_{\{S/S\subseteq W,|S|\leq\Gamma\}}\{\eta\rho\hat{\rho}\sum_{w\in S}\hat{g}_w p_w\}\\
&\qquad s.t.\\
&\qquad (5.2)\text{–}(5.11).
\end{aligned}$$

4. Similarly, we define the set $\widetilde{\mathcal{U}}^{\Gamma}_{1,2} = \{g\in\mathbb{N}^{|W|} : \exists S\subseteq W,\, |S|\leq\Gamma, \forall s\in S,\, g_s\in(g_s^{min},\tilde{g}_s] = (\tilde{g}_s-\hat{g}_s,\tilde{g}_s]\}$, , that is, at most Γ entries g_w must vary in some interval about their nominal value $\tilde{g}_w$. At this case, the worst case in $(\tilde{g}_s-\hat{g}_s,\tilde{g}_s]$ is obtained on a subset of W (with at most Γ elements) which gives the worst possible deviation for the objective function.

The corresponding robust counterpart can be formulated for $\widetilde{\mathcal{U}}_{1,2}^\Gamma$ is defined as follows:

$$(RC(\widetilde{\mathcal{U}}_{1,2}^\Gamma))$$

$$\max_x\{\,\eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w - \sum_{i\in N} c_i y_i - \sum_{\{i,j\}\in E} c_{ij}x_{ij}\} - \max_{\{S/S\subseteq W,|S|\leq\Gamma\}}\{\eta\rho\hat{\rho}\sum_{w\in S}\hat{g}_w p_w\}$$

$s.t.$

(5.2)–(5.11).

Theorem 5.2 *For $\Gamma \geq 1$, the robust network design problem with uncertainty set $\widetilde{\mathcal{U}}_{1,2}^\Gamma$, is equivalent to*

$$(SR(\widetilde{\mathcal{U}}_{1,2}^\Gamma))$$

$$\max_x\{\,\eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w - \sum_{i\in N} c_i y_i - \sum_{\{i,j\}\in E} c_{ij}x_{ij} - \theta\Gamma - \sum_{w\in W}\pi_w\}$$

$s.t.$

$\theta + \pi_w \geq \eta\rho\hat{\rho}\hat{g}_w p_w,\ w \in W$

$\theta \geq 0$

$\pi_w \geq 0,\ w \in W$

(5.2)–(5.11).

Proof.- To this end, we transform the objective function of $(RC(\widetilde{\mathcal{U}}_{1,2}^\Gamma))$ to a linear one as follows. Given a vector $x \in \mathbb{R}^{\bar{n}}$, we define:

$$\beta_{1,2}(x,\Gamma) = \min_{\{S/S\subseteq W,|S|\leq\Gamma\}}\{-\eta\rho\hat{\rho}\sum_{w\in S}\hat{g}_w p_w\} =$$
$$= \min\{-\eta\hat{\rho}\rho\sum_{w\in W}\hat{g}_w p_w\alpha_w : \sum_{w\in W}\alpha_w \leq \Gamma,\ \alpha \in \{0,1\}^{|W|}\}, \tag{5.24}$$

representing the worst possible deviation for the objective function.

Obviously, $\beta_{1,2}(x,\Gamma)$ is equal to:

$$\beta_{1,2}(x,\Gamma) = \min_{\{S/S\subseteq W,|S|\leq\Gamma\}}\{-\eta\rho\hat{\rho}\sum_{w\in S}\hat{g}_w p_w\} =$$
$$= \min\{-\eta\rho\hat{\rho}\sum_{w\in W}\hat{g}_w p_w\alpha_w : \sum_{w\in W}\alpha_w \leq \Gamma,\ 0 \leq \alpha_w \leq 1\}, \tag{5.25}$$

relaxing the variable α_w. So, the dual of (5.21) is:

$$\beta^*_{1,2}(x,\Gamma) = \max\{-\theta\Gamma - \sum_{w\in W} \pi_w : \theta + \pi_w \geq \eta\rho\hat{\rho}\hat{g}_w p_w;\ \pi_w,\ \theta \geq 0, w \in W\}.$$

By strong duality, since (5.21) is feasible and bounded for all Γ, the dual (5.22) is also feasible and bounded and the solution in each case coincide.

Substituting this result on the objective function of $(RC(\mathcal{U}^{\Gamma}_{1,2}))$ the proof can be concluded.

5. We describe the set $\mathcal{U}^{\Gamma}_1$ where at most Γ entries g_w can vary in some interval about their nominal value $\tilde{g}_w$. Concretely, this set is defined as

$$\{g \in \mathbb{N}^{|W|} : \exists S \subseteq W,\ |S| \leq \Gamma, \forall s \in S,\ g_s \in [g^{min}_s, g^{max}_s] = [\tilde{g}_s - \hat{g}_s, \tilde{g}_s + \hat{g}_s]\}.$$

In $[\tilde{g}_s - \hat{g}_s, \tilde{g}_s + \hat{g}_s]$, the worst case is obtained on a subset of W (with at most Γ elements) which gives the worst possible deviation for the objective function. The corresponding robust counterpart for $\mathcal{U}^{\Gamma}_1$ is the same as in $\mathcal{U}^{\Gamma}_{1,2}$.

6. Finally, we define the set $\widetilde{\mathcal{U}}^{\Gamma}_1 = \{g \in \mathbb{N}^{|W|} : \exists S \subseteq W,\ |S| \leq \Gamma, \forall s \in S,\ g_s \in (g^{min}_s, g^{max}_s) = (\tilde{g}_s - \hat{g}_s, \tilde{g}_s + \hat{g}_s)\}$, that is, at most Γ entries g_w must vary in some interval about their nominal value $\tilde{g}_w$. In $(\tilde{g}_s - \hat{g}_s, \tilde{g}_s + \hat{g}_s)$, the worst case is obtained on a subset of W (with at most Γ elements) which gives the worst possible deviation for the objective function. The corresponding robust counterpart can be formulated as $\widetilde{\mathcal{U}}^{\Gamma}_{1,2}$.

5.6. Light robustness

The Light Robustness (LR) was defined in Fischetti and Monaci (2009) and developed in Schöbel (2014). Concretely, given an uncertain optimization problem $P(\xi)$, $\xi \in \mathcal{U}$ and a fixed nominal scenario $\bar{\xi} \in \mathcal{U}$, the problem consists of finding a solution x which is feasible for the nominal scenario $\bar{\xi}$ (i.e. $x \in \Upsilon(x, \mathcal{D}(\bar{\xi}))$) with an aceptable objective value. Let $z^* = \max\{z_{NET}(x, \mathcal{D}(\bar{\xi}))\ :\ \Upsilon(x, \mathcal{D}(\bar{\xi}))$ is satisfied$\}$ be the optimal objective value for the nominal problem and ρ be a parameter to balance the quality of the solution. So, we are interested in finding a solution x verifying that $z_{NET}(x, \mathcal{D}(\bar{\xi})) \geq z^* - \rho$ and $\Upsilon(x, \bar{\xi})$. In order to formulate the problem we need to introduce several slack variables and the reliability concept. Schöbel (2014) introduces the reliability concept as an extension of reliability defined in Ben-Tal and Nemirovski (2000). Specifically, the reliable of a solution x of $P(\xi)$ with respect to constraint i, $i = 1, \ldots, \bar{\bar{n}}$ is defined as $rel_i(x) = \max\{0, \sup_{\xi\in\mathcal{U}} F_i(x, \xi)\}$. Thus, x is reliable with respect to a vector $\Xi \in \mathcal{R}^{\bar{\bar{n}}}_+$ if and only if $rel_i(x) \leq \Xi_i$, $i = 1, \ldots, \bar{\bar{n}}$. The reliability of a solution x is defined by means of a weighted 1-norm, i.e., $rel(x) = \sum_i w_i rel_i(x)$, where each weight w_i represents the importance of constraint i.

The Light Robust Counterpart (LRC) associated to an uncertain problem $P(\xi), \xi \in \mathcal{U}$ is given as (LRC) $\min\{rel(x):\ x \in \mathcal{T}\}$, where $\mathcal{T} = \{x \in \mathbb{R}^{\bar{n}}:\ \Upsilon(x, \bar{\xi})$ and $z_{NET}(x, \bar{\xi}) \geq z^* - \rho\}$.

Depending on the uncertainty set considered (Strict robustness or Bertsimas and Sim robustness), we have different light robustness counterpart.

5.6.1. On the uncertainty $\mathcal{U}_1$

Now we will show the light robust counterpart $LR(\mathcal{U}_1)$ associated to (P) on $\mathcal{U}_1$.

Lemma 5.2 *Let $\bar{g} \in \mathcal{U}_1$ be the fixed nominal scenario and z^* be the optimal solution associated to the nominal problem. Then, for a given ρ, the light robustness approach to the rapid transit network design problem with uncertainty $\mathcal{U}_1$ corresponds the following program:*

$$(LR(\mathcal{U}_1)) : \begin{cases} \min z - \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i \\ s.t. \\ z \geq (1-\rho)z^* \\ z - \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i \geq 0 \\ (5.2)\text{–}(5.11). \end{cases}$$

Proof.- From Lemma 5.1, we know the strictly robust network design problem for $\mathcal{U}_1$. Due to that the uncertainty only affects to one constraint, only one slack variable Ξ is needed. This yields to the following light robust counterpart:

$$(LR(\mathcal{U}_1)) : \begin{cases} \min \quad \Xi \\ \quad \text{s.t.} \\ \quad z - \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij}x_{ij}\} \leq \Xi \\ \quad z \geq (1-\rho)z^* \\ \quad z, \Xi \geq 0 \\ \quad (5.2)\text{–}(5.11). \end{cases}$$

or equivalently,

$$(LR(\mathcal{U}_1)) : \begin{cases} \min z - \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i \\ \text{s.t.} \\ z \geq (1-\rho)z^* \\ z - \eta\rho\hat{\rho}\sum_{w\in W} g_w^{min} p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i \geq 0 \\ (5.2)\text{–}(5.11). \end{cases}$$

5.6.2. On the uncertainty $\mathcal{U}^{\Gamma}_{1,2}$

Next lemma shows the light robust counterpart $LR(\mathcal{U}^{\Gamma}_{1,2})$ associated to (P) on $\mathcal{U}^{\Gamma}_{1,2}$.

Lemma 5.3 *Let $\bar{g} \in \mathcal{U}^{\Gamma}_{1,2}$ be the fixed nominal scenario and z^* be the optimal solution associated to the nominal problem. Then, for a given ρ, the light robustness approach to the rapid transit network design problem with uncertainty $\mathcal{U}^{\Gamma}_{1,2}$ corresponds the following program:*

$$(LR(\mathcal{U}^{\Gamma}_{1,2})) : \begin{cases} \min z - \eta\rho\hat{\rho} \sum_{w\in W} \tilde{g}_w p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i - \beta_{1,2}(x,\Gamma) \\ s.t. \\ z \geq (1-\rho)z^* \\ z - \eta\rho\hat{\rho} \sum_{w\in W} \tilde{g}_w p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i - \beta_{1,2}(x,\Gamma) \geq 0 \\ (5.2)\text{–}(5.11). \end{cases}$$

Proof.-

As mentioned, the robust counterpart associated to (P) for $\mathcal{U}^{\Gamma}_{1,2}$:

$$(RC(\mathcal{U}^{\Gamma}_{1,2})) : \begin{cases} \max \eta\rho\hat{\rho} \sum_{w\in W} \tilde{g}_w p_w - \sum_{i\in N} c_i y_i - \sum_{\{i,j\}\in E} c_{ij}x_{ij} + \beta_{1,2}(x,\Gamma) \\ \text{s.t.} \\ \beta_{1,2}(x,\Gamma) = - \max\limits_{\{S/S\subseteq W, |S|\leq\Gamma\}} \{\eta\rho\hat{\rho} \sum\limits_{w\in S} \hat{g}_w p_w\} \\ (5.2)\text{–}(5.11). \end{cases}$$

It can be observed that this problem is equivalents to the following problem:

$$(LRC(\mathcal{U}^{\Gamma}_{1,2})) : \begin{cases} & \max z \\ & \text{s.t.} \\ (*) & z - \eta\rho\hat{\rho} \sum_{w\in W} \tilde{g}_w p_w + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij}x_{ij} - \beta_{1,2}(x,\Gamma) \leq 0 \\ & \beta_{1,2}(x,\Gamma) = - \max\limits_{\{S/S\subseteq W, |S|\leq\Gamma\}} \{\eta\rho\hat{\rho} \sum\limits_{w\in S} \hat{g}_w p_w\} \\ & (5.2)\text{–}(5.11). \end{cases}$$

Note that the only constraint affected by uncertainty is $(*)$, and, therefore, only one slack variable Ξ is needed. Thus, following the program structure presented in Schöbel (2014) and Goerigk (2012) for the light robustness counterpart, the light robust counterpart for the rapid transit network design is defined as:

$$(LRC(\mathcal{U}^{\Gamma}_{1,2})): \begin{cases} \min \quad \Xi \\ \qquad \text{s.t.} \\ \qquad z - \eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w + \sum_{i\in N} c_i y_i + \sum_{\{i,j\}\in E} c_{ij}x_{ij}\} - \beta(x,\Gamma) \leq \Xi \\ \qquad z \geq (1-\rho)z^* \\ \qquad \Xi \geq 0 \\ \qquad (5.2)\text{–}(5.11) \end{cases}$$

or equivalently,

$$(LRC(\mathcal{U}^{\Gamma}_{1,2})): \begin{cases} \min z - \eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i - \beta(x,\Gamma) \\ \text{s.t.} \\ z \geq (1-\rho)z^* \\ z - \eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i - \beta(x,\Gamma) \geq 0 \\ (5.2)\text{–}(5.11). \end{cases}$$

Theorem 5.3 *For $\Gamma \geq 1$, the light robustness design problem with $\mathcal{U}^{\Gamma}_{1,2}$, is equivalent to*

$$(RTNDP\text{–}light): \begin{cases} \min z - \eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i + \theta\Gamma + \eta\rho\hat{\rho}\sum_{w\in W}\pi_w \\ s.t. \\ z \geq (1-\rho)z^* \\ z - \eta\rho\hat{\rho}\sum_{w\in W}\tilde{g}_w p_w + \sum_{ij} c_{ij}x_{ij} + \sum_i c_i y_i + \theta\Gamma + \eta\rho\hat{\rho}\sum_{w\in W}\pi_w \geq 0 \\ \theta + \pi_w \geq \hat{g}_w p_w,\ w \in W \\ \theta \geq 0 \\ \pi_w \geq 0,\ w \in W \\ (5.2)\text{–}(5.11). \end{cases}$$

Proof.- By applying the same procedure in Theorem 5.2 for $\beta_{1,2}(x,\Gamma)$ on the program presented in Lemma 5.3, the formulation follows.

5.7. Conclusions and further work

In this chapter we have studied the problem of designing the infrastructure of a RTS taking into account a competing mode with uncertainty in the demand. For the proposed, we have described several possible uncertainty sets on this problem. We have analyzed robust approaches according to Strict robustness, Bertsimas and Sim robustness and Light robustness. In a further work we would like to introduce other robustness measure such as adjustable robustness, recovery robustness and regret on this problem, as well as, to extend the problem to the general problem presented in Chapter 2. Another interesting aspect is to consider the alternative travel time and cost as uncertain parameters and to analyze how the type of uncertainty set influence in the rapid transit network design.

Conclusions

In this thesis, we have concentrated on rapid transit network design and line planning phases. First, in Chapter 1, we have described robustness measures and we have reviewed the existing literature on the rapid transit network design measures. For this purpose, we have represented a transportation network by means of Graph and Hypergraph theory. We have described and analyzed measures in the transport context. Moreover, we have summarized works that have recently derived into several papers, which are joint works with members of our group.

On the other hand, we have proposed several mathematical programming models. A general model which integrates both rapid transit network design problems, determining the infrastructure network, set of lines, the frequency of each line as well as the capacity of services for a rapid transit network.

Moreover, we have described a new mathematical programming model for solving this problem considering different variables and constraints. Several techniques for improving the efficiency of this model have been presented. An adaptive neighborhood search heuristic (ALNS) for metro network design has been developed in Chapter 4. The results obtained in the tested networks have been satisfactory. In our experimentation we have tested several small networks with 7 nodes and 12 edges. We have compared our ALNS heuristic algorithm against the optimal solution obtained by the mathematical model, on a set of instances, obtaining good results in a very small CPU time. Furthermore, we have tested our ALNS in a medium-network with 100 nodes, 275 edges and 9900 OD pairs, producing high quality solutions within reasonable computing times.

A mathematical model for solving the problem of selecting simultaneously the frequency and the number of carriages for each line is presented in Chapter 3. To this end, we have assumed that the line network, infrastructure and itineraries of lines are given. We have described two different versions of this problem: with an unlimited number of carriages (uncapacitated) and the capacitated problem. In the first one we have described two algorithms for solving the problem: an exact and a heuristic algorithm. The heuristic technique is a procedure based on a Local Search Algorithm. The modifications are done by means of an appropriated neighborhood structure and movements. We have carried out experiments on small and medium networks.

The second one can lead to congested networks since the number of possible carriages is a limited value. We have included in-vehicle crowding function in our model, adding new variables and constraints on the first model. We have tested it on small networks showing the effect of the congestion on the solutions. The congestion impact have been studied by means of a congestion function which measures the level of in-vehicle crowding.

The input data related to costs and trains operation have been based on real data in order to calibrate all parameters that appear in our computational experiments.

Finally, we have incorporated robustness concepts in the rapid transit network design. This problem is motivated by the fact that, in realistic situations, the input data may be affected by uncertainty. Indeed, the origin-destination matrix, travel times by the alternative mode and costs can be uncertain. We have analyzed robust approaches according to Strict robustness, Bertsimas and Sim robustness and Light robustness.

List of Figures

1.1. Different layers associated to the metro network of Seville. 4
1.2. Computation of the mean connectivity for a sample network. 6
1.3. Computation of the toughness for a sample network. 8
1.4. Computation of the clustering coefficient for a small network. 12
1.5. Computation of the clustering coefficient for a small network. 12
1.6. The affected line of Barcelona metro network. 16

2.1. Cost structure. 45

3.1. Example with two lines. 81
3.2. Phase 1. Initialization. 96
3.3. Phase 2. Neighborhood. 96
3.4. Phase 3. A local search by movements. 97
3.5. Phase 4: a local search by movements. 98
3.6. Representation of 6×2-configuration. 106
3.7. Representation of 7×3-configuration. 107
3.8. Representation of 8×3-configuration. 107
3.9. Representation of 15×5-configuration. 107
3.10. Representation of 20×6-configuration. 108

C.1. Revenue and profit for first scenario of 6×2-configuration. 114
C.2. Revenue and profit for fourth scenario of 6×2-configuration. 114
C.3. Revenue and profit of each instance for first scenario of 15×5-conf. 121
C.4. Revenue and profit of each instance for fourth scenario of 15×5. 121
C.5. In-vehicle crowding function. 125
C.6. Profit for 7×3-configuration. 128

4.1. Flow chart for the ALNS and the HLSA algorithms. 136
4.2. Representation of a logit function 154

4.3. Flow chart for our ALNS algorithm. 157
4.4. Network with 7 nodes and 12 edges. 164
4.5. Underlying network with 100 nodes and 275 edges. 165
4.6. Initial solution for the mathematical program. 168
4.7. Solutions ALNS for network with 7 nodes, 12 edges and 10 OD pairs. 170
4.8. Initial solutions considered in the experiments. 172
4.9. Solutions ALNS. 173
4.10. Solutions N1 and N2 provided by our ALNS. 174

List of Tables

2.1. Possible combinations of terms to define objective functions. 48
2.2. Terms at each perspective. 49
2.3. Classification of main papers on network design 67
2.4. Characteristics of main papers addressing the network design problem 67

3.1. Model parameters for RNFCSP. 106
3.2. Number of possible trips at each instance. 109
3.3. Results for the uncapacitated problem. 110

C.1. Results for 6×2-configuration at uncapacitated case. 115
C.2. Comparison exact and heuristic for 6×2-conf. at uncapacitated case. 116
C.3. Results for 7×3 configuration at uncapacitated case. 117
C.4. Comparison exact and heuristic for 7×3-conf. at uncapacitated case. 118
C.5. Results for 8×3-configuration at uncapacitated case. 119
C.6. Comparison exact and heuristic for 8×3-conf. at uncapacitated case. 120
C.7. Results for 15x5-configuration at uncapacitated case 122
C.8. Comparison exact and heuristic for 15×5-conf. at uncapacitated case 123
C.9. Results for 20×6-configuration at uncapacitated case. 124
C.10.Comparison exact and heuristic for 20×6 at uncapacitated case. 125
C.11.Results for 6×2-configuration at capacitated case. 127
C.13.Results for uncapacitated optimal solutions of 7×3 and $\delta^{max} = 5$. 128
C.12.Results for 7×3-configuration at capacitated case. 129
C.14.Results for uncapacitated optimal solutions of 7×3 and $\delta^{max} = 6$. 130
C.15.Results for 8×3 at capacitated case for seed1 to seed5-instances. 131
C.16.Results for 8×3 at capacitated case for seed6 to seed10-instances. 132
C.17.Results for 8×3-configuration at capacitated case. 132
C.18.Results for 8×3-configuration at capacitated case. 133

4.1. Main solution methods found in the literature of rapid transit network design. . . 148
4.2. Demand and alternative travel time data for each OD pairs. 165
4.3. Max_{time}, Max_{iter} and $\tilde{\varphi}$ in the ALNS. 166
4.6. Line configuration solutions with the mathematical program. 168
4.7. Initial solutions for the ALNS. 168
4.8. Testing the ALNS for small network with 10 OD pairs. 169
4.9. Testing the ALNS for small network with 42 OD pairs. 169
4.10. Comparing the ALNS and the optimal solution 171
4.11. Best solutions provided from the ALNS algorithm. 171
4.12. Best solution ALNS. 172
4.13. Best solutions provided from the ALNS algorithm. 173
4.4. Model parameters for RTNDP. 177
4.5. Results for networks with 7 nodes and 12 edges. 177

List of Algorithms

1. Pseudocode for the naive algorithm. 90
2. The exact algorithm for RTS frequency and capacity setting problem. 91
3. HLSA heuristic. 93
4. Pseudocode for Loop II which is used in phase 3 and 4. 94
5. Pseudocode for Loop I which is used in phase 1 and 2. 95
6. Pseudocode for the phase 3. 95
7. Pseudocode for the phase 4. 95
8. Checking the nominal capacity constraint. 102
9. Testing the fixed point with nominal capacity. 102
10. An algorithm for the RTS under congestion with nominal capacity. 103
11. Checking the seat capacity constraint. 103
12. Testing the fixed point with seat capacity. 104
13. An algorithm for the RTS under congestion with seat capacity. 104
14. Pseudocode for the initial phase in the ALNS. 163
15. Pseudocode for the ALNS heuristic. 176

Bibliography

Ahuja, R., Ergun, O., Orlin, J., and Punnen, A. (2002). A survey of very large-scale neighborhood search techniques. *Discrete Applied Mathematics*, 123(1-3):75–102.

Albert, R., Jeong, H., and Barabási, A.-L. (1999). Internet: Diameter of the world-wide web. *Nature*, 401(6749):130–131.

Alfieri, A., Groot, R., Kroon, L., and Schrijver, A. (2006). Efficient circulation of railway rolling stock. *Transportation Science*, 40(3):378–391.

Ali, S., Maciejewsk, A. A., Siegel, H., and Kim, J.-K. (2003). Definition of a robustness metric for resource allocation. *Parallel and Distributed Processing Symposium, International*, 0:42.

Baaj, H. M. and Mahmassani, H. S. (1995). Hybrid route generation heuristic algorithm for the design of transit networks. *Transportation Research Part C: Emerging Technologies*, 3(1):31–50.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286:509–512.

Barabási, A.-L., Albert, R., and Jeong, H. (1999). Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications*, 272:173–187.

Barrena, E., De-Los-Santos, A., Mesa, J. A., and Perea, F. (2012). Technical proofs of paper "analyzing connectivity in collective transportation line networks by means of hypergraphs". Technical report, Universidad de Sevilla, http://grupo.us.es/transfers/.

Barrena, E., De-Los-Santos, A., Mesa, J. A., and Perea, F. (2013). Analyzing connectivity in collective transportation line networks by means of hypergraphs. *The European Physical Journal Special Topics*, 215(1):93–108.

Ben-Tal, A. and Nemirovski, A. (2000). Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical programming A*, 88:411–424.

Bertsimas, D. and Sim, M. (2003). Robust discrete optimization and network flows. *Mathematical Programming Series B*, 98:49–71.

Bertsimas, D. and Sim, M. (2004). The price of robustness. *Operational Research*, 51(1):35–53.

Blanco, V., Puerto, J., and Ramos, A. (2011). Expanding the spanish high-speed railway network. *Omega*, 39(2):138–150.

Bruno, G., Gendreau, M., and Laporte, G. (2002). A heuristic for the location of a rapid transit line. *Computers & Operations Research*, 29:1–12.

Bruno, G., Ghiani, G., and Improta, G. (1998). A multi-modal approach to the location of a rapid transit line. *European Journal of Operational Research*, 104(2):321–332.

Bruno, G. and Laporte, G. (2002). An interactive decision support system for the design of a rapid public transit networks. *Canadian Operational Research Society*, 104(2):321–332.

Bussieck, M. (1998). *Optimal Line Plans in Public Rail Transport. Ph.D.* Technical University Braunschweig.

Bussieck, M., Kreuzer, P., and Zimmermann, U. (1997). Optimal lines for railway systems. *European Journal of Operational Research*, 96:54–63.

Cadarso, L. and Marín, A. (2012). Recoverable robustness in rapid transit network design. In Aguiléra, V., Bhouri, N., Farhi, N., Leurent, F., and Seidowsky, R., editors, *Proceedings of EWGT2012. 15th Meeting of the EURO Working Group on Transportation*, volume 54 of *Procedia - Social and Behavioral Sciences*, pages 187–199. Elsevier Science, Paris.

Ceder, A. (2007). *Public Transit Planning and Operation: Theory, Modelling, and Practice.* Elsevier, Butterworth-Heinemann, Oxford.

Chvátal, V. (2006). Tough graphs and hamiltonian circuits. *Discrete Mathematics*, 306:910–917.

Claessens, M. T., van Dijk, N. M., and Zwaneveld, P. J. (1998). Cost optimal allocation of rail passenger lines. *European Journal of Operational Research*, 110(3):474–489.

Coelho, L., Cordeau, J.-F., and Laporte, G. (2012). The inventory-routing problem with transshipment. *Computers and Operations Research*, 39(11):2537–2548.

Cordeau, J.-F., Soumis, F., and Desrosiers, J. (2000). A Benders decomposition approach for the locomotive and car assignment problem. *Transportation Science*, 34(2):133–149.

Criado, R., del Amo, A. G., Hernández-Bermejo, B., and Romance, M. (2006). New results on computable efficiency and its stability for complex networks. *J. Comput. Appl. Math.*, 192:59–74.

Criado, R., Flores, J., Hernández-Bermejo, B., Pello, J., and Romance, M. (2005). Effective measurement of network vulnerability under random and intentional attacks. *Journal of Mathematical Modelling and Algorithms*, 4:307–316.

Criado, R., Hernández-Bermejo, B., Marco-Blanco, J., and Romance, M. (2007). Asymptotic estimates for efficiency, vulnerability and cost for random networks. *J. Comput. Appl. Math.*, 204:166–171.

Crucitti, P., Latora, V., Marchiori, M., and Rapisarda, A. (2002). Efficiency of scale-free networks: Error and attack tolerance. *Physica A*, pages 320–642.

Current, J., ReVelle, C., and Cohon, J. (1985). *The Median Shortest Path Problem: A Multiobjective Approach to Analyze Cost Vs. Accessibility in the Design of Transportation Networks*. Working paper series (Ohio State University. College of Administrative Science). College of Administrative Science, Ohio State University.

De-Los-Santos, A., Laporte, G., Mesa, J. A., and Perea, F. (2012). Evaluating passenger robustness in a rail transit network. *Transportation Research Part C: Emerging Technologies*, 20(1):34–46.

De Palma, A., Kilani, M., and Proost, S. (2010). The localization and pricing of mass transit stations. In Proceedings of the 12th World Conference on Transport Research Society.

Dechter, R. and Pearl, J. (1989). Tree clustering for constraint networks (research note). *Artificial Intelligence*, 38(3):353–366.

Dekker, A. H. and Colbert, B. D. (2004). Network robustness and graph topology. In *Twenty-Seventh Australasian Computer Science Conference (ACSC2004)*. ACS.

Douglas, N. and Karpouzis, G. (2005). Estimating the cost to passengers of station crowding. In 28th Australasian Transport Research Forum (ATRF).

Dufourd, H., Gendreau, M., and Laporte, G. (1996). Locating a transit line using tabu search. *Location Science*, 4:1–19.

Escudero, L. and Muñoz, S. (2009). An approach for solving a modification of the extended rapid transit network design problem. *TOP*, 17:320–324.

Euler, L. (1741). Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, 8:128–140.

Fan, W. and Machemehl, Y. (2004). A tabu search based heuristic method for the transit route network design problem. In *In 9th International Conference on Computer Aided Scheduling of Public Transport*.

Fan, W. and Machemehl, Y. (2006). Using a simulated annealing algorithm to solve the transit route network design problem. *Journal of the Transportation Engineering*, 132(2):122–132.

Feifei, Q. (2014). Investigating the in-vehicle crowding cost functions for public transit modes. *Mathematical Problems in Engineering*, 2014:1–13.

Feifei, Q. and Haicheng, J. (2012). Remodeling in-vehicle crowding cost functions for public transit: linear or non-linear? In *Transportation Research Board 91th Annual Meeting*, Washington D.C.

Fischetti, M. and Monaci, M. (2009). Light robustness. In Ahuja, R., Möhring, R., and Zaroliagis, C., editors, *Robust online large-scale optimization. Lecture note on computer science*, volume 5868, pages 61–84.

Gallo, M., Montella, B., and D, L. (2011). The transit network design problem with elastic demand and internalisation of external costs: An application to rail frequency optimisation. *Transportation Research Part C: Emerging Technologies*, 19(6):1276–1305.

García, A. and Martín, M. P. (2012). Diseño de los vehículos ferroviarios para la mejora de su eficiencia energética. In *Monografías ElecRail*, volume 6. Fundación de los Ferrocarriles Españoles.

García-Archilla, B., Lozano, A., Mesa, J. A., and Perea, F. (2013). Grasp algorithms for the robust railway network design problem. *Journal of Heuristics*, 19(2):1–24.

García-Ródenas, R., Garzón-Astolfi, A., Marín, A., Mesa, J. A., and Ortega, F. A. (2006). Analysis of the parameters of transfers in rapid transit network design. In Kroon, L. and Möhring, R., editors, *5th Workshop on Algorithmic Methods and Models for Optimization of Railways*, Schloss Dagstuhl, Germany.

Garrison, W. and Marble, D. (1962). *The structure of transportation networks*, volume 62-II. The University of Chicago Press.

Garrison, W. and Marble, D. (1964). Factor-analytic study of the connectivity of a transportation network. *Papers in Regional Science*, 12:231–238.

Gattuso, D. and Miriello, E. (2005). Compared analysis of metro networks supported by graph theory. *Networks and Spatial Economics*, 5(4):395–414.

Gendreau, M., Laporte, G., and Mesa, J. A. (1995). Locating rapid transit lines. *Journal of Advanced Transportation*, 29(2):145–162.

Gendreau, M. and Potvin, J.-Y. (2010). *Handbook of Metaheuristics.* Springer Publishing Company, Incorporated, 2nd edition.

Geraci, A. (1991). *IEEE Standard Computer Dictionary: Compilation of IEEE Standard Computer Glossaries.* IEEE Press, Piscataway, NJ, USA.

Goerigk, M. (2012). *Algorithms and Concepts for Robust Optimization. Ph.D.* Universität Göttingen.

Goerigk, M., Schachtebeck, M., and Schöbel, A. (2013). Evaluating line concepts using travel times and robustness: Simulations with the lintim toolbox. *Public Transport*, 5(3):267–284.

Goossens, J.-W., van Hoesel, S., and Kroon, L. (2004). A branch-and-cut approach for solving railway line-planning problems. *Transportation Science*, 38:379–393.

Goossens, J.-W., van Hoesel, S., and Kroon, L. (2006). On solving multi-type railway line planning problems. *European Journal of Operational Research*, 168(2):403–424.

Gribble, S. (2001). Robustness in complex systems. In *HotOS*, pages 21–26.

Guan, J., Yang, H., and Wirasinghe, S. (2006). Simultaneous optimization of transit line configuration and passenger line assignment. *Transportation Research Part B: Methodological*, 40(10):885–902.

Guihaire, V. and Hao, J. (2008). Transit network design and scheduling: A global review. *Transportation Research Part A: Policy and Practice*, 42(10):1251–1273.

Gutiérrez-Jarpa, G., Obreque, C., Laporte, G., and Marianov, V. (2013). Rapid transit network design for optimal cost and origin destination demand capture. *Computers & Operations Research*, 40(12):3000–3009.

Holmgren, J. (2007). A framework for vulnerability assessment of electric power systems. In Murray, A. and Grubesic, T., editors, *Critical Infrastructure*, Advances in Spatial Science, pages 31–55. Springer Berlin Heidelberg.

Ide, J. (2014). *Concepts of Robustness for Uncertain Multi-Objective Optimization. Ph.D.* Universität Göttingen.

Immers, L., Stada, J., Yperman, I., and Bleukx, A. (2004). Robustness and resilience of transportation networks. In *In Proceedings of the 9th International Scientific Conference MOBILITA*.

Jansson, K. (1996). Welfare aspects on organization of passenger transport. In *World Transport Research. Proceedings of the 7th World Conference on Transport Research*, volume 3.

Jara-Díaz, S. and Gschwender, A. (2003). Towards a general microeconomic model for the operation of public transport. *Transport Reviews*, 23(4):453–469.

Jeong, H., Maso, S., Barabási, A.-L., and Oltvai, Z. N. (2001). Lethality and centrality in protein networks. *Nature*, 411:41–42.

Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N., and Barabási, A.-L. (2000). The large-scale organization of metabolic networks. *Nature*, 407:651–654.

Kansky, K. (1963). *Structure of transportation networks: relationships between network geometry and regional characteristics*. The University of Chicago Press, Chicago.

Kepaptsoglou, K. and Karlaftis, M. (2009). Transit route network design problem: Review. *Journal of Transportation Engineering*, 135(8):491–505.

Kermanshahi, S., Shafahi, Y., Mollanejad, M., and Zangui, M. (2010). Rapid transit network design using simulated annealing. In *12th World Conference on Transportation Research*, Lisbon, Portugal.

Laporte, G., Marín, A., Mesa, J. A., and Ortega, F. A. (2007). An integrated methodology for the rapid transit network design problem. In Geraets, F., Kroon, L., Schöebel, A., Wagner, D., and Zaroliagis, C., editors, *Algorithmic Methods for Railway Optimization*, volume 4359 of *Lecture Notes in Computer Science*, pages 187–199. Springer, Berlin, Heidelberg.

Laporte, G., Marín, A., Mesa, J. A., and Perea, F. (2012). Designing robust rapid transit networks with alternative routes. *Journal of Advanced Transportation*, 45(1):54–65.

Laporte, G., Mesa, J. A., and Ortega, F. (2002). Locating stations on rapid transit lines. *Computers & Operations Research*, 29:741–759.

Laporte, G., Mesa, J. A., Ortega, F., and Perea, F. (2011). Planning rapid transit networks. *Socio-Economic Planning Sciences*, 45(3):95–104.

Laporte, G., Mesa, J. A., Ortega, F., and Pozo, M. (2009). Locating a metro line in a historical city centre: application to sevilla. *Journal of the Operational Research Society*, 60:1462–1466.

Laporte, G., Mesa, J. A., Ortega, F., and Sevillano, I. (2005). Maximizing trip coverage in the location of a single rapid transit alignment. *Annals OR*, 136(1):49–63.

Laporte, G., Mesa, J. A., and Perea, F. (2010). A game theoretic framework for the robust railway transit network design problem. *Transportation Research Part B: Methodological*, 44(4):447–459.

Lastusilta, T., Bussieck, M., and Westerlund, T. (2009). Comparison of some high-performance minlp solvers. Technical report, Process Design Laboratory, Abo Akademi University, Finland and GAMS Software.

Latora, V. and Marchiori, M. (2001). Efficient Behavior of Small-World Networks. *Physical Review Letters*, 87(19).

Latora, V. and Marchiori, M. (2002). Is the Boston subway a small-world network? *Physica A*, 314(1):109–113.

Latora, V. and Marchiori, M. (2004). How the science of complex networks can help developing strategies against terrorism. *Chaos Solitons Fract.*, 20:69–75.

Latora, V. and Marchiori, M. (2008). Economic small-world behavior in weighted networks. *The European Physical Journal B*, 32:249–263.

Latora, V. and Massimo, M. (2005). Vulnerability and protection of infrastructure networks. *Physical Review E*, 71(1).

Li, Z., Ding, R., and Floudas, C. (2011a). A comparative theoretical and computational study on robust counterpart optimization: I. robust linear and robust mixed integer linear optimization. *Industrial & Engineering Chemistry Research*, (18):10567–10603.

Li, Z.-C., Lam, W. H. K., Wong, S. C., and Sumalee, A. (2011b). Design of a rail transit line for profit maximization in a linear transportation corridor. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):50–70.

Lin, T. and Wilson, N. (1992). Dwell time relationships for light rail systems. *Transportation Research Record*, 1361:287–295.

Marín, A. (2007). An extension to rapid transit network design problem. *TOP*, 15(2):231–241.

Marín, A. and García-Ródenas, R. (2009). Location of infrastructure in urban railway networks. *Computers & Operations Research*, 36(5):1461–1477.

Marín, A. and Jaramillo, P. (2008). Urban rapid transit network capacity expansion. *European Journal of Operational Research*, 191(1):45–60.

Marín, A. and Jaramillo, P. (2009). Urban rapid transit network design: accelerated benders decomposition. *Annals Operational Research*, 169(1):35–53.

Marín, A., Mesa, J. A., and Perea, F. (2009). Integrating robust railway network design and line planning under failures. In Ahuja, R., editor, *Robust and Online Large-Scale Optimization*, pages 273–292. Springer.

Mauttone, A. and Urquhart, M. (2009a). A multi-objective metaheuristic approach for the transit network design problem. *Public Transport*, 1(4):253–273.

Mauttone, A. and Urquhart, M. (2009b). A route set construction algorithm for the transit network design problem. *Computers and Operations Research*, 36(8):2440–2449.

Michaelis, M. and Schöbel, A. (2009). Integrating line planning, timetabling, and vehicle scheduling: A customer-oriented approach. *Public Transport*, 1(3):211–232.

Nagurney, A. and Qiang, Q. (2009). A relative total cost index for the evaluation of transportation network robustness in the presence of degradable links and alternative travel behavior. *International Transactions in Operational Research*, 16:49–67.

Newman, M. E. J. (2001). Ego-centered networks and the ripple effect. Technical report, Technical report, Santa Fe Institute.

Ng, A. and Efstathiou, J. (2006). Structural robustness of complex networks. In *Proceedings of the international workshop and conference on network science (NetSci2006)*.

Ngamchai, S. and Lovell, D. (2003). Optimal time transfer in bus transit route network design. *Journal of the Transportation Engineering*, 129(5):510–521.

Oldfield, R. and Bly, P. (1988). An analytic investigation of optimal bus size. *Transportation Research Part B*, 22(5):319–337.

Ortúzar, J. and Willumsem, L. (1990). *Modelling Transport*. Wiley, Chichester.

Perea, F., Laporte, G., and Mesa, J. A. (2014). Adding a new station and a road link to a road-rail network in the presence of modal competition. *Transportation Research B*, 68:1–16.

Ranjbari, A., Shairat Mohaymany, A., and Mahdi Amipour, S. (2011). Transit network design: The necessity of elastic demand consideration. *Applied Mechanics and Materials*, 97–98:1117–1122.

Ropke, S. and Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4):455–472.

Roth, C., Kang, S. M., Batty, M., and Barthelemy, M. (2012). A long-time limit for world subway networks. *Journal of the Royal Society. Interface*, 9(75):2540–2550.

Schöbel, A. (2014). Generalized light robustness and the trade-off between robustness and nominal quality. *Mathematical Methods of Operations Research*, pages 1–31.

Schöbel, A. and Scholl, S. (2006). Line planning with minimal traveling time. In Kroon, L. and Möhring, R., editors, *5th Workshop on Algorithmic Methods and Models for Optimization of Railways, ATMOS 2005*, Dagstuhl, Germany. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany.

Schmidt, M. and Schöbel, A. (2010). The complexity of integrating routing decisions in public transportation models. In Erlebach, T. and Lübbecke, M., editors, *ATMOS10*, volume 14 of *OASICS*, pages 156–169. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

Schöbel, A. (2011). Line planning in public transportation: models and methods. *OR Spectrum*, 34(3):491–510.

Scott, D., Novak, D., Aultman-Hall, L., and Guo, F. (2006). Network robustness index: A new method for identifying critical links and evaluating the performance of transportation networks. *Journal of Transport Geography*, 14(3):215–227.

Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems.

Stützle, T. (1999). *Local Search Algorithms for Combinatorial Problems: Analysis, Improvements, and New Applications.* Dissertationen zur Küntlichen Intelligenz. Infix.

Tainitier, M. (1975). Statistical theory of connectivity: Basic definitions and properties. 13:391–398.

Tom, V. and Mohan, S. (2003). Transit route network design using frequency coded genetic algorithm. *Journal of the Transportation Engineering*, 129(2):186–195.

Van Nes, R. (2002). *Design of multimodal transport networks: a hierarchical approach.* PhD thesis, TRAIL Research School for TRAnsport, Infrastructure and Logistics, Delft University of Technology, The Netherlands.

Van Nes, R. and Bovy, P. H. L. (2000). Importance of objectives in urban transit-network design. *Transportation Research Record: Journal of the Transportation Research Board*, 1735:25–34.

Vuchic, V. R. (2005). *Urban Transit Operations, Planning and Economics.* Wiley, Hoboken, New Jersey.

Watts, D. and Strogatz, S. (1998). Collective dynamics of small-world networks. *Nature*, 393(6684):440–442.

Westerlund, T. and Lundqvist, K. (2005). Alpha-ecp, version 5.101. an interactive minlp-solver based on the extended cutting plane method. Technical report, Process Design Laboratory, Abo Akademi University.

Zhu, E. (2011). *Scheduled Services Network Design for Integrated Planning of Rail Freight Transportation*. PhD thesis, CIRRELT 2011-01, Montreal.

Index

$(N, \mathcal{L})$, 82
A, 40, 82, 180
B_ℓ, 86
C_{max}, 41, 181
E, 40
E', 83
$G = G(N, A')$, 40
G_E, 40
$G_{E'}$, 40, 83
I_{carr}, 42, 83
I_{loc}, 42, 83
N, 24, 40, 82, 180
$N(i)$, 40
ORC_{ij}, 41
OSC_i, 41
W, 41, 83, 181
Δ, 83
Θ, 42, 83
$\bar{\xi}_i^{w\ell\ell'}$, 155
$\bar{d}_{ij}^{\ell}$, 99
$\bar{f}_w^{RTS}$, 43
$\delta^{\max}$, 42
$\delta^{\min}$, 42, 83
δ_ℓ, 83
δ_ℓ, 84, 150
ℓ_{max}, 41
ℓ_{min}, 41
ϵ, 181
η, 83, 181
γ, 41
$\hat{\rho}$, 42, 181
κ_{ij}^{ℓ}, 98
λ, 40, 83
$\mathcal{F}$, 83
$\mathcal{L}$, 24, 41, 82
$\widehat{\mathcal{F}}$, 149
ν_ℓ, 83
ψ^{max}, 41
ψ^{min}, 41
ψ_ℓ, 43, 84
ρ, 42, 83, 181
σ, 99
τ, 43
$\tilde{f}_i^{w\ell\ell'}$, 84, 150
$\tilde{f}_{ij}^{w\ell}$, 84, 150
ϱ_{ij}^{ℓ}, 98
$\xi_{ij}^{w\ell}$, 89, 154
b_w^{ℓ}, 43
c_{carr}, 42, 83
c_{crew}, 42, 83
c_{ij}, 41
c_i, 41, 181
c_{loc}, 41, 83
d_{ij}, 40, 82

d'_{ij}, 40, 83
f_w^{ALT}, 43
f_w^{RTS}, 43, 84, 150
$f_i^{w\ell\ell'}$, 43
$f_{ij}^{w\ell}$, 43
g_w, 41, 83
h_ℓ, 42, 150
u_w, 43
u_w^{ALT}, 41, 83, 181
u_w^{RTS}, 84, 150
uc_i, 83
x_{ij}, 42, 150, 181
x_{ij}^ℓ, 42, 150
y_i, 42, 150, 181
y_i^ℓ, 42, 150
Nb_ℓ, 98

Curriculum Vitae

Academy background

- January 2011- today: Member in training in IMUS (Instituto de Matemáticas de la Universidad de Sevilla).
- Master Degree: Advanced Studies in Mathematics, 2007.
- Bachelor Degree: Mathematics, 2005.
- Java Programmer Master, 2007.
- Internal student at the Algebra Department (2001-2002).
- Internal student at the Geometry and Topology Department (2004-2005).

Fellowships

- Ph.D fellowship associated to Excellence project P09-TEP-5022 funded by Junta de Andalucía.
- Fellowship associated to the European project: Algorithms for Robust and online Railway optimization: Improving the Validity and reliAbility of Large scale systems (ARRIVAL), from June 2007 to August 2009.

Profesional experience

- June 2007- Today: PhD Student and substitute professor at the Applied Mathematics II Department (University of Seville).

- Substitute professor at the Applied Mathematics II department (University of Seville) from September 2009 to October 2010.

- January 2007-May 2007: Java Computer Programmer, Guadaltel.

Main courses & workshops received

- Arrival Fall School 2007, 14 and 15 November, Seville (Spain).

- ATMOS 2007 Conference, 10 to 12 November, Seville (Spain).

- Course on Mathematical Programming: Stochastic Integer Programming 2009, 18 and 19 June, Universidad Rey Juan Carlos, Madrid.

- Course on Mathematical Programming: Lagragean Relaxation 2009, 21 and 22 May, Universidad Rey Juan Carlos, Madrid.

- Winter School on Network Optimization 2009, 19 -23 January, Estoril (Portugal).

- XVII ELAVIO summer school 8 to 12 September 2013, Valencia (Spain). Title of the presentation: : "A new approach for solving the rapid transit network design, frequency setting and capacity".

Research stays

- Canada Research Chair in Distribution Management and the CIRRELT (Interuniversity research Centre on Enterprise Networks, Logistics and Transportation), Montrèal, Canada.

  - Visiting Ph.D. candidate, September-December 2012. An adaptive neighborhood search heuristic for metro network design.

- Departamento de Estadística e Investigación Operativa Aplicadas y Calidad. Universitat Politècnica de València, Valencia, Spain.
  - Visiting Ph.D. candidate, May-June 2013. Simultaneous frequency and capacity problem

- Institute for Numerical and Applied Mathematics, University of Göttingen, Göttingen, Germany.

  - Visiting Ph.D. candidate, August-September 2013. Robust network design.

Membership of I+D+i Projects

- Optimización de la robustez en análisis de localizaciones y diseño de redes. Ref.: MTM2009-14243. From January 2010 to December 2012.

- Metodologías para el diseño, la planificación robusta de redes y la operación mixta del transporte por ferrocarril. Aspectos intermodales y convergencia con las políticas de la UE. Ref.: P09-TEP-5022. From February 2010 to January 2013.

- Modelos de optimización aplicados a la planificación robusta y la gestión de los servicios metropolitanos de transporte público en caso de emergencia. Ref.: PT-2007-003-08CCPP. From January 2008 to December 2010.

- Algorithms for Robust and online Railway optimization: Improving the Validity and reliability of Large scale systems (Arrival) Entidad financiadora: EC (Cominsión Europea), Sixth Framework Programme. From February 2006 to January 2009. Ref.: ARRIVAL FTP2006-021235-2.

- Análisis de Localizaciones y Diseño y Operación de Redes: Aspectos de Eficiencia, Robustez y Fiabilidad. Ref.: MTM2012-37048.

Presentations at Conferences

1. Authors: De Los Santos, A. and Laporte, G. and Mesa, J.A. and Perea, F.
   Title: Simultaneous frequency and capacity setting in uncapacitated metro lines in presence of a competing mode.
   Conference: EURO working group on transportation (EWGT 2014), Sevilla, Spain.

2. Authors: De Los Santos, A. and Laporte, G. and Mesa, J.A. and Perea, F.
   Title: Simultaneous frequency and capacity setting in uncapacitated metro lines in presence of a competing mode.
   Conference: 20th Conference of the International Federation of Operational Research Societies (IFORS) 2014, Barcelona, Spain.

3. Authors: Barrena, E. and De Los Santos, A. and Mesa, J.A. and Perea, F.
   Title: Análisis y mejora de las medidas de robustez de redes de ferrocarriles
   Conference: X Congreso de Ingeniería del Transporte (CIT 2012), Granada, Spain.

4. Authors: Barrena, E. and De Los Santos, A. and Mesa, J.A. and Pere, F.
   Title: Analyzing the Transfers System in a Railway Network by Means of Hypergraphs
   Poster: Exploratory Workshop on Locational Analysis (EWLA), 2011, Sevilla, Spain.

5. Authors: De Los Santos, A. and Mesa, J.A. and Perea, F.
Title: Una panorámica de las aplicaciones de la teoría de grafos en la optimización para la planificación ferroviaria.
Conference: More 2009-Segundas Jornadas sobre modelos de optimización aplicados a la planificación robusta y la gestión de los servicios de transporte, 2009, Sevilla

6. Authors: De Los Santos, A. and Mesa, J.A. and Ortega, F.A. and Perea, F.
Title: Análisis de robustez de redes de ferrocarriles.
Conference: VIII Congreso de ingeniería del transporte (CIT 2008), A Coruña, Spain

Papers

1. A. De-Los-Santos, G. Laporte, J.A. Mesa and F. Perea. Simultaneous frequency and capacity setting in uncapacitated metro lines in presence of a competing mode. Transportation Research Procedia Journal, 2014.

2. A. De-Los-Santos, G. Laporte, J.A. Mesa and F. Perea. Simultaneous frequency and capacity setting for rapid transit systems with a competing mode and capacity constraints. 12th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems (ATMOS2014) which will be published in Dagstuhl Open Access Series in Informatics (OASIcs).

3. E. Barrena, A. De Los Santos, G. Laporte, J.A. Mesa. Transferability of collective transportation line networks from a topological and passenger demand perspective (submitted to a JCR Journal).

4. D. Canca, A. De Los Santos, G. Laporte, J.A. Mesa. A General Rapid Network Design, Line Planning and Fleet Investment Integrated Model, accepted except by two minor changes in Annals of Operations Research.

5. E. Barrena, A. De Los Santos, G. Laporte and J.A. Mesa. Passenger flow connectivity in collective transportation line networks. International Journal of Complex Systems, 3(1): 1–10, 2013.

6. E. Barrena, A. De Los Santos, J.A. Mesa, F. Perea. Analyzing connectivity in collective transportation line networks by means of hypergraphs. The European Physical Journal Special Topics, 215(1):93–108, 2013.

7. A. De Los Santos, G. Laporte, J.A. Mesa, F. Perea. Evaluating passenger robustness in a rail transit network. Transportation Research Part C: Emerging Technologies, 20(1):34–46, 2012.