FELIPE MACHADO URBAN

MODELING THE RESPONSE TO UNFORESEEN EVENTS IN TRAIN SCHEDULES

São Paulo
2024

FELIPE MACHADO URBAN

MODELING THE RESPONSE TO UNFORESEEN EVENTS IN TRAIN SCHEDULES

Graduation Thesis for Escola Politécnica da Universidade de São Paulo to obtain the Degree of Production Engineering.

São Paulo
2024

FELIPE MACHADO URBAN

MODELING THE RESPONSE TO UNFORESEEN EVENTS IN TRAIN SCHEDULES

Graduation Thesis for Escola Politécnica da Universidade de São Paulo to obtain the Degree of Production Engineering.

Supervisor:

Professor Dr. Leonardo Junqueira

São Paulo
2024

To my parents, who are my biggest role models

ACKNOWLEDGMENTS

I would like to thank everyone who has supported me personally and academically.

I thank my brother and my parents for their love and support and all their efforts to educate me and take care of me.

I would like to thank my girlfriend, Fernanda, who has supported me throughout this work, sometimes even being my trusted reviewer, helping me to improve my writing and probably being responsible for the best written parts of this thesis.

I would like to thank my friends who I met in the first semester of university and who helped me survive the first three years of graduation, Arthur, Rodrigo, Joao and Bruno.

Thank you to my friends from my exchange program, who have been like a family to me for almost two years Bia, Chris, Gaia, Kato, Paiva, Crete, Giovanni and Angelo.

And a big thank you to my supervisors Leonardo from Universidade de São Paulo and Hongjun Wu from Technische Universität Darmstadt. For giving me guidance and helping me to develop this work.

“

The important thing is not to stop questioning. Curiosity has its own reason for existing.

”

– Albert Einstein

ABSTRACT

Train transportation promises many benefits when compared to its counterpart road, bringing efficiency by transporting heavier loads faster and emitting less greenhouse gases. For this reason growing the use of trains in transportation is part of plans from many countries to lower the carbon dioxide emissions. To bring these benefits the rail system has to maintain a higher utilization of tracks through the use of schedules. However, similar to other transportation modes these trains are subject to unforeseen events, such as passengers getting sick, maintenance stops and tracks breaking. These delay the trains and make the planned schedule. The process of building a new schedule after these events minimizing the effects and avoiding propagation of delays is called rescheduling. This is currently done in many rail companies by a experienced worker without a optimization software support. This graduation thesis focuses in studying a mathematical model that optimizes the rescheduling process to assist these workers during these events. With this objective an alternative graph model was chosen to do the rescheduling of trains using mixed linear programming. Two other variations of this model are also proposed: one simplifies the model by taking the binary variables that made the original model a mixed integer linear problem and turning it into a pure linear problem; the second extending the model by including a rescheduling measure, the rerouting, by adding binary variables and making the model more complex. The three models were implemented in the general purpose programming language python, using the libraries igraph and pyomo as supporting packages, the first to built routes and the second to actually create the mathematical model for a general purpose solver to optimize. The state of the art mixed integer and linear programming solver Gurobi was used to test the models in several instances made of data available on the internet and of data from a real-world infrastructure from a company in the city of São Paulo. The tests showed that the original and the extended models might not be able to solve the problem in a bigger real-world infrastructure in the restricted time available, however, the simplified model is more than fast enough and might help those workers that do not have any assistance in this process.

Keywords: Train scheduling, Schedule recovery, Rescheduling, Train transportation, Mixed integer linear programming.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

This chapter provides a brief introduction to the research topic and its motivation, as well as an overview of the main objectives and intended contributions of this work. Sequentially, the structure of the remaining thesis is presented.

1.1 Topic and motivation

In the field of logistics, there is a transportation mode that promises to deliver significant value to the sector. This is the rail transportation mode, which offers substantial advantages over road transportation in many aspects. In 2023, DHL, a company with a comprehensive portfolio of delivery products, published an article detailing the major advantages and disadvantages of rail freight transport when compared to its counterpart road transport (STEPPER, 2023). Table 1 presents these advantages and disadvantages.

Table 1: Advantages and disadvantages of rail transportation.

Advantages	Disadvantages
Fewer greenhouse gas emissions	Low level of flexibility of time and location
Efficiency thanks to high cargo capacity	Transport costs
Safety	Noise emissions
“Plannability”	Non-uniform standards
Intermodality	Infrastructure in need of expansion

Source: (STEPPER, 2023).

The two main advantages that deserve attention in Table 1 are the lower carbon emission of trains and the “plannability” of this transportation mode. “Plannability” refers to the quality or degree of being possible to be planned. In other words, it is the capacity of something being organized and prepared for in advance. This characteristic is present in rail transportation because its operation is controlled by timetables, which prevents sudden traffic jams from happening. By establishing specific travel times for every train, timetables ensure a smooth flow of trains, unlike road transportation where such congestion is common (STEPPER, 2023). This may appeal to the customer as he can rely on trains being on time and being a comfortable way to travel.

The biggest appeal of rail transportation, however is not its “plannability”, rather it is its greenhouse gases emissions. A 2021 study made by the European Environment Agency found that a freight train emits an average of 24 grams of greenhouse gases per ton/kilometer transported (MAHNKEN, 2023). This is less than one-fifth of the emissions from road transportation. This difference in emissions is not only related to the energy source but also to the efficiency of trains.

The rail system is already predominantly powered by electricity. In Europe, approximately 80% of all freight kilometers are covered electrically (STEPPER, 2023).In Germany, however, this proportion is reduced to 62% of the state owned tracks and 54% of all tracks in the country. Interestingly, 90% of the train transportation is made using electricity powered vehicles in Germany, indicating a concentration of train travels on this electrically covered infrastructure. The country has the objective of growing the present electrical coverage of tracks from 62% to 70% of the state-owned tracks by 2025, with the goal of further reducing the greenhouse gases emissions (ALLIANZ PRO SCHIENE, 2024).

Furthermore, trains can transport larger and heavier loads than its road counterparts. This can result in greater efficiency, both environmentally and economically, over long-distance trips. This results in improvements in both logistics costs and in carbon emissions, which are notable advantages when using rail instead of road transportation (STEPPER, 2023).

Moreover, to achieve the European Green Deal initiated by the EU Commission, Europe has a target to raise the rail share of freight transportation to 30% by 2030. Now it is only responsible for 16.8%, with a heterogeneous distribution from 3.2% in Greece to 64.7% in Lithuania (MAHNKEN, 2023). This indicates a big demand for growth in this transportation mode. However, this growth is being slowed down by its disadvantages and the present situation of the infrastructure, which has effects on punctuality.

The two main disadvantages of rail transportation that deserve attention are the low level of flexibility of time and the infrastructure in need of expansion and maintenance work. The use of timetables in operations has a drawback: a resulting inflexibility in time. To follow a schedule, trains must adhere to small time windows in order to carry out their journey (STEPPER, 2023). Add that to the fact that not everything can be known in advance, and all transportation modes face uncertainty in their operation, including rail transportation, which may result in unforeseen delays or even cancellations, which may make the planned timetable infeasible. This means that other trains will also become delayed, making trains less attractive as their punctuality becomes worse and consequently their service level reduces. Examples of these unforeseen events include late train arrivals or departures, unexpected track maintenance, vehicle breakdowns or even bad weather (VISENTINI et al., 2014).

Infrastructure is also slowing down rail transportation. In many countries there is not only a problem of tracks in need for expansion, but also the existing infrastructure is outdated and need urgent maintenance in addition to the expansion measures (STEPPER, 2023). For example, in Germany, 16% of the existing tracks are out of service (BALSER, 2018) and because of maintenance work trains are becoming frequently delayed (DEUTSCHE WELLE, 2024). On the other hand, Brazil has very few tracks, which can hardly meet demand, making them frequently crowded or out of service (G1, 2024). This lack of available tracks makes busy areas even more inflexible in time, as a result of a even higher demand, leading to busier tracks which are more susceptible to propagating delays.

With all that said, rail transportation is a highly attractive mode of transportation and should be used more frequently due to its advantages. However, the disadvantages discussed contribute to the current scenario of low service levels and unreliable timetables. Therefore, more efficient operations management is needed to ensure that the advantages outweigh the drawbacks, facilitating a transition towards more widespread use of rail transportation for both people and goods.

An important task is to improve the process of rescheduling, which involves finding a new feasible schedule that satisfies operational and safety constraints while also minimizing an objective in the time-constrained environment of operations (FANG; YANG; YAO, 2015). This can enhance time flexibility by providing a fast and reliable way to create a new schedule when the original has become infeasible.

Nowadays, most rescheduling is done by an experienced worker called *dispatcher* in an operation center, using only some rule of thumb or a contingency plan (QU; CORMAN; LODEWIJKS, 2015; GHAEMI; CATS; GOVERDE, 2017). Alternatively, some cases use an automated system, however, in the literature, there are only two cases of this implementation (LAMORGESE; MANNINO, 2015).

This work seeks to further evaluate the possibility of implementing optimization models to assist dispatchers in the rescheduling process, hopefully making this process more efficient and consequently improving the current scenario in train operations.To do so, the author communicated with a company of urban trains of São Paulo, the Companhia Paulista de Trens Metropolitanos (CPTM), to further understand the problem and applicability of algorithms, as well as to obtain data to test the algorithm in real-world cases.

1.2 Research questions

With the exposed motivation and context in mind, the following research questions are made:

1. How is the real time rescheduling problem modeled and optimized in the literature?
2. Can these optimization models solve complex instances, including real-world ones, in real time?
3. Is there a way to improve these process?

To address these questions, the following tasks are proposed for the development of this work:

I Conduct a literature research to develop a better understanding of the topic and identify the main factors that influence rescheduling.

II Study a mathematical model of the rescheduling problem to address question 1.

III Implement a model found in the literature, as well as two variations of that model, with the objective of improving the solution. This step prepares for the next task and attempts to address question 3.

IV Conduct experiments using instances, available on the internet and from real-world examples, to determine if the models and the exact solution can solve the problem within the time limit, thus addressing question 2.

The objectives of this thesis are:

I To study the rescheduling problem, focusing in a real-world application.

II To translate the problem into a mathematical optimization model that minimizes the effects of perturbations in train operations.

III To implement different solution approaches in computational language and compare the solutions in data available in the literature and the CPTM context.

This thesis hopes to bring the following contributions:

1. Testing if these solution approaches implemented can solve the rescheduling problem in a real-world scenario.

2. Further improving the model found in the literature.

1.3 Structure

The structure of this work is organized into 6 chapters, according to the following structure:

- Chapter 1 - Introduction: Presents the motivation for studying the subject, the proposed activities, and the questions they aim to answer.
- Chapter 2 - Theoretical Background: Provides the theoretical knowledge necessary to understand the problem based on the literature research.
- Chapter 3 - Problem Description: Introduces the problem studied and the real-world examples used to test the solution.
- Chapter 4 - Mathematical Model: Describes the mathematical models implemented in this thesis.
- Chapter 5 - Model implementation and solution approaches: Details the additional approaches, alongside the mathematical modeling, necessary to the solution.
- Chapter 6 - Computational experiments: Details the computational tests conducted comparing results from the different approaches proposed.
- Chapter 7 - Conclusions: Summarizes the thesis, answers the research questions based on the work, and identifies possible critiques to the methodology as well as opportunities for future research.

2 THEORETICAL BACKGROUND

The information presented in this chapter is derived from a literature research employing the snowballing method. The research started with the reading of a literature review on the subject of train rescheduling (CACCHIANI et al., 2014), and then proceeded with the reading of numerous other papers that were either cited in this review or were considered relevant papers in the subject, because more recent and contained one or more key words such as trains scheduling, rescheduling, operations, etc. This chapter is divided into 3 sections: train operations theory, categories of unforeseen events in rail track and rescheduling theory.

2.1 Train operations theory

In this section the knowledge needed to understand how trains operations work is briefly presented. This will cover the information gathered from papers on the scheduling and rescheduling of trains.

2.1.1 Resources

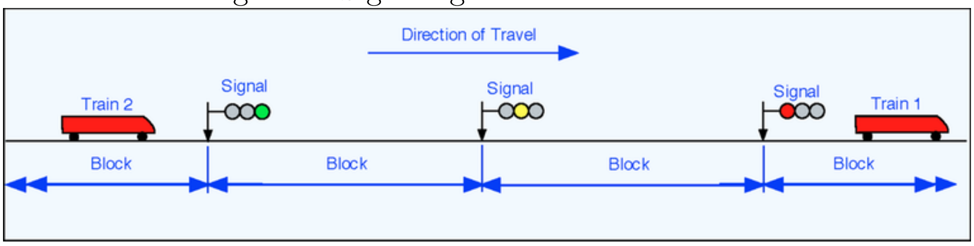
Resources are the main source of constraints in most of optimization problems since all solutions must be restricted to using only the available resources. Train management is no different. With that said, the two main resources in a rail system are the track infrastructure and the trains. Because rail tracks are a shared resource, they impose tighter restrictions to the problem as the infrastructure of other transportation modes (VISENTINI et al., 2014).

Track infrastructure can be divided into elements, which are rail track, rail unit, stations and communication (NARAYANASWAMI; RANGARAJ, 2011). The rail track is composed from multiple rail units that can be called by rail segments, track section, track segments, rail blocks or block section. Stations also contain these track units, but are special because are points where trains interact with their clients deserving special attention. Communication devices called signals are usually placed in both the beginning and end of every block section (D'ARIANO et al., 2008).

2.1.1.1 Signaling and blocking

To start to explain this resources an illustration can help clarify the idea. Figure 1 shows how this division of rail tracks works in a train line. This illustration is taken from a very informative website on trains operations and shows that in every end of a block section of rail track there is a signaling device that is used to communicate to the conductor the status of the next track, if it is occupied or free (PRC Rail Consulting Ltd, 30/01/2023).

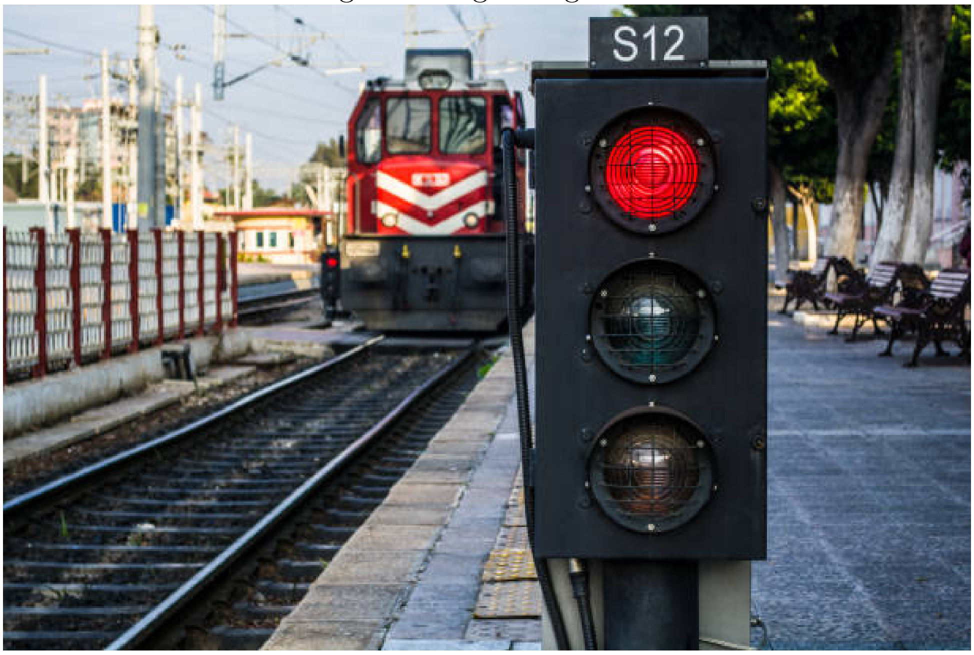
Figure 1: Signaling and blocks illustration



Source: PRC Rail Consulting Ltd (30/01/2023).

Each block section can have a different length and for safety reasons can only have one train traveling through it at a time on the timetable. To further ensure safety the signaling system composed by the signaling devices are used (LUSBY et al., 2011). These devices can display either red, yellow or green lights. Red indicates that the next block is either out of service or occupied by another train, which means a train can not enter it. Yellow means the next block section is empty, but the one after it is occupied by another train, meaning the speed is also limited for safety reasons. Green indicates that the two subsequent block sections are empty and the train can run through it at maximum speed if needed (D'ARIANO; PACCIARELLI; PRANZO, 2007). An example of these possibilities is illustrated in Figure 1.

Figure 2: Signalling device



Source: Toker (2024).

Figure 2 shows a real-world example of a signaling device on a CPTM track section. These devices assist train traffic by communicating the track occupancy status to the conductor, thereby indicating the appropriate speed limits (CAPPART; SCHAUS, 2017). This system is designed to enhance safety in train operations. When two or more trains attempt to occupy the same track simultaneously, a conflict arises (BAI et al., 2023), representing the risk of collision. Such conflicts must be resolved by a dispatcher through a rescheduling process (FANG; YANG; YAO, 2015).

With this division of the rail track infrastructure the rail system can be represented as a collection of block sections connected to each other. Their attributes would be length and their connections. A route of a train is then represented as a list of ordered block sections through which it travels through.

2.1.1.2 Resulting problem elements

To guarantee that the solution is adherent to the available resources and operations run smoothly with the safety measure, of only one train permitted to occupy each block section, the following restrictions are imposed to the model: Minimal travel time and Blocking.

The first constraint refers to the minimal time a train takes to travel through a block section. That is the result of a function that involves both the speed a train can travel through a block, and the length of the block. These can be dependent on both the train and the track section, for example a train can be slower than others or the track can have a curve which would also require trains to travel slower through it, or the track section could be bigger in length, making the minimal time the train takes to travel through this block higher (GODWIN; GOPALAN; NARENDRAN, 2007).

This constraint usually uses a simplified method to model speed. It models the speed profile of a train as a uniformly fixed function equal to the average speed that the train can achieve in a block section. This must be slower than the maximum speed that the train can travel through that section of track (NARAYANASWAMI; RANGARAJ, 2011; REYNOLDS, 2021). In other words, by imposing that the train takes more time than the time required to run through the block with the maximum speed, the constraint ensures that the train travels in a speed smaller than the maximum.

This minimal travel time starts when the front of the train enters the block section and ends when that train has entered the next block section. The limit of the speed of that train is its maximal speed, so the minimal travel time is the length of that block section divided by this maximal speed. Trains normally travel with their scheduled speed, and increasing that speed can be a measure to recovering a disturbed schedule, and the maximum speed of the resources is the limit to the new scheduled speed (CORMAN et al., 2012a).

Then the second constraint enforces that in each block section only one train is allowed to be traveling through it in a given time. This is required, only during the scheduling, due to safety reasons as was explained in the previous Subsection 2.1.1.1. During operation the signaling device communicates this occupation to the conductor and trains may travel in the same block section but with reduced speed. With that said this restriction of only one train occupying a block is translated into the model of the problem of rescheduling, with the purpose of avoiding conflicts and consequently possible collisions between trains.

The resources do also describe important parameters for the model, which are Setup time, Dwell time and Priority. Setup time is the time between the front of the train entering the next block section and the back of the train leaving the previous block section (CORMAN et al., 2012a). This setup time enforces that trains run with a safe distance between them at all times. This distance divided by the speed of the train is also refereed to as headway. This may vary according to the safety standards with which the company operates the rail system.

Then dwell time is the time trains spend in a station. The lower limit to this time, when considering a passenger train, is the time passengers take to board or leave the train. There can also be an upper limit to this parameter as an attempt to increase the utilization of a station capacity (LUSBY et al., 2011). However, usually only the minimum dwell time is imposed so passengers can board the train with ease, while the dwell time is minimized as an effect of the objective function.

Priority is an element connected to trains, which reflects their importance as a service to their clients. In a system where both passenger and freight share the same infrastructure, a delay brings different consequences to each client, and passengers tend to have a higher priority when compared with freight trains (QU; CORMAN; LODEWIJKS, 2015; BAI et al., 2023). The higher the priority, the more the delay will be felt and the higher the associated cost when rescheduling. With that said, priority is usually modeled as a delay cost that multiplies a priority factor to the delay experienced by the train.

2.1.2 Management structure

This subsection will address some relevant points in the management structure of rail systems to the rescheduling problem. It is divided in timetables, planning levels and dispatchers coordination.

2.1.2.1 Timetables

Due to its importance to this work timetable has to be defined. It is the name given to trains schedule. This means that timetabling is to provide a time value to departure and arrival for a set of trains in a rail system. Moreover, the timetable is supposed to be an easy to understand summary of all trains that travel through the rail system, specifying all station on a train route, and the arrival and departure time of that train for each block section in its path. By doing so it also indicates traveling time and dwell time on stations (NARAYANASWAMI; RANGARAJ, 2011).

Another definition could be that a timetable specifies the paths of trains including track lines used, junctions and stations traveled and the various interactions between trains, including the planned time for all events, which comprises the time trains arrive in each block section (HARROD, 2012). This means planning a schedule can be divided into two tasks: first defining the path of the trains, then defining the arrival and departure time from each block section in that path. Both tasks effect significantly the operation of the rail system (QU; CORMAN; LODEWIJKS, 2015). With that said this work, as well as most of the works in the literature, assumes that railroads always operate with timetables (KRAAY; HARKER, 1995).

The timetable can be either cyclic or non-cyclic. This means that it can either repeat every given period, or change every time according to demand (CACCHIANI et al., 2014). Non cyclic timetables are usually used for long distance passenger trains and freight trains (NARAYANASWAMI; RANGARAJ, 2011), but the most common timetable is the cyclic which is used for regional transport. However there is no mention of a pattern in the time period in which these schedules repeat, meaning this could be specific to each rail system.

2.1.2.2 Planning levels

In the timetabling process there are, at least 3 (sometimes 4) levels (FANG; YANG; YAO, 2015), in which different tasks are performed. These levels are: strategic, tactical, operational and real time. These are shown in Figure 3.

Figure 3: Planing levels

Source: Narayanaswami & Rangaraj (2011).

Figure 3 summarizes well these levels and give the main factors that are different in them. These are the planning and execution time horizon, stakeholders, criticality, decision impacts and performance threats changes. As the level comes closer to execution the tasks are each time more concrete and closer to the operation, while farther from it the tasks are more strategic and abstract.

The farther from operation the bigger are both planning and execution horizon, which means that actions taken will have a longer result and that the tasks have more time to be done, respectively. Moreover, whenever the planning horizon changes, the tasks and who performs them also change, for example the dispatcher who does the rescheduling in the real time level is usually not the same person who plans the infrastructure design, even though this second task has a huge effect on the rescheduling. Furthermore, the farther the plan is from execution the bigger impact and criticality it has, because it affects a longer period of time and consequently has a bigger yield of performance. For example, although both building a new track and re-timing a train are essential tasks for the best operation on rail systems, the first task has a bigger impact than the second in the overall rail system, but it also takes much longer to be done. And as the planning gets closer to execution more performance threats are in the system and must be managed making the complexity of the tasks bigger.

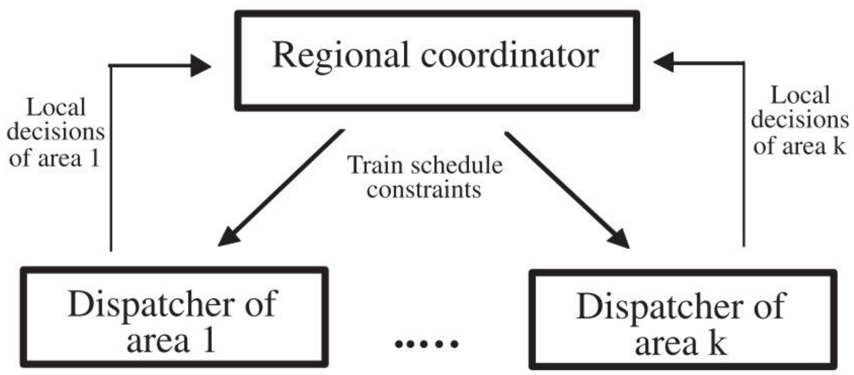
The strategic level has the characteristics of long time horizon typically involving resources changing, either buying or selling. In the tactical level the task is to allocate the available resources assuming that the infrastructure is fixed, some of the problems solved in this level are planning lines, routing trains, scheduling rolling stock and crew to build the original timetable (LUSBY et al., 2011; NARAYANASWAMI; RANGARAJ, 2011). And at the operational level, tasks that are performed daily, close to the time of operation, are performed by local traffic managers (FANG; YANG; YAO, 2015). Finally, real time tasks must be executed extremely quickly and focus on handling perturbations to the plan with contingency tools, mainly summarized by the rescheduling task (FANG; YANG; YAO, 2015; TÖRNQUIST, 2006). Some papers also stipulate the time available to these tasks, that ranges from seconds to a few minutes (FISCHETTI; MONACI, 2017; CACCHIANI et al., 2014).

2.1.2.3 Dispatchers coordination

Dispatchers are experienced workers who have the job of completing the operational and real time planning level tasks so operations run as smoothly as possible (D'ARIANO; PACCIARELLI; PRANZO, 2007). However, in many real-world scenarios the rail system is too large for a single dispatcher. This is solved by a decentralized control, which means that the rail network is divided into regions or sub-networks that are connected at border stations called interchange points. Each of these regions is controlled by a dispatcher that usually work in a operation control center (CORMAN et al., 2012b).

This division requires a coordination between the dispatchers so the optimization of the individual regions combines into the optimization of the whole network. This is ensured by the imposition of border constraints by a coordinator, who has the task of ensuring global feasibility of schedules and to pursue the overall quality of the global solution (CORMAN et al., 2012b).

Figure 4: Dispatcher coordination illustration.



Source: Corman et al. (2012b).

Figure 4 illustrates how this dynamic between dispatchers and coordinator works, with an example of network divided into k regions. For a further understanding of how this is done refer to the article written by Corman et al. (2012b).

2.2 Categories of unforeseen events in rail track

The unforeseen events, previously mentioned, have not been defined in this work until now. In this section they will be defined based on the literature on the train rescheduling problem.

2.2.1 Perturbations, disturbances and disruptions

In the train rescheduling literature the unforeseen events that makes the original schedule of the trains infeasible are called by three names: perturbation, disturbance or disruptions. Perturbation is the name used to refer to any of these events that change the schedule. Some examples of perturbations can be delays in tasks performed on trains, like boarding, crew changes, or operational delays such as signal failures and engineering works in progress, or catastrophes such as trains derailing, vandalism, severe weather, cable theft and so on (BLUM; ESKANDARIAN, 2002; Office of Rail and Road, 2023).

Disturbances are usually related to smaller perturbations in the railway system, such as small delays, while disruptions are larger incidents that require more measures to return the system to a schedule (CACCHIANI et al., 2014). With that said, some authors differentiate them based on the deviation from the timetable, meaning that both terms refer to delay, however, disturbances are smaller delays than disruptions, although there is no sharp distinction between the terms (SHARMA et al., 2023; QU; CORMAN; LODEWIJKS, 2015).

Moreover, some disruptions may require rescheduling not only the timetable but also other resources such as crew and train wagons (CACCHIANI et al., 2014). These disruptions also have a bigger economic impact introducing additional costs and decreasing service level (VISENTINI et al., 2014).

A simplification used by many of the papers in the subject is assuming that the duration of the perturbation is known and fixed (DOLLEVOET et al., 2017). This is done so the problem can be solved in a deterministic approach, and is only necessary when handling disruptions, since they require rescheduling before ending, while smaller delays can be rescheduled after the fact.

Both perturbations types, disturbance and disruption, create a delay that if not dealt with efficiently can propagate and delay other trains in the system. This results in a amplifying the original delay which may result in some undesired consequences for clients, such as the example depicted in Figure 5.

Figure 5: Crowd waiting for delayed train in germany

Source: DPA picture alliance (2022).

In Germany delays have become more common with the increase of maintainance needs of the infrastructure due to its age. Therefore, it has become a relevant issue improving the responses to delays in operations (HÖLZL, 2023).

2.2.2 Solutions

Until now the discussion has been focused on the context of the problem studied, but the literature in this problem presents two possible solutions: one preventive and another reactive. Both of these solutions will be briefly presented in this subsection.

2.2.2.1 Robust schedule

One possible solution to minimize consequences from unexpected events is through robust scheduling. The robustness of a schedule is defined as the ability of that schedule in absorbing perturbations. This is achieved through inserting buffers or slacks between operations in the schedule. These buffers avoid conflicts when disturbances occur because they keep trains far from each other even with the resulting delay (VISENTINI et al., 2014).

With that said, robust scheduling is the preventive solution, which means that the solution is implemented before the problem occurs. However, robustness is inversely correlated to track utilization, because one of the few measures to increase capacity is to reduce buffers in operations (NARAYANASWAMI; RANGARAJ, 2011; FANG; YANG; YAO, 2015). Since high utilization is usually the operation standard to most railroads, these are very suitable to conflicts and not very robust (BLUM; ESKANDARIAN, 2002).

The slacks and buffers, that result in robustness, are also correlated to the stability of the schedule. Schedule stability refers to the ability to returning to normal operation after a perturbation. In other words, the more robust is the schedule the more feasible will be the reschedule (NARAYANASWAMI; RANGARAJ, 2011). That means the robust scheduling is a problem strongly correlated to the rescheduling problem, even though they are treated as isolated and independent (VISENTINI et al., 2014).

2.2.2.2 Rescheduling

The other solution to unexpected events is rescheduling, which is making a new timetable with adjustments to account for the delays that resulted from these events and make the train schedule feasible again (CACCHIANI et al., 2014). This means this is the reactive solution approach to the problem because it takes place after the problem has already happened.

The rescheduling problem was chosen to be studied because even with a robust schedule, there may always be a need for rescheduling, especially in a disruption. Moreover, there is still little to no optimization software in the field to assist dispatchers with rescheduling. In other words, in most cases, they have to rely on their experience, intuition, and some general contingency plans to deal with all the perturbations that occur during operations (GHAEMI; CATS; GOVERDE, 2017; VISENTINI et al., 2014; CACCHIANI et al., 2014; QU; CORMAN; LODEWIJKS, 2015). In contrast, the scheduling process already uses many computational optimization resources (TÖRNQUIST, 2006).

This opportunity space to help operators on improving the rescheduling quality have motivated the literature to focus on this problem. Some of the names used in the literature to refer to this problem is summarized in Table 2.

Table 2: Some of the possible names for the train rescheduling problem.

Name	Abreviation	Papers
Conflict resolution problem	CRP	D'Ariano, Pacciarelli & Pranzo (2007), D'Ariano, Pacciarelli & Pranzo (2008)
Train dispatching problem	TD	Lamorgese & Mannino (2013)
Train rescheduling problem	TRP	Visentini et al. (2014)
Train timetable rescheduling	TTR	Cacchiani et al. (2014), Bai et al. (2023)
Train timetable rescheduling problem	TTRP	Reynolds (2021)

Source: Own representation.

However, there are few examples listed in the literature of real-world implementations of computational support in train rescheduling. These are: a proprietary decision support system from Alston called ICONIS (FISCHETTI; MONACI, 2017) and two implementations of optimization based dispatching systems; one in Italy and the other in Norway (LAMORGESE; MANNINO, 2015).

2.3 Rescheduling theory

Now that the operational tasks and the unexpected events have been explained, next some important topics to understand the train rescheduling problem will be presented. These points are the measures taken in the rescheduling process, how is the original schedule considered during this process and the optimization models used in this problem.

2.3.1 Measures

In this work what is referred to as measures are the changes done to the timetable in order to make it feasible again and minimize the consequences of the perturbations. Most papers break the train rescheduling problem into 2 decisions which are choosing the route of trains and than deciding the new time in which trains depart from each point (CAPPART; SCHAUS, 2017; GODWIN; GOPALAN; NARENDRAN, 2007). In this last part two rescheduling measures are applied which are the retiming and the reordering of trains (QU; CORMAN; LODEWIJKS, 2015; CACCHIANI et al., 2014). With that said, some other papers list further possible measures, some of which are summarized in Table 3.

Table 3: Train rescheduling measures.

Possible rescheduling measures
Retiming
Reordering
Rerouting
Adding Stop
Cancellation
Emergency Train
Skip stop
Short-turning
Rolling Stock Rescheduling
Speed Control
Changing dwell times of trains at stations
Breaking connections

Source: Sharma et al. (2023).

Most of the measures presented in Table 3 are self explanatory, however it is important to explain the most important ones. Retiming means adjusting the time trains arrive and depart from each block section in their route. Reordering is choosing the best order of trains in each track section so the slower trains pass the track section after faster ones if they would delay the faster trains. Rerouting is changing the route of trains, making them follow a different order of track sections when compared to the original schedule (SHARMA et al., 2023). The other measures may also be used in the practice, but will not be modeled in the present work.

2.3.2 Original schedule consideration

With the measures to rescheduling presented, it is interesting to understand the difference between scheduling and rescheduling and how does the rescheduling process considers the original schedule in its decision process. While scheduling is creating a timetable from start, rescheduling is modifying an existing, infeasible, schedule into a new one (TÖRNQUIST, 2006).

With that said both problems are usually formulated very similarly in mathematical models, with some key differences. In the scheduling problem, the objective function is maximizing customer satisfaction or minimizing costs. On the other hand, in rescheduling models, the objective is minimizing the effects of perturbations, which means recovering the initial schedule as soon as possible (NARAYANASWAMI; RANGARAJ, 2011).

The key differences come from the fact that the scheduling is a tactical planning level problem while the rescheduling is done in the operational level. For this reason while there is plenty of time to solve the scheduling problem there is only a few seconds to a few minutes to solve the rescheduling problem (FISCHETTI; MONACI, 2017; CACCHIANI et al., 2014). Furthermore, the rescheduling horizon is either until the end of the day, for non cyclic schedules, or until the end of a cycle in a cyclic schedule, which are more common to suburban traffic (CACCHIANI et al., 2014; NARAYANASWAMI; RANGARAJ, 2011), while the scheduling horizon is at least until the end of the day, if not longer.

Moreover the train rescheduling problem must take under consideration the original schedule as part of its objective function, as well as the current position of each train in the rail system as constraints (QU; CORMAN; LODEWIJKS, 2015; VISENTINI et al., 2014). This will reduce the feasible solution space due to the conflicts that are present on the current operation making the original timetable infeasible and requiring conflict resolution through a reschedule (D'ARIANO et al., 2008; NARAYANASWAMI; RANGARAJ, 2011). On the other hand, the scheduling problem can only take under consideration the customer demand, since it is performed before the position and the original schedule exists.

The idea of recovering the original schedule as being the rescheduling objective function comes from the assumption that the original schedule is already the optimal timetable for operations, since it was developed during the tactical planning level with optimization tools (KRAAY; HARKER, 1995).

2.3.3 Models

After presenting the previous information on the train rescheduling problem, this subsection is meant to present two of the most common models used to solve the problem, giving a brief comparison and describing how the one chosen to be implemented works. This subsection is divided into three parts: a categorization of the models by how it considers the rail infrastructure, a division of the total delay into two parts: an unavoidable delay and a propagation delay, A comparison between the two main models and a description of one of them.

2.3.3.1 Infrastructure consideration

In the literature there are two categories of models that are different on how they represent tracks. These are “microscopic” or “macroscopic”. “Microscopic” models consider the infrastructure in more detail and each individual block section is considered separately. Most of this models use *Alternative Graph* mathematical formulations. On the other hand, macroscopic models aggregate many blocks together, creating a higher level view of the railway network. Most of these models use the *Event-Activity* mathematical formulation to model the railroad network (CACCHIANI et al., 2014; QU; CORMAN; LODEWIJKS, 2015; SHARMA et al., 2023; DOLLEVOET et al., 2017).

In the context of the train rescheduling problem, choosing how to consider the infrastructure represent a trade-off between solution time and quality. Because solution time is scarce on operational level it is an important point to be considered, requiring a balance with the solution detail (FANG; YANG; YAO, 2015). The network representation, however, also describes how vehicles can be rescheduled (VISENTINI et al., 2014). However, the aggregations performed in the macroscopic models both speed up the solution and restrict the ability to reschedule trains. In that sense a vast part of macroscopic models simplifies the representation of stations and do not consider the potential of train paths crossing and the allocation of tracks within the stations, making them less representative of the reality (TÖRNQUIST, 2006).

With all of that in mind, this work studies a microscopic model, since it is more detailed, can have a better quality solution with the information closer to what a dispatcher would need to implement it during operations. Regarding the trade-off with the solution time, solution approaches are considered to reduce the solution time needed, for example reducing the area of rescheduling to only the area that a single dispatcher controls.

2.3.3.2 Primary and secondary delays

In the train rescheduling problem the objective is to minimize the delay resulting from a unforeseen event. In this work the delay is defined as the difference between the time a train arrives in a important point, usually a station, at the original schedule and time in the new schedule. This delay can be divided into primary and secondary.

The primary delay is the direct result of a perturbation, being the original delay. This part cannot be prevented and is part of the initial condition of the system in the problem (CACCHIANI et al., 2014; CORMAN et al., 2012a). With the interaction between trains and the surrounding traffic it can propagate through conflicts to other trains, making them also delayed (TÖRNQUIST, 2006). This propagated part is called secondary delay (CACCHIANI et al., 2014; CORMAN et al., 2012a).

With that said the train rescheduling process can only minimize the secondary delay, because the primary is already part of the initial condition. To do so measures of rescheduling are applied such as retiming, reordering and rerouting.

2.3.3.3 Comparison of main models

The two main microscopic models presented in the literature to solve the rescheduling problem are the *Discretized time model* and the *Alternative graph model*. The *Discretized time model* can also be called Binary integer occupancy model. It is an integer programming problem, where variables are binary, and each one of them indicates the occupancy of a track by a train during a time span (HARROD, 2012). By making time discrete, the constraints are simplified and implicitly modeled (TÖRNQUIST, 2006). On the other hand, it also may make some conflicts undetectable if the time period is not small enough to consider all train interactions (GHAEMI; CATS; GOVERDE, 2017). And when the time discretization step gets finer, in other words as the time periods become smaller, the number of variables and constraints grows exponentially making the problem more complex and consequently slower to solve (MANINO, 2011; TÖRNQUIST, 2006). For a doctor dissertation on this model refer to Reynolds (2021).

The other model is an adaptation of a job shop scheduling problem modeling technique, with blocking and no-wait constraints, to the case of train rescheduling (MASCIS; PACCIARELLI, 2002). The job shop scheduling problem is a well-known problem in the field of operational research where jobs are scheduled to go through some machines in a certain order, which are shared resources (YAMADA; NAKANO, 1997). Therefore the problem literature has a strong theoretical background, which was leveraged to solve the rescheduling problem. In the adaptation the trains are the jobs that must go through the track sections, which represent the machines, in a certain order. The processing time is the running time required by the train to travel through a block section. The no-wait constraint means that the operations must be followed without interruption and the blocking constraint means there is no storage between machines (MASCIS; PACCIARELLI, 2002). The solution technique used to solve this problem is the alternative graph, which is a special case of a disjunctive graph commonly used to solve job shop problems (LUSBY et al., 2011). With this formulation, the model can use continuous variables to the times trains arrive in each block section improving the speed of solving the problem. However this formulation also makes the reordering decision be made comparing two trains at a time with binary variables, hence making the number of trains probably a driver of complexity to the model.

With both models in mind, the weaknesses and strengths of each problem make them better in different situations. The *Discretized time model* is better for smaller time horizon reschedules and the *Alternative graph model* is better to reschedule a smaller number of trains. With that said, the literature tends to prefer the *Alternative graph model* for rescheduling, for the benefits of using continuous time variables. For this reason this is the model chosen to be studied in this work.

2.3.3.4 Solutions approaches

As both models described previously can become complex very quickly for even medium size instances, the literature proposes a few solution approaches to try to minimize the solution time. From these solution approaches, two of them are important: the division of the network into subareas and the rolling horizon approach.

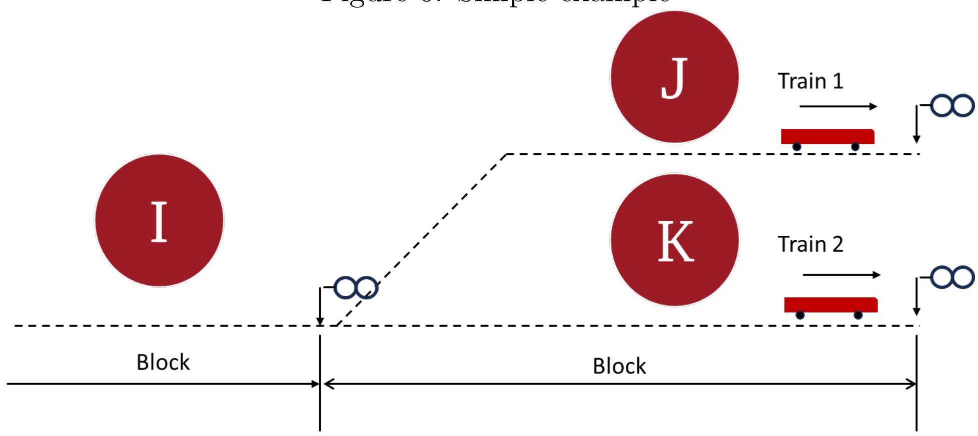
The division of the network implies in solving the problem in the space that only one dispatcher supervises as described in Subsection 2.1.2.3. This way there are fewer trains and the model is simplified. However to have the optimal solution for the entire network, there are some boundary restrictions.

The rolling horizon on the other hand is a common solution approach for models used to solve problems in long time horizons. In this approach, instead of solving the model for the entire time horizon, this is divided into several smaller periods which are then solved in sequence (BISCHI et al., 2019). This is specially useful in the case of this problem, because trains that run in distinct times and share a block section end up creating a alternative arc even though in practice the conflict is very unlikely. Therefore by using the rolling horizon solution approach the model is simplified lowering the processing time without making the solution worse.

2.3.3.5 Alternative graph

In this section the most common alternative graph model in the literature is presented. This model uses both retiming and reordering to solve the problem of rescheduling. To explain the alternative graph model, the concept of the graph has to be explained. More specifically, how it is built and how it is translated to a mathematical model. To do so a simple example of an operation situation is depicted in Figure 6.

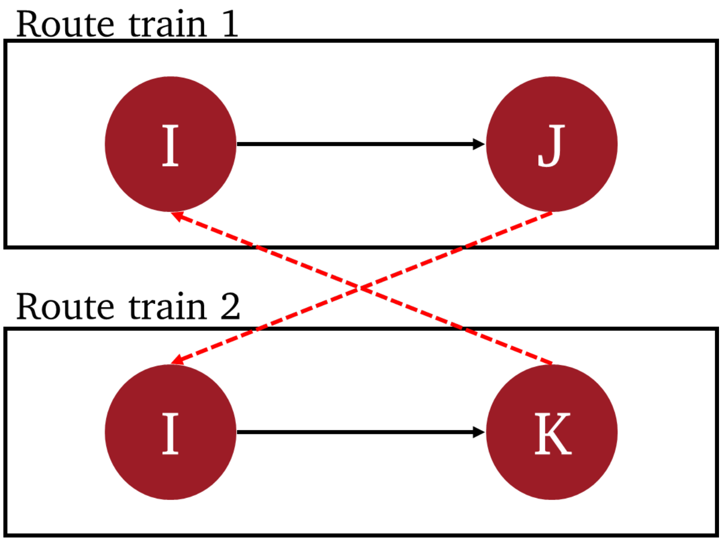
Figure 6: Simple example



Source: Own representation.

In this example a block section, I, is connected to two other block sections, J and K. With that said, two trains share the block section I in their routes, the first train goes from block section I to block section J and the other train goes from I to K. For this small example the part of a alternative graph depicted in Figure 7 is built.

Figure 7: Alternative graph



Source: Own representation based on the paper by Espinosa-Aranda & García-Ródenas (2013).

In a alternative graph each node represents a block section in a train route. In Figure 7 there are two nodes representing the block section I because both trains have this block in their routes. There are also two sets of arcs that connects these nodes: fixed arcs and alternative arcs. The fixed arcs set connect the nodes in the route of a train, these are represented by a black arrow in Figure 7. The alternative arcs set, on the other hand, are created whenever two trains share a block section. These arcs are always presented in pairs connecting the succeeding node of one of the trains routes to the shared block node in the other trains route. These are represented in Figure 7 by red dashed arrows. An example of a fixed arc in Figure 7 is the arrow connecting the nodes I and J in the route of the train 1. An example of a alternative arc is the arc connecting the node J in the route of train 1 with the node I in the route of the train 2.

With the alternative graph built the next step is to translate it into the mathematical model. In this context, each node in the alternative graph will be translated into one or more variable and each arc is translated into a constraint connecting these variables. Because each node represents a block section in a route of a train, they are translated into a continuous variable, which represents the time that train has arrived in that block section. In the example of Figure 7 there would be 4 continuous variables, one for train 1 and node I, one for train 1 and node J, and so on for the other nodes. The arcs are translated into precedence constraints. Fixed arcs are translated into the constraint that the next node will only be arrived in a time bigger or equal to the time the train has arrived in the previous node plus the time it takes to travel through that node. For example, train 1 will only arrive at node J after it has arrived and run through node I.

Alternative arcs, on the other hand, are translated into disjunctive constraints. These are constraints that represent a choice, where only one of them needs to be satisfied in order to satisfy both constraints. In this problem the alternative arcs represent the choice of order of trains, which means it is the application of the reordering measure. These arcs are translated into a constraint of precedence between trains, which means one of the nodes will only be allowed to be reached after the other one plus a setup time. In the example of Figure 7 the alternative arc that connects node J to I is translated into the constraint that train 2 will only be allowed to arrive at node I after train 1 has arrived node J and left node I. In other words, the time train 2 arrives at node I must be bigger than the time train 1 has arrived node J plus the setup time of train 1. And the same idea can be applied to the other alternative arc. Because these are disjunctive arcs, only one of the two constraints must be active, and its restriction satisfied, which indicates the choice of which train should go first. If the arc connecting node J to I is active, then train 1 will go first.

To translate the alternative graph into a model the elements presented in the tables below are necessary. The indexes and sets are presented in Table 4, then the decision variables are presented in Table 5 and the parameters are presented in Table 6.

Table 4: Indexes and sets of the model from the literature.

Elements	**Description**
$i, j \in T$	i and j are the indexes of trains in the network, and T is the set containing all train indexes.
$u, v, x \in N$	u, v, x are block sections in the rail network N.
$s \in S_i$	s is a station in the route of the train i, where S_i is the set of stations that train must visit.
$(u, v), (u, x) \in F_i$	(u, v) and (u, x) are fix arcs, where u, v, x are the block sections of the network N connected by fixed arcs, and the set of all fixed arcs in the route of train i is F_i.
$(i, j, u) \in A$	i and j are the indexes of trains and u is the block section of possible conflict involved in the pair of alternative arc (i, j, u), where A is the set that contains all pairs of alternative arcs.

Source: Own representation, based on model from paper (ESPINOSA-ARANDA; GARCÍA-RÓDENAS, 2013; CACCHIANI et al., 2014).

Every block section in the rail network is contained in N. One point that has not been mentioned yet is that every train goes through a set of stations in its route, these are all present in the set S_i for the train i. These nodes are important as they are the important points of the network where the delay is calculated, which is used in the objective function that the problem minimizes. And the alternative arcs are here summarized in a list of 3 numbers (i, j, u), where the first two numbers are the trains ids that share a block section and the last number is that block id.

Table 5: Decision variables from the model from the literature.

Elements	**Description**
$t_{i,u}$	Continuous decision variable that indicates the time of arrival from train i in block u.

Source: Own representation, based on model from paper (ESPINOSA-ARANDA; GARCÍA-RÓDENAS, 2013; CACCHIANI et al., 2014).

Table 5 presents the continuous variable used to determine the time a train arrives in a block section. This variable is created for each node in the alternative graph, as it was explained previously.

Table 6: Parameters from the model from the literature.

Elements	Description
$SC_{i,s}$	Original schedule time that train i arrived in station s.
P_i	The priority of the train i.
$TOT_{i,u}$	Minimum time on track that train i spends on block section u.
$ST_{i,u}$	Setup time from train i in block u.

Source: Own representation, based on model from paper (ESPINOSA-ARANDA; GARCÍA-RÓDENAS, 2013; CACCHIANI et al., 2014).

Table 6 present the parameters that bring both the original timetable information, as well as necessary information of the resources to build the constraints and the objective function of the model. The minimum time on track is given by the division of the length of the block section by the maximum speed that train can travel, which is the translation of the speed restriction presented in the resource Section 2.1.1.2. The setup time of train i in block u is given by the division of that trains length by the maximum speed that train can travel in that node. With all these elements presented in these tables the following non linear model is built to perform the rescheduling in a rail system (ESPINOSA-ARANDA; GARCÍA-RÓDENAS, 2013; CACCHIANI et al., 2014).

$$Minimize \quad Z = \sum_{i \in T} \sum_{s \in S_i} P_i \max(0, t_{i,s} - SC_{i,s}) \tag{2.1}$$

Subject to:

$$t_{i,v} - t_{i,u} \geq TOT_{i,u}, \qquad \forall i \in T; (u,v) \in F_i \tag{2.2}$$

$$(t_{i,u} - t_{j,v} \geq ST_{j,u}) \vee (t_{j,u} - t_{i,x} \geq ST_{i,u}), \quad \forall (i,j,u) \in A; (u,v) \in F_j; (u,x) \in F_i \tag{2.3}$$

The objective function (2.1) minimizes the cost function of delays in stations, which is given by multiplying the train priority by the time difference between the time a train would arrive in a station in the original timetable and the new time calculated by the reschedule. Constraints (2.2) is the translation of fixed arcs, which enforces that a train may only arrive in the next block of its route after the arrival time in the previous block plus the minimum time on track of that previous block. In the constraint (2.3) there are two disjunctive inequations, they are merged by a or logic, which means that if one inequations is satisfied, the constraint is satisfied. These inequations represent the alternative arcs, which enforce that after a train leaves a shared block section, the next train can only enter it after a setup time. This constraint also translates the blocking restriction presented in Section 2.1.1.2.

With that said this model is not linear, because it has a or logic connecting two inequations in a disjunctive restriction. To make it linear, the system is expanded adding a binary decision variable for every alternative arc pair and breaking inequation (2.3) into two constraints (ESPINOSA-ARANDA; GARCÍA-RÓDENAS, 2013; CACCHIANI et al., 2014). The new decision variables are $y_{i,j,u}$, which represent the decision of ordering the trains. And the new constraint inequations are:

$$t_{i,u} - t_{j,v} \geq ST_{j,u} - My_{i,j,u} \qquad \forall (i,j,u) \in A; i,j \in T; u,v \in N; (u,v) \in F_j \tag{2.4}$$

$$t_{j,u} - t_{i,v} \geq ST_{i,u} - M(1 - y_{i,j,u}) \qquad \forall (i,j,u) \in A; i,j \in T; u,v \in N; (u,v) \in F_i \tag{2.5}$$

These constraints are built using the method of the "big M", which uses a big number, the parameter M, to activate and deactivate restrictions based on a binary decision variable value. This is a very popular method in mathematical modeling used to transform disjunctive restrictions into linear restrictions. In the example model if the variable $y_{i,j,u}$ is equal to 1 constraint (2.4) is deactivate and constraint (2.5) is activated enforcing that train i comes before train j (Wayne Winston, 2003).

This model applies both retiming and reordering. Retiming is done with the continuous variables $t_{i,u}$ and rerouting is done with the binary variables $y_{i,j,u}$. To also consider rerouting the model must be expanded. A possible expansion is presented in the paper written by D'Ariano et al. (2014). An important point to be taken under consideration is that the inclusion of new possible routes makes the model more complex. A previous work with this model has shown that including too many routes makes the model too complex to be solved in the real time context even with small instances (URBAN, 2023).

3 PROBLEM DESCRIPTION

After explaining the theoretical background, the minimum knowledge necessary to understand the problem is available to the reader. Therefore, describing the problem is the next step in solving it. In this chapter the problem studied is described, explaining some specific details and showing the perspective of the Companhia Paulista de Trens Metropolitanos (CPTM) as a case studie of the problem in a real-world context. The chapter is divided in the following sections: general setting and real-world example.

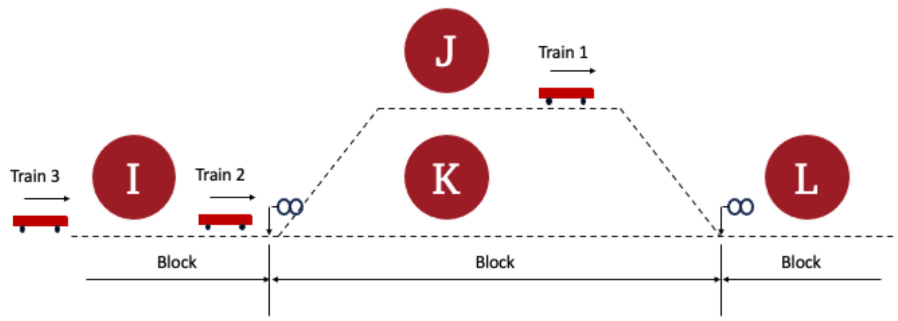
3.1 General setting

The idea in this problem is to assist the work of dispatchers on how to proceed with train operations after a perturbation has occurred in the rail system. This is done by creating a new feasible timetable with minimum changes done to the original schedule. This is an interesting problem to study because currently there is almost no optimizing software or algorithm assistance in this process in the real-world, and it is mostly done based on the experience of the staff or with some general use contingency plan.

The inputs to the problem are the original timetable, as well as the present status of the rail network, and the output is a new feasible timetable for the network. In this work the focus is on cases of disturbances, because this is the main research area in this topic and although a disturbance represents a smaller event it occurs more frequently than disruptions on rail systems. The studied disturbances events are translated into cases where the train takes longer than expected to travel through a block section. This is modeled by significantly increasing the minimum time on track of a train in a track section, which is represented by the parameter $TOT_{i,u}$ in Table 6, where i is the index of a train and u is the index of a block section, both presented in Table 4. With this new time on track, that represents the disturbance event, the rescheduling process is run and the minimization is applied to the sum of the delays of all trains in the stations they visit. This is chosen as objective function because it is the result of the unforeseen event that the client of the rail system can observe.

To illustrate this problem a new infrastructure is introduced in Figure 8 based on the simple example from Figure 6.

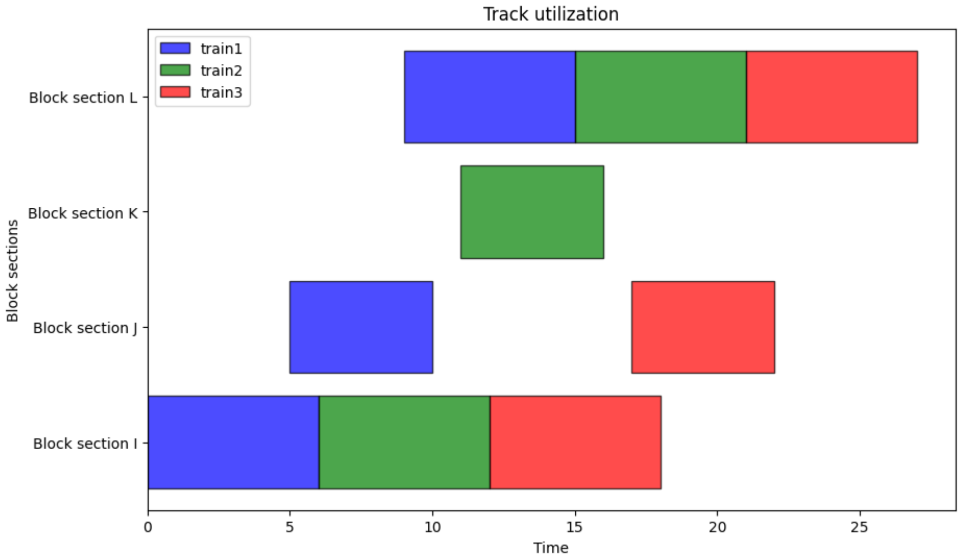
Figure 8: Extension of the example



Source: Own representation.

In Figure 8 there are only four block sections in the network, with blocks I and L at the extremes of the network and blocks J and K connecting them. The scenario consists of this network and three trains that must travel from block I to block L, which represent station block sections visited by the trains. Figure 9 shows the original timetable for this scenario in a gant chart. In it each block section is represented in the vertical axis and the time is represented in the horizontal axis, the bars are colored according to the train they represent, and these bars mean that the train is in the corresponding block section of the vertical axis during the time of the horizontal axis.

Figure 9: Original timetable

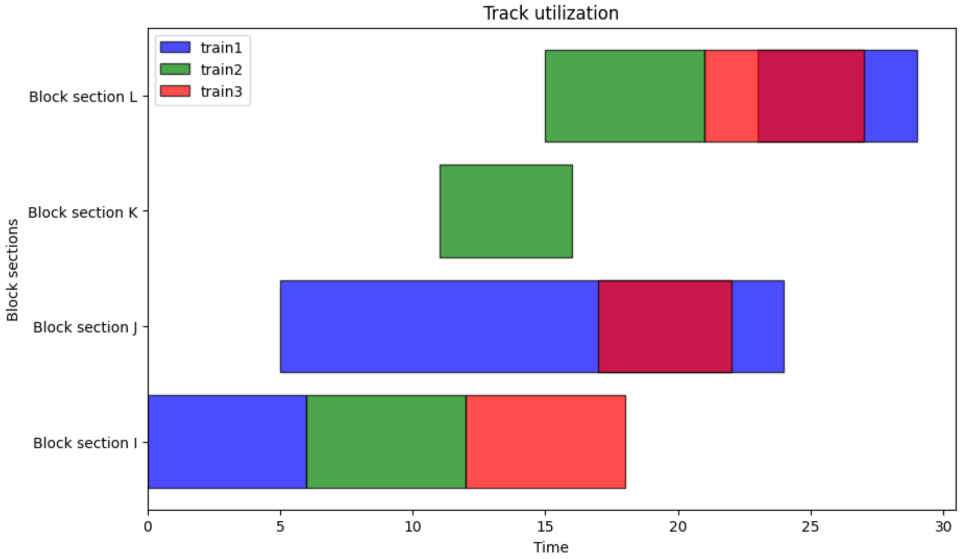


Source: Own representation.

In the original timetable, the first and third trains follow the route of block sections I, J, L, while the second train takes the route I, K, L. The setup time in this scenario is one unit of time, which means that the trains leave the previous block section one unit of time after they have arrived at the next one. For example, train one starts traveling block section I at time zero and arrives at block section J at time five, but it does not leave block I until time six, when train two enters this block.

Then a disturbance occurs during the original timetable while train one was in block section J. This disturbance caused the first train to spend eighteen units of time in block section J instead of the originally scheduled five units. The new scenario with the disturbance is shown in the Gantt chart in Figure 10.

Figure 10: Perturbed timetable



Source: Own representation.

With this perturbation, the original timetable is now infeasible, because trains one and three try to occupy the same block section in both blocks J and L at the same time. These situations are the so called conflicts and are shown in Figure 10 as the overlapping of the blue and red bars on the respective blocks. This perturbed schedule now requires a reschedule so that there is no more conflicts and the new timetable can be followed. To solve this problem, the train rescheduling measures introduced in Table 3 can be applied.

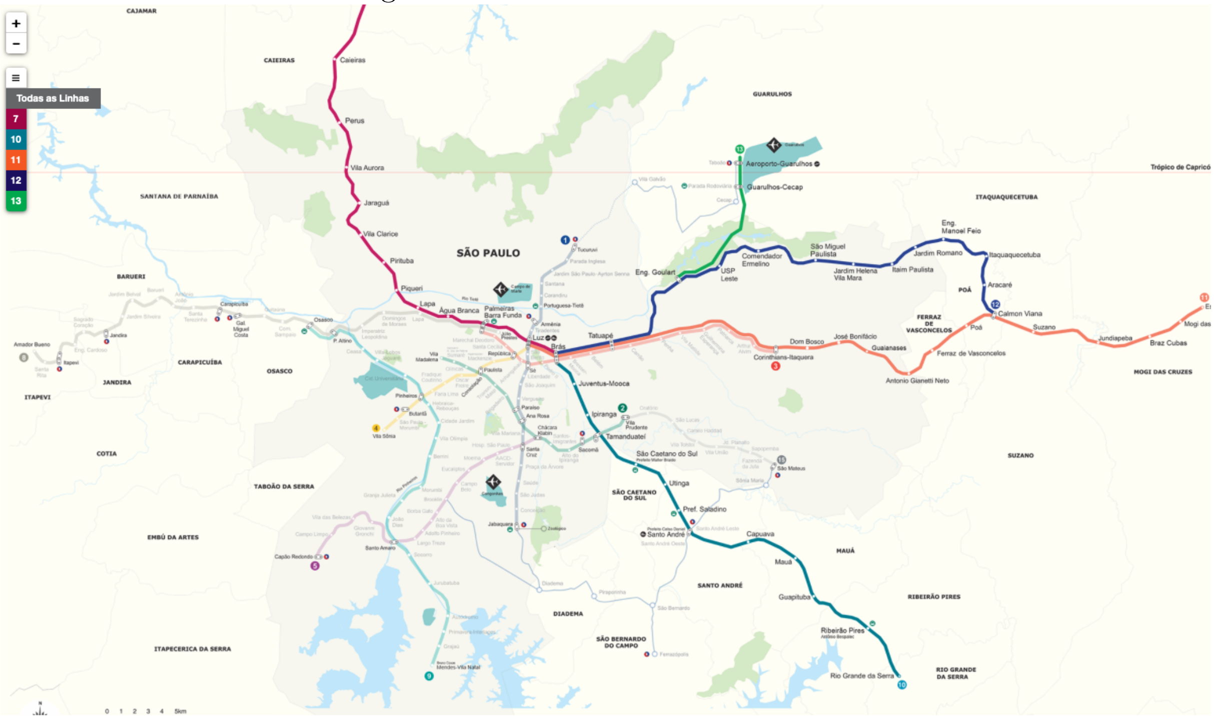
3.2 Real-world example

With the general setting of the problem presented, in this section, the context of the studied real-world case is presented. This is divided in: a brief description of the rail system of the Metropolitan Company of Trains of São Paulo (CPTM) then a summary of the contact with the company is presented.

3.2.1 Rail system

The CPTM operates currently five lines presented in Figure 11, from which lengths summed is equal to a hundred and ninety six kilometers long. All of which start in the city of São Paulo and usually end in another city, reaching in total eighteen cities. Moreover, there are fifty seven stations in these lines (CPTM, 2024a).

Figure 11: The CPTM infrastructure



Source: CPTM (2024c).

In these lines an average of thirty three million passengers enter the rail system every month (CPTM, 2024b). This means approximately one million and nine hundred thousand passengers enter the system on average during a weekday. From which five hundred and thirty three thousand enter the system in line eleven, representing the line with the biggest flow of passengers (CPTM, 2024a).

3.2.2 Communication with CPTM

After some email exchange with the company and a proof of the connection of this work to the university, a meeting was scheduled to better understand the context of the company as a real-world example and to answer some questions related to the subject. Seventeen questions were prepared for this meeting divided into 2 topics: Operations and Rescheduling. All these questions were answered by one of the operators of the company, whose duties include the work of the dispatcher described in this thesis.

3.2.2.1 Operations

The first questions were made to understand if the management of the rail system is done according to what has been studied in the literature. In this sense, the rail infrastructure is divided into block sections, which have a signaling device at each end that looks like a traffic light. These devices automatically change their color according to the occupancy of the following block sections, indicating the maximum speed at which trains can travel. This is consistent with the description in the Theoretical Background Section 2.1.1.1. Some other confirmations of the theoretical background were: the company has a fixed dwell time in stations, which can be changed according to demand, the tracks are bi-directional, being possible to change the direction as well as the speed of the flow of trains, and the operation is done according to a timetable made in advance by a tactical department. In addition, the dispatchers are in direct contact with the conductors by radio, which makes it possible to make changes to the timetable in real time.

Now, to understand the resources of the company, the following topics were discussed: each block section can have a different size and the controller has the real time location of the trains based on the block section it is. For this reason, they try to make all block sections as small as possible to have finer control over train movement and optimize track utilization. With regard to trains, the main attributes considered for scheduling are power and carrying capacity. In this respect, the fleet of the company is very homogeneous, having mostly the same speed limit, the same number of cars and therefore approximately the same capacity. The only speed restrictions that may occur are in cases of red or yellow signals, curved tracks or some types of programmed maintenance, but these are rare.

Other issues discussed were: the goal of the timetable, which is to meet the demand of the population with only the available resources, so profit is not the explicit objective function of the scheduling problem. The division of the system into zones controlled by an operator is based on the flow of trains and not on the size of the tracks, which means that operators can control smaller areas if more trains pass through them. The scheduling process for schedule changes, usually due to maintenance, takes up to 15 days to be used in operations.

3.2.2.2 Rescheduling

In the rescheduling process, the most common perturbation to service are freight trains delaying passenger trains and passengers getting sick and needing assistance, which can both be interpreted as disturbances. On the other hand, the most common disruptions are flooding, track breakage, copper wire theft, and maintenance outages. Having said that, the rescheduling measures used by the company are usually; in case of major disruptions that stop operations, a contingency plan that uses buses to transport passengers between stations. In the case of smaller disturbances, the most common schedule recovery measure is to add a train to the system, referred to as an emergency train in the Table 2.3.1.

The time available for rescheduling trains in the case of minor disturbances is almost non-existent, in other words, changes to operations must be made immediately. On the other hand, for major disruptions, there is usually 10 to 15 minutes to create a new schedule. Therefore, the processing time limit for a solution that can be implemented in the real-world is only seconds.

In addition, software is being developed in the company that may contain the solution to the problem studied, but there is no official estimated date for the software to be completed and implemented. With that said, it is hoped that this work can help provide some insight into the problem and how to solve it.

3.2.2.3 Conclusions of the meeting

From the description of the company and the answers to the questions the conclusion was that the company operates according to what was described in the literature. Furthermore, there are common cases of disturbances that occur in the rail system indicating that studying the problem of rescheduling in this kind of perturbations would help the company in their operations. Considering the size of the company, its context is a good case study because it represents a complex real-world case of the problem and operations would benefit from this study.

4 MATHEMATICAL MODELING

This chapter presents three mathematical models for solving the problem studied. Each of them adds a new train reschedule measure to the previous model. The first model uses only retiming to solve the problem. The second model introduces reordering into the first model. The third model adds the rerouting measure to the second model. However, each measure introduces more complexity into the solution approach, which should yield better solution values in exchange for more processing time. The second model was introduced in Chapter 2, and the first model was built by simplifying it, while the third model is an extension of it.

Each model is presented in a separate section, which is divided into two subsections: firstly, notation, and secondly, objective function and constraints. For ease of reading, the entire model is presented in all sections, even when there are repetitive elements, so that the reader does not have to refer to the previous model to understand the current one.

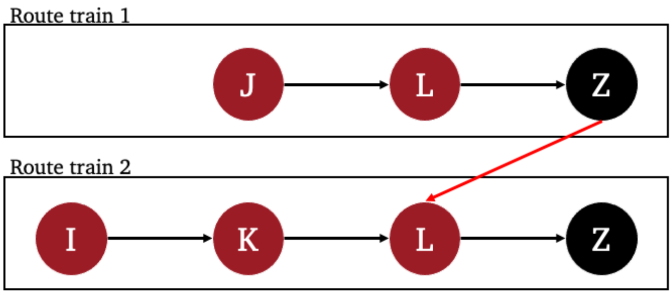
To illustrate that each model creates a different alternative graph for the solution, the extended example shown in Figure 8 is used to create the alternative graph of each model. This example can show the differences between the solution of each model. In it, there are four block sections I, J, K, L. Both trains go from block segment I to block segment L. Originally, train one would go first and get to block section L via block section J, while train two would go travel through block section K instead of J. However, train one is delayed in block section J and now the system needs rescheduling.

Similar to the alternative graph in Figure 7, the graphs in this chapter represent fixed arcs, arcs that indicate a train route, as black arrows. Arcs that indicate the order of trains are represented by red arrows. The dotted arrows mean that there is a decision, either of route or order selection, which indicates disjunctive constraints.

4.1 First model

The first model presented is a simplification of the model available in the literature. In this first model, only the retiming measure is applied, so the model is simplified to contain only the necessary elements for this measure. This simplifies the alternative graph and constrain the alternatives. To illustrate the graph used to represent this model, consider the extended example in Figure 8. In this model, the following graph is constructed:

Figure 12: Model 1 graph



Source: Own representation.

In the graph in Figure 12, there are only fixed arcs because the reordering measure is not applied. This preserves the order of the trains in the blocks as in the original timetable. In this example, train one travels through the block section L before train two. This simplifies the model because it does not need the integer variables.

One addition in this graph is that because the whole route is represented, a dummy node Z is added at the end of all routes, representing the exit of the train from the network. It is used to build the order constraint between trains in case the last block section of the routes is shared between them. Also, the route of train one starts in block section J because the train was already there when the rescheduling started.

In Figure 12, there is an arc between block Z in the path of train one and block L in the path of train two because these trains share this block L. Furthermore, this model does not apply reordering due to simplification, making this arc fixed and forcing train one to pass through block L before train two, even if train one is delayed, because this was the original order.

4.1.1 Notation

Next, the tables containing the elements of the model are presented in the following order. First, the indexes and sets used in the model are introduced, then the main parameters are presented, and finally the decision variables are summarized in the third table.

Table 7: Indexes and sets of the first model.

Elements	Description
$i, j \in T$	i and j are the index of trains in the network, and T is the set of trains.
$u, v \in N$	u and v are blocks in the set of blocks sections N of the rail network.
$s \in S_i$	s is a station in the route of train $i \in T$, where $S_i \in N$ is the set of stations in that train route.
$(u, v) \in F_i$	(u, v) is a fixed arc, where u and v are block sections of the set N connected by the arc, and the set of fixed arcs in the route of train i is F_i.
$(i, j, u) \in A$	i and j are the indexes of trains and u is the block section of possible conflict involved in the pair of alternative arcs (i, j, u), where set A is the set that contains all alternative arcs.

Source: Own representation based on model from paper by Espinosa-Aranda & García-Ródenas (2013).

All blocks shared by two trains are considered a possible conflict block section, and the alternative arcs are built for the trains and shared block sections. In this first model the alternative arcs is simplified to a fixed arc where the original order is imposed. In Figure 12 the trains only share the block section L making the only fixed arc that imposes the order the one between block section Z in the path of train one and the block section L of train two.

Table 8: Parameters of the first model.

Elements	Description
$SC_{i,s}$	Original scheduled time that train i should arrive at station $s \in S_i$.
$TOT_{i,u}$	Minimum time on track that train i spends on block section u.
P_i	Priority assigned to train i.
$ST_{i,u}$	Setup time in block section u for train i.

Source: Own representation based on model from paper by Espinosa-Aranda & García-Ródenas (2013).

Table 9: Decision variables of the first model.

Elements	Description
$t_{i,u}$	Continuous decision variable that indicates the arrival time of train i in block section u.
$d_{i,s}$	Continuous decision variable that indicates the delay of train i at station s.

Source: Own representation based on model from paper by Espinosa-Aranda & García-Ródenas (2013).

4.1.2 Objective function and constraints

With the sets, parameters, and variables presented, the objective function and the constraints can now be presented.

Objective function:

$$Minimize\ Z = \sum_{i \in T} \sum_{s \in S_i} P_i \times d_{i,s} \tag{4.1}$$

Constraints:

$$d_{i,s} \geq t_{i,s} - SC_{i,s} \qquad \forall i \in T; s \in S_i \tag{4.2}$$

$$t_{i,v} - t_{i,u} \geq TOT_{i,u} \qquad \forall i \in T; u, v \in N; (u,v) \in F_i \tag{4.3}$$

$$t_{i,u} - t_{j,v} \geq ST_{j,u} \qquad \forall (i,j,u) \in A; i,j \in T; u,v \in N; (u,v) \in F_j \tag{4.4}$$

$$d_{i,s} \geq 0 \qquad \forall i \in T, s \in S_i \tag{4.5}$$

The objective function (4.1) minimizes a cost function of delay, where the delay of each train is multiplied by the priority it has. This is subject to:

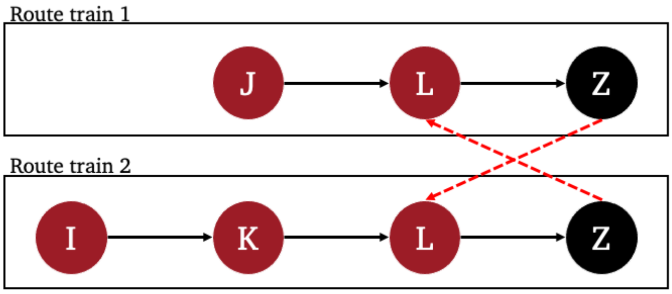
- Constraint (4.2) defines the value of the delay at a particular station as the difference between the time the train arrives at that station and the time it should have arrived according to the original schedule. If this difference is lower than zero, in other words the train arrives earlier in that station, the delay becomes zero.
- Constraint (4.3) enforces the maximum speed for the train, where the arrival time in the next block section of a train route minus the arrival time in the previous block section must not be less than the smallest time it would take that train to travel through that previous block. This minimum time is calculated by the length of that block divided by the maximum speed that train can travel. This is part of the fixed arc constraint, which also dictates the route that each train follows by enforcing the order of the nodes that the train will pass through.
- Constraint (4.4) This is a constraint, which impose the order of trains in shared block sections. Unlike the other models, this is not a disjunctive constraint, since there is no binary variable needed and alternative constraint.
- Constraint (4.5) enforce the domain for the decision variable of delay. The variable d must be non-negative.

In order to build the constraint (4.4) a preprocessing of the data was necessary when building the set A. That is, the original schedule was used to compare the trains with the original arrival time to the shared block sections, and whoever was first in the first shared block section would be the first to pass through all the other block sections these trains shared.

4.2 Second model

The second model is found in the literature presented in Chapter 2. It applies both retiming and reordering to make the schedule feasible again and to minimize the effects of perturbations. The alternative graph for this model has already been illustrated in Figure 7. However, to solve the extended example from Figure 8, the graph displayed in Figure 13 is built to represent the second model.

Figure 13: Model 2 graph



Source: Own representation.

In this graph there is again the use of the alternative arcs, as described in Subsection 2.3.3.5 in contrast to the previous model that had only fixed arcs.

4.2.1 Notation

With the alternative graph built the following Tables 10, 11 and 12 present the elements needed to build the second model.

Table 10: Indexes and sets of the second model.

Elements	Description
$i, j \in T$	i and j are indexes of trains in the network, which T is the set of all trains.
$u, v \in N$	u and v are block sections in the set of blocks N of the network.
$s \in S_i$	s is a station in the route of train $i \in T$, where $S_i \in N$ is the set of stations in that train route.
$(u, v) \in F_i$	(u, v) is a fixed arc, where u and v are the block sections of the network N connected by the arc, and the set of fixed arcs of train $i \in T$ is F_i.
$(i, j, u) \in A$	i and j are the indexes of trains and u is the block of possible conflict involved in the pair of alternative arc (i, j, u), where set A is the set that contains all pairs of alternative arcs.

Source: Adapted from Espinosa-Aranda & García-Ródenas (2013).

Table 11: Parameters of the second model.

Elements	Description
$SC_{i,s}$	Original schedule time that train i arrived at stations s.
M	A sufficiently large positive number.
$TOT_{i,u}$	Minimum time on track that train i spends on block section u.
P_i	Priority assigned to train i.
$ST_{i,u}$	Setup time in block section u for train i.

Source: Adapted from Espinosa-Aranda & García-Ródenas (2013).

Table 12: Decision variables of the second model.

Elements	Description
$t_{i,u}$	Continuous decision variable that indicates the time of arrival from train i at block section u.
$d_{i,s}$	Continuous decision variable that indicates the delay of train i at station s.
$y_{i,j,u}$	Binary decision variable of alternative arc pair that indicates the order of trains i and j at block section u.

Source: Adapted from Espinosa-Aranda & García-Ródenas (2013).

4.2.2 Objective function and constraints

With the sets, parameters, and variables presented, the objective function and the constraints can now be presented.

Objective function:

$$Minimize\ Z = \sum_{i \in T} \sum_{s \in S_i} P_i \times d_{i,s} \tag{4.6}$$

Constraints:

$$d_{i,s} \geq t_{i,s} - SC_{i,s} \qquad \forall i \in T; s \in S_i \tag{4.7}$$

$$t_{i,v} - t_{i,u} \geq TOT_{i,u} \qquad \forall i \in T; u, v \in N; (u,v) \in F_i \tag{4.8}$$

$$t_{i,u} - t_{j,v} \geq ST_{j,u} - My_{i,j,u} \qquad \forall (i,j,u) \in A; i,j \in T; u,v \in N; (u,v) \in F_j \tag{4.9}$$

$$t_{j,u} - t_{i,v} \geq ST_{i,u} - M(1 - y_{i,j,u}) \qquad \forall (i,j,u) \in A; i,j \in T; u,v \in N; (u,v) \in F_i \tag{4.10}$$

$$d_{i,s} \geq 0 \qquad \forall i \in T, s \in S_i \tag{4.11}$$

$$y_{i,j,u} \in \{0,1\} \qquad \forall (i,j,u) \in A; i,j \in N \tag{4.12}$$

The objective function, as mentioned in the first model, works by minimizing the delay of the trains. This is done by Equation 4.6 which minimizes the sum of all delays of the trains at their most important points, the stations, weighted by their priority. This was chosen because it gives the model the freedom to optimally adjust the time in any block section between stations, but minimizes the delay with which trains arrive at the stations that are most important to those trains' customers.

Then the constraints are explained in the topics:

- Constraint (4.7) defines the value of the delay at a particular station as the difference between the time the train arrives at that station and the time it should have arrived according to the original schedule. If this value is negative, which means that the train arrived sooner than scheduled, the delay will be zero because it is the lower limit to the solution space of the variable.

- Constraint (4.8) enforces the maximum speed for the train, where the arrival time in the next block section of a trains route minus the arrival time in the previous block must not be less than the smallest time it would take that train to travel through that previous block. In other words, the length of that block divided by the maximum speed that train can travel. This is the fixed arc constraint, which also dictates the route that each train follows by enforcing the order of the nodes that the train will pass through.

- Constraint (4.9) appears in pair with restriction 4.10. They are the alternative arc constraints. They enforce that only one of each pair of alternative arcs in set A become active at the solution of the model. This is done using the so-called big M constraint technique, which is very important for integer programming. This technique uses a binary variable (y in this case) to ensure that only one of the constraints is imposed, depending on whether its value is 1 or 0. In the model, when y equals 1, constraint 4.9 is deactivated by the big M parameter, while 4.10 remains active, this means train i would travel before train j. On the other hand, if y is equal to 0, the opposite happens.

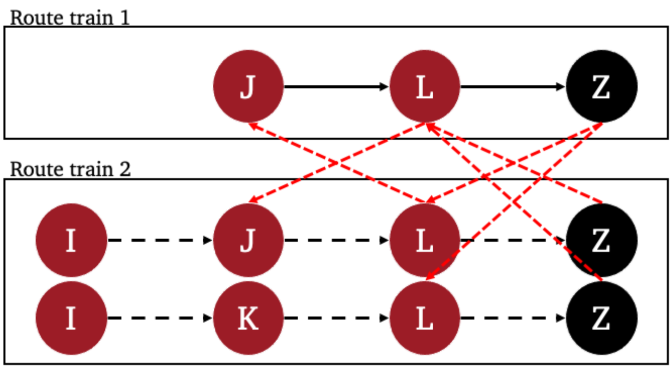
- Constraint (4.11) enforce the domain for the following decision variables. The delay must be positive.

- Constraint (4.12) enforce the domain for the following decision variables. The variable y must be binary (either have the value 1 or 0).

4.3 Third model

The third and final model incorporates the rerouting measure into the second model. This is done by extending the second model, adding some new binary variables, and extending some constraints. If model three was applied to the extended example from Figure 8 the alternative graph depicted in Figure 14 would be built.

Figure 14: Model 3 graph



Source: Own representation.

In this alternative graph, there are two possible routes for train two: either the I, J, L, Z route or the I, K, L, Z route. The fixed arcs for this train are dashed lines because in this case there is a choice between the two routes represented by the new binary variables, while for train one the only possible route is J, L, Z because the perturbation occurred while it was already in block J. The alternative arcs are also constructed between the shared block segments and the dummy node Z, which in turn is used to create the alternative arcs for the block segment L, which would be the last block in the route of both trains. An important point to note is that even for this small example, this model builds an expressively larger alternative graph.

4.3.1 Notation

Table 13: Indexes and sets of the third model.

Elements	Description
$i, j \in T$	i and j are indexes of trains in the network, and T is the set containing all trains.
$u, v \in N$	u and v are block sections in the set N of all blocks in the rail network.
$s \in S_i$	s is a station in all the routes of train $i \in T$, where $S_i \in N$ is the set of stations in that train route.
$k, l \in R_i$	k and l are routes in the set of routes from train $i \in T$, and R_i is the set of all routes that train can use.
$(u, v) \in F_{i,k}$	(u, v) is a fixed arc, where u and v are the blocks of the network N connected by the arc, and the set of fixed arcs is $F_{i,r}$ for the train $i \in T$ and route $k \in R_i$.
$(i, j, k, l, u) \in A$	i and j are the indexes of trains, $k \in R_i$ and $l \in R_j$ are the routes that those trains are following and u is the block section of possible conflict involved in the pair of alternative arc (i, j, k, l, u), where set A is the set that contains all pairs of alternative arcs.

Source: Adapted from D'Ariano et al. (2014).

Table 14: Parameters of the third model.

Elements	Description
$SC_{i,s}$	Original schedule time that train i arrived at station s.
M	A sufficiently large positive number.
$TOT_{i,u}$	Minimum time on track that train i spends on block section u.
P_i	Priority assigned to train i.
$ST_{i,u}$	Setup time of train i in node u.

Source: Adapted from D'Ariano et al. (2014).

Table 15: Decision variables of the third model.

Elements	Description
$t_{i,k,u}$	Continuous decision variable that indicates the time of arrival from train i, following route k, at block section u.
$d_{i,s}$	Continuous decision variable that indicates the delay of train i at station s.
$r_{i,k}$	Binary decision variable for route selection which, if set to one, means that train i should follow route k.
$y_{i,j,k,l,u}$	Binary decision variable of alternative arc pair that indicates the order of trains i and j at block section u.

Source: Adapted from D'Ariano et al. (2014).

4.3.2 Objective function and constraints

With the sets, parameters, and variables presented, the objective function and the constraints can now be presented.

Objective function:

$$Minimize\ Z = \sum_{i \in T} \sum_{s \in S_i} P_i \times d_{i,s} \tag{4.13}$$

Constraints:

$$d_{i,s} \geq \sum_{k \in R_i} t_{i,k,s} - SC_{i,s} \qquad \forall i \in T; s \in S_i \tag{4.14}$$

$$t_{i,k,v} - t_{i,k,u} \geq TOT_{i,u} - M(1 - r_{i,k}) \qquad \forall i \in T; u, v \in N; k \in R_i; (u,v) \in F_{i,k} \tag{4.15}$$

$$\begin{aligned} t_{i,k,u} - t_{j,l,v} \geq ST_{j,u} - My_{i,j,k,l,v} - M(1 - r_{i,k}) - M(1 - r_{j,l}) \\ \forall (i,j,k,l,u) \in A; i,j \in T; u,v \in N; (u,v) \in F_{j,l} \end{aligned} \tag{4.16}$$

$$\begin{aligned} t_{j,l,u} - t_{i,k,v} \geq ST_{i,u} - M(1 - y_{i,j,k,l,v}) - M(1 - r_{i,k}) - M(1 - r_{j,l}) \\ \forall (i,j,k,l,u) \in A; i,j \in T; u,v \in N; (u,v) \in F_{j,l} \end{aligned} \tag{4.17}$$

$$\sum_{k \in R_i} r_{i,k} = 1 \qquad \forall i \in T \tag{4.18}$$

$$d_{i,s} \geq 0 \qquad \forall i \in T, s \in S_i \tag{4.19}$$

$$y_{i,j,k,l,u} \in \{0,1\} \qquad \forall (i,j,k,l,u) \in A; i,j \in T; u \in N \tag{4.20}$$

$$r_{i,k} \in \{0,1\} \qquad \forall i \in T, k \in R \tag{4.21}$$

The objective function (4.13) remains unchanged from the first model and second model, with the goal of minimizing the sum of delays experienced by trains at the stations they visit. This is possible because, although more routes are considered, all routes are built to pass through the exact same stations as the original route, in the same order. Thus, the objective function is still applicable.

- Constraint (4.14) defines the value of the delay at a particular station as the difference between the time the train arrives at that station and the time it should have arrived according to the original schedule. The time of arrival in the solution is calculated by the sum of the time of arrival variable of all routes in that station. This is based on the assumption that only the variable of the chosen route will assume a value different from zero, while the rest of the routes will have their variables minimized to zero.

- Constraint (4.15) is the fixed arc constraint. In addition to enforcing that the train travels below its maximum speed, it also enforces the route that each train follows by dictating the order of the nodes that the train will pass through. This is altered in this model by adding the factor of more than one route to each train, which is solved by the big M method to deactivate the constraints from not selected routes, leaving only the constraints from the chosen route active.

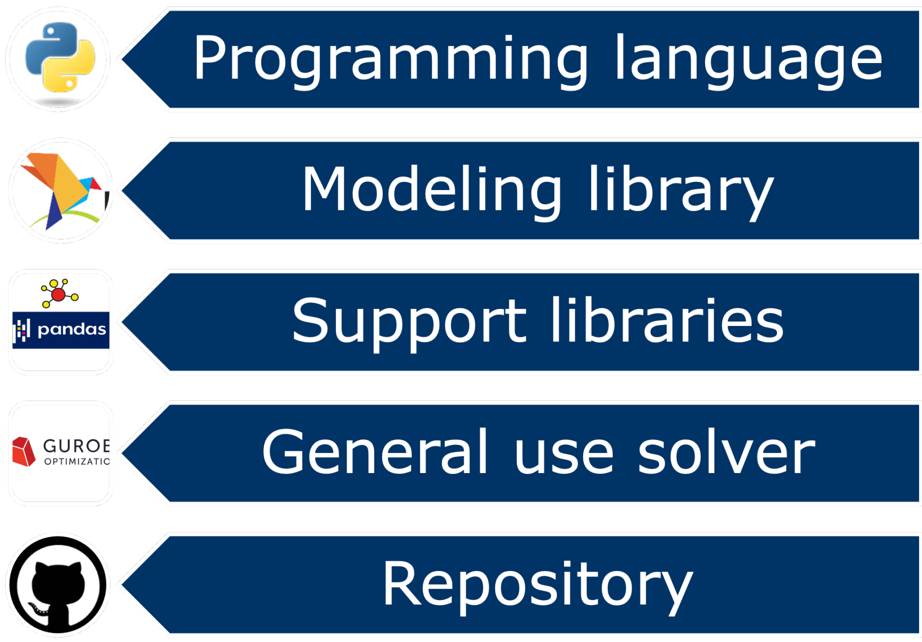
- Constraint (4.16) is the alternative arc constraints, it is always presented in pairs with (4.17). In this model, the difference here is the choice of route, which also deactivates the constraints from routes that were not chosen. To achieve this goal, the same big M method is utilized, where the value of both binary decision variables, y and r, defines whether the constraint will remain active or will be deactivated.

- Constraint (4.18) ensures that only a single route per train is selected, thereby preventing multiple routes, or no route to be chosen to each train.
- Constraint (4.20) imposes restrictions on the domain for the decision variables y. The alternative arc variable y is binary, which means that it must assume either the value of 1 or 0, indicating the order of trains i and j in block section u.
- Constraint (4.19) impose restrictions on the domain for the delay decision variables. This variables must be non-negative.
- Constraint (4.21) impose restrictions on the domain for the decision variables of route choice. The route variable r is binary, which means that it can assume either the value of 1 or 0, indicating if train i follows route k.

5 MODEL IMPLEMENTATION AND SOLUTION APPROACHES

The mathematical models were implemented in a programming language, with some helper packages to preprocess the data, and solved with a general-purpose mixed-integer-linear programming solver. All the work was saved and versioned in a repository for later use in future researches. Figure 15 shows all the implementation tools used.

Figure 15: Implementation tools



Source: Own representation.

The programming language used was Python, a general purpose programming language. It was chosen for its adaptability to solve many different problems, excellent support and great packages, as well as its growing popularity as it has become the main programming language for many of the current popular problems such as AI and working with data.

Pyomo was chosen as the modeling package for the implementation because it is an independent package designed to be compatible with most of the popular general-purpose state-of-the-art linear programming solvers such as CPLEX and Gurobi. To change the solver, only one line of code would need to be changed, which brings an interesting versatility to experiments with different solvers. All models have been programmed with this package (BYNUM et al., 2021).

Igraph and *Pandas* were used as auxiliary packages. Igraph is a Python library designed to solve complex network problems. It was used to create routes for the trains. Pandas is a library for working with data, very popular in the Python community. It can work with very large databases and is very efficient in working with tabular data. It was mostly used to preprocess the instances data and store the results, being another important package for the implementation (TEAM, 2024; CSáRDI, 2006).

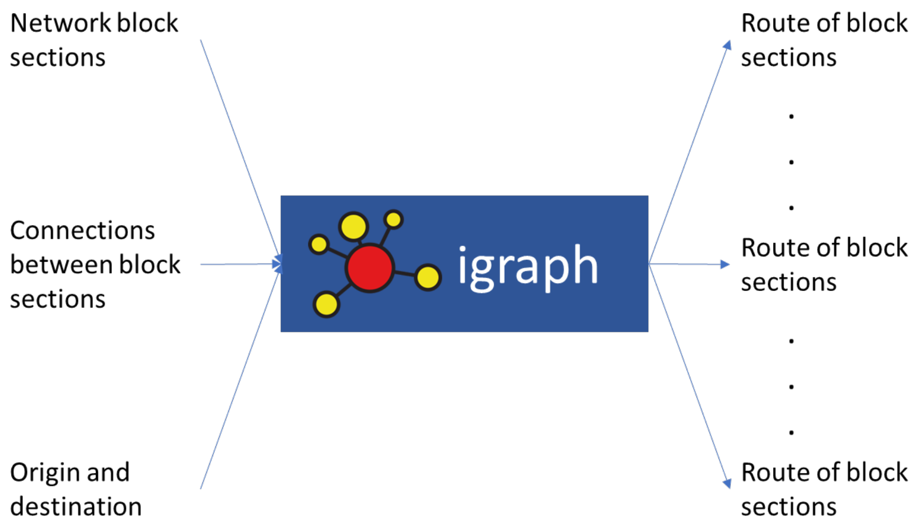
Gurobi is the general purpose linear programming solver used. It is a state of the art solver used to tackle a wide range of linear programming problems. It was used to solve the models presented to obtain the best solution available to the problem of rescheduling trains (Gurobi Optimization, LLC, 2024).

All of this was versioned using git, a versioning software commonly used for code projects. It was all stored in a repository on GitHub that was kept private during the project, with the intention of making it visible when the work was done. The repository contains most of the data used, as well as all of the code implementing the models, making it easy for future researchers to access the results of this work (URBAN, 2024). This is how the present work intends to achieve the objective of making the results of the tests available for future researchers to compare, as this is a reported gap in the literature (FANG; YANG; YAO, 2015). The only data, that are not saved in the repository, are the real-world instances, which were not allowed to be published due to CPTM rules.

5.1 Route creation

In this problem, what was called a route is usually called a path in the field of operations research. New paths were generated for the third model as well as for the real-world instances. This section explains the path generation process used. The shortest path problem is a classic problem in operations research that can be mathematically modeled and optimized. It studies how to find the shortest path of nodes in a graph between a source and a sink. However, this problem was not studied in this work, which focused on the rescheduling of trains. The problem was solved using a “black box” approach. A “black box” approach is a commonly used expression in engineering that refers to using a tool to solve a problem without getting into the details of what the tool does, the only parts we are interested in are the inputs and the outputs. The Igraph package was used to create the routes. Figure 16 depicts this approach to the routing problem.

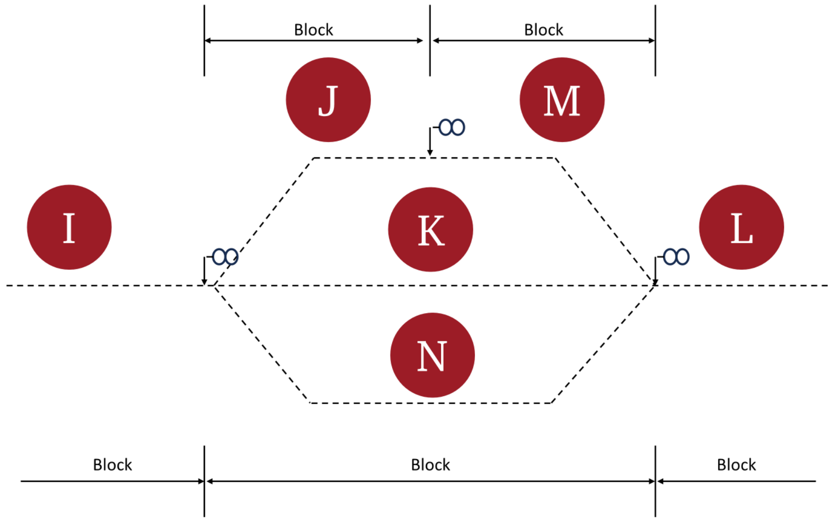
Figure 16: Routing method



Source: Own representation.

As shown in Figure 16, in order to create the paths, the package was first given the characteristics of the rail network. These are the block sections present in the network and the connections of each block. With this information Igraph built a graph with these block sections as nodes and the arcs as their connections. To illustrate this better, let us introduce a new infrastructure and then build the graph with it.

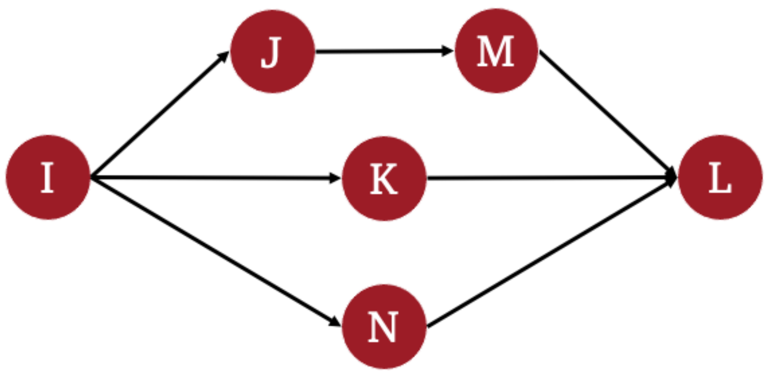
Figure 17: New infrastructure



Source: Own representation.

Figure 17 shows an infrastructure where to get from block section I to L there are three options, either go I, J, M, L or I, K, L, or I, N, L. This is translated into the graph in Figure 18, which is used to create the paths.

Figure 18: Graph from new infrastructure



Source: Own representation based on the graph created by igraph (CSáRDI, 2006).

With the graph built, the Igraph package received an origin and a destination node as input to one of two commands: all shortest path and all simple paths. Both create paths between these two nodes, where a path refers to a list of ordered nodes connected by arcs, with the first and last nodes being the origin and destination, respectively, and not repeating nodes. The all shortest path command returns the paths with the least number of block sections between the origin and the destination, this did not take into account the length of the tracks because they were not provided to the package in the input. The All Simple Paths command, on the other hand, returns all possible paths between the origin and the destination.

Now, with the graph from Figure 18, the all simple paths command between I and L returns the paths I, K, L and I, N, L, and I, J, M, L. It would not return a list like I, K, I, K, L because that list repeats nodes, and it would not return a list like I, L or I, J, L because those lists contain nodes that are not connected by arcs. These examples are not paths. The all shortest paths command, on the other hand, will return I, K, L and I, N, L, because these paths are one node shorter when compared to the other path I, J, M, L. With that explained, the next two subsections explain how these commands were used to create paths for the third model and for the real-world instances.

5.1.1 Paths created for new instances

To build the instances, data from the rail network as well as stations and train schedules were written into tables. Then the data on the infrastructure was translated and given to the Igraph package to build the route of each train using the shortest path command. All trains start their route in one station in the studied instances.

To build the route, the first station visited by a train was used as the origin, and the second station was used as the destination of the "all shortest paths" command. This returned all the shortest paths between them, if there were more than one, one was selected. Then the second and third stations were used as inputs to the same command, which returned all the shortest paths between them. Again, if there was more than one path, one was selected and added to the end of that route, and this was iterated until all the stations visited by the train were in the route.

With all the available data of the routes and the scheduled arrival time of the train at the stations, the second model was solved without perturbing the original schedule. This was done to find the original schedule time in all other block sections of each train route to use as input for the tests.

5.1.2 Paths created for the third model

To build the routes for the third model, the "all simple paths" command was initially used to solve the problem with instances available on the Internet. However, due to the complexity of the network presented on the real-world instances, this command was changed to "all shortest paths". This is because using "all simple paths" made the iterations to generate the routes for the third model too large and therefore took too long to test.

A process similar to that used to create instances was used to create the routes for the model. That is, for each train, the route started at the first node of that train's route in that instance then all the shortest paths between that first node and the next station it would visit were given. Then, for each of these paths, all shortest path between that station and the next one was added to the end of the route. This process was repeated until the last station visited by the train was in every route of the train.

After all routes were created using the process described, the model would be solved using a chosen number of routes per train. If this number was less than the number of possible routes created for a train, the routes would be selected based on the number of block sections in the route. This was done by adding all routes created for that train to a list, then sorting the list in ascending order based on the number of block sections on the routes, and then selecting the first routes to use in the model. For example, if four routes were created for a train, but only two routes per train were used to solve the third model, then the two routes with the smallest number of block sections would be used.

One simplification added to the routing problem so it would better represent the real-world example is that trains cannot change platforms in stations when being rerouted. This is different from how trains usually operate in other countries, because as an urban passenger train company, operations are organized by line, and these lines do not change platforms in stations. This restriction is taken into account when solving the routing problem for the rerouting measure.

5.2 Solution approaches

In Chapter 2, where the theoretical background was presented, two different solution approaches were introduced to help solve the model faster. These were the division of the network into subareas and the rolling horizon approach. In the present work, only the rolling horizon approach was tested in a simplified way.

The rolling horizon approach was simplified. This was done by dividing the schedule into several one-hour periods. With these instances created, the combined solution of all of them can be compared to the solution of the entire schedule at once, showing the potential processing time saved by this approach, even though the solution objective value is not fully comparable to the traditional solution because it does not introduce the constraints between solution periods.

As the subdivision of the network into subareas would require the introduction of new constraints into the model, this solution approach was not tested, as these new constraints could mean a more complex model, which might take longer to solve.

5.3 Implementation simplification

To solve the three models some simplifications were applied implicitly to the data used in the tests. In this section the simplifications are presented explicitly. The first simplification was the way in which the dwell time was handled. This time, when given in the datasets, was introduced to the minimum time on track of the block sections from the station. In this way, the model enforces minimum dwell time and maximum speed with the same constraint, but they are implied in the solution output because there is no distinction between them.

Another simplification used to build the third model was to split the problem into two steps: firstly a routing problem, secondly a scheduling problem. This simplification was already introduced in Chapter 2, as a common strategy. The routing problem has been tackled using the process described. Then the rescheduling process is done using the studied model of alternative graph.

The final simplification was done to the modeling of the schedule. A schedule of a train is composed of the time it arrives and departs from every block section in its route. However, only the arrival time was used in the model, where the departure times are calculated according to the following rule: the departure of the train from a block section is equal to the arrival time in the next block plus the setup time.

6 COMPUTATIONAL EXPERIMENTS

This chapter presents the computational experiments that were performed. This chapter is divided into three sections. Section 6.1 solves a disturbance in an example scenario created based in the infrastructure illustrated by Figure 8. Section 6.2 presents the solutions to the instances found in the literature of the Madrid railway infrastructure. Section 6.3 presents the problem solved in a real system of CPTM with some randomly created disturbances.

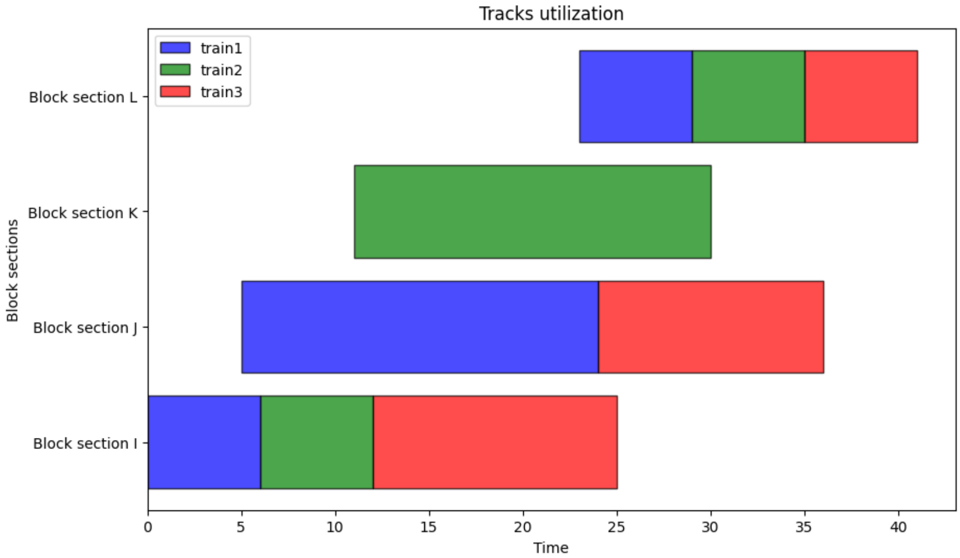
The first two sections could be used to validate the models, since these instances as well as the models are available for future reference in the Github repository (URBAN, 2024). And the third section has the purpose to show the possibility of implementing the models in the operation of a real railway company, as well as to show the potential savings in applying the more complex models compared to the solution of simpler models.

All these tests were performed on a Dell G5 laptop equipped with a CPU Core i7 9750H and 16 GB of RAM. The solution time reported here is the time taken by the solver to optimize the problem. This definition was based on the assumption that when implementing these models, the pre- and post-processing would be insignificant compared to the solver time. Furthermore, since there was no indication of priority in any of the cases, all the tests, from the example to the real case, were performed assuming that all the trains had the same priority. In other words, the priority parameters presented in Tables 8, 11, and 14 were equal to one for all the trains.

6.1 Example

In this section, the solutions from each model to the scenario presented in Section 3.1 to illustrate the problem are presented as Gantt charts. This scenario has the original timetable shown in Figure 9 and the perturbed schedule shown in Figure 10. The disturbance that occurred after train one was on block section J caused that train to be delayed fourteen units of time on that block section, creating a conflict with train three on both block sections J and L. In the original schedule train one arrived the last station at time nine, train two at fifteen and train three at twenty one.

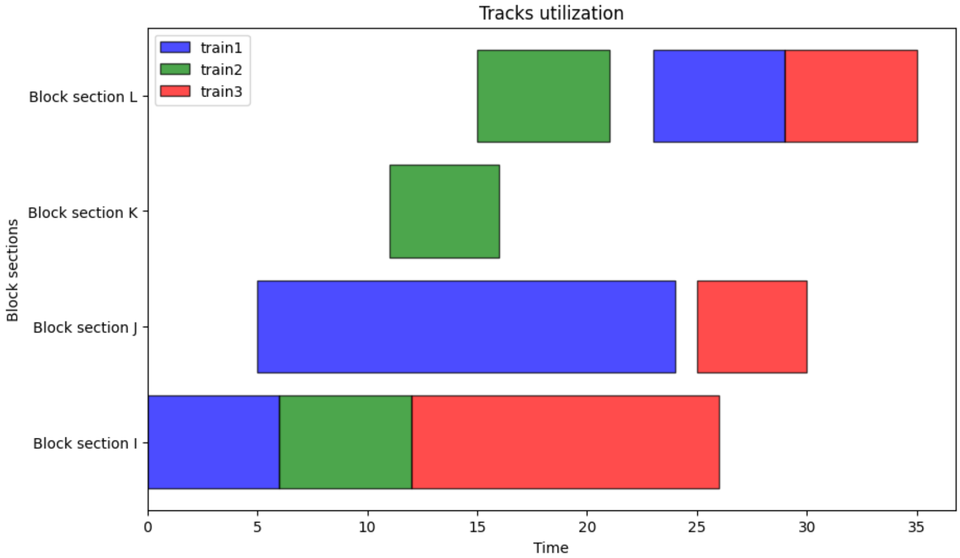
Figure 19: Solution of model 1



Source: Own representation.

The first model changes only the time at which each train arrives and leaves the block sections. The order of the trains and the routes remain the same. With only these changes, the resulting timetable is shown in Figure 19. In this solution, both trains two and three are delayed so that they do not conflict with train one. This means that train one arrives fourteen time units later than originally planned in the block section L, which was a station. This delay is also propagated to both train two and train three. The total delay of all trains in this block is forty two time units, because the arrival time in block section L is: train one at time twenty three, train two at time twenty nine, and train three at time thirty five. The only factor here that could reduce the total delay would be buffers in the original timetable, but since there is no slack in the schedule of the two block sections I and L, the delay propagates to the other trains and becomes three times the primary delay. Nevertheless, the new timetable is feasible.

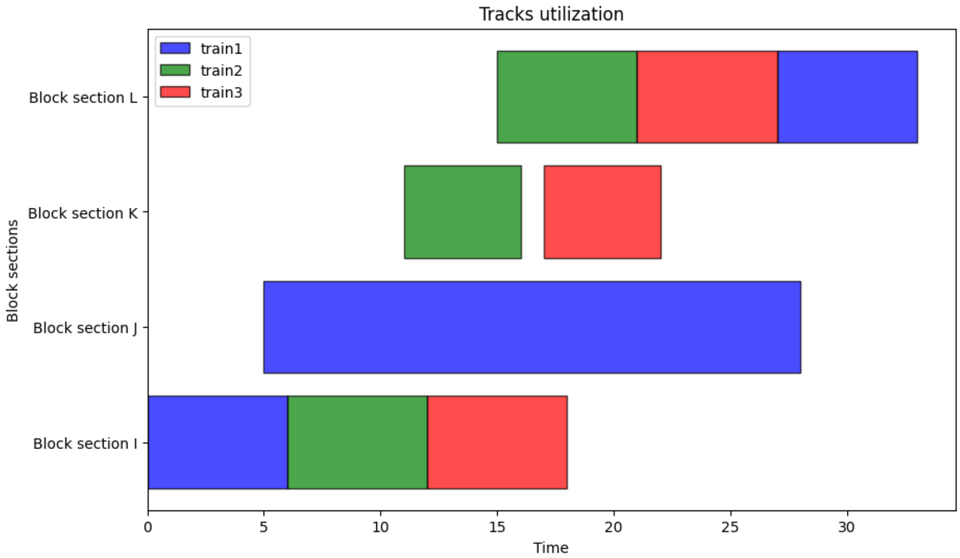
Figure 20: Solution of model 2



Source: Own representation.

Figure 20 shows the solution of the second model, which, in addition to changing the time that each train arrives at and leaves the block sections, also changes the order of the trains. This addition changes the order of the trains in the block section L, where the order was train one, then train two, and finally train three. In this solution, train two passes through this block section before train one. This change avoids the delay of train two and improves the overall delay of the system. Train three cannot be ordered before train one because its route uses the same block section that train one occupies when the disturbance occurs, block section J, which prevents the solution from improving. In this solution found by the second model, the total delay is twenty two, which means that in addition to the fourteen units of time from the primary delay, the secondary delay of train three is equal to eight units of time. This is because the arrival time in block section L is: for train one twenty three, for train two fifteen, and for train three twenty nine.

Figure 21: Solution of model 3



Source: Own representation.

Figure 21 shows the solution from the third model. This model adds the ability to reroute trains to the measures of model two. With this addition, the solution is further improved by rerouting train three to follow the same route as train two. This makes it possible to again change the order in block section L, where train three travels it before train one. This change makes the total delay of the system to be minimized to eighteen units of time. This means that this model delayed the first train four more units of time to avoid delaying the other trains. This means that the new arrival time in block section L is: for train one twenty seven, for train two fifteen, and for train three twenty one. This was the solution found for equal priority between trains. Perhaps by changing the priority the optimal solution could also be changed.

This example was designed to show the difference between each model in an ideal example to illustrate the potential improvements from using a more complex model. In this example, the solution of the third model was fifty seven percent better than the solution of the first model. However, in a real-world situation, these improvements may not be as easy to achieve, and solution time becomes an important factor in the solution approach as instances become more complex to solve. Nevertheless, the following sections present some more realistic examples of the application of the models.

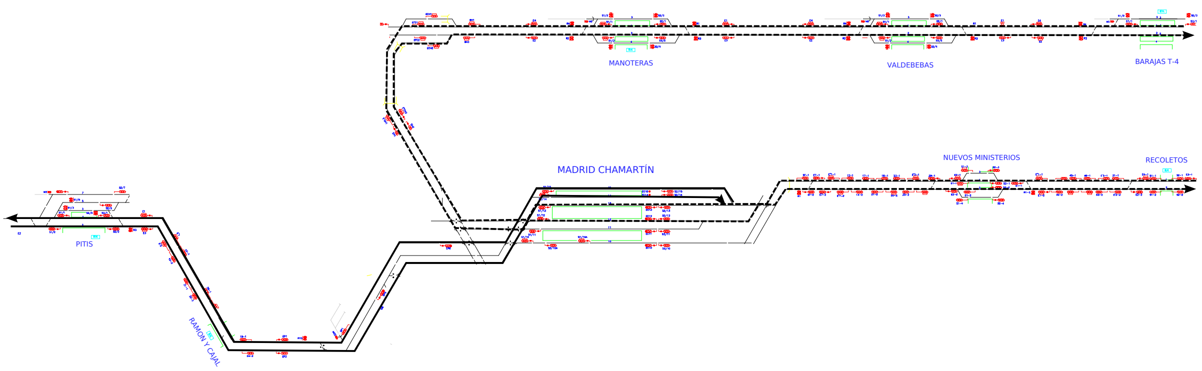
6.2 Madrid infrastructure

Here the tests carried out in the Madrid infrastructure, available online and first presented in the paper of Espinosa-Aranda & García-Ródenas (2013), are presented. The data were taken from the *HINT* project and no preprocessing was done, only a Python script was run to transform the edb documents into inputs for the models (FRÜHSTÜCK, 2023).

6.2.1 Data available on the internet

This section gives a brief description of the dataset available on the internet that was used to test the implemented models. The dataset represents the railway system of Madrid, as shown in Figure 22, and was first presented in the paper by Espinosa-Aranda & García-Ródenas (2013). Moreover, a project called *Heuristic intelligence (HINT)* also used it, providing new versions of the same dataset but already preprocessed with perturbations in it, with twenty instances of one hour of the available schedule as well as one instance of the entire twenty hours schedule (FRÜHSTÜCK, 2023). Both sources solved the train rescheduling problem, the paper using the alternative graph model and the project using a developed algorithm.

Figure 22: Madrid infrastructure



Source: Espinosa-Aranda & García-Ródenas (2013).

The dataset consists of 93 block sections, 475 trains and 20 hours of schedule. This means an average of 24 trains per hour in the system, with a peak of 36 and a minimum of 3 trains. The data is available in the paper as a XML document containing the infrastructure and train data. On the infrastructure, there is a description of each block section, including the start and end points, the size, if it is a station, the name of the station, the maximum allowed speed, if there is a signaling device, and two other attributes not relevant to the problem. There are also the connections between block sections, in other words which block sections follow each other. On the trains there is information on the time of entering the system, the maximum speed, the route assigned to it, its origin and destination stations, as well as the time of arrival planned for each of the stations on its route (ESPINOSA-ARANDA; GARCÍA-RÓDENAS, 2013).

On the site of the project that used the same data, the information is available in documents of edb or edbz format. In addition, all of the above information is summarized in the infrastructure distribution of the block section, without attributes, and the trains with their route, time on track, and an initial schedule given to each block section on its route. The instances used in this work to test the implemented models are the ones available in the project website, because they are available completely including perturbations, allowing future works to compare the results by only having access to the data provided by the *Heuristic intelligence (HINT)* project and not having to generate new perturbations (FRÜHSTÜCK, 2023).

This choice was made because each data source presents a problem for future comparison of solutions. The paper did not provide the preprocessed data, that is, the data with the perturbations, but only the data of the original timetable (ESPINOSA-ARANDA; GARCÍA-RÓDENAS, 2013). Also, the paper is not clear on how to work with dwell time as well as setup time. Therefore, the results are not exactly comparable. On the other hand, the project has not made available on its website the solutions found for the cases presented (FRÜHSTÜCK, 2023). They did publish the instances and the code to solve the problem on their website, but the results of the solution, including objective function value and solution time, were not provided. This is why the data from the project was chosen, and why the work presented here is available on a repository, including the code, data, and results, so that future work can validate the solution studied here (URBAN, 2024).

In these instances, the dwell time was already included in the station time. There is also no indication of the setup time, so for these tests the setup time was considered to be zero, meaning that as soon as a train entered a new block section, another train could enter the previous block section.

6.2.2 Solution comparison

In this subsection the first comparison between solutions is done. In Table 16 there is a summary of the solutions of the one hour long instances.

Table 16: Madrid one hour long results

			Model 1		Model 2		Model 3	
Hour of schedule	(HINT) Project instance name	Number of trains	O.V.	S.T.	O.V.	S.T.	O.V.	S.T.
1	3600-input	5	0	0,02	0	0,04	0	0,04
2	7200-input	26	6	0,01	6	0,13	6	0,12
3	10800-input	34	169	0,03	169	0,27	169	0,25
4	14400-input	36	262	0,02	262	0,42	262	0,62
5	18000-input	32	228	0,01	228	0,19	228	0,20
6	21600-input	24	304	0,05	304	0,11	304	0,11
7	25200-input	21	12	0,01	12	0,10	12	0,07
8	28800-input	22	108	0,02	108	0,06	108	0,06
9	32400-input	20	10	0,01	10	0,04	10	0,06
10	36000-input	30	349	0,01	349	0,14	349	0,14
11	39600-input	26	34	0,01	34	0,09	34	0,08
12	43200-input	27	111	0,02	111	0,10	111	0,11
13	46800-input	24	70	0,01	70	0,08	70	0,08
14	50400-input	27	35	0,01	35	0,14	35	0,12
15	54000-input	29	218	0,02	218	0,13	218	0,15
16	57600-input	27	151	0,01	151	0,11	151	0,10
17	61200-input	24	49	0,01	49	0,07	49	0,07
18	64800-input	20	12	0,02	12	0,06	12	0,06
19	68400-input	18	88	0,01	88	0,06	88	0,06
20	72000-input	3	0	0,01	0	0,01	0	0,03

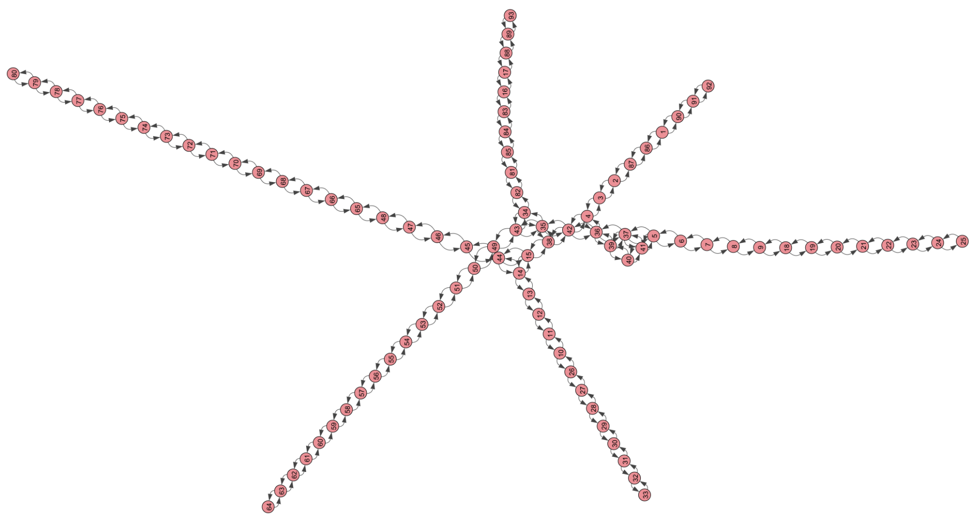
Source: Own representation with data from the paper by Espinosa-Aranda & García-Ródenas (2013).

The first two columns are references to the instance that was solved. The first one is the hour of the schedule on which the instance was built, for example, the first row with the number one means that this instance was built on the first hour of the available schedule. The second column is the name of the instance in the *HINT* project documents, which also refers to the schedule time it was built on, for example the first line is called 3600-input, which was built on the first 3600 seconds of the schedule, in other words the first hour. The next columns show the Solution Time (S.T.), in other words, the time it took the solver to reach the solution, and the final value of the objective function (O.V.).

Unfortunately, the results of the paper by Espinosa-Aranda & García-Ródenas (2013) are not comparable to the results presented here, even though the same model is applied in that paper and in the present work. This is because in the paper the disturbances applied to the original schedule are randomly generated and not made available. Furthermore, there is no indication whether the solution time also takes into account any pre- or post-processing, which would make these values different as well. Therefore, it is hoped that by making the code and solutions presented here available on Github, future works can compare their results with those found here (URBAN, 2024).

On the comparison between the models studied on the present work, all three models present the same objective value for each instance, and model one is the fastest, while model two and model three take similar time to solve each instance. This could be explained by analyzing the infrastructure of the instances. Figure 23 shows the infrastructure described in all the Madrid instances studied.

Figure 23: Madrid infrastructure graph



Source: Own representation with data from the project by Frühstück (2023).

Based on the graph, Madrid's infrastructure could be described as predominantly line based, with few block sections connected to more than two other block sections. This means that there is little possibility to change the route and order of trains. Thus, if a train was delayed, the other trains would not be able to overtake it by reordering or rerouting, making the propagation of delays much more difficult to prevent. For this reason, the objective value of the solutions of the three models is the same. The solution time of the second and third models are so similar because all instances were solved considering only one route for each train, the routing method did not create other route options for the third model. Therefore, the third model behaved like the second model, but with an additional binary variable for each train that had to have the value one. Considering that model two was taken from the literature, model one was a better fit for this type of infrastructure because it maintained the same objective function while taking less time to solve the instances.

6.2.3 Simplified rolling horizon approach

An additional analysis to make is to compare the approximation to the rolling horizon solution with the solution of the entire schedule. Table 17 brings these solution values for comparison. Here the rolling horizon approach is approximated as the sum of the solution time and objective value of all one hour long instances solutions.

Table 17: Rolling horizon comparison

Solution Approach	Model	Objective value	solution time	# of binary variables	Termination condition
Complete schedule	1	2218	2,51	0	optimal
Rolling horizon	1	2216	0,33	0	optimal
Complete schedule	2	2168	101,38	540740	optimal
Rolling horizon	2	2216	2,35	28746	optimal
Complete schedule	3	2168	275,74	541215	optimal
Rolling horizon	3	2216	2,53	29221	optimal

Source: Own representation with data from the project by Frühstück (2023).

The first column brings the information if the solution was done using the entire schedule or if it is the approximation to the rolling horizon solution approach. The second column shows the model used to solve the problem. Then the objective value and the solution time are presented in the next two columns. The last two columns show the number of binary variables generated by the models in each case and the termination condition, if the model was solved to optimality or if the time limit of six hundred seconds stopped the solver.

This table shows that for the more complex models, two and three, the rolling horizon has a great potential to reduce the solution time while maintaining most of the solution quality. This is because in the case of the third model, while the time was reduced by ninety-nine percent, the objective value was only two percent larger. However, it is interesting to note that model one could even solve the entire schedule in less than three seconds, thus being able to solve the problem in the required time without the need for the solution approach. The other models could only solve the problem in "real time" with the rolling horizon approach. The full schedule solution could have been worse than the simplified rolling horizon solution with model one because the delay from one train could have propagated to other trains, while models two and three were able to avoid this propagation in the full schedule.

6.3 Real-world tests

The instances created to test the models in a real-world scenario were inspired by the infrastructure shown in Figure 24, where they represent the real-world infrastructure and schedule with randomly generated perturbations.

Figure 24: Real-world instances infrastructure

Source: CPTM (2024c).

For the tests, data from both lines eleven and thirteen of CPTM were used to create instances that were then solved. Even though mapped, line twelve was not used for the tests. Twenty and a half hour schedules were available for each line. Some of the assumptions used to create the instances were that all trains had a maximum speed of ninety kilometers per hour, which means that the minimum time on track for all trains in all block sections was the length of that block section divided by ninety kilometers per hour. The average length of a CPTM train was approximately one hundred seventy meters, so the setup time was that length divided by ninety kilometers per hour. There was no priority differentiation between trains, so all delays were treated the same. That is, the train priority parameter was all equal to one. These assumptions were chosen based on the statement that the train fleet is homogeneous.

To create the test instances for each hour of the schedule, five trains in the system were randomly selected without repetition, for which a random block section of their route was selected. With these choices, a disturbance of one thousand to five thousand percent was applied to the time on track of the selected pair, which means that the train spent eleven to fifty one times the minimum time it was supposed to spend in that block section. This was done because the time on track in the infrastructure described were usually very small, so to create a real disturbance a high percentage of that time on track was needed, the absolute disturbance sought was about ten minutes, which was approximately found using this percentage through an empirical search. Then an instance was created with the delayed train and one hour of the rest of the system following the original schedule. Thus, by multiplying the twenty one, which was the original schedule time available rounded up, by the five instances created for each hour, one hundred and five one hour instances were created for each line. Each of these instances was combined with others that had the same percentage of disturbances by adding the trains present in each of these instances into a single instance of twenty one delayed trains. This created the five complete schedule instances for each line, which, when added to one hour instances, resulted in a total of one hundred and ten instances.

All of these one hundred and ten instances created for each line were solved by the three models with a time limit of six hundred seconds, or in other words, ten minutes, which is the time limit required by the problem for rescheduling disruptions. Tables 18 and 19 show some data about the perturbations applied to the instances.

Table 18: Perturbation data for instances from line 11

Index	Percentage perturbation	Total absolute perturbation
1	1000	7299,6
2	2000	13080,8
3	3000	18679,2
4	4000	24707,2
5	5000	21600,0

Source: Own representation.

In Table 18, the first column represents the index of the instance, which refers to the value of the percentage perturbation that was applied. This means that for the index one, the first train in each hour of the schedule was randomly selected and a thousand percent perturbation was applied to a random block section of its route, then no perturbation was applied to the remaining trains and an hour long instance was created. This process was repeated for the remaining indices. The second column shows the percentage of disturbance applied to the selected train block section pair. This means that in the index one, once the train and block section were selected for one hour of the timetable, a perturbation of one thousand percent of the minimum time on track of that train in that block section was applied. That is, the original time on track used in the maximum speed constraint was multiplied by eleven and used as input to the rescheduling problem. If the index was two, the time on track was multiplied by twenty one. The last column is the sum of all the absolute perturbations applied to this index, that is, for index one, the sum of the multiplication of ten by the time on track of the train and block section selected in each hour of the schedule was seven thousand two hundred and ninety nine seconds. This is because the disturbance was applied by multiplying the original time on track by eleven, but if we consider that the original time on track was already scheduled, the disturbance is the time on track multiplied by eleven minus one.

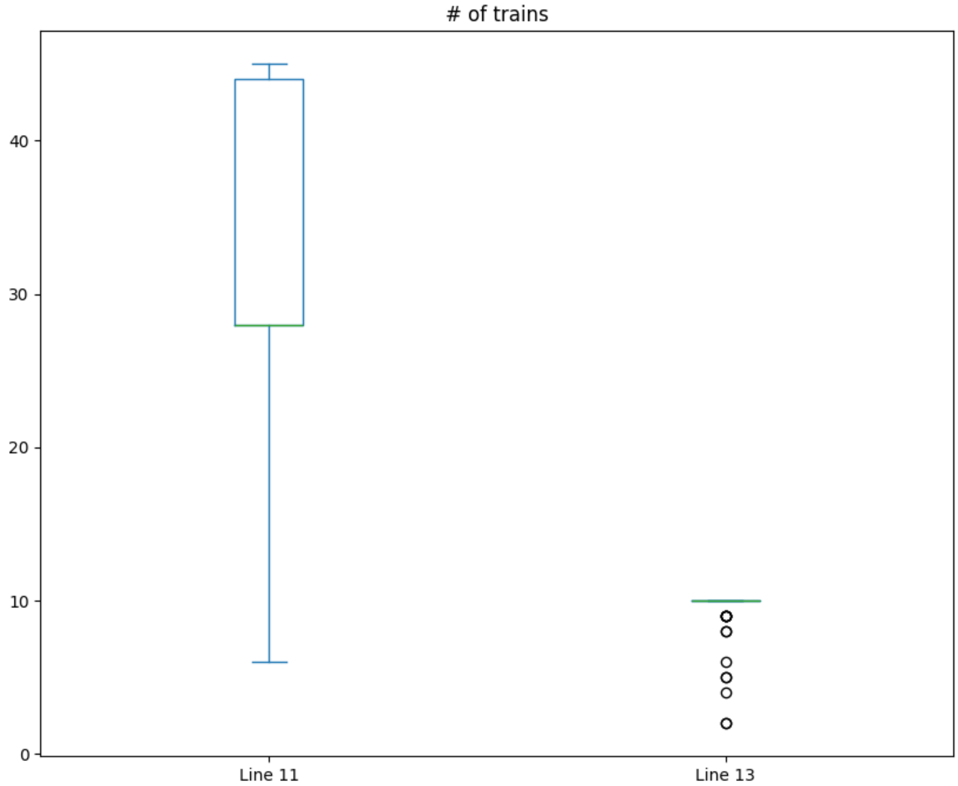
Table 19: Perturbation data for instances from line 13

Index	Percentage perturbation	Total absolute perturbation
1	1000	3824,8
2	2000	5253,6
3	3000	10935,6
4	4000	16865,6
5	5000	14276,0

Source: Own representation.

When comparing the two Tables, 18 and 19, it is noticeable that the block sections of line eleven tend to be larger, because all the combined complete schedule instances had a larger absolute perturbation on line eleven than on line thirteen, even though the speed of the trains was considered the same and the percentage perturbation applied was the same. Furthermore, line eleven had five hundred and nineteen trains passing through it in the twenty hours, while line thirteen had only one hundred and sixty four. All this means that line eleven is a more difficult line to reschedule, with a more complex problem. This is further emphasized by the Figure 25.

Figure 25: Histogram of number of trains in the one hour long instances by lines



Source: Own representation.

This Figure shows the box plot of trains in the one hour instances on both lines. Line eleven has significantly more trains with an average of about thirty one trains per instance, while line thirteen has an average of about nine trains per instance.

6.3.1 Simplified rolling horizon approach

As for the comparison between rolling horizon and solving the problem with the full schedule, Tables 20 and 21 show the results of both. Asterisk indicates that the problem was to large to be solved.

Table 20: Results from rolling horizon and complete schedule tests in line 11

		Rolling horizon			Complete schedule		
Index	**model**	**Objective value**	**solution time**	**number of binary variables**	**Objective value**	**solution time**	**number of binary variables**
0	1	45.240,08	0,42	-	20.578,16	5,79	-
1	1	110.875,64	0,39	-	101.048,72	5,60	-
2	1	205.825,60	0,39	-	232.618,32	5,74	-
3	1	224.701,44	0,36	-	212.137,72	6,89	-
4	1	243.162,40	0,36	-	232.988,84	6,13	-
0	2	32.870,76	8,23	73.730,00	15.471.232,60	600,47	958.367,00
1	2	87.059,08	43,82	74.662,00	42.629.464,36	623,16	959.466,00
2	2	119.077,48	896,30	75.970,00	39.080.052,24	642,28	947.195,00
3	2	198.072,96	989,95	75.791,00	39.756.272,08	600,45	993.249,00
4	2	224.302,00	1.465,85	72.196,00	39.779.763,44	633,68	989.721,00
0	3	263.288,20	8.993,24	1.908.945,00	*	*	*
1	3	40.550,16	9.690,43	1.927.170,00	*	*	*
2	3	7.066.578,36	10.633,16	2.012.726,00	*	*	*
3	3	85.765,28	9.923,25	1.976.221,00	*	*	*
4	3	233.939,32	10.605,78	1.907.722,00	*	*	*

Source: Own representation.

The first two columns show both the model used and the index of instances. The next six columns are divided into two groups of three, where the first presents the results from the simplified rolling horizon approach, then the second from solving the problem with the complete schedule. The first column of these groups is the objective values, the second is the solution time, and the third is the number of binary variables present in the solution. The third model could not be solved for the instances of the complete schedule in line eleven because the problem was too large to be preprocessed, causing the computer to crash in the middle of the solution, probably due to running out of memory. Also, many of the one hour long instances could not be solved by the third model in the six hundred second time limit; these had their objective value considered zero in the aggregation done to build the table, making the objective value of model three a bit deceptive. The solution of all the instances are presented in tables in the Appendix 6.3

Table 21: Results from rolling horizon and complete schedule tests in line 13

		Rolling horizon			Complete schedule		
Index	model	Objective value	solution time	number of binary variables	Objective value	solution time	number of binary variables
0	1	3.988,80	0,31	-	3.988,80	0,13	-
1	1	6.199,36	0,29	-	6.045,44	0,16	-
2	1	16.766,72	0,32	-	15.926,72	0,14	-
3	1	22.051,36	0,29	-	20.951,04	0,14	-
4	1	17.002,24	0,32	-	16.433,92	0,16	-
0	2	3.988,80	0,62	3.241,00	3.988,80	7,00	61.633,00
1	2	6.199,36	0,61	3.140,00	6.045,44	7,82	84.834,00
2	2	16.766,72	0,62	3.260,00	15.926,72	17,87	82.787,00
3	2	22.051,36	0,57	3.157,00	20.951,04	19,76	63.661,00
4	2	17.002,24	0,60	3.304,00	16.433,92	6,38	63.179,00
0	3	3.988,80	0,93	3.849,00	3.988,80	8,45	66.459,00
1	3	5.815,36	0,89	4.907,00	5.661,44	355,36	139.600,00
2	3	16.766,72	0,82	4.533,00	No objective value	600,20	152.268,00
3	3	22.051,36	0,93	3.754,00	20.951,04	11,29	68.613,00
4	3	17.002,24	0,90	3.921,00	16.433,92	9,86	68.348,00

Source: Own representation.

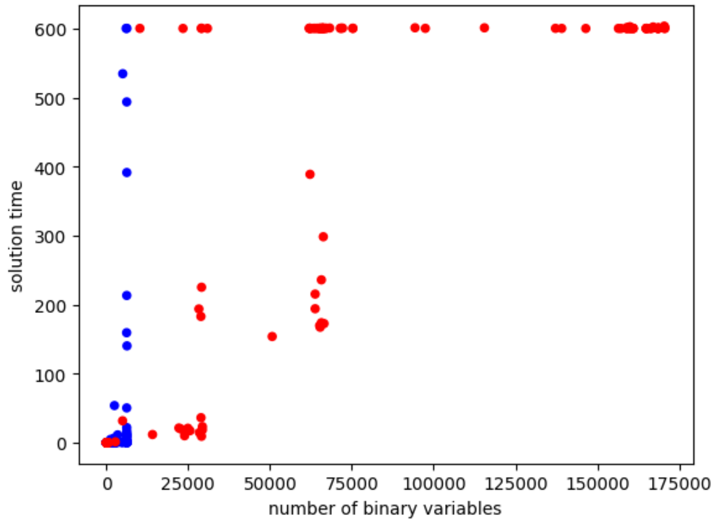
Both tables show some interesting results when comparing the solution approaches. In Table 21 the rolling horizon starts to make sense only in the second model, because in the first model, instead of speeding up the solution, it makes the solution slower, even if only by some fractions of seconds. This means that it starts to make sense to use the rolling horizon approach when the problem starts to take more than a second to solve, otherwise it makes no sense to use it because the solution of the complete schedule is already in the time constraint of the problem.

Table 20, on the other hand, shows the solution to problems much larger than the previous table, and the simplified rolling horizon approach allows these problems to be solved with more complex models, some of which even reach optimality. For example, the third model can only solve the instances from line eleven with the simplified rolling horizon approach, and the second model can only obtain solutions comparable to the first model with the simplified rolling horizon approach. However, it is interesting to note that this approach sometimes ends up taking longer to solve than the full schedule approach because the solution time was calculated by adding the time to solve each one hour instance in this simplification. This added up to some large solution times because some one hour instances took too long to solve, making the simplified total solution time with this approach larger than the proposed time limit. The first model, on the other hand, is capable of solving all sizes of problems to optimality in just a few seconds, with or without the rolling horizon approach.

6.3.2 Models comparisons

When comparing all the models, the first model stands out in the tests performed in this work. This is because it is able to obtain solutions not substantially worse than the other models much faster. This could be explained, among other factors, by the absence of integer and binary variables in this model. Because it is composed only of continuous variables the simplex method is able to solve it much faster than the methods of branch-and-bound used to solve the other models.

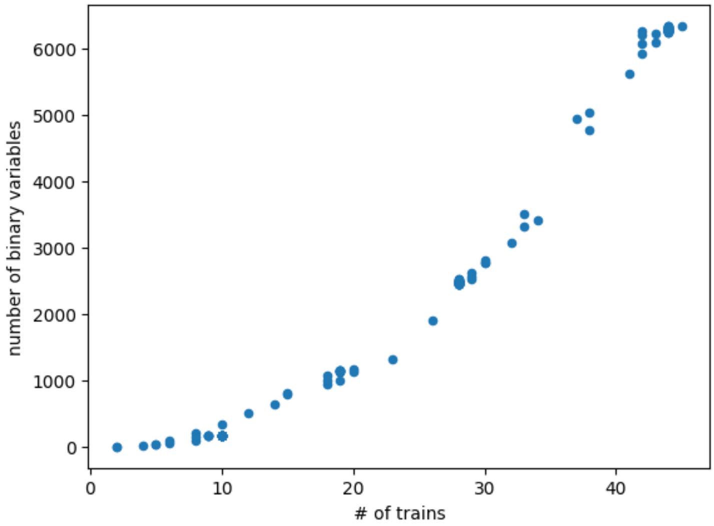
Figure 26: Binary variables and solution time of the second and third models



Source: Own representation.

The points present in the chart of Figure 26 are solutions of one hour long instances, where the blue points are the solutions of the model two while the red points are the solutions of model three. The assumption is partially illustrated by Figure 26, which shows a bound on the number of binary variables with which the models can be solved to optimality in less than six hundred seconds. Moreover, when making a correlation in the one hour long instances, the solution time has a correlation of 0.82 with the number of binary variables. This indicates that with more complex models and instances, more binary variables are needed to describe the problem, and with more of these variables, the solver takes more time to optimize the model.

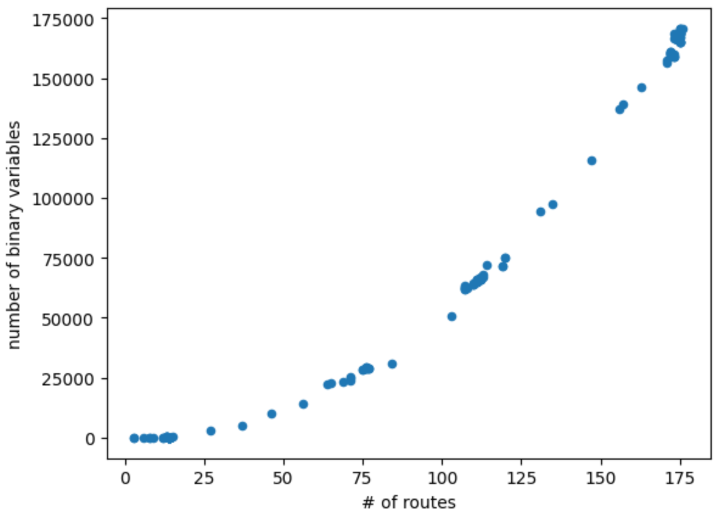
Figure 27: Binary variables by number of trains in the second model



Source: Own representation.

In this sense, Figures 27 and 28 show some strong correlations between the number of binary variables and the number of trains in the second model, and the number of routes in the third model. Therefore, reducing these factors would significantly reduce the number of binary variables in the solution and possibly accelerate the solution of the problem.

Figure 28: Binary variables by number of routes in the third model



Source: Own representation.

The solution approach studied in this work that reduces these factors in the solution was the simplified rolling horizon. An alternative would be to make the period even smaller to a few minutes in an attempt to simplify the model solution and improve the objective value. However, this has the risk of diminishing returns of the rolling horizon approach, as was shown in line thirteen with the first model, where the solver took longer to solve the problem in multiple instances than to solve one more complex instance.

In conclusion, with the results presented from both the Internet available data of Madrid and the real-world data of CPTM, the first model returns a solution in the constrained time of the real time train rescheduling problem with a solution quality that is not far from the other models. This difference in solution speed is probably due to the difference in the number of binary variables, and there are approaches to mitigate this by reducing the number of binary variables in the more complex models. Nevertheless, the difference in solution quality in the tests was not so significant, due to the infrastructure of the tests in both cases studied be based in lines, which tends to offer less opportunities to improve the solution with the rescheduling measures implemented in models two and three.

7 CONCLUSION AND FUTURE PERSPECTIVES

To summarize the present work, the problem studied was the rescheduling of trains. This is a challenging problem because trains have rigid schedules with a high probability of propagating delays to other trains. This problem is faced when unforeseen events occur in the railway infrastructure and is currently solved in most companies without the help of an optimization algorithm.

To solve this problem, an adaptation of the scheduling model for flow shops, called the alternative graph model, has been studied in the literature. With the understanding of this model, two other ones were proposed, a simplified model that eliminated the need for integer variables, and an extension that introduced the decision of choosing the route a train would take.

With the implementation of these three models, all of them were tested in the instances available on the Internet, as well as in instances created based in the infrastructure of a large urban railway company in São Paulo. Since all instances were based in a line organization, the simplified model not only had the fastest solution, being able to be solved in real time without any additional approach, but also the solution quality was similar to the other models.

Although there are solution approaches that simplify the instances to allow more complex models to solve the problem in a reasonable time, the first model is a good compromise to be implemented in the cases studied. This is because the line organization reduces the ability of rerouting and reordering measures to improve the solution. Moreover, the implementation of the first model can already contribute to the solution of the problem by suggesting a new feasible solution.

7.1 Research questions answered

The first research question was how the rescheduling problem is modeled and optimized in the literature. This is more commonly done by the alternative graph model presented in the theory background and implemented in this work as the second model.

With this answer, the second question was whether these models could solve complex instances, including real-world ones, in real time. The answer to this question is more complex. While the second model was able to solve the Madrid instances as well as the instances from line thirteen in real time, it had to use the rolling horizon solution approach, which means that the data had to be divided into smaller instances of one hour duration in order to be solved in real time. Thus, the second model was able to solve the entire schedule of most instances in real time. However, line eleven had a few instances that could not be solved by this model in real time even with the solution approach, which means that this model cannot solve all real-world instances in real time.

The final question was whether there was a way to improve this process. To improve this process, the proposed simplified model is the best option to be used in an implementation in a line-based infrastructure. Morever, other options to improve the applicability of the models studied to the problem are presented in the next section as opportunities for future work.

With that said, this work hopes to have contributed by introducing some real-world examples that challenge the most popular model in the literature, as well as introducing a simplified model that could solve these examples in real time.

Furthermore, the objective of studying the train rescheduling problem has been achieved by researching the literature and understanding the context of the problem, then understanding the model that translates the problem into mathematical models, and even introducing some slight variations in the first and third models. Lastly, applying the models to instances created based on the real-world example of the CPTM infrastructure in the sixth chapter. All the code and results of the data available on the Internet are stored in a Github repository so that future works can use it to build on the work done here (URBAN, 2024).

7.2 Opportunities for future work

There are opportunities for future work on two main fronts: improving the models and working to implement them. To improve the models, there are a few points that could be studied. First, there are many rescheduling measures, some of which have been presented in Table 3, that have not yet been mathematically modeled. Modeling these measures is an interesting way to improve the quality of the solution. There are also possibilities to explore in the sense of reducing the solution time of the more complex models. Some alternatives to be studied are to make a real implementation of the rolling horizon approach, instead of the simplification applied in the present work, and to reduce the periods of the rolling horizon approach to less than one hour. In addition, there is the possibility of modifying the model to have a binary variable for each pair of trains instead of each pair of alternative arcs, thus possibly reducing significantly the number of binary variables. Another point is that the models tend to reduce the time trains spend on the track to the minimum time on the track when this is not necessary. To make the model more consistent with the real-world, modifying the model so that the train uses all of the time available to travel through the block section could be an interesting avenue for future work, as it would improve the applicability of the models to the real-world problem.

When considering the implementation of these models, there are many other areas that need to be investigated. First, there is the optimization of the pre- and post-processing of the data for the models, which was assumed to be irrelevant in this work, but would need to be optimized in an actual implementation. There is also the perspective of giving the tool to the dispatchers so that they can use it and give feedback on the results, as well as validate that this computational tool would help them.

REFERENCES

ALLIANZ PRO SCHIENE. *Elektrifizierung*. 2024. Disponível em: ⟨https://www.allianz-pro-schiene.de/themen/infrastruktur/elektrifizierung-bahn/⟩.

BAI, Z. et al. A rescheduling approach for freight railway considering equity and efficiency by an integrated genetic algorithm. *Journal of Advanced Transportation*, v. 2023, p. 1–22, 2023. ISSN 0197-6729.

BALSER, M. Deutsche bahn hat 16 prozent ihrer schienen stillgelegt. *Süddeutsche Zeitung*, 2018. Disponível em: ⟨https://www.sueddeutsche.de/wirtschaft/deutsche-bahn-deutsche-bahn-hat-16-prozent-ihrer-schienen-stillgelegt-1.4268351⟩.

BISCHI, A. et al. A rolling-horizon optimization algorithm for the long term operational scheduling of cogeneration systems. *Energy*, v. 184, p. 73–90, 2019. ISSN 03605442.

BLUM, J.; ESKANDARIAN, A. Enhancing intelligent agent collaboration for flow optimization of railroad traffic. *Transportation Research Part A: Policy and Practice*, v. 36, n. 10, p. 919–930, 2002. ISSN 0965-8564. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/s0965856402000022⟩.

BYNUM, M. L. et al. *Pyomo–optimization modeling in python*. Third. [S.l.]: Springer Science & Business Media, 2021. v. 67.

CACCHIANI, V. et al. An overview of recovery models and algorithms for real-time railway rescheduling. *Transportation Research Part B: Methodological*, v. 63, p. 15–37, 2014. ISSN 0191-2615.

CAPPART, Q.; SCHAUS, P. Rescheduling railway traffic on real time situations using time-interval variables. In: *Lecture Notes in Computer Science*. Springer, Cham, 2017. p. 312–327. Disponível em: ⟨https://link.springer.com/chapter/10.1007/978-3-319-59776-8_26⟩.

CORMAN, F. et al. Bi-objective conflict detection and resolution in railway traffic management. *Transportation Research Part C: Emerging Technologies*, v. 20, n. 1, p. 79–94, 2012. ISSN 0968090X.

CORMAN, F. et al. Optimal inter-area coordination of train rescheduling decisions. *Transportation Research Part E: Logistics and Transportation Review*, v. 48, n. 1, p. 71–88, 2012. ISSN 13665545.

CPTM. *A CPTM*. 2024. Disponível em: ⟨https://www.cptm.sp.gov.br/a-companhia/Pages/a-companhia.aspx⟩.

CPTM. *Embarcados Acumulados - 2023*. 2024. Disponível em: ⟨https://www.cptm.sp.gov.br/Transparencia/Pages/Embarcados-Acumulados.aspx⟩.

CPTM. *Linhas CPTM*. 2024. Disponível em: ⟨https://www.cptm.sp.gov.br/sua-viagem/Pages/Linhas.aspx⟩.

CSáRDI, T. N. G. The igraph software package for complex network research. *InterJournal*, Complex Systems, p. 1695, 2006. Disponível em: ⟨https://igraph.org⟩.

D'ARIANO, A. et al. Reordering and local rerouting strategies to manage train traffic in real time. *Transportation Science*, v. 42, n. 4, p. 405–419, 2008. ISSN 0041-1655.

D'ARIANO, A.; PACCIARELLI, D.; PRANZO, M. A branch and bound algorithm for scheduling trains in a railway network. *European Journal of Operational Research*, v. 183, n. 2, p. 643–657, 2007. ISSN 03772217. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/s0377221706010678⟩.

D'ARIANO, A.; PACCIARELLI, D.; PRANZO, M. Assessment of flexible timetables in real-time traffic management of a railway bottleneck. *Transportation Research Part C: Emerging Technologies*, v. 16, n. 2, p. 232–245, 2008. ISSN 0968090X.

D'ARIANO, A. et al. Evaluating the applicability of advanced techniques for practical real-time train scheduling. *Transportation Research Procedia*, v. 3, p. 279–288, 2014. ISSN 23521465.

DEUTSCHE WELLE. Atrasos de trens põem pontualidade alemã em xeque. *Deutsche Welle*, 2024. Disponível em: ⟨https://www.dw.com/pt-br/atrasos-de-trens-p%C3%B5em-pontualidade-alem%C3%A3-em-xeque/a-67965916⟩.

DOLLEVOET, T. et al. Application of an iterative framework for real-time railway rescheduling. *Computers & Operations Research*, v. 78, p. 203–217, 2017. ISSN 03050548.

DPA picture alliance. Deutsche bahn war 2022 so unpünktlich wie nie. *MDR*, 2022. Disponível em: ⟨https://www.mdr.de/nachrichten/deutschland/wirtschaft/deutsche-bahn-unpuenktlich-wie-nie-100.html⟩.

ESPINOSA-ARANDA, J. L.; GARCÍA-RÓDENAS, R. A demand-based weighted train delay approach for rescheduling railway networks in real time. *Journal of Rail Transport Planning & Management*, v. 3, n. 1-2, p. 1–13, 2013. ISSN 22109706.

FANG, W.; YANG, S.; YAO, X. A survey on problem models and solution approaches to rescheduling in railway networks. *IEEE Transactions on Intelligent Transportation Systems*, v. 16, n. 6, p. 2997–3016, 2015. ISSN 1524-9050.

FISCHETTI, M.; MONACI, M. Using a general-purpose mixed-integer linear programming solver for the practical solution of real-time train rescheduling. *European Journal of Operational Research*, v. 263, n. 1, p. 258–264, 2017. ISSN 03772217.

FRÜHSTÜCK, M. *Heuristic Intelligence (HINT): Train Rescheduling Problem*. 2023. Disponível em: ⟨http://isbi.aau.at/hint/hint/2-uncategorised/10-train-rescheduling-problem.html⟩.

G1. *Atrasos, superlotação, acidentes: veja o sufoco de quem depende de trens em São Paulo*. 2024. Disponível em: ⟨https://g1.globo.com/fantastico/noticia/2023/05/14/atrasos-superlotacao-acidentes-veja-o-sufoco-de-quem-depende-de-trens-em-sao-paulo.ghtml⟩.

GHAEMI, N.; CATS, O.; GOVERDE, R. M. P. Railway disruption management challenges and possible solution directions. *Public Transport*, v. 9, n. 1-2, p. 343–364, 2017. ISSN 1613-7159.

GODWIN, T.; GOPALAN, R. A.; NARENDRAN, T. T. Freight train routing and scheduling in a passenger rail network: Computational complexity and the stepwise dispatching heuristic. *Asia-Pacific Journal of Operational Research*, v. 24, n. 04, p. 499–533, 2007. ISSN 0217-5959.

Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*. 2024. Disponível em: ⟨https://www.gurobi.com⟩.

HARROD, S. S. A tutorial on fundamental model structures for railway timetable optimization. *Surveys in Operations Research and Management Science*, v. 17, n. 2, p. 85–96, 2012. ISSN 1876-7354.

HÖLZL, V. Deutsche bahn: Bahn so unpünktlich wie seit acht jahren nicht. *Die Zeit*, 2023. Disponível em: ⟨https://www.zeit.de/gesellschaft/2023-12/deutsche-bahn-puenktlichkeit-baustellen-werte-vergleich-jahre⟩.

KRAAY, D. R.; HARKER, P. T. Real-time scheduling of freight railroads. *Transportation Research Part B: Methodological*, v. 29, n. 3, p. 213–229, 1995. ISSN 0191-2615. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/019126159400034w⟩.

LAMORGESE, L.; MANNINO, C. The track formulation for the train dispatching problem. *Electronic Notes in Discrete Mathematics*, v. 41, p. 559–566, 2013. ISSN 15710653.

LAMORGESE, L.; MANNINO, C. An exact decomposition approach for the real-time train dispatching problem. *Operations Research*, v. 63, n. 1, p. 48–64, 2015. ISSN 0030-364X.

LUSBY, R. M. et al. Railway track allocation: models and methods. *OR Spectrum*, v. 33, n. 4, p. 843–883, 2011. ISSN 0171-6468.

MAHNKEN, D. The future of rail freight in europe. *DHL Freight Connections*, 2023. Disponível em: ⟨https://dhl-freight-connections.com/en/trends/the-future-of-rail-freight-in-europe/⟩.

MANINO, C. (Ed.). *Real-time traffic control in railway systems: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany*. [S.l.: s.n.], 2011.

MASCIS, A.; PACCIARELLI, D. Job-shop scheduling with blocking and no-wait constraints. *European Journal of Operational Research*, v. 143, n. 3, p. 498–517, 2002. ISSN 03772217. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/s0377221701003381⟩.

NARAYANASWAMI, S.; RANGARAJ, N. Scheduling and rescheduling of railway operations: A review and expository analysis. *Technology Operation Management*, v. 2, n. 2, p. 102–122, 2011. ISSN 2249-2364.

Office of Rail and Road. Passenger rail performance – april to june 2023. 2023.

PRC Rail Consulting Ltd. *Signalling — The Railway Technical Website — PRC Rail Consulting Ltd*. 30/01/2023. Disponível em: ⟨http://www.railway-technical.com/signalling/⟩.

QU, W.; CORMAN, F.; LODEWIJKS, G. (Ed.). *A Review of Real Time Railway Traffic Management During Disturbances*. Cham: Springer International Publishing, 2015. v. 9335. (Lecture Notes in Computer Science, v. 9335). ISBN 978-3-319-24263-7.

REYNOLDS, E. *Modelling, solution and evaluation techniques for Train Timetable Rescheduling via optimisation*. [s.n.], 2021. Disponível em: ⟨https://search.proquest.com/openview/7337885d0825fdf1f2e2074805ae17bc/1?pq-origsite=gscholar&cbl=2026366&diss=y⟩.

SHARMA, B. et al. A review of passenger-oriented railway rescheduling approaches. *European Transport Research Review*, v. 15, n. 1, 2023. ISSN 1866-8887.

STEPPER, M. Rail freight – pros and cons of rail transport. *DHL Freight Connections*, 2023. Disponível em: ⟨https://dhl-freight-connections.com/en/trends/rail-freight-pros-and-cons-of-rail-transport/⟩.

TEAM, T. pandas development. *pandas-dev/pandas: Pandas*. Zenodo, 2024. Disponível em: ⟨https://doi.org/10.5281/zenodo.13819579⟩.

TOKER, B. *Blue Sinalização Com Bandeiras Na Estação De Trem fotos de stock, imagens e fotos royalty-free - iStock*. 2024. Disponível em: ⟨https://www.istockphoto.com/br/search/2/image-film?phrase=blue+sinalizao+com+bandeiras+na+estao+de+trem⟩.

TÖRNQUIST, J. (Ed.). *Computer-based decision support for railway traffic scheduling and dispatching: A review of models and algorithms: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany*. [S.l.: s.n.], 2006.

URBAN, F. M. *Netzweite Fahrplananpassung bei Zeitlicher Unsicherheit in einem Schienennetz*. 2023.

URBAN, F. M. *Felipe-MU/Train-Rescheduling-Problem-TCC*. GitHub: [s.n.], 2024. Disponível em: ⟨https://github.com/Felipe-MU/Train-Rescheduling-Problem-TCC⟩.

VISENTINI, M. S. et al. Review of real-time vehicle schedule recovery methods in transportation services. *Journal of Scheduling*, v. 17, n. 6, p. 541–567, 2014. ISSN 1099-1425.

Wayne Winston. *Operations Research Applications and Algorithms*. 4th. ed. [S.l.: s.n.], 2003.

YAMADA, T.; NAKANO, R. (Ed.). *Job shop scheduling*. Stevenage: Inst. of Electrical Engineers, 1997. v. 55. (Institution of Electrical Engineers: IEE control engineering series, v. 55). ISBN 9780852969021.

APPENDIX A – REAL-WORLD TESTS RESULTS

In this appendix the results from the tests in the real data are presented. These are here so the reader can explore the results, because the company did not allowed their data to be published for future reference.

In the first column of the next Tables the time of the instances are presented. This is similar to the Table 16. Then the index of the perturbation is on the next column. The third column presents the absolute perturbation applied in the instance. Than the next column presents the model that was used to solve the problem. Than the number of trains, the objective value of the solution and the solution time are presented in the next three columns. Finally the number of routes in average for the trains used for the solution, as well as the termination condition and the number of binary variables are shown for each instance solved.

Table 22: Line 11 model 1

Time	Index	percentage perturbation	absolute perturbation	model	# of trains	Objective value	solution time	# of routes	Termination condition	number of binary variables
4	0	1000	254,8	model 1	44	955,36	0,023	1	optimal	0
4	1	2000	912	model 1	44	10990,52	0,022	1	optimal	0
4	2	3000	840	model 1	44	9380	0,018	1	optimal	0
4	3	4000	1019,2	model 1	42	13642,88	0,018	1	optimal	0
4	4	5000	1480	model 1	45	28014,72	0,018	1	optimal	0
5	0	1000	267,2	model 1	44	1042,16	0,019	1	optimal	0
5	1	2000	271,2	model 1	44	1133,88	0,021	1	optimal	0
5	2	3000	764,4	model 1	44	6957,6	0,017	1	optimal	0
5	3	4000	1979,2	model 1	44	25613,84	0,017	1	optimal	0
5	4	5000	678	model 1	44	6214,88	0,019	1	optimal	0
6	0	1000	1000	model 1	41	23185,84	0,023	1	optimal	0
6	1	2000	160	model 1	44	240,48	0,019	1	optimal	0
6	2	3000	240	model 1	44	1281,68	0,017	1	optimal	0
6	3	4000	889,6	model 1	44	5405,76	0,018	1	optimal	0
6	4	5000	530	model 1	44	5985,04	0,017	1	optimal	0
7	0	1000	222,4	model 1	38	427,68	0,038	1	optimal	0
7	1	2000	2000	model 1	29	39340,48	0,016	1	optimal	0
7	2	3000	748,8	model 1	32	5091,48	0,018	1	optimal	0
7	3	4000	560	model 1	34	2738,68	0,019	1	optimal	0
7	4	5000	400	model 1	29	750,48	0,012	1	optimal	0
8	0	1000	224	model 1	29	472,12	0,016	1	optimal	0
8	1	2000	600	model 1	28	2291,76	0,016	1	optimal	0
8	2	3000	840	model 1	28	4706,64	0,016	1	optimal	0
8	3	4000	542,4	model 1	28	1945,68	0,013	1	optimal	0
8	4	5000	1274	model 1	28	9272,32	0,017	1	optimal	0
9	0	1000	135,6	model 1	28	271,2	0,016	1	optimal	0
9	1	2000	912	model 1	28	4791,2	0,013	1	optimal	0
9	2	3000	3000	model 1	28	40700	0,015	1	optimal	0
9	3	4000	1307,2	model 1	28	9737,12	0,017	1	optimal	0
9	4	5000	100	model 1	28	300	0,017	1	optimal	0
10	0	1000	182,8	model 1	28	365,6	0,025	1	optimal	0
10	1	2000	624,8	model 1	28	2465,36	0,013	1	optimal	0
10	2	3000	398,4	model 1	28	1157,36	0,015	1	optimal	0
10	3	4000	542,4	model 1	28	1945,68	0,015	1	optimal	0
10	4	5000	1634	model 1	28	13985,36	0,015	1	optimal	0
11	0	1000	868	model 1	28	3527,2	0,014	1	optimal	0
11	1	2000	989,6	model 1	28	8376,32	0,018	1	optimal	0
11	2	3000	420	model 1	28	1264,08	0,014	1	optimal	0
11	3	4000	1627,2	model 1	28	13876,56	0,017	1	optimal	0
11	4	5000	1480	model 1	28	12284,28	0,014	1	optimal	0
12	0	1000	140	model 1	28	280	0,028	1	optimal	0
12	1	2000	160	model 1	28	480	0,015	1	optimal	0
12	2	3000	312	model 1	28	811,76	0,015	1	optimal	0
12	3	4000	531,2	model 1	28	1878,48	0,017	1	optimal	0
12	4	5000	700	model 1	28	2991,76	0,017	1	optimal	0
13	0	1000	494,8	model 1	28	2494,88	0,015	1	optimal	0
13	1	2000	641,6	model 1	28	3872,68	0,015	1	optimal	0
13	2	3000	840	model 1	28	4706,64	0,021	1	optimal	0
13	3	4000	731,2	model 1	28	3232,16	0,015	1	optimal	0
13	4	5000	1500	model 1	28	11940,08	0,014	1	optimal	0
14	0	1000	296	model 1	28	760,12	0,032	1	optimal	0
14	1	2000	499,2	model 1	30	2534,48	0,016	1	optimal	0
14	2	3000	318	model 1	28	1247,72	0,016	1	optimal	0
14	3	4000	1068,8	model 1	28	6692,6	0,018	1	optimal	0
14	4	5000	914	model 1	30	5036,32	0,025	1	optimal	0
15	0	1000	249,6	model 1	42	299,36	0,022	1	optimal	0
15	1	2000	560	model 1	43	4948,92	0,016	1	optimal	0
15	2	3000	3000	model 1	42	78300,64	0,02	1	optimal	0
15	3	4000	531,2	model 1	44	3948,8	0,023	1	optimal	0
15	4	5000	1562	model 1	33	13906,88	0,018	1	optimal	0
16	0	1000	182,8	model 1	44	558,08	0,016	1	optimal	0
16	1	2000	989,6	model 1	44	6637,28	0,034	1	optimal	0
16	2	3000	60	model 1	44	60	0,015	1	optimal	0
16	3	4000	1019,2	model 1	44	12004,48	0,017	1	optimal	0
16	4	5000	1274	model 1	43	21333,92	0,019	1	optimal	0
17	0	1000	374	model 1	44	1869,2	0,02	1	optimal	0
17	1	2000	641,6	model 1	44	8048,08	0,016	1	optimal	0
17	2	3000	1220,4	model 1	44	17033,44	0,019	1	optimal	0
17	3	4000	16	model 1	44	16	0,017	1	optimal	0
17	4	5000	100	model 1	44	120,16	0,019	1	optimal	0
18	0	1000	456	model 1	44	2664	0,02	1	optimal	0
18	1	2000	560	model 1	44	4948,92	0,022	1	optimal	0
18	2	3000	398,4	model 1	44	2280,32	0,018	1	optimal	0
18	3	4000	80	model 1	44	80,16	0,022	1	optimal	0
18	4	5000	1870	model 1	44	45616,08	0,025	1	optimal	0
19	0	1000	406,8	model 1	30	2172	0,013	1	optimal	0
19	1	2000	624,8	model 1	42	4765,68	0,023	1	optimal	0
19	2	3000	60	model 1	37	180	0,019	1	optimal	0
19	3	4000	1627,2	model 1	44	31194,64	0,021	1	optimal	0
19	4	5000	1870	model 1	38	40139	0,018	1	optimal	0
20	0	1000	267,2	model 1	26	1042,16	0,014	1	optimal	0
20	1	2000	592	model 1	19	1644,12	0,018	1	optimal	0
20	2	3000	937,2	model 1	33	10958,88	0,02	1	optimal	0
20	3	4000	3472	model 1	23	47229,12	0,015	1	optimal	0
20	4	5000	940	model 1	28	9210,08	0,017	1	optimal	0
21	0	1000	374	model 1	19	772,16	0,015	1	optimal	0
21	1	2000	208	model 1	19	416	0,014	1	optimal	0
21	2	3000	1484,4	model 1	19	11359,52	0,015	1	optimal	0
21	3	4000	1307,2	model 1	19	6181,88	0,013	1	optimal	0
21	4	5000	1562	model 1	20	8257,6	0,016	1	optimal	0
22	0	1000	374	model 1	19	772,16	0,015	1	optimal	0
22	1	2000	534,4	model 1	20	1401,36	0,015	1	optimal	0
22	2	3000	764,4	model 1	19	2437,28	0,055	1	optimal	0
22	3	4000	1824	model 1	19	10952,4	0,016	1	optimal	0
22	4	5000	100	model 1	19	300	0,011	1	optimal	0
23	0	1000	280	model 1	18	560	0,016	1	optimal	0
23	1	2000	8	model 1	15	24	0,03	1	optimal	0
23	2	3000	548,4	model 1	18	1457,36	0,015	1	optimal	0
23	3	4000	560	model 1	18	1396,12	0,015	1	optimal	0
23	4	5000	520	model 1	15	1343,76	0,017	1	optimal	0
24	0	1000	249,6	model 1	14	748,8	0,025	1	optimal	0
24	1	2000	592	model 1	8	1524,12	0,013	1	optimal	0
24	2	3000	1484,4	model 1	6	4453,2	0,013	1	optimal	0
24	3	4000	3472	model 1	10	24988,4	0,013	1	optimal	0
24	4	5000	1112	model 1	12	6159,68	0,012	1	optimal	0
linha11 20 hours	0	1000	7299,6	model 1	519	20578,16	5,787	1	optimal	0
linha11 20 hours	1	2000	13080,8	model 1	519	101048,7	5,596	1	optimal	0
linha11 20 hours	2	3000	18679,2	model 1	519	232618,3	5,743	1	optimal	0
linha11 20 hours	3	4000	24707,2	model 1	519	212137,7	6,89	1	optimal	0
linha11 20 hours	4	5000	21600	model 1	519	232988,8	6,128	1	optimal	0

Table 23: Line 11 model 2

Time	Index	percentage perturbation	absolute perturbation	model	# of trains	Objective value	solution time	# of routes	Termination condition	number of binary variables
4	0	1000	254,8	model 2	44	955,36	0,365	1	optimal	6248
4	1	2000	912	model 2	44	10005,24	10,249	1	optimal	6275
4	2	3000	840	model 2	44	8877,28	1,99	1	optimal	6258
4	3	4000	1019,2	model 2	42	11378,36	50,322	1	optimal	6249
4	4	5000	1480	model 2	45	24155,44	140,234	1	optimal	6326
5	0	1000	267,2	model 2	44	1042,16	0,605	1	optimal	6274
5	1	2000	271,2	model 2	44	1133,88	0,609	1	optimal	6310
5	2	3000	764,4	model 2	44	6882,12	11,958	1	optimal	6248
5	3	4000	1979,2	model 2	44	22830,84	391,383	1	optimal	6274
5	4	5000	678	model 2	44	5995,56	7,362	1	optimal	6310
6	0	1000	1000	model 2	41	10816,52	2,104	1	optimal	5609
6	1	2000	160	model 2	44	240,48	0,378	1	optimal	6299
6	2	3000	240	model 2	44	1281,68	0,384	1	optimal	6284
6	3	4000	889,6	model 2	44	5405,76	12,939	1	optimal	6266
6	4	5000	530	model 2	44	4544,96	0,621	1	optimal	6249
7	0	1000	222,4	model 2	38	427,68	0,314	1	optimal	4772
7	1	2000	2000	model 2	29	20776,52	1,609	1	optimal	2611
7	2	3000	748,8	model 2	32	4310,12	0,216	1	optimal	3068
7	3	4000	560	model 2	34	2738,68	0,374	1	optimal	3409
7	4	5000	400	model 2	29	750,48	0,149	1	optimal	2557
8	0	1000	224	model 2	29	472,12	0,174	1	optimal	2522
8	1	2000	600	model 2	28	2291,76	0,249	1	optimal	2458
8	2	3000	840	model 2	28	4706,64	0,432	1	optimal	2453
8	3	4000	542,4	model 2	28	1945,68	0,175	1	optimal	2462
8	4	5000	1274	model 2	28	9272,32	2,353	1	optimal	2482
9	0	1000	135,6	model 2	28	271,2	0,109	1	optimal	2451
9	1	2000	912	model 2	28	4791,2	0,564	1	optimal	2503
9	2	3000	3000	model 2	28	39695,6	53,736	1	optimal	2497
9	3	4000	1307,2	model 2	28	9697,8	1,664	1	optimal	2483
9	4	5000	100	model 2	28	300	0,132	1	optimal	2468
10	0	1000	182,8	model 2	28	365,6	0,118	1	optimal	2520
10	1	2000	624,8	model 2	28	2465,36	0,273	1	optimal	2468
10	2	3000	398,4	model 2	28	1157,36	0,125	1	optimal	2462
10	3	4000	542,4	model 2	28	1945,68	0,159	1	optimal	2451
10	4	5000	1634	model 2	28	13843,52	5,919	1	optimal	2483
11	0	1000	868	model 2	28	3527,2	0,305	1	optimal	2457
11	1	2000	989,6	model 2	28	7829,36	0,428	1	optimal	2462
11	2	3000	420	model 2	28	1264,08	0,156	1	optimal	2472
11	3	4000	1627,2	model 2	28	13741,52	6,758	1	optimal	2509
11	4	5000	1480	model 2	28	12284,28	1,563	1	optimal	2473
12	0	1000	140	model 2	28	280	0,115	1	optimal	2480
12	1	2000	160	model 2	28	480	0,114	1	optimal	2461
12	2	3000	312	model 2	28	811,76	0,127	1	optimal	2458
12	3	4000	531,2	model 2	28	1878,48	0,129	1	optimal	2462
12	4	5000	700	model 2	28	2991,76	0,192	1	optimal	2472
13	0	1000	494,8	model 2	28	2494,88	0,174	1	optimal	2462
13	1	2000	641,6	model 2	28	3872,68	0,24	1	optimal	2513
13	2	3000	840	model 2	28	4706,64	0,381	1	optimal	2453
13	3	4000	731,2	model 2	28	3232,16	0,419	1	optimal	2457
13	4	5000	1500	model 2	28	11874,56	3,81	1	optimal	2458
14	0	1000	296	model 2	28	760,12	0,117	1	optimal	2473
14	1	2000	499,2	model 2	30	2534,48	0,211	1	optimal	2801
14	2	3000	318	model 2	28	1247,72	0,139	1	optimal	2531
14	3	4000	1068,8	model 2	28	6692,6	0,86	1	optimal	2483
14	4	5000	914	model 2	30	5036,32	0,303	1	optimal	2775
15	0	1000	249,6	model 2	42	299,36	0,234	1	optimal	5928
15	1	2000	560	model 2	43	3832,84	0,684	1	optimal	6091
15	2	3000	3000	model 2	42	No objective value	600,203	1	maxTimeLimit	6075
15	3	4000	531,2	model 2	44	3948,8	1,662	1	optimal	6335
15	4	5000	1562	model 2	33	13602,04	2,716	1	optimal	3310
16	0	1000	182,8	model 2	44	558,08	0,404	1	optimal	6266
16	1	2000	989,6	model 2	44	6637,28	16,761	1	optimal	6274
16	2	3000	60	model 2	44	60	0,263	1	optimal	6308
16	3	4000	1019,2	model 2	44	11215,6	21,891	1	optimal	6248
16	4	5000	1274	model 2	43	17032	159,286	1	optimal	6225
17	0	1000	374	model 2	44	1869,2	0,75	1	optimal	6290
17	1	2000	641,6	model 2	44	6319,2	1,992	1	optimal	6249
17	2	3000	1220,4	model 2	44	14830,32	213,217	1	optimal	6257
17	3	4000	16	model 2	44	16	0,256	1	optimal	6275
17	4	5000	100	model 2	44	120,16	0,291	1	optimal	6308
18	0	1000	456	model 2	44	2664	1,08	1	optimal	6327
18	1	2000	560	model 2	44	4337,92	1,566	1	optimal	6274
18	2	3000	398,4	model 2	44	2280,32	0,765	1	optimal	6335
18	3	4000	80	model 2	44	80,16	0,274	1	optimal	6308
18	4	5000	1870	model 2	44	42209,32	600,296	1	maxTimeLimit	6290
19	0	1000	406,8	model 2	30	2172	0,707	1	optimal	2773
19	1	2000	624,8	model 2	42	4501,28	7,437	1	optimal	6196
19	2	3000	60	model 2	37	180	0,305	1	optimal	4937
19	3	4000	1627,2	model 2	44	24900,16	493,695	1	optimal	6257
19	4	5000	1870	model 2	38	36173,88	534,527	1	optimal	5037
20	0	1000	267,2	model 2	26	1042,16	0,16	1	optimal	1901
20	1	2000	592	model 2	19	1644,12	0,107	1	optimal	1148
20	2	3000	937,2	model 2	33	9216,56	11,273	1	optimal	3498
20	3	4000	3472	model 2	23	32905,88	4,728	1	optimal	1318
20	4	5000	940	model 2	28	8960,8	5,591	1	optimal	2485
21	0	1000	374	model 2	19	772,16	0,096	1	optimal	1130
21	1	2000	208	model 2	19	416	0,087	1	optimal	1140
21	2	3000	1484,4	model 2	19	9221,44	0,249	1	optimal	1144
21	3	4000	1307,2	model 2	19	6181,88	0,238	1	optimal	1148
21	4	5000	1562	model 2	20	7454,4	0,208	1	optimal	1166
22	0	1000	374	model 2	19	772,16	0,097	1	optimal	1130
22	1	2000	534,4	model 2	20	1401,36	0,081	1	optimal	1131
22	2	3000	764,4	model 2	19	2437,28	0,155	1	optimal	1144
22	3	4000	1824	model 2	19	10952,4	1,24	1	optimal	1127
22	4	5000	100	model 2	19	300	0,091	1	optimal	990
23	0	1000	280	model 2	18	560	0,096	1	optimal	1068
23	1	2000	8	model 2	15	24	0,104	1	optimal	792
23	2	3000	548,4	model 2	18	1457,36	0,181	1	optimal	995
23	3	4000	560	model 2	18	1396,12	0,109	1	optimal	935
23	4	5000	520	model 2	15	1343,76	0,113	1	optimal	816
24	0	1000	249,6	model 2	14	748,8	0,102	1	optimal	649
24	1	2000	592	model 2	8	1524,12	0,074	1	optimal	206
24	2	3000	1484,4	model 2	6	4453,2	0,044	1	optimal	93
24	3	4000	3472	model 2	10	24988,4	0,671	1	optimal	335
24	4	5000	1112	model 2	12	6056,44	0,09	1	optimal	516
linha11 20 hours	0	1000	7299,6	model 2	519	15471233	600,471	1	maxTimeLimit	958367
linha11 20 hours	1	2000	13080,8	model 2	519	42629464	623,164	1	maxTimeLimit	959466
linha11 20 hours	2	3000	18679,2	model 2	519	39080052	642,278	1	maxTimeLimit	947195
linha11 20 hours	3	4000	24707,2	model 2	519	39756272	600,447	1	maxTimeLimit	993249
linha11 20 hours	4	5000	21600	model 2	519	39779763	633,677	1	maxTimeLimit	989721

Table 24: Line 11 model 3

Time	Index	percentage perturbation	absolute perturbation	model	# of trains	Objective value	solution time	# of routes	Termination condition	number of binary variables
4	0	1000	254,8	model 3	44	90372,68	600,165	175	maxTimeLimit	164965
4	1	2000	912	model 3	44	No objective value	602,329	175	maxTimeLimit	170552
4	2	3000	840	model 3	44	No objective value	600,098	176	maxTimeLimit	170639
4	3	4000	1019,2	model 3	42	No objective value	600,084	173	maxTimeLimit	166438
4	4	5000	1480	model 3	45	No objective value	602,672	173	maxTimeLimit	159940
5	0	1000	267,2	model 3	44	No objective value	600,102	173	maxTimeLimit	158975
5	1	2000	271,2	model 3	44	No objective value	600,219	172	maxTimeLimit	160944
5	2	3000	764,4	model 3	44	No objective value	600,092	175	maxTimeLimit	164965
5	3	4000	1979,2	model 3	44	No objective value	600,961	172	maxTimeLimit	160252
5	4	5000	678	model 3	44	No objective value	600,087	172	maxTimeLimit	160944
6	0	1000	1000	model 3	41	No objective value	600,133	163	maxTimeLimit	146476
6	1	2000	160	model 3	44	No objective value	600,858	175	maxTimeLimit	168591
6	2	3000	240	model 3	44	77994	600,497	172	maxTimeLimit	160074
6	3	4000	889,6	model 3	44	No objective value	602,301	175	maxTimeLimit	167050
6	4	5000	530	model 3	44	No objective value	600,121	171	maxTimeLimit	157217
7	0	1000	222,4	model 3	38	No objective value	601,053	147	maxTimeLimit	115516
7	1	2000	2000	model 3	29	20156,84	600,283	111	maxTimeLimit	64927
7	2	3000	748,8	model 3	32	16993,92	600,392	119	maxTimeLimit	71484
7	3	4000	560	model 3	34	3591,32	600,518	119	maxTimeLimit	71734
7	4	5000	400	model 3	29	935,12	600,464	113	maxTimeLimit	67076
8	0	1000	224	model 3	29	1410,36	600,35	108	maxTimeLimit	62220
8	1	2000	600	model 3	28	2680,48	600,366	112	maxTimeLimit	66136
8	2	3000	840	model 3	28	9646,24	600,332	112	maxTimeLimit	66110
8	3	4000	542,4	model 3	28	3011,12	600,284	110	maxTimeLimit	63830
8	4	5000	1274	model 3	28	10363,52	600,375	111	maxTimeLimit	65229
9	0	1000	135,6	model 3	28	271,2	194,031	110	optimal	63809
9	1	2000	912	model 3	28	1891,76	388,779	107	optimal	62242
9	2	3000	3000	model 3	28	10356,2	600,266	107	maxTimeLimit	62226
9	3	4000	1307,2	model 3	28	3399,24	600,424	112	maxTimeLimit	66546
9	4	5000	100	model 3	28	3414,76	600,405	112	maxTimeLimit	67234
10	0	1000	182,8	model 3	28	365,6	169,558	111	optimal	65178
10	1	2000	624,8	model 3	28	1688,24	600,449	111	maxTimeLimit	65367
10	2	3000	398,4	model 3	28	3834,6	600,357	111	maxTimeLimit	65964
10	3	4000	542,4	model 3	28	1945,68	215,268	110	optimal	63809
10	4	5000	1634	model 3	28	No objective value	600,322	112	maxTimeLimit	66533
11	0	1000	868	model 3	28	5657,52	600,288	111	maxTimeLimit	65302
11	1	2000	989,6	model 3	28	No objective value	600,458	107	maxTimeLimit	62035
11	2	3000	420	model 3	28	1264,08	235,89	111	optimal	65711
11	3	4000	1627,2	model 3	28	4386,68	600,235	111	maxTimeLimit	65032
11	4	5000	1480	model 3	28	5520,08	600,292	111	maxTimeLimit	66110
12	0	1000	140	model 3	28	2534,12	600,379	111	maxTimeLimit	65604
12	1	2000	160	model 3	28	626,72	600,413	110	maxTimeLimit	64393
12	2	3000	312	model 3	28	811,76	167,221	111	optimal	65337
12	3	4000	531,2	model 3	28	4799,56	600,377	111	maxTimeLimit	65964
12	4	5000	700	model 3	28	2363,76	173,765	111	optimal	65711
13	0	1000	494,8	model 3	28	2775,04	600,287	107	maxTimeLimit	62035
13	1	2000	641,6	model 3	28	6612,84	600,259	107	maxTimeLimit	62786
13	2	3000	840	model 3	28	No objective value	600,697	112	maxTimeLimit	66110
13	3	4000	731,2	model 3	28	272,64	298,324	112	optimal	66298
13	4	5000	1500	model 3	28	44096,04	600,249	112	maxTimeLimit	66136
14	0	1000	296	model 3	28	1448,12	600,241	111	maxTimeLimit	66110
14	1	2000	499,2	model 3	30	2534,48	600,619	114	maxTimeLimit	72202
14	2	3000	318	model 3	28	1273	600,366	107	maxTimeLimit	63438
14	3	4000	1068,8	model 3	28	-6,5E-09	172,526	112	optimal	66513
14	4	5000	914	model 3	30	3635,04	600,218	120	maxTimeLimit	75308
15	0	1000	249,6	model 3	42	125048,4	600,567	172	maxTimeLimit	159770
15	1	2000	560	model 3	43	No objective value	600,357	172	maxTimeLimit	159142
15	2	3000	3000	model 3	42	No objective value	600,324	173	maxTimeLimit	168460
15	3	4000	531,2	model 3	44	No objective value	600,484	175	maxTimeLimit	170429
15	4	5000	1562	model 3	33	No objective value	601,103	131	maxTimeLimit	94313
16	0	1000	182,8	model 3	44	No objective value	600,343	173	maxTimeLimit	158987
16	1	2000	989,6	model 3	44	No objective value	600,126	172	maxTimeLimit	160252
16	2	3000	60	model 3	44	No objective value	600,446	175	maxTimeLimit	168608
16	3	4000	1019,2	model 3	44	No objective value	600,104	175	maxTimeLimit	164965
16	4	5000	1274	model 3	43	No objective value	600,09	174	maxTimeLimit	166158
17	0	1000	374	model 3	44	No objective value	600,295	175	maxTimeLimit	168574
17	1	2000	641,6	model 3	44	No objective value	600,202	171	maxTimeLimit	156447
17	2	3000	1220,4	model 3	44	No objective value	600,373	175	maxTimeLimit	164990
17	3	4000	16	model 3	44	No objective value	601,044	175	maxTimeLimit	167067
17	4	5000	100	model 3	44	152127,1	600,833	175	maxTimeLimit	168608
18	0	1000	456	model 3	44	No objective value	600,379	172	maxTimeLimit	160992
18	1	2000	560	model 3	44	No objective value	601,515	173	maxTimeLimit	159033
18	2	3000	398,4	model 3	44	No objective value	603,497	175	maxTimeLimit	170429
18	3	4000	80	model 3	44	No objective value	600,648	175	maxTimeLimit	168608
18	4	5000	1870	model 3	44	No objective value	600,092	175	maxTimeLimit	168574
19	0	1000	406,8	model 3	30	29509,92	600,481	120	maxTimeLimit	75326
19	1	2000	624,8	model 3	42	No objective value	600,25	172	maxTimeLimit	160021
19	2	3000	60	model 3	37	No objective value	600,322	156	maxTimeLimit	137214
19	3	4000	1627,2	model 3	44	No objective value	600,095	175	maxTimeLimit	164990
19	4	5000	1870	model 3	38	No objective value	600,259	157	maxTimeLimit	139119
20	0	1000	267,2	model 3	26	1042,16	153,777	103	optimal	50683
20	1	2000	592	model 3	19	1201,32	225,159	76	optimal	29105
20	2	3000	937,2	model 3	33	6928293	600,322	135	maxTimeLimit	97462
20	3	4000	3472	model 3	23	32918,52	600,247	84	maxTimeLimit	30905
20	4	5000	940	model 3	28	No objective value	600,759	113	maxTimeLimit	68243
21	0	1000	374	model 3	19	772,16	18,512	76	optimal	29364
21	1	2000	208	model 3	19	416	9,209	77	optimal	29085
21	2	3000	1484,4	model 3	19	9221,44	600,237	76	maxTimeLimit	29097
21	3	4000	1307,2	model 3	19	2614,4	14,635	75	optimal	28451
21	4	5000	1562	model 3	20	3783,68	193,692	75	optimal	28295
22	0	1000	374	model 3	19	772,16	23,248	76	optimal	29364
22	1	2000	534,4	model 3	20	1401,36	36,17	77	optimal	28941
22	2	3000	764,4	model 3	19	2437,28	600,478	77	maxTimeLimit	29050
22	3	4000	1824	model 3	19	4007,92	182,842	76	optimal	28916
22	4	5000	100	model 3	19	300	9,986	71	optimal	23921
23	0	1000	280	model 3	18	560	17,312	71	optimal	25572
23	1	2000	8	model 3	15	24	21,238	64	optimal	22192
23	2	3000	548,4	model 3	18	-1,5E-10	20,858	71	optimal	24933
23	3	4000	560	model 3	18	1396,12	600,181	69	maxTimeLimit	23441
23	4	5000	520	model 3	15	1343,76	19,741	65	optimal	22785
24	0	1000	249,6	model 3	14	748,8	11,738	56	optimal	14123
24	1	2000	592	model 3	8	1316,12	1,168	27	optimal	2777
24	2	3000	1484,4	model 3	6	4453,2	0,098	15	optimal	425
24	3	4000	3472	model 3	10	23422,08	31,672	37	optimal	4983
24	4	5000	1112	model 3	12	6056,44	600,252	46	maxTimeLimit	10268

Table 25: Line 13 model 1

Time	Index	percentage perturbation	absolute perturbation	model	# of trains	Objective value	solution time	# of routes	Termination condition	number of binary variables
4	0	1000	154	model 1	10	154	0,015	1	optimal	0
4	1	2000	172	model 1	10	212,32	0,011	1	optimal	0
4	2	3000	462	model 1	10	462	0,013	1	optimal	0
4	3	4000	616	model 1	10	1100,32	0,018	1	optimal	0
4	4	5000	480	model 1	8	960	0,014	1	optimal	0
5	0	1000	154	model 1	10	176,32	0,015	1	optimal	0
5	1	2000	497,6	model 1	10	863,52	0,015	1	optimal	0
5	2	3000	462	model 1	10	462	0,013	1	optimal	0
5	3	4000	622,4	model 1	10	1113,12	0,014	1	optimal	0
5	4	5000	770	model 1	10	1408,32	0,012	1	optimal	0
6	0	1000	248,8	model 1	10	248,8	0,012	1	optimal	0
6	1	2000	308	model 1	10	308	0,012	1	optimal	0
6	2	3000	428,4	model 1	10	856,8	0,025	1	optimal	0
6	3	4000	622,4	model 1	10	1113,12	0,015	1	optimal	0
6	4	5000	770	model 1	10	770	0,013	1	optimal	0
7	0	1000	154	model 1	10	176,32	0,013	1	optimal	0
7	1	2000	198,4	model 1	10	265,12	0,011	1	optimal	0
7	2	3000	462	model 1	10	462	0,017	1	optimal	0
7	3	4000	995,2	model 1	10	1163,68	0,013	1	optimal	0
7	4	5000	758	model 1	10	758	0,013	1	optimal	0
8	0	1000	154	model 1	10	176,32	0,013	1	optimal	0
8	1	2000	308	model 1	10	484,32	0,012	1	optimal	0
8	2	3000	2078,4	model 1	10	5448,48	0,014	1	optimal	0
8	3	4000	606,4	model 1	10	606,4	0,018	1	optimal	0
8	4	5000	496	model 1	10	860,32	0,011	1	optimal	0
9	0	1000	65,2	model 1	10	65,2	0,014	1	optimal	0
9	1	2000	285,6	model 1	9	153,92	0,014	1	optimal	0
9	2	3000	258	model 1	10	384,32	0,026	1	optimal	0
9	3	4000	616	model 1	10	616	0,017	1	optimal	0
9	4	5000	770	model 1	10	770	0,014	1	optimal	0
10	0	1000	99,2	model 1	10	99,2	0,014	1	optimal	0
10	1	2000	130,4	model 1	10	130,4	0,014	1	optimal	0
10	2	3000	457,2	model 1	10	782,72	0,013	1	optimal	0
10	3	4000	616	model 1	10	616	0,011	1	optimal	0
10	4	5000	700	model 1	10	568,32	0,014	1	optimal	0
11	0	1000	154	model 1	10	154	0,022	1	optimal	0
11	1	2000	303,2	model 1	10	474,72	0,01	1	optimal	0
11	2	3000	462	model 1	10	462	0,013	1	optimal	0
11	3	4000	344	model 1	10	556,32	0,013	1	optimal	0
11	4	5000	326	model 1	10	520,32	0,015	1	optimal	0
12	0	1000	154	model 1	10	154	0,015	1	optimal	0
12	1	2000	130,4	model 1	10	130,4	0,014	1	optimal	0
12	2	3000	462	model 1	10	792,32	0,022	1	optimal	0
12	3	4000	616	model 1	10	1100,32	0,012	1	optimal	0
12	4	5000	480	model 1	10	960	0,012	1	optimal	0
13	0	1000	152,4	model 1	10	173,12	0,015	1	optimal	0
13	1	2000	198,4	model 1	10	265,12	0,028	1	optimal	0
13	2	3000	428,4	model 1	9	296,72	0,013	1	optimal	0
13	3	4000	616	model 1	10	1100,32	0,012	1	optimal	0
13	4	5000	770	model 1	10	770	0,015	1	optimal	0
14	0	1000	154	model 1	10	176,32	0,012	1	optimal	0
14	1	2000	308	model 1	10	308	0,011	1	optimal	0
14	2	3000	428,4	model 1	10	856,8	0,015	1	optimal	0
14	3	4000	616	model 1	10	616	0,014	1	optimal	0
14	4	5000	326	model 1	10	520,32	0,013	1	optimal	0
15	0	1000	154	model 1	10	154	0,014	1	optimal	0
15	1	2000	308	model 1	10	484,32	0,013	1	optimal	0
15	2	3000	462	model 1	10	462	0,013	1	optimal	0
15	3	4000	344	model 1	10	556,32	0,014	1	optimal	0
15	4	5000	770	model 1	10	770	0,013	1	optimal	0
16	0	1000	248,8	model 1	10	248,8	0,014	1	optimal	0
16	1	2000	192	model 1	10	384	0,011	1	optimal	0
16	2	3000	420	model 1	10	840	0,013	1	optimal	0
16	3	4000	616	model 1	10	1100,32	0,014	1	optimal	0
16	4	5000	700	model 1	10	568,32	0,02	1	optimal	0
17	0	1000	248,8	model 1	10	365,92	0,015	1	optimal	0
17	1	2000	172	model 1	10	212,32	0,013	1	optimal	0
17	2	3000	462	model 1	10	792,32	0,012	1	optimal	0
17	3	4000	2771,2	model 1	10	2639,52	0,013	1	optimal	0
17	4	5000	770	model 1	10	770	0,022	1	optimal	0
18	0	1000	65,2	model 1	10	65,2	0,015	1	optimal	0
18	1	2000	192	model 1	9	192	0,014	1	optimal	0
18	2	3000	428,4	model 1	9	296,72	0,016	1	optimal	0
18	3	4000	396,8	model 1	10	396,8	0,013	1	optimal	0
18	4	5000	770	model 1	10	770	0,018	1	optimal	0
19	0	1000	692,8	model 1	10	561,12	0,017	1	optimal	0
19	1	2000	198,4	model 1	10	198,4	0,014	1	optimal	0
19	2	3000	454,8	model 1	10	454,8	0,013	1	optimal	0
19	3	4000	616	model 1	10	1100,32	0,011	1	optimal	0
19	4	5000	770	model 1	10	1408,32	0,016	1	optimal	0
20	0	1000	154	model 1	10	154	0,016	1	optimal	0
20	1	2000	280	model 1	10	148,32	0,014	1	optimal	0
20	2	3000	466,8	model 1	10	801,92	0,017	1	optimal	0
20	3	4000	616	model 1	10	1100,32	0,013	1	optimal	0
20	4	5000	770	model 1	10	770	0,016	1	optimal	0
21	0	1000	154	model 1	10	176,32	0,02	1	optimal	0
21	1	2000	308	model 1	10	484,32	0,013	1	optimal	0
21	2	3000	462	model 1	10	462	0,012	1	optimal	0
21	3	4000	616	model 1	10	1100,32	0,018	1	optimal	0
21	4	5000	770	model 1	10	770	0,016	1	optimal	0
22	0	1000	154	model 1	10	176,32	0,013	1	optimal	0
22	1	2000	192	model 1	9	192	0,013	1	optimal	0
22	2	3000	462	model 1	10	462	0,013	1	optimal	0
22	3	4000	616	model 1	10	1100,32	0,014	1	optimal	0
22	4	5000	770	model 1	10	770	0,013	1	optimal	0
23	0	1000	155,6	model 1	10	179,52	0,017	1	optimal	0
23	1	2000	285,6	model 1	5	153,92	0,015	1	optimal	0
23	2	3000	466,8	model 1	10	466,8	0,013	1	optimal	0
23	3	4000	616	model 1	8	616	0,013	1	optimal	0
23	4	5000	770	model 1	10	770	0,017	1	optimal	0
24	0	1000	154	model 1	4	154	0,013	1	optimal	0
24	1	2000	285,6	model 1	5	153,92	0,02	1	optimal	0
24	2	3000	462	model 1	2	462	0,013	1	optimal	0
24	3	4000	2771,2	model 1	2	2639,52	0,013	1	optimal	0
24	4	5000	770	model 1	6	770	0,021	1	optimal	0
linha13 20 hours	0	1000	3824,8	model 1	164	3988,8	0,127	1	optimal	0
linha13 20 hours	1	2000	5253,6	model 1	164	6045,44	0,159	1	optimal	0
linha13 20 hours	2	3000	10935,6	model 1	164	15926,72	0,142	1	optimal	0
linha13 20 hours	3	4000	16865,6	model 1	164	20951,04	0,141	1	optimal	0
linha13 20 hours	4	5000	14276	model 1	164	16433,92	0,159	1	optimal	0

Table 26: Line 13 model 2

Time	Index	percentage perturbation	absolute perturbation	model	# of trains	Objective value	solution time	# of routes	Termination condition	number of binary variables
4	0	1000	154	model 2	10	154	0,043	1	optimal	160
4	1	2000	172	model 2	10	212,32	0,03	1	optimal	160
4	2	3000	462	model 2	10	462	0,041	1	optimal	164
4	3	4000	616	model 2	10	1100,32	0,033	1	optimal	160
4	4	5000	480	model 2	8	960	0,045	1	optimal	156
5	0	1000	154	model 2	10	176,32	0,036	1	optimal	160
5	1	2000	497,6	model 2	10	863,52	0,03	1	optimal	160
5	2	3000	462	model 2	10	462	0,026	1	optimal	164
5	3	4000	622,4	model 2	10	1113,12	0,024	1	optimal	160
5	4	5000	770	model 2	10	1408,32	0,032	1	optimal	160
6	0	1000	248,8	model 2	10	248,8	0,024	1	optimal	164
6	1	2000	308	model 2	10	308	0,031	1	optimal	164
6	2	3000	428,4	model 2	10	856,8	0,035	1	optimal	164
6	3	4000	622,4	model 2	10	1113,12	0,037	1	optimal	160
6	4	5000	770	model 2	10	770	0,034	1	optimal	164
7	0	1000	154	model 2	10	176,32	0,025	1	optimal	160
7	1	2000	198,4	model 2	10	265,12	0,027	1	optimal	160
7	2	3000	462	model 2	10	462	0,026	1	optimal	164
7	3	4000	995,2	model 2	10	1163,68	0,03	1	optimal	164
7	4	5000	758	model 2	10	758	0,025	1	optimal	164
8	0	1000	154	model 2	10	176,32	0,029	1	optimal	160
8	1	2000	308	model 2	10	484,32	0,025	1	optimal	160
8	2	3000	2078,4	model 2	10	5448,48	0,039	1	optimal	164
8	3	4000	606,4	model 2	10	606,4	0,025	1	optimal	164
8	4	5000	496	model 2	10	860,32	0,024	1	optimal	160
9	0	1000	65,2	model 2	10	65,2	0,023	1	optimal	160
9	1	2000	285,6	model 2	9	153,92	0,033	1	optimal	164
9	2	3000	258	model 2	10	384,32	0,028	1	optimal	160
9	3	4000	616	model 2	10	616	0,028	1	optimal	164
9	4	5000	770	model 2	10	770	0,025	1	optimal	164
10	0	1000	99,2	model 2	10	99,2	0,029	1	optimal	160
10	1	2000	130,4	model 2	10	130,4	0,029	1	optimal	164
10	2	3000	457,2	model 2	10	782,72	0,037	1	optimal	160
10	3	4000	616	model 2	10	616	0,028	1	optimal	164
10	4	5000	700	model 2	10	568,32	0,025	1	optimal	160
11	0	1000	154	model 2	10	154	0,029	1	optimal	164
11	1	2000	303,2	model 2	10	474,72	0,026	1	optimal	160
11	2	3000	462	model 2	10	462	0,03	1	optimal	164
11	3	4000	344	model 2	10	556,32	0,026	1	optimal	160
11	4	5000	326	model 2	10	520,32	0,027	1	optimal	160
12	0	1000	154	model 2	10	154	0,039	1	optimal	164
12	1	2000	130,4	model 2	10	130,4	0,024	1	optimal	160
12	2	3000	462	model 2	10	792,32	0,024	1	optimal	160
12	3	4000	616	model 2	10	1100,32	0,027	1	optimal	160
12	4	5000	480	model 2	10	960	0,029	1	optimal	164
13	0	1000	152,4	model 2	10	173,12	0,028	1	optimal	160
13	1	2000	198,4	model 2	10	265,12	0,028	1	optimal	160
13	2	3000	428,4	model 2	9	296,72	0,035	1	optimal	164
13	3	4000	616	model 2	10	1100,32	0,026	1	optimal	160
13	4	5000	770	model 2	10	770	0,031	1	optimal	164
14	0	1000	154	model 2	10	176,32	0,027	1	optimal	160
14	1	2000	308	model 2	10	308	0,041	1	optimal	164
14	2	3000	428,4	model 2	10	856,8	0,026	1	optimal	164
14	3	4000	616	model 2	10	616	0,025	1	optimal	164
14	4	5000	326	model 2	10	520,32	0,029	1	optimal	160
15	0	1000	154	model 2	10	154	0,033	1	optimal	164
15	1	2000	308	model 2	10	484,32	0,028	1	optimal	160
15	2	3000	462	model 2	10	462	0,024	1	optimal	164
15	3	4000	344	model 2	10	556,32	0,023	1	optimal	160
15	4	5000	770	model 2	10	770	0,023	1	optimal	164
16	0	1000	248,8	model 2	10	248,8	0,03	1	optimal	164
16	1	2000	192	model 2	10	384	0,024	1	optimal	164
16	2	3000	420	model 2	10	840	0,023	1	optimal	164
16	3	4000	616	model 2	10	1100,32	0,036	1	optimal	160
16	4	5000	700	model 2	10	568,32	0,023	1	optimal	160
17	0	1000	248,8	model 2	10	365,92	0,025	1	optimal	160
17	1	2000	172	model 2	10	212,32	0,025	1	optimal	160
17	2	3000	462	model 2	10	792,32	0,024	1	optimal	160
17	3	4000	2771,2	model 2	10	2639,52	0,036	1	optimal	160
17	4	5000	770	model 2	10	770	0,025	1	optimal	164
18	0	1000	65,2	model 2	10	65,2	0,024	1	optimal	160
18	1	2000	192	model 2	9	192	0,052	1	optimal	164
18	2	3000	428,4	model 2	9	296,72	0,028	1	optimal	164
18	3	4000	396,8	model 2	10	396,8	0,027	1	optimal	164
18	4	5000	770	model 2	10	770	0,026	1	optimal	164
19	0	1000	692,8	model 2	10	561,12	0,023	1	optimal	160
19	1	2000	198,4	model 2	10	198,4	0,041	1	optimal	164
19	2	3000	454,8	model 2	10	454,8	0,038	1	optimal	164
19	3	4000	616	model 2	10	1100,32	0,028	1	optimal	160
19	4	5000	770	model 2	10	1408,32	0,024	1	optimal	160
20	0	1000	154	model 2	10	154	0,023	1	optimal	164
20	1	2000	280	model 2	10	148,32	0,025	1	optimal	160
20	2	3000	466,8	model 2	10	801,92	0,027	1	optimal	160
20	3	4000	616	model 2	10	1100,32	0,024	1	optimal	160
20	4	5000	770	model 2	10	770	0,037	1	optimal	164
21	0	1000	154	model 2	10	176,32	0,024	1	optimal	160
21	1	2000	308	model 2	10	484,32	0,025	1	optimal	160
21	2	3000	462	model 2	10	462	0,027	1	optimal	164
21	3	4000	616	model 2	10	1100,32	0,026	1	optimal	160
21	4	5000	770	model 2	10	770	0,025	1	optimal	164
22	0	1000	154	model 2	10	176,32	0,026	1	optimal	160
22	1	2000	192	model 2	9	192	0,021	1	optimal	164
22	2	3000	462	model 2	10	462	0,036	1	optimal	164
22	3	4000	616	model 2	10	1100,32	0,024	1	optimal	160
22	4	5000	770	model 2	10	770	0,03	1	optimal	164
23	0	1000	155,6	model 2	10	179,52	0,03	1	optimal	160
23	1	2000	285,6	model 2	5	153,92	0,022	1	optimal	34
23	2	3000	466,8	model 2	10	466,8	0,025	1	optimal	164
23	3	4000	616	model 2	8	616	0,024	1	optimal	93
23	4	5000	770	model 2	10	770	0,025	1	optimal	164
24	0	1000	154	model 2	4	154	0,049	1	optimal	17
24	1	2000	285,6	model 2	5	153,92	0,025	1	optimal	34
24	2	3000	462	model 2	2	462	0,02	1	optimal	0
24	3	4000	2771,2	model 2	2	2639,52	0,017	1	optimal	0
24	4	5000	770	model 2	6	770	0,034	1	optimal	60
linha13 20 hours	0	1000	3824,8	model 2	164	3988,8	6,996	1	optimal	61633
linha13 20 hours	1	2000	5253,6	model 2	164	6045,44	7,821	1	optimal	84834
linha13 20 hours	2	3000	10935,6	model 2	164	15926,72	17,868	1	optimal	82787
linha13 20 hours	3	4000	16865,6	model 2	164	20951,04	19,764	1	optimal	63661
linha13 20 hours	4	5000	14276	model 2	164	16433,92	6,38	1	optimal	63179

Table 27: Line 13 model 3

Time	Index	percentage perturbation	absolute perturbation	model	# of trains	Objective value	solution time	# of routes	Termination condition	number of binary variables
4	0	1000	154	model 3	10	154	0,033	14	optimal	190
4	1	2000	172	model 3	10	212,32	0,055	14	optimal	190
4	2	3000	462	model 3	10	462	0,034	14	optimal	194
4	3	4000	616	model 3	10	1100,32	0,059	14	optimal	190
4	4	5000	480	model 3	8	960	0,03	12	optimal	184
5	0	1000	154	model 3	10	176,32	0,05	14	optimal	190
5	1	2000	497,6	model 3	10	863,52	0,037	14	optimal	190
5	2	3000	462	model 3	10	462	0,034	14	optimal	194
5	3	4000	622,4	model 3	10	1113,12	0,042	14	optimal	190
5	4	5000	770	model 3	10	1408,32	0,053	14	optimal	190
6	0	1000	248,8	model 3	10	248,8	0,037	14	optimal	194
6	1	2000	308	model 3	10	308	0,056	14	optimal	194
6	2	3000	428,4	model 3	10	856,8	0,036	14	optimal	198
6	3	4000	622,4	model 3	10	1113,12	0,029	14	optimal	190
6	4	5000	770	model 3	10	770	0,045	14	optimal	194
7	0	1000	154	model 3	10	176,32	0,052	14	optimal	190
7	1	2000	198,4	model 3	10	265,12	0,046	14	optimal	190
7	2	3000	462	model 3	10	462	0,047	14	optimal	194
7	3	4000	995,2	model 3	10	1163,68	0,037	14	optimal	194
7	4	5000	758	model 3	10	758	0,04	14	optimal	194
8	0	1000	154	model 3	10	176,32	0,049	14	optimal	190
8	1	2000	308	model 3	10	484,32	0,031	14	optimal	190
8	2	3000	2078,4	model 3	10	5448,48	0,065	14	optimal	194
8	3	4000	606,4	model 3	10	606,4	0,04	14	optimal	194
8	4	5000	496	model 3	10	860,32	0,065	14	optimal	190
9	0	1000	65,2	model 3	10	65,2	0,032	14	optimal	190
9	1	2000	285,6	model 3	9	153,92	0,032	13	optimal	525
9	2	3000	258	model 3	10	384,32	0,046	14	optimal	190
9	3	4000	616	model 3	10	616	0,035	14	optimal	194
9	4	5000	770	model 3	10	770	0,041	14	optimal	194
10	0	1000	99,2	model 3	10	99,2	0,043	14	optimal	190
10	1	2000	130,4	model 3	10	130,4	0,039	14	optimal	194
10	2	3000	457,2	model 3	10	782,72	0,045	14	optimal	190
10	3	4000	616	model 3	10	616	0,035	14	optimal	194
10	4	5000	700	model 3	10	568,32	0,065	14	optimal	190
11	0	1000	154	model 3	10	154	0,035	14	optimal	194
11	1	2000	303,2	model 3	10	474,72	0,066	14	optimal	190
11	2	3000	462	model 3	10	462	0,044	14	optimal	194
11	3	4000	344	model 3	10	556,32	0,078	14	optimal	190
11	4	5000	326	model 3	10	520,32	0,036	14	optimal	190
12	0	1000	154	model 3	10	154	0,036	14	optimal	194
12	1	2000	130,4	model 3	10	130,4	0,047	14	optimal	190
12	2	3000	462	model 3	10	792,32	0,032	14	optimal	190
12	3	4000	616	model 3	10	1100,32	0,048	14	optimal	190
12	4	5000	480	model 3	10	960	0,048	14	optimal	198
13	0	1000	152,4	model 3	10	173,12	0,046	14	optimal	190
13	1	2000	198,4	model 3	10	265,12	0,047	14	optimal	190
13	2	3000	428,4	model 3	9	296,72	0,035	13	optimal	525
13	3	4000	616	model 3	10	1100,32	0,054	14	optimal	190
13	4	5000	770	model 3	10	770	0,039	14	optimal	194
14	0	1000	154	model 3	10	176,32	0,048	14	optimal	190
14	1	2000	308	model 3	10	308	0,034	14	optimal	194
14	2	3000	428,4	model 3	10	856,8	0,046	14	optimal	198
14	3	4000	616	model 3	10	616	0,033	14	optimal	194
14	4	5000	326	model 3	10	520,32	0,032	14	optimal	190
15	0	1000	154	model 3	10	154	0,038	14	optimal	194
15	1	2000	308	model 3	10	484,32	0,056	14	optimal	190
15	2	3000	462	model 3	10	462	0,034	14	optimal	194
15	3	4000	344	model 3	10	556,32	0,051	14	optimal	190
15	4	5000	770	model 3	10	770	0,051	14	optimal	194
16	0	1000	248,8	model 3	10	248,8	0,037	14	optimal	194
16	1	2000	192	model 3	10	384	0,038	14	optimal	198
16	2	3000	420	model 3	10	840	0,035	14	optimal	194
16	3	4000	616	model 3	10	1100,32	0,056	14	optimal	190
16	4	5000	700	model 3	10	568,32	0,05	14	optimal	190
17	0	1000	248,8	model 3	10	365,92	0,034	14	optimal	190
17	1	2000	172	model 3	10	212,32	0,053	14	optimal	190
17	2	3000	462	model 3	10	792,32	0,036	14	optimal	190
17	3	4000	2771,2	model 3	10	2639,52	0,044	14	optimal	190
17	4	5000	770	model 3	10	770	0,035	14	optimal	194
18	0	1000	65,2	model 3	10	65,2	0,036	14	optimal	190
18	1	2000	192	model 3	9	-4,3E-09	0,037	13	optimal	515
18	2	3000	428,4	model 3	9	296,72	0,04	13	optimal	525
18	3	4000	396,8	model 3	10	396,8	0,033	14	optimal	194
18	4	5000	770	model 3	10	770	0,042	14	optimal	194
19	0	1000	692,8	model 3	10	561,12	0,056	14	optimal	190
19	1	2000	198,4	model 3	10	198,4	0,044	14	optimal	194
19	2	3000	454,8	model 3	10	454,8	0,042	14	optimal	194
19	3	4000	616	model 3	10	1100,32	0,049	14	optimal	190
19	4	5000	770	model 3	10	1408,32	0,043	14	optimal	190
20	0	1000	154	model 3	10	154	0,07	14	optimal	194
20	1	2000	280	model 3	10	148,32	0,044	14	optimal	190
20	2	3000	466,8	model 3	10	801,92	0,033	14	optimal	190
20	3	4000	616	model 3	10	1100,32	0,049	14	optimal	190
20	4	5000	770	model 3	10	770	0,037	14	optimal	194
21	0	1000	154	model 3	10	176,32	0,076	14	optimal	190
21	1	2000	308	model 3	10	484,32	0,031	14	optimal	190
21	2	3000	462	model 3	10	462	0,043	14	optimal	194
21	3	4000	616	model 3	10	1100,32	0,042	14	optimal	190
21	4	5000	770	model 3	10	770	0,035	14	optimal	194
22	0	1000	154	model 3	10	176,32	0,045	14	optimal	190
22	1	2000	192	model 3	9	-4,3E-09	0,042	13	optimal	515
22	2	3000	462	model 3	10	462	0,04	14	optimal	194
22	3	4000	616	model 3	10	1100,32	0,033	14	optimal	190
22	4	5000	770	model 3	10	770	0,032	14	optimal	194
23	0	1000	155,6	model 3	10	179,52	0,035	14	optimal	190
23	1	2000	285,6	model 3	5	153,92	0,034	8	optimal	144
23	2	3000	466,8	model 3	10	466,8	0,034	14	optimal	194
23	3	4000	616	model 3	8	616	0,042	12	optimal	117
23	4	5000	770	model 3	10	770	0,041	14	optimal	194
24	0	1000	154	model 3	4	154	0,037	6	optimal	25
24	1	2000	285,6	model 3	5	153,92	0,021	8	optimal	144
24	2	3000	462	model 3	2	462	0,017	3	optimal	3
24	3	4000	2771,2	model 3	2	2639,52	0,036	3	optimal	3
24	4	5000	770	model 3	6	770	0,036	9	optimal	75
linha13 20 hours	0	1000	3824,8	model 3	164	3988,8	8,455	238	optimal	66459
linha13 20 hours	1	2000	5253,6	model 3	164	5661,44	355,355	239	optimal	139600
linha13 20 hours	2	3000	10935,6	model 3	164	No objective value	600,204	236	maxTimeLimit	152268
linha13 20 hours	3	4000	16865,6	model 3	164	20951,04	11,287	240	optimal	68613
linha13 20 hours	4	5000	14276	model 3	164	16433,92	9,86	241	optimal	68348