Robustness for a Single Railway Line: Analytical and Simulation Methods

Miguel A. Salido, Federico Barber, Laura Ingolotti

Universidad Politecnica de Valencia

Valencia, Spain

Abstract

Railway scheduling has been a significant issue in the railway industry. Over the last few years, numerous approaches and tools have been developed to compute railway scheduling. However, robust solutions are necessary to absorb short disruptions. In this paper, we present the robustness problem from the point of view of railway operators and we propose analytical and simulation methods to measure robustness in a single railway line. In the analytical approach, we have developed some formulas to measure robustness based on the study of railway line infrastructure topology and buffer times. In the simulation approach, we have developed a software tool to assess the robustness for a given schedule. These methods have been inserted in MOM[1], which is a project in collaboration with the Spanish Railway Infrastructure Manager (ADIF).

Keywords:
Robustness, Railway Timetabling, analytical measures, simulation tool

1. Introduction

It is well known that solving a scheduling problem is a difficult and time-consuming task. The main goal of researchers is to develop algorithms that satisfy the following properties: correctness, optimality and efficiency. However, in many real problems, the quality of the solution obtained (robustness) is very valuable . Robustness is best understood not as a property of a problem solver, but rather of the solutions that it produces. It is not that an algorithm is robust, but that an algorithm produces robust solutions.

In many decision processes, it is necessary to offer solutions with a certain level of robustness in order to maintain their feasibility in frameworks with incomplete or imprecise data. These data can be:

---

[1]More information can be found at the MOM web page http://www.dsic.upv.es/users/ia/gps/MOM

- Actual data or knowledge about the problem, which cannot be known with the desired detail or level of certainty at the moment of making a decision.

- Future data, in a dynamic problem, since the evolution of the problem is not exactly known in advance.

The term "robust" refers to the ability to resist "imprecision".

The idea of robust schedules consists of solutions that can tolerate a certain degree of uncertainty during execution. In other words, they should be able to absorb dynamic variations in the problem due to both external reasons (exogenous events) and internal reasons (false definitions in the problem) [32].

In the context of railway scheduling, traffic operators not only need to optimize the use of railway infrastructures [36] but also maintain robustness in railway timetabling. Thanks to developments in computer science and advances in the fields of optimization and intelligent resource management, railway managers can optimize the use of available infrastructures and study important railway features such as robust timetables and railway capacity.

Our aim is to obtain a measure of robustness in railway timetabling. A timetable can have characteristics that can mean that delayed trains lead to considerable knock-on effects, whereas another configuration of the timetable may be able to absorb such effects more readily. Ensuring that the service can recover quickly when one train is delayed not only makes traveling more reliable, but it also means that train operators can, in some cases, avoid substantial penalties. To this end, we have identified the main parameters that are directly related to robustness. Some of them have been identified by railway operators based on their experience. These parameters have been considered in our analytical methods to give us a measure of robustness. Furthermore, these methods have been compared with a simulation method to verify the measurements obtained. Figure 1 shows our study of robustness in railway timetabling. Given a real timetable, we obtain analytical measures of robustness. These measures give us a model of robustness that we can use to identify new criteria in optimization processes. Furthermore, we insert random delays in the real timetable in order to replan and study a simulation measure of robustness. Both analytical and simulation measures will be checked in order to verify the proposed analytical methods. This study gives us the degree of robustness of the real timetable.

In this paper, we present two different ways of measuring robustness in railway timetabling: analytical measures, and simulation measures. In section 2, we present some definitions of robustness in general scheduling; section 3 shows some works related to robustness in railway scheduling. In section 4, we present some factors that are directly related to robustness in railway scheduling. To measure robustness, we briefly present in section 5 some methods such as stochastic programming, analytical methods and simulation methods. Section 6 presents analytical approaches to measure robustness in four different environments: with homogeneous and heterogeneous trains, in which overtaking is or is not allowed. In section 7, we present our simulation tool for measuring robustness. Section 8 shows the evaluation obtained from both, the analytical and
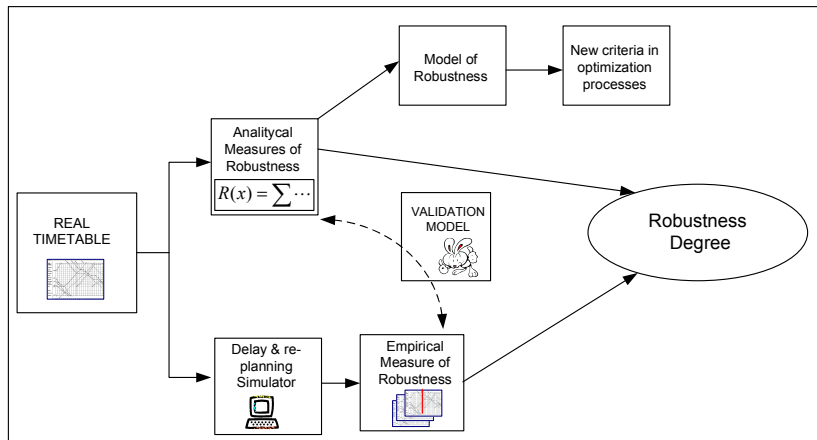
2

Figure 1: Degree of robustness in railway timetabling by checking analytical and simulation methods.

the simulation approaches. Finally in section 9, some conclusions and further works are presented.

2. Robustness in Scheduling

Research efforts aimed at generating solutions and quality robust schedules in combination with an effective reactive scheduling mechanism are still in a burn-in phase.

Even though different approaches have been pursued so far, the concept of robustness for scheduling solutions, as well as in other areas, remains vague and is not well defined. In fact, stability and robustness are two different concepts [22]. It is possible to note different definitions of robustness in scheduling with respect to the different aspects that are taken into account. Some of these definitions have emphasized the ability to preserve some level of solution quality. For instance, in [26], the robustness of a solution is defined with respect to its ability to preserve the solution quality, that is, the completion time or makespan.

An intuitive approach to obtain robust schedules consists of adding redundancy to the solution. A first example is represented by [26] where a genetic algorithm for producing robust schedules in the case of the job shop problem is described. The authors define an evaluation function, which is used in the algorithm to synthesize robust solutions, according to the actual makespan of the schedule during the execution and the schedule delay. A typical technique from general scheduling theory consists of introducing time slacks in the execution of the tasks [10]. Another alternative is to build an explicit set of complementary solutions and at each moment to use the most suitable solution according to the current state [11]. The method proposed in [35] and [34] consists of the creation of a partial order schedule which provides a certain degree of robustness due to the fact that it maintains temporal flexibility.

3

Nicola in [32] aims to increase robustness by introducing flexibility in the scheduling generation phase. Flexible solutions consist of a set of possible schedules that can be followed during the execution and that at the same time guarantee the easy and fast recovery of the current situation if necessary. In fact, they are concerned with the generation of schedules that offer some degree of robustness in the face of a dynamic and uncertain execution environment.

In [37], the authors provide a distinction of robustness according to the following two characteristics: quality robustness and solution robustness. The first is a property of a solution whose quality does not change much when small changes in the problem occur. The latter occurs when, in the same situation (small changes), a solution does not deviate much. Thus, the authors conclude that the two robustness concepts can be viewed in two different spaces: the objective function space and solution space, respectively, for quality and solution robustness.

3. Robustness in Railway Timetabling

In the last few years many researchers have focused their attention in railway timetabling [5],[19],[30] and in the robustness for railway timetables [9], [27], [12].

As we pointed out above, there are some definitions of robustness in scheduling and also robustness in railway timetabling, which are understood in many different ways. Robustness (in [24]) means that conflicting train routes are spread in time as much as possible. Norio in [38] gives another definition of robustness: "A timetable is robust if we can cope with unexpected troubles without significant modifications". Furthermore, this author considers several robustness indexes for each level (see Figure 2). In most cases, the main disruptions occur in level 0 due to small disruptions. Higher levels do not occur frequently, and they require railway re-scheduling.

Several methods have already been proposed to evaluate timetable stability on railway networks, mainly for cyclic timetables. Many of these methods are based on Petri nets and max-plus algebra ([14]). In [15], Goverde deals with general higher-order max-plus linear systems to model periodic railway timetables as scheduled max-plus linear systems and develops a generic max-plus system analysis theory that has been proved valuable in evaluating railway timetables on stability and robustness. This approach has been implemented in the analytical tool PETER [16] to evaluate timetable stability.

In [6], Carey and Carville describe various heuristic measures of stability that can be employed at early planning stages. In [7], Carey and Carville present a simulation model that is used for testing schedule performance regarding the probability distribution of so-called secondary delays (knock-on effects) caused by the primary delays, given the occurrence of these and a schedule. The model is used for evaluating schedules with respect to the ability to absorb delays. The super-models were introduced in [13] as a scheme to measure the degree of stability of a solution. This idea is extended to constraint satisfaction problems in [17] with the super-solution concept.
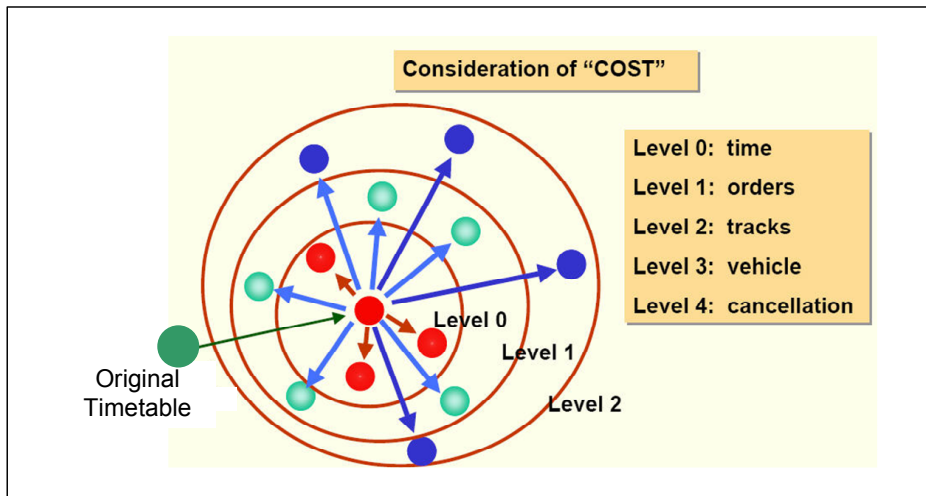
4

Figure 2: The main levels of robustness.

Nevertheless, due to the fuzzy definition of robustness in scheduling in general and in railway timetabling in particular, some simulation studies have been generated to evaluate robustness. In [41], Vromans, Dekker and Kroon present concepts of reliability in public railway systems. Using simulations, they test the effect of homogenizing lines and the number of stops in timetables. Mattsson [28] presents a literature study on how secondary delays are related to the amount of primary delay and the capacity utilization of the rail network. Hooghiemstra and Tunisse [20] present a prototype of a simulator used for studying the robustness of timetables on the Dutch railway network. The simulation prototype is called the DONS-simulator and is used for generating timetables. Then, Middelkoop and Bouwman [29] present SIMONE which is linked to the DONS-simulator in order to generate simulation models of national networks and timetables in an automated way. SIMONE is used for analyzing timetable robustness. It simulates a complete network and is used to identify bottlenecks.

Railway timetabling is one of the main stages in railway management (see Figure 3), where many feedback loops may be necessary:

Specifically, the timetable generation phase has two interrelated steps:

1. *Operator phase:* This process is performed by railway operators in order to obtain optimal timetables for their own trains with the objectives of: minimizing waiting times among passenger connections, optimizing rolling-stock and crew scheduling, optimizing customer requirements, etc. This is user-oriented timetabling.

2. *Infrastructure manager phase:* The request of all operators should be optimized thus satisfying robustness, and operational and traffic constraints.

The concept of robustness should be considered in relation to each one of
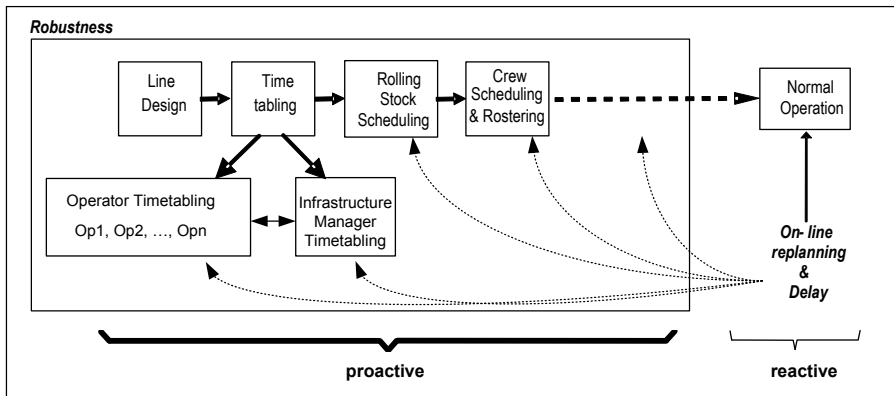
5

Figure 3: Main railway management phases

the above phases. In general, there are two approaches to prevent and manage the disruptions that can occur in the execution phase of any plan:

1. *Proactive:* it is directly related to the concept of robustness [18]. The goal is to obtain robust plans. A commonly accepted definition of robustness is: *"A robust plan is a plan that, given typical incidences in the environment, maintains its feasibility and, as much as possible, the optimality of the initial solution, at least with few meaningful changes".* However, due to the fuzzy definition, some questions are straightforward:

   - What are 'typical incidents'? According to railway manager experience, it is very difficult to characterize causes of incidents and disruptions, except for the fact that the most common disruptions are between 1 and 5 minutes long.
   - What are 'few meaningful changes'? They are changes that are necessary for restoring the plan to the initial plan. So, possible measures of robustness could be the required time interval until the initial plan is restored, or how many trains are affected, or the average delay of the trains.

2. *Reactive:* usually performed on-line when incidents can not be absorbed by a robust plan. Therefore, the goal of this approach is the re-planning of the previous plan according to the incidents in order to minimize the effects of incidents such as secondary delays, and the recovery of the normal plan/operation as soon as possible. Thus, the reactive approach is related to the on-line re-planning and delay management concepts.

4. How to Obtain Robust Timetables

Several methods can be applied in order to obtain robust timetables. Following, we present four influencing parameters that are related to each other.

6

These parameters affect the robustness of a timetable. Figure 5 shows these parameters as related axes. If we increase one of the parameters, we must decrease some of the others. For instance, decreasing speed or heterogeneity both influence capacity and robustness, and increasing optimality means decreasing robustness and heterogeneity, etc.

1. **Adding recovery time**

   The robustness of the timetable against small disturbances can be improved by optimally allocating time supplements and buffer times in the timetable. This is a common method to add time supplements to the travel times of each train in each section of track along its journey and buffer times to increase the minimum headway time. For instance, Figure 4 shows a real timetable for a train.

Estations	Dis	Tmín	Ttip	Toc	Tco	H lle	Tpa	F	P	t	H sal
Estation 1						08003	002				08023
Estation 2	1.4	0049	0223		023						08050
Estation 3	3.9	0216	0219		023						08073
Estation 4	1.8	0102	0106		013						08090
Estation 5	1.5	0039	0049		010						08100
Estation 6	1.4	0036	0040		010						08110
Estation 7	2.1	0048	0053		010						08120
Estation 8	1.5	0033	0037		010						08130
Estation 9	7.0	0237	0238	020	050						08180
Estation 10	6.3	0242	0247		030						08210
Estation 11	6.5	0248	0253		030						08240
Estation 12	7.3	0244	0245		030						08270
Estation 13	7.4	0247	0334		040	08310	001				08320

Figure 4: Real timetable.

The columns are:

- *Dis* (in kilometers) is the distance for each section of track.
- *Tmin* (in mmss) is the minimum travel time for each section of track (obtained from physical data and dynamic models of tracks and trains).
- *Ttip* (in mmss) is the basic travel time. *Ttip* introduces a user-defined slack (obtained from manager's experience) for robustness purposes. Infrastructure managers call it the 'security margin'.
- *Toc* (mss) is an extra-time for current maintenance operations in tracks.
- *Tco* (in mms) is the final rounded travel time for the train on the track (in this example, operational constraints require travel times rounded to 30").

The added slacks in *Ttip* give some robustness $R(x)$ to the timetable x. However, these slacks increase the global travel time $F(x)$. Thus, there is a direct relation between the global travel time to be optimized and the

7

timetable robustness $R(x)$ [4]: robustness is increased $R(x) - R(x^*)$ at the expense of a loss of optimality $F(x^*) - F(x)$.

Currently, slacks are introduced according to the historical experience of the infrastructure manager. Therefore, some questions are:

- Where should the slacks be placed in order to maximize $R(x) - R(x^*)$ versus $F(x^*) - F(x)$?

- Should they be placed uniformly?

- What relation exists between $R(x)$ and $F(x)$ in each timetable x?, etc.

2. **Decreasing capacity for a given traffic pattern**

Robustness is obtained by decreasing the capacity of the infrastructure for a fixed traffic pattern. It is especially useful in the case of double tracks. Decreasing capacity is also a common method applied by railway infrastructure managers and is used in the UIC (Union Internationale des Chemins de Fer) methods proposed by the International Union of Railways [40],[39]. Decreasing theoretical capacity for a given traffic pattern is related to the previous point due to the fact that it implies adding buffer time to minimum headways.

Railway managers know that scheduling as many trains as the theoretical capacity indicates is not viable. However, they have to make the best use of the expensive railway infrastructures. Trade-off between capacity and reliability/robustness, in other words, between the 'physical maximum' level of capacity and the 'economically optimal' level of capacity is a key point in operational management contexts [1].

For instance, Figure 5 extends a well-known graph for railway managers. Clearly, the greater the capacity, the greater the risk of secondary delays due to incidents. Thus, the idea is not to use the infrastructure at the maximum level of capacity (theoretical capacity), but within a practical capacity limit. Thus, Robustness is increased $R(x) - R(x^*)$ at the expense of a loss of capacity $C(x^*) - C(x)$.

However, there are no clear standards about how this trade-off can be modeled. For this purpose, the application of simulation methods seems to be an appropriate way to evaluate the trade-off between capacity and reliability. A stochastic model relating delays and capacity can be found in [21].

3. **Decreasing heterogeneity**

Railway traffic is considered homogeneous if all trains have similar characteristics, especially the same average speed per track segment, resulting from the running times and the stopping times. However, for large railway networks, railway traffic cannot be fully homogeneous as freight and passenger trains share the same infrastructure. If there are large differences
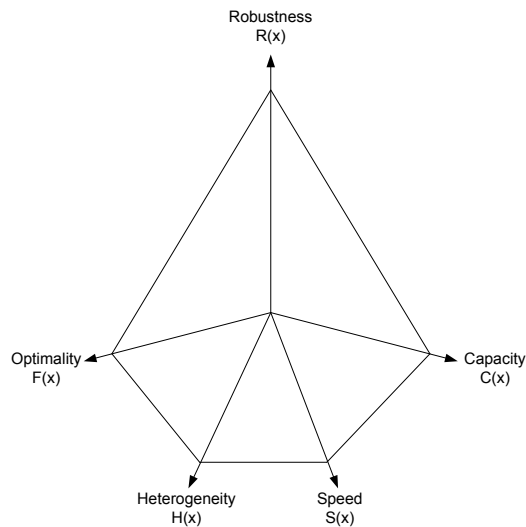
Figure 5: Main parameters related to Robustness

in the timetable characteristics of the trains on the same track, then the railway traffic is called heterogeneous [41]. Due to heterogeneous trains sharing the same infrastructure over large distances, timetabling becomes very complicated. Heterogeneity usually leads to many small headway times, which may increase delay propagation in the operations so that robustness decreases. It is generally accepted in practice that the heterogeneity resulting from the line plan and the timetable has a negative influence on the punctuality and the reliability of a railway system (UIC, 2004). Thus, homogeneous trains increase robustness and heterogeneous trains decrease robustness.

4. **Decreasing average speed** It is well known in any human-managed transportation system (without external help) that as the average speed increases, the headway time between two consecutive vehicles also increases, due to the fact that more time is needed to react/adapt to the system. For instance, when two cars are running at 5 km/h on a congested road, the headway time between them is reduced to a few meters, and this solution is robust due to the fact that a short disruption of the first car can be absorbed by the system. However, if the congestion disappears and the cars accelerate, the headway time must increase. If the headway time is not increased, this solution is not robust and a short disruption of the first car will not be adsorbed by the system and both cars will crash. Thus, in railway timetabling, this analogy with road traffic would be possible if there was not any signalling system, so the speed in inversely proportional to robustness. However, a direct relationship between train speed and minimum headway only holds for optimized signal distances with re-

spect to the allowed speed, and this in turn only works for homogeneous traffic.

Thus, according to the points made above, there is a clear trade-off between optimality/capacity and robustness, as the known price of robustness [4][9]. All these parameters: robustness, capacity, optimality and also heterogeneity are directly related (see Figure 5). As we have pointed out, robustness can be increased by decreasing capacity, optimality and heterogeneity.

Furthermore, different timetables with near-to-optimal travel times can be obtained due to the existence of many traffic operations (commercial stops, overtaking, crossings, replacements, etc.) which are responsible for generating different timetables. Thus, we should choose the most robust timetable among alternative timetables with similar and near-to-optimal travel times. Here, we need some analytical or simulation functions to characterize the robustness of each timetable. The goal is to obtain a better configuration of timetables, without penalizing the travel time of trains.

5. How to measure Robustness?

As was pointed out, a clear trade-off among capacity, optimality and robustness exists. We can obtain the theoretical capacity of a rail network $C(x)$, as well as the used capacity of a given solution. However,

- How can the robustness of a solution $R(x)$ be measured?

- How can we quantify the robustness of timetables?

- How can we ensure that one timetable is more robust than another?

We can point out several ways to do this:

1. **Stochastic programming models**

   Stochastic programming is an approach for modeling optimization problems that involve uncertainty. Whereas deterministic optimization problems are formulated with known parameters, real world problems almost invariably include parameters that are unknown at the time a decision should be made. Stochastic programming models are similar in style but try to take advantage of the fact that probability distributions governing the data are known or can be estimated.

   In railway timetabling, it is assumed that the process times, in the timetable generation phase, are deterministic. However, delays of trains occur since real-time railway operations are subject to external stochastic disturbances. For instance, in [25], the authors describe a Stochastic Optimization Model that can be used to allocate the time supplements and the buffer times in a given timetable in such a way that the timetable becomes maximally robust against stochastic disturbances. However, difficulties of

the stochastic programming method, in railway timetabling, are pointed out in [21]. Although it can be applied in simpler contexts, we consider it very difficult to obtain a model and apply stochastic programming to periodic and no-periodic timetables with very complex capacity and traffic constraints.

2. **Analytical methods**

   These consist of measuring 'certain characteristics' of timetables to evaluate their robustness.

   There are several analytical methods for evaluating robustness in general scheduling problems. These methods require metrics that characterize the robustness of solutions. In [2], a metric is proposed for counting the number of activity pairs that are not directly related; In [8] a metric that measures the roominess associated to each activity is defined; Nicola et al. [33] give a metric to measure the impact caused by an incident; etc.

   Although these analytical methods could provide a quicker way to measure robustness, they are not valid in a railway context and they do not correctly describe the robustness of a railway timetable.

   In a more specific railway context, analytical measures such as SSHR and SAHR have been developed to evaluate the homogeneity of a timetable, assuming the relation between this factor and the propagation of delays due to interdependencies between trains [41]. In [15], a linear system description of a railway timetable in max-plus algebra is presented; in [31], Joern presents microscopic models based on blocking time theory. Carey [6] proposes the use of dispersion measures of the inequality of headway times. These measures of robustness are especially useful in the case of double track lines. However, additional measures have to be developed to allow for cases of overtaking and single track lines. They must be based on the existing slack time with respect to the minimum reception time between two crossing trains.

3. **Simulation methods**

   This method is related to "what-if" analysis. The method consists of simulating incidents in a timetable, then re-planning and evaluating the effect on the final timetable (as the difference with the initial one). Some simulation models of railway processes are described in [29], [3] and [42].

   There are basically two different simulation models: Microscopic and macroscopic simulation models.

   - Microscopic simulation models which are related to a detailed topological level including train dynamics, signalling logic, etc. These simulation models are very accurate and they are very realistic and close to the real timetable. Microscopic operational simulation enables the railway operator to determine the consequences of infras-

tructure or operational changes in railway systems before investing funds or putting an unfeasible timetable into operation.

- Macroscopic simulation tools generalize network operations. Macroscopic simulation is focused on a level of stations, connecting tracks with simplified running time calculations, minimum headways instead of signal logic modeling, etc. These simulation methods lose accuracy, but they are more time-efficient.

Nevertheless, some works are based in the integration of both models, which would give several advantages for railway companies. In [23], the infrastructure data has to be maintained only once in the microscopic model and can be effectively reused in the macroscopic one. In order to be able to reuse the data of the microscopic model, the generation of the macroscopic network has to be automated.

5.1. Parameters for measuring robustness

In this subsection, we will identify the main parameters that will be taken into consideration to measure robustness in railway timetabling. A part of the time-distance diagram is shown in Figure 6. A time-distance diagram is a train diagram where a set of trains are scheduled. The vertical axis represents the space (locations). The horizontal axis represents the time. Horizontal dotted lines represent halts or junctions, while solid lines represent stations. It can be observed that the trains go from left-top to right-bottom. In Figure 6 (center), we shows the departure of train T_i from location l_j and arrival to location l_{j+1}. Figure 6 (right) shows the actual journey of this train, which stops in location l_j, departs from this location, accelerates to achieve the travel speed, and finally brakes to stop in location l_{j+1}. For simplicity, we will draw a straight line to represent this journey. This figure points out some of the parameters that will be used in the rest of the paper.

- **Incident** (I). The incident is the amount of time that a train wastes due to a disruption on the train, at the station, etc. This time I includes the braking time, the disruption time, and the acceleration time to achieve the travel speed. Figure 6 (center-bottom) shows that the incident I is the sum of the braking time plus the disruption time and the acceleration time. This time is called *primary delay*. If the incident is propagated to other trains, the amount of time that these trains waste is called *secondary delay*.

- **Point of the incident** (p) is the point at the track at which the incident occurs. This point will be relevant for studying robustness in some cases, and irrelevant in other cases. In Figure 6, this point corresponds to the location l_k.

- **Interval Time** ($ITime$) is the established time in which two contiguous trains are separated (see Figure 6). This time is split into two different parameters:
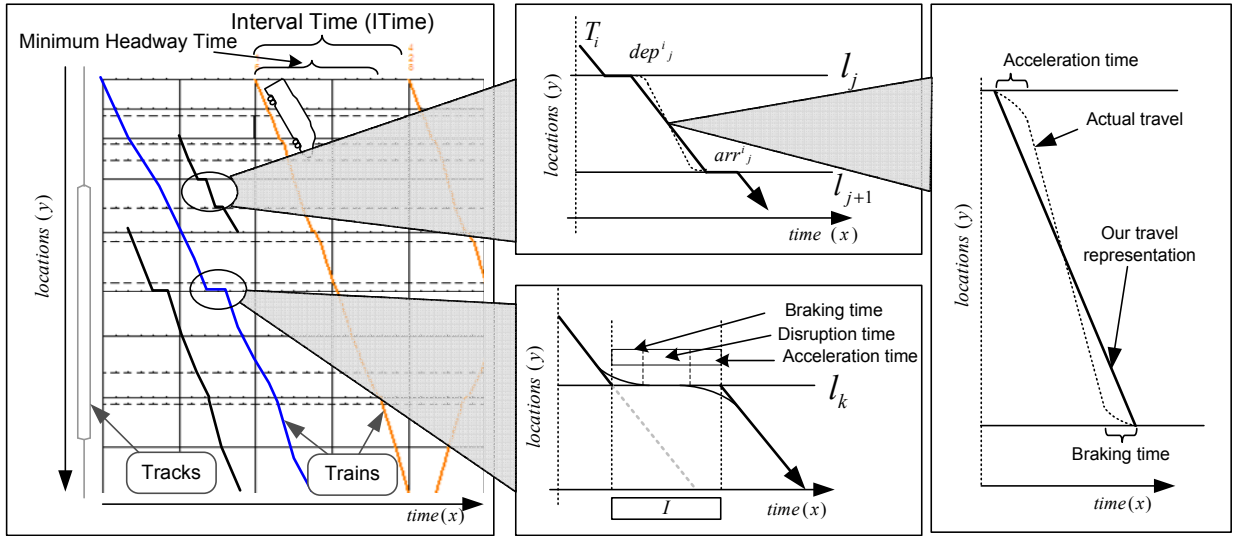
12

Figure 6: Example of a time-distance diagram.

(a) **Minimum Headway Time ($MHWT$)** is the minimum time needed between two contiguous trains to guarantee traffic operations.

(b) **Buffer Time** is the additional time included in the interval time between each pair of trains. This time is added to the minimum headway time to absorb short incidents.

- **Time Supplements** is the additional time assigned to the travel times of each train in each track along its journey.

- **Number of trains**. In highly loaded timetables, where capacity utilization is high, a disruption may be propagated to the rest of the trains in the timetable. Indeed, capacity utilization and robustness are opposing terms, so that a lower capacity utilization (smaller number of trains) implies higher robustness.

6. Analytical Methods for Measuring Robustness in Railway Timetabling

In this section, we present some analytical methods for estimating the robustness of single-line railway timetables on double-tracks. The Robustness of a timetable will be estimated by means of the primary and secondary delay that a disruption produces. For simplicity, we consider that the minimum headway time between two trains is constant throughout the journey, and the angle of each train is the average angle (pondered by distance) of all tracks.

A timetable A will be more robust than a timetable B if the estimated total delay of timetable A, due to an incident, is lower than the estimated total delay of timetable B.

In this section, we study the total delay generated in a timetable due to a disruption. We classify four types of timetable:

- Homogenous trains where overtaking **is not** allowed.

- Homogenous trains where overtaking **is** allowed.

- Heterogeneous trains where overtaking **is not** allowed.

- Heterogeneous trains where overtaking **is** allowed.

By homogeneous trains, we means trains with the same speed, meanwhile heterogeneous trains is related to trains with different speeds. In all of these cases, we will study the dependence of all above parameters. For instance, we will see that in some cases, the total delay is not dependent on the point of the incident. It is obvious that the time supplement is crucial for the robustness of a timetable.

6.1. Homogenous trains where overtaking is not allowed

In this subsection, we study the total primary and secondary delay in homogeneous trains where overtaking is not allowed. Due to the fact that all of the trains are of the same type, we consider them to be uniformly distributed along the planning horizon. Thus, the minimum headway time between any two trains is the same.
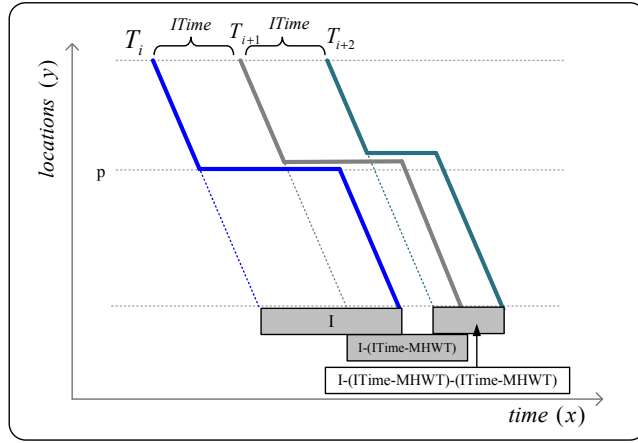
Figure 7: The incident generates primary and secondary delays due to the fact that overtaking is not allowed.

If a train T_i has an incident I at point p, secondary delays will affect k successive trains until the difference between their interval time and minimum

Figure 8: Equations to calculate the sum of delays generated by the incident I.

$$RtotalHomWO(I, i) = \sum_{j=i}^{i+k} R_j = \begin{cases} I & \left\lfloor \frac{I}{ITime-MHWT} \right\rfloor = 0 \\ \left\lceil \frac{I \cdot (I + (ITime-MHWT))}{2 \cdot (ITime-MHWT)} \right\rceil & \left\lfloor \frac{I}{ITime-MHWT} \right\rfloor > 0 \end{cases} \quad (1)$$

headway time absorbs the incident at point p. Figure 7 shows an example in which train T_i has an incident I (primary delay) and it is propagated to train T_{i+1} and T_{i+2} (secondary delays) until the disruption is absorbed in point p. In general, the disruption of each train is presented in Table 1. The first train (disrupted train) is delayed I time units, the second one is delayed in $I - (ITime - MHWT)$ time units, the third one is delayed in $I - 2(ITime - MHWT)$ time units, and so on, until the disruption is absorbed. The sum of disruptions generated for each train is developed in Figure 8. For simplicity, we denote the buffer time $BT = ITime - MHWT$.

Thus, the amount of primary and secondary delay generated by incident I in train T_i is given by equation (1).

Note that the global delay does not depend on the position along the train journey where the disruption occurs.

6.2. Homogenous trains where overtaking is allowed

If an incident occurs in a homogeneous train where overtaking is allowed, this incident is not propagated directly to the following trains in the initial timetable, but the secondary delay will affect the next trains once the incident has finished. Figure 9 shows an example in which train T_i has an incident. As overtaking is allowed, this incident does not affect train T_{i+1} and train T_{i+2}. The main problem is allocating the disrupted train (T_i) between two consecutive trains (T_{i+2} and T_{i+3}), so train T_{i+3} and the following trains may be affected by this incident. As in the previous case, we consider them to be homogeneously distributed by the same interval time. Thus, the minimum headway time between any two trains is equal.

Table 1: Propagation of the incident I

R_i	R_{i+1}	R_{i+2}	R_{i+3}	...	R_{i+k}	R_{i+k+1}
I	I-(ITime-MHWT)	I-2(ITime-MHWT)	I-3(ITime-MHWT)	...	I-k(ITime-MHWT)	I-(k+1)(ITime-MHWT)$\leq$ 0

$$RtotalHomO(I, i) = \begin{cases} RtotalHomWO(I, i) & \left\lfloor \frac{I}{ITime} \right\rfloor = 0 \\ \left\lfloor \frac{I}{ITime} \right\rfloor \cdot ITime + RtotalHomWO(MHWT, i) & I\%ITime < MHWT \\ \left\lfloor \frac{I}{ITime} \right\rfloor \cdot ITime + RtotalHomWO(I\%ITime, i) & I\%ITime \geq MHWT \end{cases} \quad (2)$$
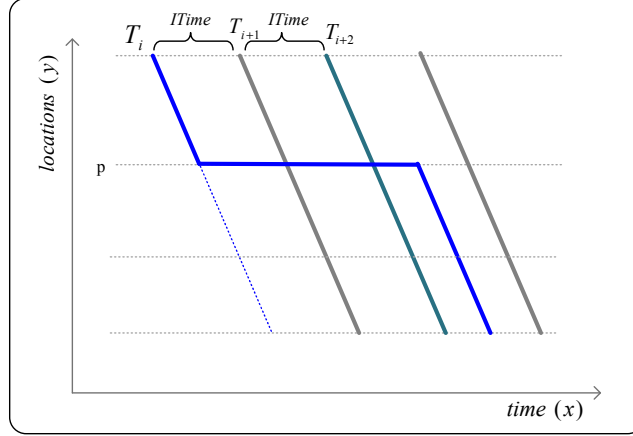

Figure 9: The incident may not affect some of the following trains when overtaking is allowed in homogeneous trains.

To study the total delay, given an incident I in train T_i, we must calculate the number of trains that the incident will not affect. This number is $nonaffec = \lfloor I/ITime \rfloor$. Thus, $nonaffec$ trains, following train T_i will not been affected by the incident. If $nonaffec = 0$, then we can use the formulas of the above section (see equation 2, first case). Otherwise, we must study two possible alternatives 1) and 2), which are represented in Figure 10.

1) The train T_i has been delayed I seconds in point p, but this train finishes its disruption too close to train T_{i+2}, so we must extend the delay to guarantee minimum headway time between train T_{i+2} and train T_i. Thus, we can apply the formulas of the previous section to study the rest of delays generated by the incident (see equation 2, second case).

2) The train T_i has been delayed I seconds in point p, but this train finishes its disruption far enough from train T_{i+2}, (more than the minimum headway time), so we must only apply the formula of the previous section to study the rest of the delays generated by the incident (see equation 2, third case).

Thus, the total delay in homogenous trains when overtaking is allowed can be obtained by using equation (2), where $I\%ITime$ is the modulo between I and $ITime$, that is $I\%ITime = I - ITime\lfloor I/ITime \rfloor$.

Note that the total delay is also independent of the point of disruption in the timetable.
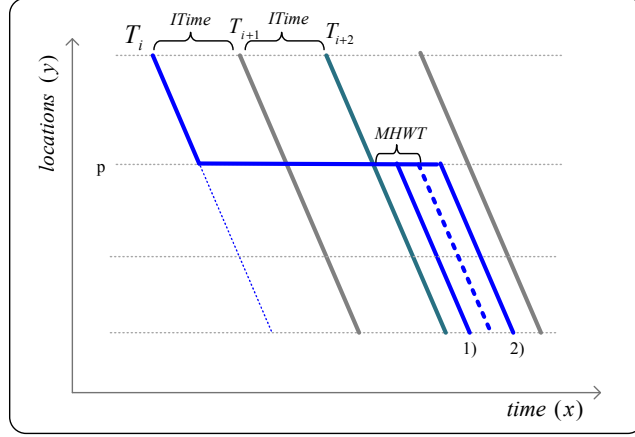
16

Figure 10: Two different cases of secondary delays in homogeneous trains.

6.3. Heterogeneous trains where overtaking is not allowed

In this case, we study the total primary and secondary delay in heterogonous trains where overtaking is not allowed. To this end, we must calculate the minimum headway time between every two trains at each point of the track. We introduce the use of angles of each train-track in the time-distance diagrams. The angle of a train is calculated taking into account the minimum travel time plus the time supplement given by the railway operators. Angles α_1 and α_2 are shown in Figure 11. It can be observed that the shorter the angle is, the faster the train goes. Thus, the angle measures the time needed to travel a given distance.

We can distinguish two different cases:

- The angle of the first train is smaller than the angle of the second train ($\alpha_1 < \alpha_2$). Thus, the first train is faster than the second train.

- The angle of the first train is larger than the angle of the second train ($\alpha_1 > \alpha_2$). Thus, the first train is slower than the second train.

Note that, if $\alpha_1 = \alpha_2$, the trains are homogeneous and the following two formulas can be applied. Let's study both cases.

Case 1 ($\alpha_1 < \alpha_2$).

The temporal distance between train (T_1) and train (T_2) in point p ($Dist_{12}(p)$), when $\alpha_1 < \alpha_2$, must be calculated as follows:

$$
\begin{aligned}
Dist_{12}(p) &= ITime_{12} + p \cdot \tan\alpha_2 - p \cdot \tan\alpha_1 \\
&\equiv ITime_{12} + \frac{p}{\tan(\frac{\pi}{2} - \alpha_2)} - \frac{p}{\tan(\frac{\pi}{2} - \alpha_1)}
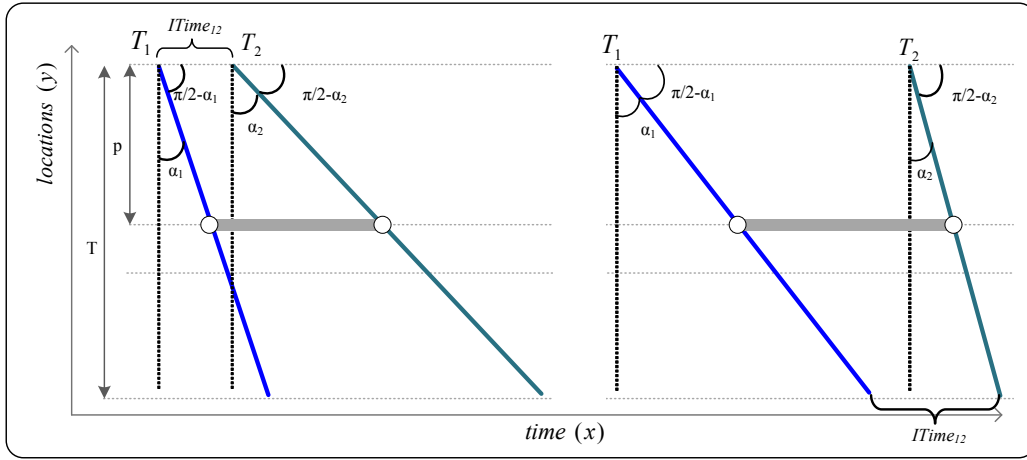\end{aligned}
\tag{3}
$$

17

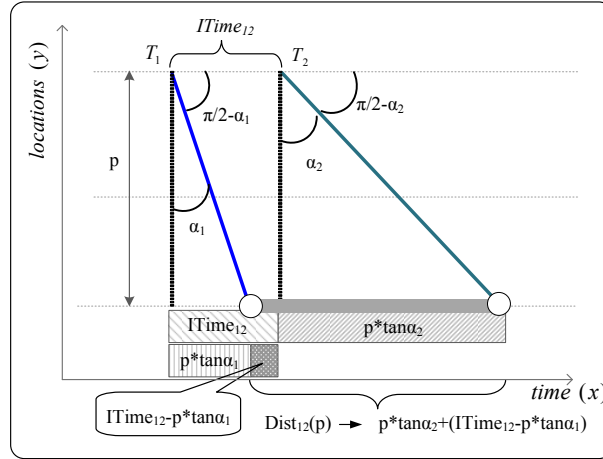Figure 11: Two different cases in heterogenous trains where overtaking is not allowed.

Figure 12: The angle of the first train is smaller than the angle of the second train ($\alpha_1 < \alpha_2$).

Figure 12 shows how $Dist_{12}(p)$ is obtained. This temporal distance can be obtained by means of α_1 and α_2 or by the complementary angles ($\pi/2 - \alpha_1, \pi/2 - \alpha_2$).

Case 2 ($\alpha_1 > \alpha_2$).

The temporal distance between train (T_1) and train (T_2) in point p ($Dist_{12}(p)$), when $\alpha_1 > \alpha_2$, must be calculated as follows:
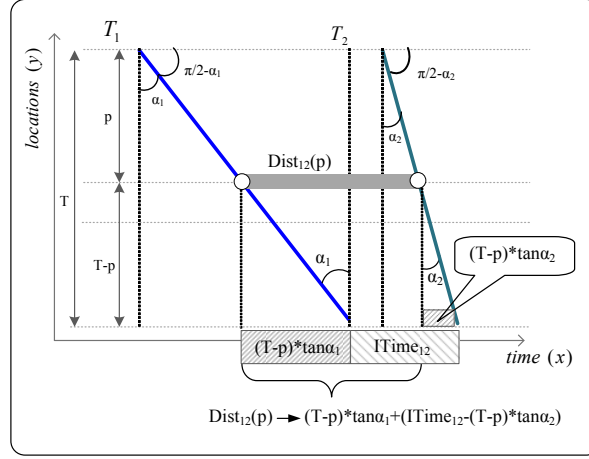
Figure 13: The angle of the first train is larger than the angle of the second train ($\alpha_1 > \alpha_2$).

$$Dist_{12}(p) = ITime_{12} + (T - p) \cdot \tan \alpha_1 - (T - p) \cdot \tan \alpha_2$$
$$\equiv ITime_{12} + \frac{(T - p)}{\tan(\frac{\pi}{2} - \alpha_2)} - \frac{(T - p)}{\tan(\frac{\pi}{2} - \alpha_2)} \tag{4}$$

Figure 13 shows how $Dist_{12}(p)$ is obtained. This temporal distance is directly related to the metric distance between the point p and the end of the track (T). This temporal distance can be obtained by means of α_1 and α_2 or by the complementary angles ($\pi/2 - \alpha_1, \pi/2 - \alpha_2$).

We study the amount of delay generated by an incident, when no overtaking is to be carried out. Without loss of generality, we consider that the incident is generated in train T_0. Given an incident I, the size of primary delay is I. Thus $Rt_0 = I$. Due to the heterogeneous nature of the train, we must study whether the following train is faster or slower than the disrupted train.

1. If the following train is faster (see Figure 14), then the delay of the second train in point p could be:

$$Rt_1(p) = Rt_0(p) - ITime_{01} + MHWT_{01} \tag{5}$$

Note that it is possible to measure the secondary delay without taking into account the temporal distance between both trains in point p due to the fact that in the previous formula $Rt_0(p) = I$. So, we can conclude that the secondary delay in a sequence of slower-train faster-train is independent of the point of incident.

However $Rt_0(p)$ could be very small and then Rt_1 may be lower than 0. To this end, the general formula will follow the following criteria:
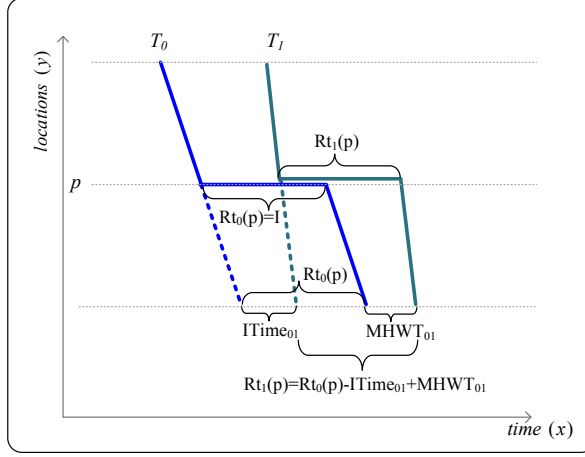
19

Figure 14: Secondary delay when the next train is faster.

$$Rt_1(p) = max\{Rt_1(p), 0\} \qquad (6)$$

2. If the following train is slower (see Figure 15), then the delay of the second train could be:

$$Rt_1(p) = Rt_0(p) - Dist_{01}(p) + MHWT_{01} \qquad (7)$$

It can observed that the secondary delay in a sequence of faster-train slower-train depends on the point of incident p. Figure 16 shows that the same incident generated in different stations, generates different secondary delays. This incident only generates primary delay in point p' (Figure 16 right) due to the fact that the temporal distance between both trains can absorb the incident, whereas this incident generates primary and secondary delays in point p (Figure 16 left). Thus, depending on the disruption point, the incident may be directly absorbed (primary delays) or not (secondary delay).

Thus, given an incident I in a train T_i at station p, the total amount of delay in heterogeneous trains where overtaking is not allowed is given by the following formula:

$$RtotalHWO(p, I, i) = \sum_{j=i}^{h} max\{Rt_j(p), 0\} \qquad (8)$$

where:

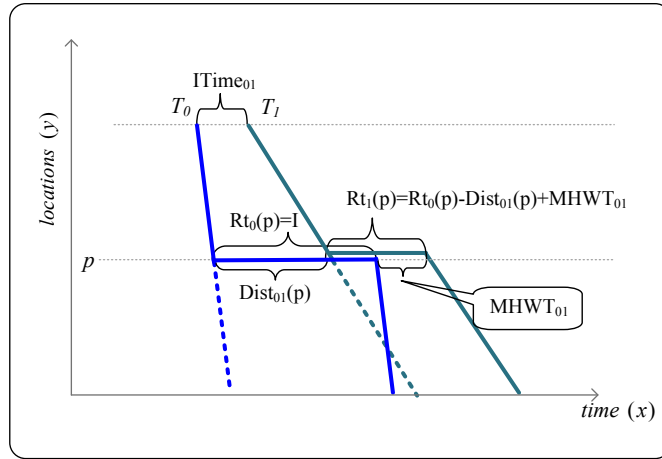$$Rt_i(p) = I; \qquad (9)$$

20

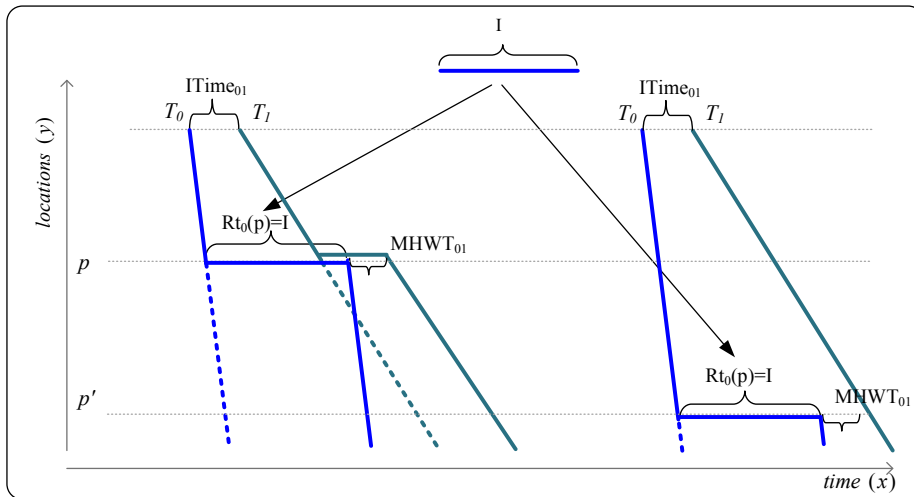Figure 15: Secondary delay when the next train is slower.

Figure 16: Secondary delay when the next train is slower.

$$Rt_j(p) = Rt_{j-1}(p) - \beta_j \cdot (ITime_{j-1,j} - MHWT_{j-1,j}) - (1 - \beta_j) \cdot$$
$$\cdot (Dist_{j-1,j}(p) - MHWT_{j-1,j}) \ \forall j > i \tag{10}$$

$$\beta_j = \begin{cases} 0 & \alpha_{j-1} < \alpha_j \\ 1 & \alpha_{j-1} \geq \alpha_j \end{cases} \tag{11}$$

Thus, the delay of each train depends on the delay of the previous train minus a convex combination between each type of sequence of trains.

If train T_j is faster that train T_{j-1}, then case 1 occurs and then $\beta_j = 1$ and so $(1 - \beta_j) = 0$; thus the above formula is reduced to:

$$Rt_j(p) = Rt_{j-1}(p) - (ITime_{j-1,j} - MHWT_{j-1,j}) \tag{12}$$

If train T_j is slower that train T_{j-1}, then case 2 occurs and then $\beta_j = 0$ and so $(1 - \beta_j) = 1$; thus the above formula is reduced to:

$$Rt_j(p) = Rt_{j-1}(p) - (Dist_{j-1,j}(p) - MHWT_{j-1,j}) \tag{13}$$

6.4. Heterogeneous trains where overtaking is allowed

In this section, we study the amount of primary and secondary delay in heterogonous trains where overtaking is allowed. To this end, we must calculate the minimum headway time between every two consecutive trains at the disruption point of the track. Thus, we use the formula of the previous section to calculate these temporal distances. Figure 17 shows an example in which Train T_0 has an incident in point p. In this figure, we have included three types of disruptions (a), (b) and (c).

- In case (a), the incident of train T_i is lower than the temporal distance between T_i and T_{i+1}. Thus, this case can be considered as a disruption in heterogeneous trains where overtaking is not allowed.

- In cases (b) and (c), the incident of train T_i (see Figure 17) is greater than the sum of temporal distance between train T_i and train T_{i+1}, train T_{i+1} and train T_{i+2}, ..., train T_{j-1} and train T_j. Thus, this case must be studied recursively in order to achieve the base case: the disrupted train T_i does not overtake the train T_{j+1} (see Figure 17). We must take into account that case (b) is different to case (a) in the recursive study due to the fact that in case (b) train T_i must satisfy at least minimum headway time with train T_j (see Figure 17). Meanwhile, in case (a), train T_i does not have to satisfy any constraint except its own incident.

  The recursive function $RecRtotalHO(p, I, i)$ is presented in formula 15, meanwhile the base case $BCHO(p, R, j)$ of this recursive function is presented in formula 16. Nevertheless, both cases (b) and (c) in the base case of the recursive study must be treated in a different way. Let's suppose that R is the remaining time between I and the arrival time of train T_j at station p. Four different alternatives can be classified (see Figure 18):
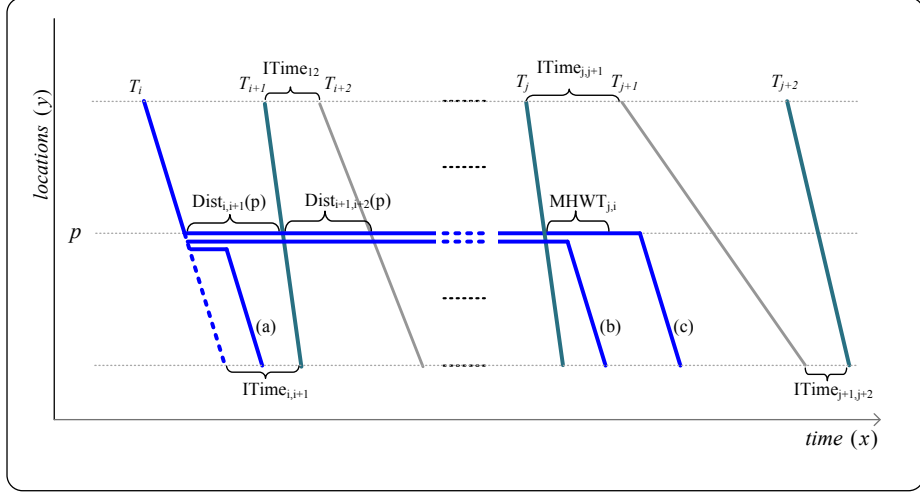
Figure 17: Three types of disruptions in heterogenous trains with overtaking.

1. The remaining time R is lower than the minimum headway time $MHWT_{j,i}$ and furthermore, the angle of train T_i is larger than the angle of train T_j ($\alpha_i \geq \alpha_j$). The base case is $MHWT_{j,i} + RtotalHWO(p, MHWT_{j,i}, j)$, that is, the minimum headway time plus the consequence of this delay in the case where no overtaking is allowed.

2. The remaining time R is not enough for train T_i to arrive at its destination with the minimum margin (minimum headway time ($MHWT_{j,i}$)), and, furthermore, the angle of train T_i is smaller than the angle of train T_j ($\alpha_i < \alpha_j$). The base case is composed of R plus the necessary additional time to arrive at its destination with the minimum margin plus the consequence of this delay in the case where overtaking is not allowed. To obtain the necessary additional time ($Extra_{j,i}(p)$), we use the formula $Dist_{12}(p)$ (see Figure 13) to adapt to our context.
   Thus, $Extra_{j,i}(p) = MHWT_{j,i} + ((T-p)tan\alpha_j - R) - (T-p)tan\alpha_i$.

3. The remaining time R is greater than the minimum headway time $MHWT_{j,i}$, and, furthermore, the angle of train T_i is larger than the angle of train T_j ($\alpha_i \geq \alpha_j$). The base case is $RtotalHWO(p, R, j)$, that is, the consequence of this delay in the case where no overtaking is allowed.

4. The remaining time R is enough for train T_i to arrive at its destination by satisfying the minimum margin (minimum headway time ($MHWT_{j,i}$)), and, furthermore, the angle of train T_i is smaller than the angle of train T_j ($\alpha_i < \alpha_j$). The base case is $RtotalHWO(p, R, j)$, that is, the consequence of this delay in the case where no overtaking is allowed.
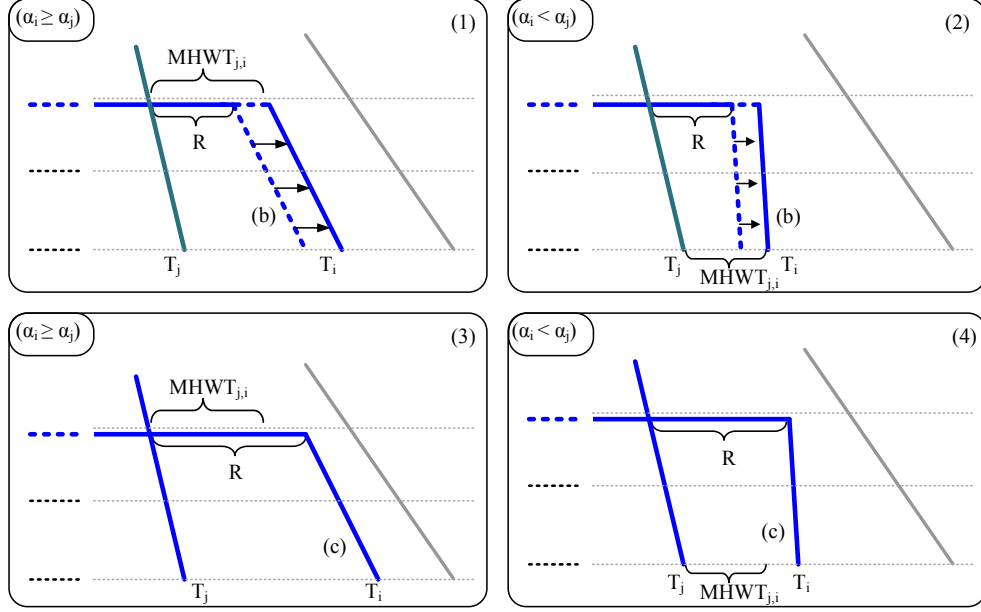
23

Figure 18: Four different alternatives for the base case for heterogeneous trains where overtaking is allowed.

Thus, given an incident I in a train T_i at station p, the total amount of delay in heterogeneous trains where overtaking is allowed is given by the formulas (14), (15) and (16).

7. A Simulation Approach for Robust Timetabling

In this section, we present a simulation approach to assess robustness of railway timetables. The approach consists of simulating disruptions to a given timetable, rescheduling it in order to repair conflicts that appear due to primary and secondary delays, and measuring the overall final delay obtained. This final delay allows us to assess the robustness of the initial timetable.

$$
RtotalHO(p, I, i) = \begin{cases} RecRtotalHO(p, I, i) & I - Dist_{i,i+1}(p) > 0 \\ RtotalHWO(p, I, i) & I - Dist_{i,i+1}(p) \le 0 \end{cases} \tag{14}
$$

$$
RecRtotalHO(p, I, i) = \begin{cases} Dist_{i,i+1}(p) + RecRtotalHO(p, I - Dist_{i,i+1}(p), i+1) & I - Dist_{i,i+1}(p) > 0 \\ BCHO(p, Dist_{i,i+1}(p) - I, i) & I - Dist_{i,i+1}(p) \le 0 \end{cases} \tag{15}
$$

$$
BCHO(p, R, j) = \begin{cases} MHWT_{j,i} + RtotalHWO(p, MHWT_{j,i}, j) & (R < MHWT_{j,i}) \wedge (\alpha_i \ge \alpha_j) \\ Extra_{j,i}(p) + RtotalHWO(p, R + Extra_{j,i}(p), j) & (Extra_{j,i}(p) > 0) \wedge (\alpha_i < \alpha_j) \\ RtotalHWO(p, R, j) & (R \ge MHWT_{j,i}) \wedge (\alpha_i \ge \alpha_j) \\ RtotalHWO(p, R, j) & (Extra_{j,i}(p) < 0) \wedge (\alpha_i < \alpha_j) \end{cases} \tag{16}
$$

24

In this section, we describe the underlying processes of the computer-based tool we have developed to implement this alternative approach.

7.1. Re-scheduling - Repairing timetables from disruptions

In practice, incidents occur and affect any initial optimized timetable. Thus, a robust timetable should be able to deal with the minor delays that occur in real time.

Incidents can occur anywhere, at anytime, and in any size. Technical failures, human factors, weather disturbances, etc., may influence the running times, dwelling events and departing events, thus causing primary delays. Due to the interaction between trains, these delays may be propagated as secondary delays to other trains in the network. A partial modification (re-scheduling) of the timetable may then be required during operation. Hence, managing railway traffic in real time requires re-scheduling train movements throughout the network, minimizing secondary delays and ensuring the feasibility of the resulting plan of operations. A real-time conflict resolution system aims to support the dispatchers in restoring the feasibility of the schedules given the real-time positions of the trains. In fact, in order to repair the timetable, only the part of the journey scheduled after the time disruption is considered.

We have developed a re-scheduling tool that is able to cope with disruptions and restore timetabling feasibility. This re-scheduling process searches for a feasible and optimized timetable that is as close as possible to the original timetable. Thus, the objective function for the re-scheduling process may be either (i) to minimize the sum of delays in destination arrival times or (ii) to minimize the sum of delays in departure times for each station with dwell time.

Optimality Criteria during the re-scheduling process

Given that:

- ω_i is the weight assigned to train T_i,

- $\widehat{arr}^i_j / \widehat{dep}^i_j$ is the original arrival time/departure time of train T_i in station j,

- arr^i_j / dep^i_j is the arrival time/departure time of train T_i in station j after the re-scheduling process,

- S is the initial timetable obtained as a solution,

- n_i is the number of stations with stopover in the journey of train T_i,

- N is the number of trains whose timetables have been rescheduled,

the objective function (i) is defined as $f(S) = \sum\limits_{i=1}^{N} \omega_i \times (arr^i_{n_i} - \widehat{arr}^i_{n_i})$, and the objective function (ii) is defined as $f(S) = \sum\limits_{i=1}^{N} \omega_i \times \sum\limits_{j=1}^{n_i} (dep^i_j - \widehat{dep}^i_j)$

Weights included in the objective function will allow different priorities to be assigned to different train types and to be considered when solving the conflicts generated by the delay propagation. The weighting is taken into account when the objective function is assessed, so that the process searches for the solution that minimizes the delays of trains whose weight is greater than the others. In this way, a priority order is specified among trains whose timetables have to be replanned.

Recovery Strategies are another important aspect in a re-scheduling procedure. Some of the most commonly used are: (i) reducing dwell times to a minimum, (ii) reducing headway to a minimum, (iii) reducing running times to a minimum, (iv) allowing overtaking in stations with available tracks, (v) cancelation of entire train lines (in case of severe disruptions).

The main strategies for allowing recovery in case of incidents are to increase headway or to consume the time supplement in the network. The re-scheduling procedure developed here reduces running times and headway and also allows overtaking in stations with available tracks.

Specifically, the re-scheduling process considers two values as the possible running time for each train and track section of its journey, rt_j^i and $cut\_rt_j^i$, where:

- The first value (rt_j^i) is the sum of: (i) the running time resulting from the dynamic model of train velocities, plus (ii) the temporal margin that human planners consider necessary for the train and track section characteristics, and (iii) time supplement that the railway manager adds for other security reasons and robust issues.

- The second value $(cut\_rt_j^i)$ is the same as the first without the time supplement.

The running time $cut\_rt_j^i$ is used by the re-scheduling process only if the train T_i is delayed in its arrival time to station $j + 1$.

We show an example in Figure 19. Figure 19.a shows a timetable as was originally planned. Suppose that train T_i suffers a disruption that causes a primary delay to its arrival at station j. This delay affects train T_{i+1} because the minimum headway time is not satisfied between these trains at station j. In this case, the re-scheduling process decides to delay train T_{i+1} until there exists the minimum headway time required between both trains. That is, the arrival time of T_{i+1} to station j is delayed until headway with respect to T_i holds. If the process uses the running times with the time supplements (rt_j^i), the rescheduled timetables are the ones shown in Figure 19.b. If the user allows recovery journey time, the process tries to reduce delays by using the allowed time suplements, and, therefore, chooses to use the running time without time supplements (cut running time) as the running time for both trains in the track section considered. The rescheduled timetables are shown in Figure 19.c.

Another additional constraint that is considered during the re-scheduling process is related to the departure time of trains from stations with dwell time. In such cases, the train cannot leave the station at a time that is earlier than
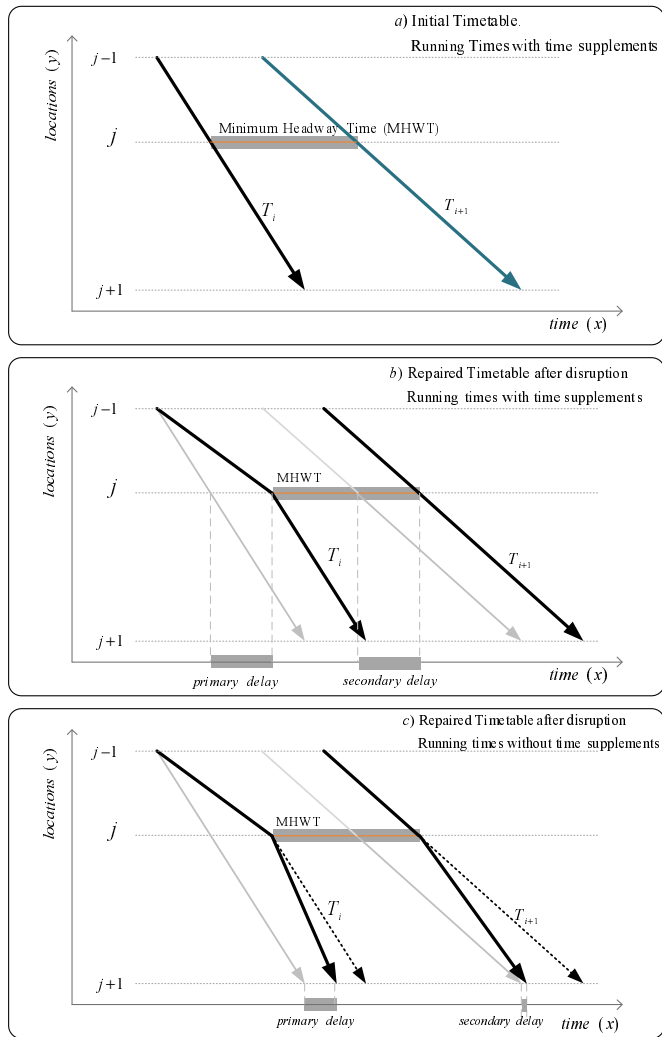
Figure 19: Cutting running times to reduce primary and secondary delays

the original scheduled departure time. This constraint prevents passengers from missing the train because of a premature departure. Thus, the use of time supplements will never allow a train to leave/arrive from/to a station before originally planned.

7.2. Disruption Simulator

In this section, we show the disruption simulator tool that was developed to assess the robustness of a given timetable. This tool introduces random incidents on a given timetable, performs a re-scheduling process and evaluates the primary and secondary delays that are caused.

The disruption simulator can be used in two main ways:

1. For assessing the robustness of real-world or user-defined timetables. Related timetable data are directly acquired by the disruption simulator from a database.

2. For evaluating the effects of several timetable parameters. In this way, the simulator is joined with a timetable generator, where several parameters can be varied, such as speed, heterogeneity, buffer times, time supplements and used capacity of generated timetables. Therefore, by using these generated timetables in the simulator, the effects of these factors on timetable robustness can be evaluated.

Figures 20, 21 and 22 show several examples of the timetables generated by way 2:

- The timetable in Figure 20 shows a homogeneous timetable where all trains belong to the same type and are uniformly distributed in the scheduling time horizon.

- Figure 21 represents a timetable of heterogeneous trains, where overtaking is possible in the configuration of initial departure time. The process that generates the initial timetable will verify that trains avoid overtaking in track sections.

- Additionally, heterogeneous trains can be grouped by sequences: Fast - Slow; Slow - Slow - Fast; Fast-Fast-Slow; etc. In Figure 22, a sequence of heterogeneous trains Fast-Slow is shown with a proposed departure time that avoids overtaking in track sections.

In the following, we describe the main stages of the disruption simulator, namely the disruption generation and the re-scheduling processes[2] (Figure 25).

---

[2]More information about the developed disruption simulator tool, functionalities and example cases can be found at: http://www.dsic.upv.es/grupos/gps/MOM/
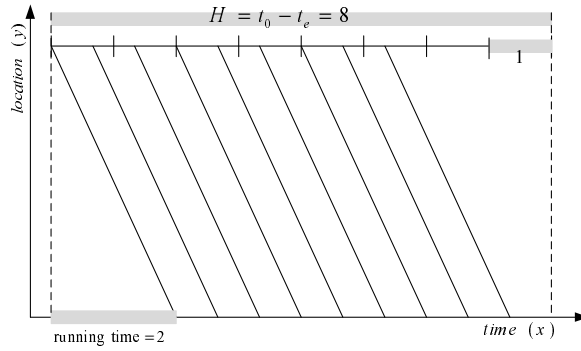
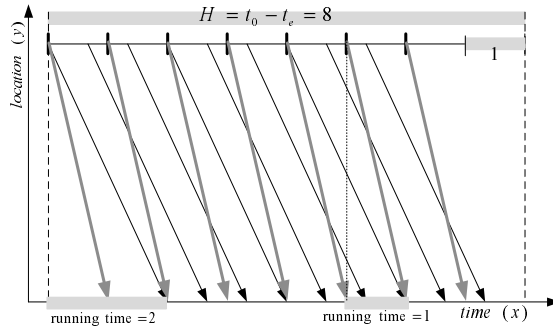Figure 20: Homogeneous Trains uniformly distributed in the Time Horizon of Scheduling

Figure 21: Departure Time of Heterogeneous Trains distributed uniformly (possible overtaking)
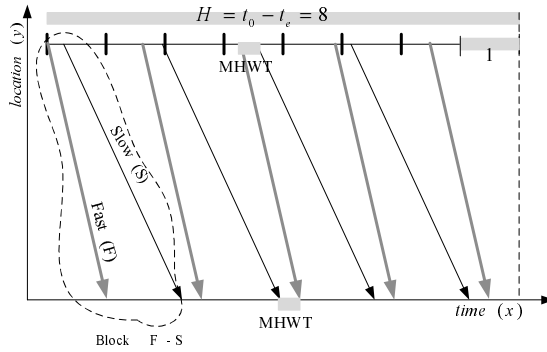

Figure 22: Heterogeneous Trains distributed by Block (without overtaking)

Disruption Generation

The simulation process assesses the robustness of a given railway timetable by evaluating the overall delay that an unexpected disruption causes. Let's assume that the initial disruption occurs on one train. Consequently, the time-
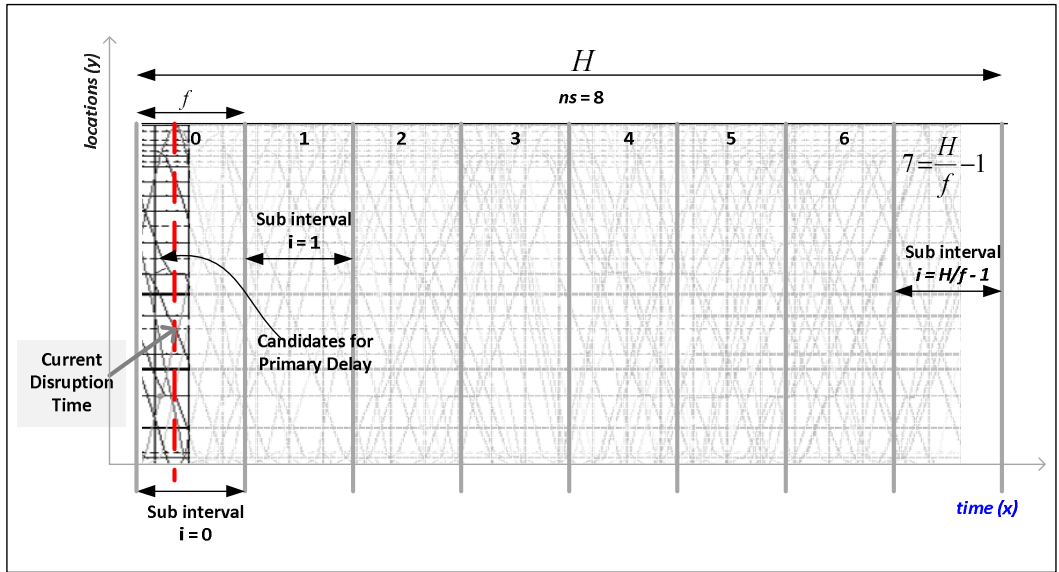
29

Figure 23: Selection of Disruption Time in the Time Horizon of Scheduling

table is re-scheduled and the primary delay is propagated to the subsequent trains of the timetable (secondary delays).

In real situations, disruptions may occur at any time and they may affect any train. Therefore, the process performs a set of sampling of disruptions over the timetable. In each sampling step, a single disruption occurs at time t_i on train $Train_j$ (primary delay); the disruption is propagated; the timetable is re-scheduled; and the overall (primary and secondary) delay for all affected trains is measured.

In each sampling step, the time when the initial disruption occurs, t_i, and the affected train, $Train_j$, are selected so that, at the end of the all sampling processes, trains in the timetable and sections of their journeys are uniformly chosen. At the end of all sampling processes, the final delay is returned as the average delay of the measured delays in each sampling step.

The main issues in each sampling step are: When does the incident occur? Which train is affected and at what point in its journey? Moreover, how many sampling steps should be performed in order to obtain a representative average delay?

Following, we describe the sampling process in more detail.

The planning horizon, $H = [t_0; t_e]$, is uniformly split into n_s subintervals, where n_s is parameterized by the user. Each $subinterval_i$ has $f = H/n_s$ time units and is defined as $s_i = [t_{0i}; t_{ei}]$. Figure 23 shows a timetable where the planning horizon is $[00:00, 08:00]$, $n_s = 8$, so that each subinterval is equal to one hour.

Once the subintervals have been defined, for each $s_i = 1..n_s$ subinterval, the

process iteratively: (i) inserts one disruption in the subinterval, (ii) re-schedules the timetable, and (iii) measures the overall delay.

In the first step, the time within the subinterval s_i in which the initial disruption is inserted, the train affected, and the size of the primary delay are randomly chosen according to the procedure described in Figure 24:

1. The time t_i, when the disruption occurs, is randomly chosen in interval $s_i = [t_{0i}; t_{ei}]$.

2. The size $delay_i$ of the primary delay is randomly chosen within a $delay\_range$ [$min\_delay..max\_delay$] defined by the user. Otherwise, a fixed user-defined primary delay can be used (for instance 1', 2', ...5').

3. At this step, the disruption time (t_i) and the size of delay ($delay_i$) are known, but it is necessary to know which train is the target of this disruption. The function $Get\_Can-\ didates\_Primary\_Delay$ identifies the train set (T) that are circulating at the time t_i when the disruption occurs. Then, the two trains from the set T whose delay affects the greatest/lowest number of following trains are the trains in the most initial/final part of its journey. The process randomly chooses one of these two worst/best case trains.

4. Once the affected train is chosen, the part of its journey that is running at time t_i is identified:

   (a) If the disruption occurs when the train is between stations k and $k+1$ of its journey, then its arrival to the station $k + 1$ is delayed.

   (b) On the other hand, if the disruption occurs when the train is stopped (for instance, in a commercial stop) at station k, then the delay is applied to its departure from this station k.

   In both cases, the sections of the train journey to be considered in the re-scheduling process begin from the station k. Likewise, as described in subsection 7.1, only the part of the timetable scheduled after the time when the initial disruption occurs should be considered. Therefore, the part of the timetable to be re-scheduled begins at time t_i, station k.

When the process $Put\_Disruption$ has been finished, the re-scheduling process begins, and the overall delay for all trains in the timetable is saved. Then, a new iteration of the sampling process on the initial timetable is performed. This loop continues until all subintervals (n_s) have been scanned over the given timetable. The final result is the average delay of all the overall delays obtained for each subinterval in which a disruption was inserted.

Description of the Re-scheduling Procedure

When a disruption occurs ($Put\_Disruption$), it causes a primary delay in the planned timetable so the original schedule may already not be feasible. For this

Algorithm 7.1: PUT_DISRUPTION($Timetable, s_i = [t_{0i}, t_{ei}], delay\_range = [min\_delay, max\_delay]$)

$t \leftarrow \text{RAND}(t_{0i}, t_{ei})$
$delay \leftarrow \text{RAND}(min\_delay, max\_delay)$
$Trains \leftarrow \text{GET\_CANDIDATES\_PRIMARY\_DELAY}(Timetable, t)$
$train_A \leftarrow \text{GET\_WORST\_CANDIDATE}(T, t)$
$train_B \leftarrow \text{GET\_BEST\_CANDIDATE}(T, t)$
$affectedTrain \leftarrow \text{RAND}(train_A, train_B)$
if $affectedTrain\_dep_k < t$ **and** $t \leq affectedTrain\_arr_{k+1}$
  then *(Case 4.a)*
$affectedTrain\_arr_{k+1} \leftarrow affectedTrain\_arr_{k+1} + delay$
if $affectedTrain\_arr_k < t$ **and** $t \leq affectedTrain\_dep_k$
  then *(Case 4.b)*
$affectedTrain\_dep_k \leftarrow affectedTrain\_dep_k + delay$

Figure 24: Procedure to simulate a disruption over a given railway timetable

reason, a re-scheduling procedure becomes necessary. It must detect and repair the conflicts that appear as a consequence of the unexpected disruption. Next, we describe the re-scheduling process that is performed at each iteration of the sampling process, after the *Put_Disruption* function inserts the disruption:

1. Propagate the primary delay to the rest of the affected train journey.

2. Generate an ordered list where each $node_j$ is a pair $[T_i; l_k]$. Each pair refers to a train in the timetable and a track section of its journey. The nodes are ordered according to the departure time of each train T_i from the corresponding station l_k. The first node in the list for the re-scheduling process is the earliest $[train, track\_section]$ departure that is after the time of the incident. The following steps start with the first node in the list ($j = 1$).

3. Verify the timetable corresponding to train and track section defined by the pair that occupies position j in the ordered list. If no conflict is detected, increase position j in one and repeat this step. Otherwise, repair it according the following rules:

   (a) If the two trains have the same running time in the track section, then the train with the greater departure time is delayed in order to satisfy the constraint. This criterion will obtain the minimum overall delay for both trains.
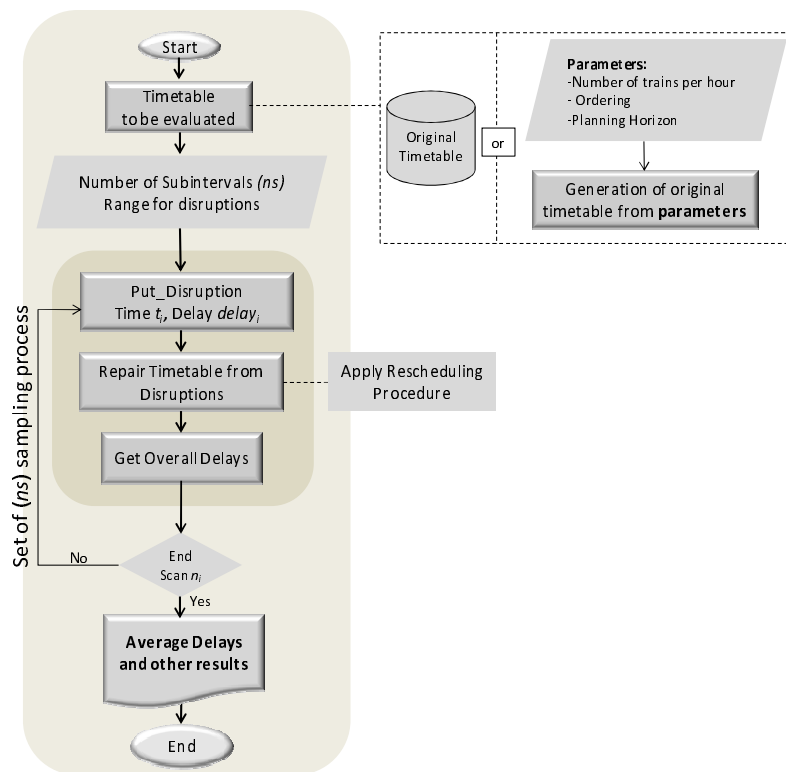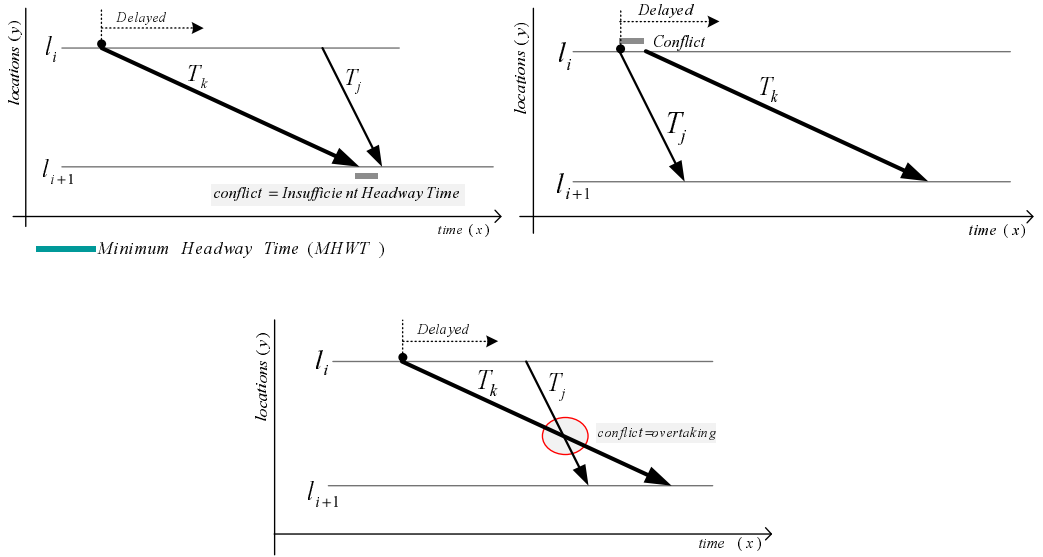
32

Figure 25: Simulator Main Stages

Figure 26: Conflict between trains with different running time in the track section

    (b) In the other case, the train that produces the minimum delay with respect its original timetable is delayed.

The technical stop that is added to the departure time of the selected train to be delayed in order to satisfy the constraint being verified is called secondary delay.

4. Propagate the secondary delay assigned to the selected train in the previous steps to the rest of its journey.

5. Reorder the list of pairs (train, track section) according to the departure time of each train. Go to step 3, until the end of the list is reached.

6. End of re-scheduling procedure.

    In step 3 of the above procedure, when a conflict between two trains appears, the process decides which of the two trains will be delayed to repair the conflict[3]. Figures 26 and 27 show the cases where the rules (a) and (b) are applied, respectively.

- In the first case of Figure 26 (upper left), a headway conflict appears at destination: the slower Train T_k is delayed in track section $l_i \rightarrow l_{i+1}$

---

[3]A weighted re-scheduling process could be applied, such that the weight assigned to each train type on the railway line is considered: trains with higher weight (higher priority) will be prioritized (like fast trains) in the re-scheduling process respect to the lower weight trains
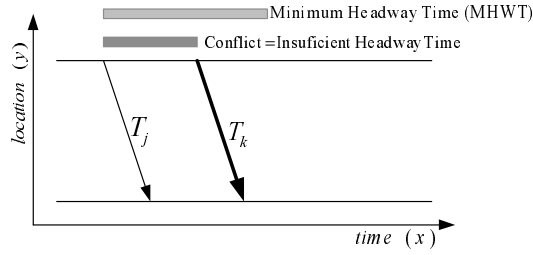
Figure 27: Conflict between trains with the same running time in the track section

such that a conflict with the faster train T_j appears at location l_{i+1}. In this case, the train T_j will be delayed at l_i in order to hold the headway constraint with respect to T_k at location l_{i+1}. The process computes the delay of each train with respect to its original timetable if it was delayed. If T_j was delayed the difference with respect its original timetable would be minor than if the train T_k was delayed. Then, T_j is delayed.

- In the second case of Figure 26 (upper right), a headway conflict appears at origin: the faster train T_j is delayed with respect to its original departure time of l_i (as case 4.b of the *Put_Disruption* procedure) such that a conflict with the slower train T_k appears at location l_i. Thus, the departure time of T_k from l_i should be delayed in order to hold the headway time with respect to T_j at location l_i.

- The third case of Figure 26 (lower center), is similar to the first case, if we consider the headway time at destination to be a negative value. The slower train T_k is delayed from l_i to l_{i+1} such that a conflict with the faster T_j appears during the journey from l_i to l_{i+1} (as a negative headway time at destination). As the first case, the process computes the delay of each train with respect to its original timetable if it was delayed. If T_j was delayed the difference with respect its original timetable would be minor than if the train T_k was delayed. Then, T_j is delayed.

We will describe each step of the re-scheduling procedure in more detail with an example. Figure 28 shows a timetable that has been planned for the circulation of trains T_k, T_{k+1}, T_{k+2}, and T_{k+3}. The minimum headway time that there must be between consecutive trains is given by the Minimum Headway Time ($MHWT$).

Let's suppose that a primary delay affects the train T_{k+1}, producing a delay (primary delay) in its departure time from the station l_{i+1}. This delay is propagated to the rest of its journey (Figure 29). At this point, a re-scheduling procedure is necessary because the timetable of one of the trains in the original timetable has been modified. We need to check if this delay produces a conflict with the rest of the trains in the original timetable.

The first step of the re-scheduling procedure consists of propagating the primary delay of the affected train to the rest of its journey. In the case of the
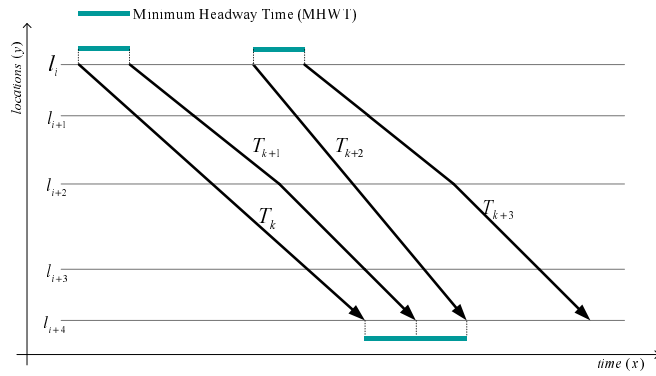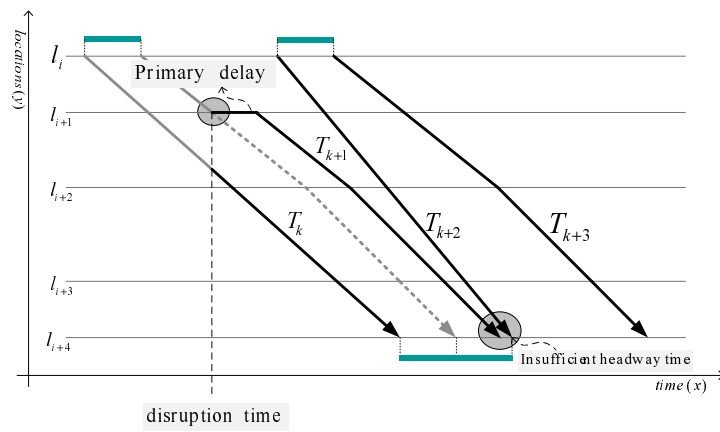
Figure 28: Original Timetable in Circulation

Figure 29: Primary delay is propagated to the rest of the train journey

affected train T_{k+1}, the dotted line indicates the original timetable, and the continuous line indicates the possible new timetable for T_{k+1} after the disruption occurs. Once we obtain a new possible timetable for train T_{k+1}, a list of pairs (train, track section) is generated in step 2. For each train and for each track section of its journey considered by the re-scheduling (i.e., after the time, when the primary delay occurs), a node is added to the list. The pairs are ordered according to the departure time of the train from the station where the corresponding track section of the pair begins. The ordered list corresponding to Figure 29 is shown in Figure 30.

The timetable is verified considering each node of the list in the same order that it appears in the list. The verification starts with the node in position $j = 1$ (in the example, the node $[T_k; l_{i+2}]$). Here, we verify that the train T_k has no conflict with any other train in the track section $l_{i+2} \rightarrow l_{i+3}$. Therefore (step 3), j is increased by one unit in order to repeat the constraint verification with the node in position $j = 2$. When this node is checked, a conflict with train T_{k+2} is detected (see Figure 31). In this case, the procedure has to decide which of the two trains, T_{k+1} or T_{k+2}, must delay its departure time from l_{i+1}, in such a way that the minimum headway time between the two trains is held. The procedure applies the rule (a) (step 3) of the algorithm assigning a technical stop to train t_{k+1} because its running time in track section $l_{i+1} \rightarrow l_{i+2}$ is greater than the running time of T_{k+2}.
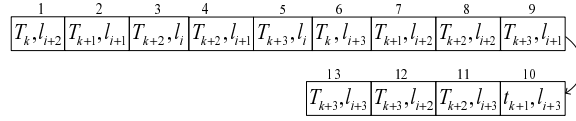

Figure 30: Ordered list of pairs [train, track-section]

Once the conflict is repaired by adding a technical stop to the selected train (T_{k+1}), it is necessary to propagate this secondary delay to the rest of the selected train journey (step 4), and the list is reordered because departure times have been modified (step 5). The position $j = 2$ in the ordered list is not modified, so the next node to be verified will be $[T_{k+2}; l_i]$. Figure 32 shows the state of the schedule after the first repair has been made.

When the end of the ordered list is reached, the re-scheduling procedure ends, and a new feasible schedule, which should be the closest possible to the original one, is obtained.

Time Supplements in Running Times: Recovery

The process of re-scheduling only changes the timetables of those trains whose departure times have been modified in order to repair conflicts that appear. When the departure time of a train from a station of its journey has been delayed with respect to the original one, the re-scheduling process projects a new arrival time to the next station. As pointed out in subsection 7.1, the re-scheduling process can use the minimum feasible running time given for trains for
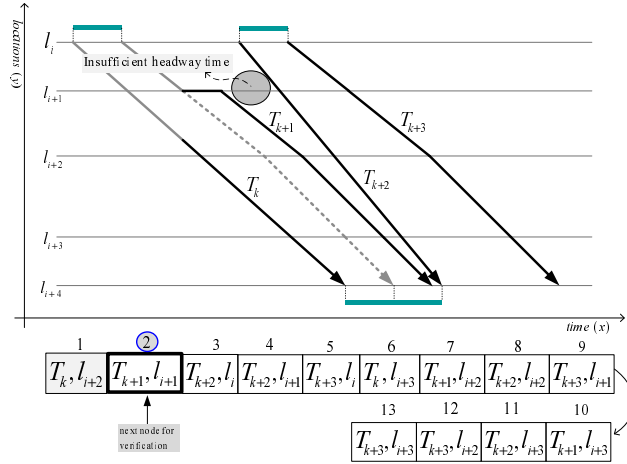
Figure 31: Verification of the timetable assigned to train T_{k+1}, in the track section $l_{i+1} \rightarrow l_{i+2}$ (node 2 in the ordered list)
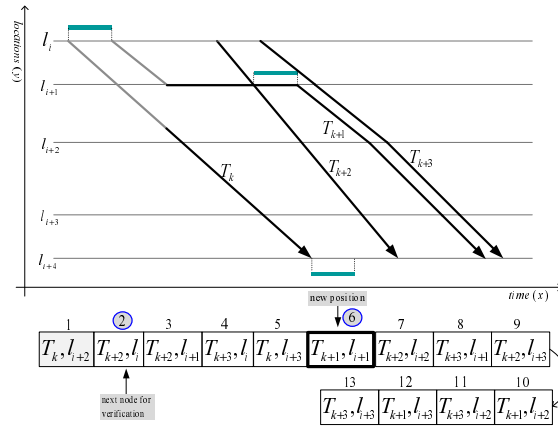

Figure 32: Repair of conflict and Reordering of [train,track-section] list

traveling track sections. In this way, if the original timetable has been planned considering rt_j^i as running time (the minimum feasible value plus a time supplement), it would be possible to decrease the delay or to recover the punctuality in the following arrival time. However, if the projection performed with the minimum feasible running time (without the time supplement) indicates that the train will arrive at the next station earlier than its original arrival time, the running time will be increased until the new arrival time is, at least, equal to the original one. In this case the train has been able to absorb the disruption and return to its original timetable. However, a train without scheduled stops during the rest of its journey could be re-scheduled such that it could arrive at destination before the arrival planned time. For instance, in Figure 33, it can
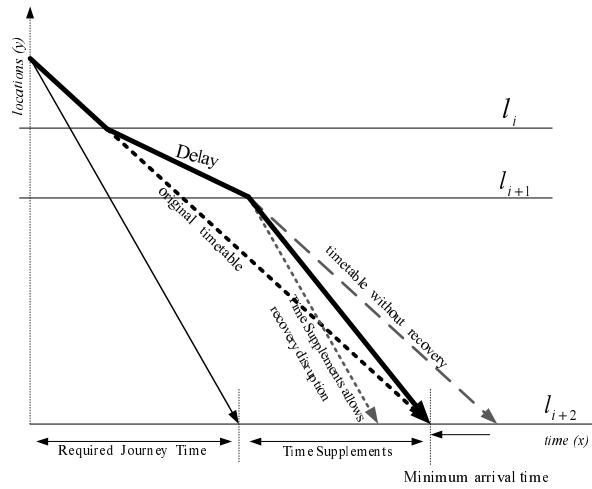
Figure 33: Example about use of buffer time by the re-scheduling process

be observed that trains can recover even more time than the disruption time
(see the curve behind the disruption linear line). However, this is not the usual
case. Trains only recover time whereas it is necessary to absorb the incident,
and it returns as soon as possible to the original timetable.

Figure 33 shows an example of the possibility of recovering running time
after delays, when time supplements have been added in train journeys. In the
track section specified in the example, the timetable for T_i has been planned
considering the minimum feasible plus a given time supplement as running time.
During the implementation of this scheduling, T_i is affected directly or indirectly
by an unexpected disruption in track section $l_i \rightarrow l_{i+1}$. When the re-scheduling
process projects the new timetable for T_i using the same running time that was
used in the original schedule, the delay is maintained and it also appears in
l_{i+2}. At this point, the re-scheduling process projects the new timetable of T_i
considering as running time the minimum feasible without the time supplement
considered by the original planning. In this way, it obtains an earlier arrival
time than the original one. For this reason the running time increases until T_i
arrives at the originally planned time at l_{i+2}.

At the planning stage, it is very important to decide where and how many
time supplements are added to the minimum running time of trains. It is a
clear trade-off between optimality and robustness. Analysis of this dependency
is not the goal of this article. However, we can see the effects of adding time
supplements in the first, middle or final part of train journeys, depending on the
size of the primary delay (see Figure 34). Obviously, if we only take into account
delays of trains at the final stations of their journeys, the more towards the end
of journeys time supplements are placed, the more useful they will be in case
of delays. Therefore, the schedule would be more robust. It can be observed in
Figure 34 that time supplements in final stations reduce the overall delay due to
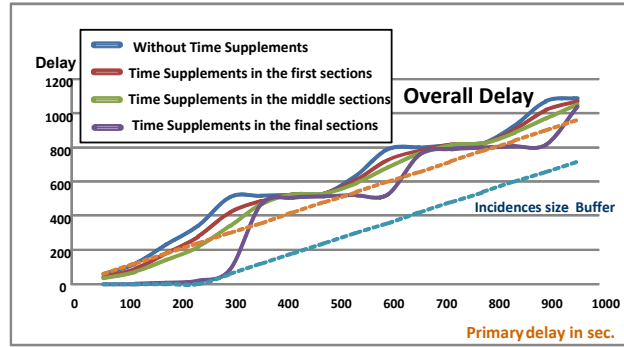
39

Figure 34: Overall delay when time supplement is located at the first stations, in the middle or at the end stations.

the fact that disruptions along the entire path can be more easily absorbed. It must be taken into account that this is the case only if we consider the overall delay of each train at the end of its journey. Moreover, note that this figure represents a timetable where overtaking is allowed. For this reason, each curve is similar to the *sin* function, that is, as the disruption increases, the secondary delay increases, until the disrupted train is overtaken by the following train. In this case, the overall delay decreases, and so on.

Other results show that in highly loaded timetables, which usually have higher secondary delays, the availability of time supplements for delay recovery becomes more important. Figure 35 shows the different overall delay (with or without time supplements) depending on whether the timetable is heavily or lightly loaded. It can be observed that in the interval [00:00, 06:00], the line is lightly loaded so a disruption can be easily absorbed and secondary delays are not relevant. However, in the interval [14:00, 20:00], the line is heavily loaded so any disruption generates higher secondary delays.

Results obtained from the Disruption Simulator

As pointed out, the developed disruption simulator tool performs an iterative set of sampling of disruptions over the timetable planning horizon. In each one of the n_s sampling processes, the three sequential steps of put-disruption, re-scheduling, and overall delay measurement for all rescheduled trains are performed. These results are obtained in each sampling process:

- (δ_i) - Average Delay: Overall delay divided by the number of trains whose timetables are being rescheduled.

- (α_i) - Affected Trains: number of affected trains (with delay with respect to their original timetables).

- (Δ_i) - Overall Delay: sum of delays for all affected trains.

- (ρ_i) - Percentage Delay: percentage of delay with respect to the planned running time of trains.
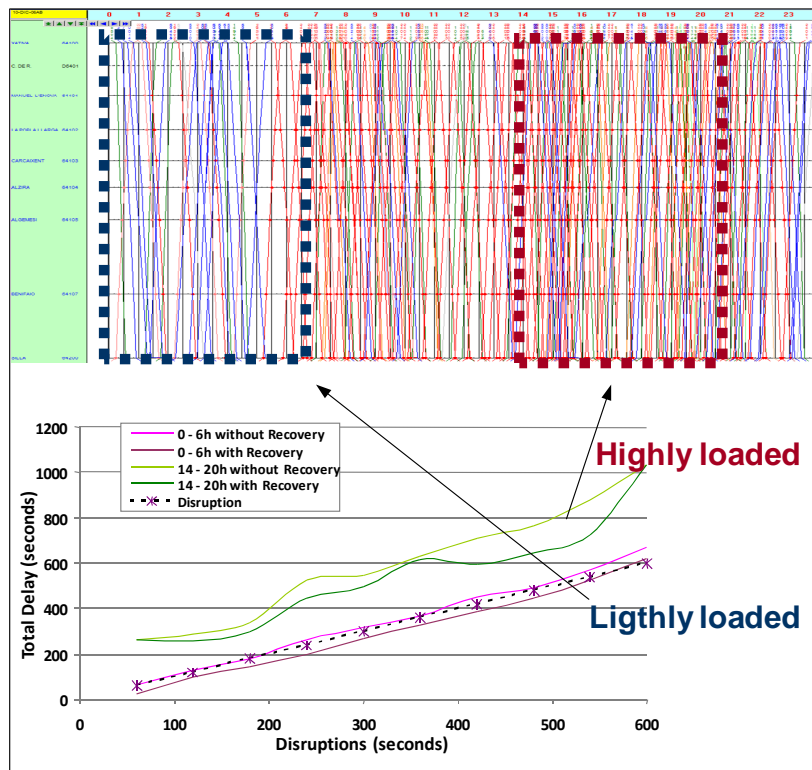
40

Figure 35: Total delay of real railway line in highly loaded time interval and lightly loaded time interval.

Number of Re-scheduling	Av. Delay	A. Trains	T.Delay	Perc. Delay	Settling Time
1	δ_1	α_1	Δ_1	ρ_1	τ_1
...					
i	δ_i	α_i	Δ_i	ρ_i	τ_i
...					
M	δ_M	α_M	Δ_M	ρ_M	τ_M
Final Results	$\dfrac{\sum\limits_{i=1}^{M} \delta_i}{M}$	$\dfrac{\sum\limits_{i=1}^{M} \alpha_i}{M}$	$\dfrac{\sum\limits_{i=1}^{M} \Delta_i}{M}$	$\dfrac{\sum\limits_{i=1}^{M} \rho_i}{M}$	$\dfrac{\sum\limits_{i=1}^{M} \tau_i}{M}$

Table 2: Response variables obtained from the disruption simulation

- (τ_i) - Settling Time: time elapsed from the disruption time until all trains run without delay (i.e. the timetable returns to its initial planned state).

At the end of all n_s sampling processes over the given timetable, several average results are obtained (Table 2).

The described processes do not consume very excessive computational time. Therefore, in order to obtain representative values, each set of n_s sampling processes can be repeated again several times. In the example shown in Figure 35, the width of intervals was around $30' - 60'$ (n_s is around 8 -16, since we usually restrict the timetable horizon up to 8 hours) and primary delay was in the range of $[1', 5']$. Typical tested timetables have around 50-80 trains, and each set of n_s sampling processes was repeated $M = 1000$ times.

This Disruption Simulator is an specific module that is integrated in a computer-based support system for railway traffic planning that is applied for generating real-world railway timetables in a commercial Railway Infrastructure Manager. The Disruption Simulator process takes into account all the constraints, and all train and infrastructure data that exist in real-world cases. For instance, the re-schedule process considers real-world running times; required times for acceleration and braking in delays; number of available tacks in stations; maintenance track constraints; types, controls and signals in headway constraints; etc. These details have not been described in this paper for reasons of simplicity.

8. Evaluation: Results from the Analytical Model vs the Disruption Simulator

In this section, we present the evaluation obtained from the two methods presented in the above sections, the analytical and simulation approaches. We use the same test cases for both methods in order to contrast the results obtained by each one.

First Test Case : Homogeneous Trains

The first tests were performed on a homogenous railway line with the following characteristics:

| Train i	RTotalHomWO (I=360,i)	RTotalHomO (I=360,i)
1	7200	6000
2	6840	5700
3	6480	5400
4	6120	5100
5	5760	4800
6	5400	4500
7	5040	4200
8	4680	3900
9	4320	3600
10	3960	3300
11	3600	3000
12	3240	2700
13	2880	2400
14	2520	2100
15	2160	1800
16	1800	1500
17	1440	1200
18	1080	900
19	720	600
20	360	360
Average Sum of Delay	3780	3153

Table 3: Analytical Results for Homogeneous Trains

- Incident (I) = 360 seconds

- Minimum Headway Time $(MHWT)$ = 300 seconds

- Number of trains = 20

- Interval time $(ITime)$ = 300 seconds

Table 3 shows the results obtained. Column 2 $(RtotalHomWO)$ shows the total delay for the case where the trains cannot overtake another train. In column 3 $(RTotalHomO)$, we present the results for the same trains but when they can overtake another train. Each row of the table indicates the order number of the train affected by the disruption. This is the train that has the primary delay. In this test case, the period of time between the trains is the same as the headway time. For this reason, the primary delay (I) is not absorbed, and all of the following trains have a secondary delay. Considering that the disruption can affect any train on this railway line, the average total delay is shown in the last row of Table 3.

Total trains in the railway line	Periodicity	Average Sum of Delay
1	4h	360
2	1h38m30s	360
3	49m40s	360
4	32m40s	360
5	24m50s	360
6	19m50s	360
7	16m30s	360
8	14m	360
9	12m30s	360
10	11m	366
11	9m	469.091
12	8m10s	515
13	7m30s	604.615
14	7m	685.714
15	6m30s	840
16	6m10s	1005
17	5m50s	1231.765
18	5m30s	1616.667
19	5m10s	2510.526
20	5m	3153
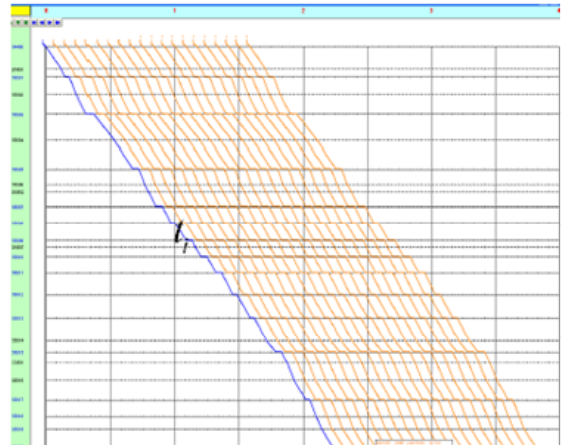
Primary Delay
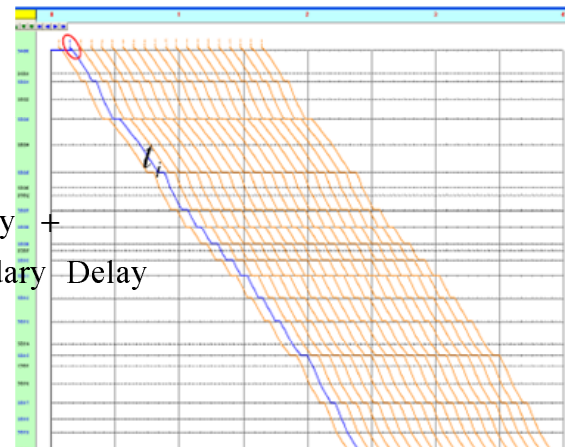
Primary + Secondary Delay

Figure 36: Results obtained with the simulation tool for a homogeneous railway line with overtaking

Figure 36 shows the railway line used by the simulation tool to evaluate the same schedule used by the analytical method. Represented in this figure is the original schedule before the disruption occurs and the new schedule after the re-scheduling process changes the timetables in order avoid future conflicts among the trains caused by the primary delay in the train T_i. Each row of the table shown in Figure 36 corresponds to a given period of time between the trains (different $ITime$). The disruption does not produce a secondary delay until the trains are separated by a period of time of 11 minutes. From this point, the secondary delays increase until the maximum capacity is reached, when the period of time between the trains is the same as the headway time. The result in the last row corresponds to the test case evaluated by the analytical method. We can observe that the average total delay produced by the simulation tool is the same as the average total delay obtained from the analytical approach.

Second Test Case : Heterogeneous Trains

We have also contrasted the results obtained in both approaches when the railway line is composed of heterogeneous trains. In this case, the railway line is composed of trains with an average speed equal to 100 Km/h and 80 Km/h, respectively. For the analytical approach, we consider the following:

- angle for the slower trains: 22.6°

- angle for the faster trains: 14.6°

- position (p) with respect to the initial point of the railway line, where the total delay is computed = 162.7 Km

- period of time between consecutive trains ($ITime$) = 300 seconds

- Minimum Headway Time ($MHWT$) = 250 seconds

The result given by the analytical approach indicates that, in the point $(p) = 162.7Km$, the trains have a total delay of 670 seconds. The following expression indicates the way that the analytical approach obtained this result.

$$
\begin{aligned}
Rt_0(162.7) &= 360 \\
Rt_1(162.7) &= Rt_0(162.7) - ITime_{01} + MHWT_{01} \\
Rt_1(162.7) &= 360 - 300 + 250 = 310 \\
Rt_2(162.7) &= Rt_1(162.7) - Dist_{1,2}(p) + MHWT_{1,2} \\
Dist_{1,2}(162.7) &= ITime_{1,2} + (p \times \tan(\alpha_2))/rel_2 - (p \times \tan(\alpha_1))/rel_1 \\
Dist_{1,2}(162.7) &= 2721.13 \\
Rt_2(162.7) &= max\{310 - 2721.13 + 250 \, , \, 0\} = 0 \\
RtotalHWO &= 360 + 310 + 0 = 670
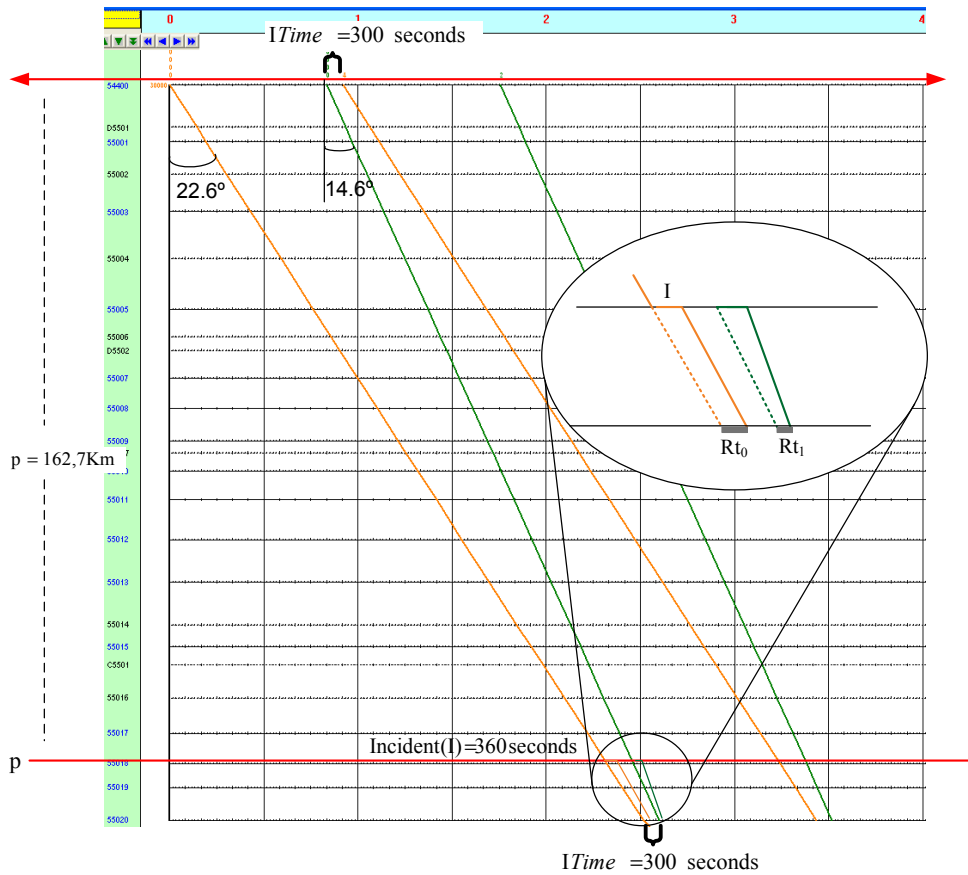\end{aligned}
$$

$$(17)$$

Figure 37: Heterogeneous Railway Line used by the simulation and analytical approaches

The values rel_1 and rel_2 indicate the relation between the expression $p \times \tan \alpha$ and the time units (in this case, seconds). The value assigned depends on the average speed assigned to each train.

When we use a simulation tool with the railway line shown in Figure 37, we obtain a total delay for these schedule in point (p) equal to 720 seconds. This difference is due to the fact that the simulation tool computes the total delay for each train at point (p) and not at destination. If we desire to study the total delay to destination by the simulator tool, then it reduces the total delay to 670 seconds due to the fact that the second train recovers the buffer time of 50 seconds (ITime-MHWT), so the total delay is 720-50=670 seconds.

Other Simulation Results

We provide results that have been obtained after simulating disruptions for different timetables. We have used the same railway line, which is composed of 17 stations with double track and automatic blocking in all its track sections. We

have followed the process defined in subsection 7.2: (i) define original timetable, (ii) insert disruptions, (iii) repair timetable (Re-scheduling), and (iv) get values for response variables.

Train delay is one of these result variables, which is defined according to expression 18, where:

- $arr_{n_i}^i$ is the arrival time of train i to its final station l_{n_i}, after the re-scheduling process is finished,

- $\widehat{arr}_{n_i}^i$ is the arrival time of train i to its final station l_{n_i} according to the original timetable.

$$\text{if } arr_{n_i}^i - \widehat{arr}_{n_i}^i < 0 \text{ then } delay_i = 0$$
$$\text{else } delay_i = arr_{n_i}^i - \widehat{arr}_{n_i}^i \tag{18}$$

We have considered two types of trains in each original timetable, called *Fast* and *Slow*, respectively. All trains of *Fast* type travel in each track section 1.5 times faster than trains of *Slow* type, in the same track section. The ordering specified for the initial departure time of trains is made in such a way that trains are distributed uniformly in the scheduling horizon with respect to trains of the same type.

The tests have been divided in two parts. In the first part, we show how the sum of delays changes when the number of trains is increased in the original timetable. In the second part, we show how the sum of delays changes when the primary delay produced by an incident is increased, considering the same original timetable.

a) Sum of delays vs. Capacity (Number of Trains)

The simulator performs an iterative process. In each iteration, it obtains the sum of delays corresponding to a given number of trains that make up the original planning. The number of trains is increased in each iteration by the lowest number that maintains the proportion among the types of trains. We have established the minimum and maximum number of trains that are considered in the first and last iterations, respectively.

We simulate incidents at different times in the scheduling horizon, and we consider that each one causes a primary delay equal to two minutes.

Figure 38 shows the percentage of the total secondary delays with respect to the primary delay according to the number of trains for three cases. These are different due to the type ratio established in each case.

- Case 1: The same number of *Fast* and *Slow* trains.

- Case 2: The number of *slow* trains is two times the number of *fast* trains.

- Case 3: The number of *fast* trains is two times the number of *slow* trains.
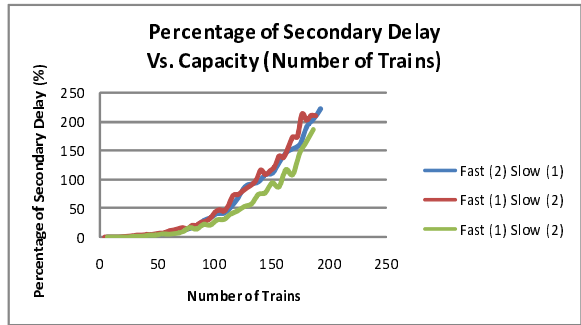
47

Figure 38: Percentage of total delays with respect to primary delay vs. Number of trains in the original timetable (capacity utilization)
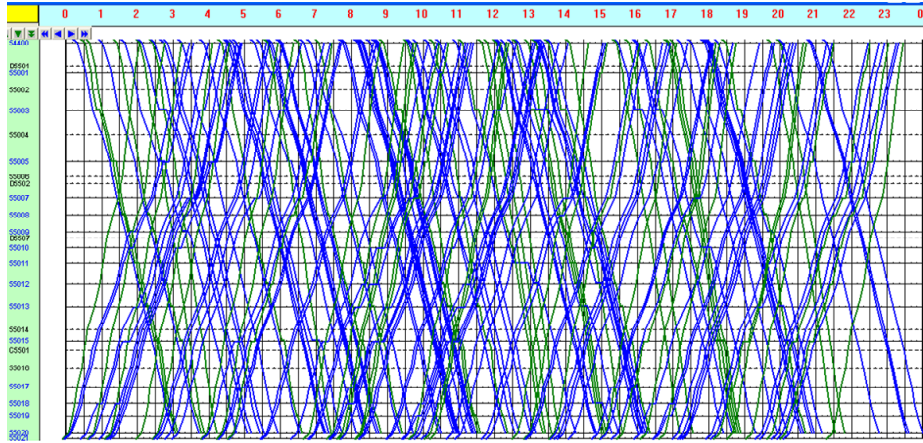

Figure 39: Original timetable with 192 trains

Figure 39 shows the original timetable used in the disruption scan process when the number of trains is equal to 192. If this timetable suffers an incident whose primary delay is equal to six minutes, then we could expect a sum of delays equal to 780 seconds, according to the results shown in Figure 38.

b) Sum of delays vs. level of Primary delay

In this case, we consider only one original timetable, and we show how changes the sum of delays when the primary delay produced by a disruption is increased. In this case, the original timetable is composed of 180 trains in the railway line. We employ the same re-scheduling procedure, but this time, we consider different primary delays. The results are shown in Figure 40, which shows that as the level of primary delay increases, the sum of delays also increases. However there exist a point in which the primary delay is big enough that the secondary delays are not affected in the same proportion so that the sum of delays decrease. This is the case when the disrupted train is overtaken
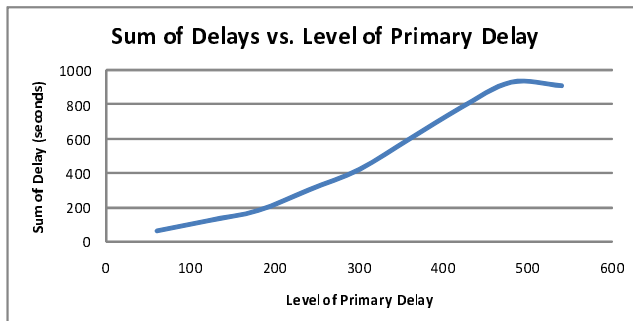
(see Figure 34).

Figure 40: Sum of delays vs. level of primary delay

9. Conclusions and further work

The robustness of a solution in a real scheduling problem is becoming more and more valuable. In railway scheduling, the robustness of a timetable is an indicative measure of how good a timetable is. It depends on several factors. From an initial point of view, it mainly depends on train heterogeneity, the capacity used in the infrastructure, and the average speed of running trains. For instance, there can be: (i) a high stability in metro-trains with homogeneous traffic, low speed and a large number of trains per time unit; or (ii) intercity train services, with a higher average speed and heterogeneous traffic so that the same reliability of the service cannot be attained even if the number of trains per time unit is reduced considerably, [28]. Timetable robustness also depends on timetable optimality, which is a basic criterion in the timetable construction process as stated above. However, we can also work on many other factors in the timetable construction process in order to obtain robust timetables. These robust timetables allow the railway operators to cope with unexpected disruptions that normally occur on a daily basis. A timetable can have the characteristic whereby delayed trains lead to considerable knock-on effects, whereas another configuration of the timetable may be able to absorb such effects more readily. In this paper, we have presented the main parameters that can be directly related to robustness. Railway operators (based on their expertise) help us to determine the most important parameters and they have been included in our analytical formulas to compare and measure robustness. According to the several real world test cases performed, the results obtained by the disruption simulator process were very close to the expected average delays obtained in real-world timetables under real world circumstances. Results obtained by the analytical model were also close to those obtained by the Disruption Simulator on the same real world timetables. For this reason, the disruption simulator verifies the evaluations obtained by the analytical model.

49

The described re-scheduling process is complex due to the real-world train and infrastructure data and constraints it manages. An exhaustive sampling of disruptions on complex timetables is a time consuming task. Performing the analytical model on timetables is very efficient and only requires a few milliseconds of computational time.

Therefore, the proposed (computationally more efficient) analytical model becomes an efficient and useful alternative for light robustness assessment in single-line railway timetables. This is a main conclusion of this work. For these reasons, we will use the analytical model for generating more robust timetables by introducing this analytical robustness assessment as a new evaluation criterion in timetable generation processes.

In further work, we will also use the results obtained by the analytical model to deduce new conclusions about robustness, its properties and relationships. Furthermore, we will also study some other features and statistical relations when measuring robustness. To this end, we must analyze the historical data about disruptions (given by railway operators) to obtain useful statistics to improve our formulas and a better robustness assessment and timetable generation.

References

[1] Abril M, Barber F, Ingolotti L, Salido M, Tormos P, Lova A (2008) An assessment of railway capacity. Transportation Research - Part E 44:774–806

[2] Aloulou M, Portmann M (2003) An efficient proactive reactive scheduling approach to hedge against shop floor disturbances. In Procof the 1st Int Conf on Scheduling, MISTA 2003 pp 337–362

[3] Bergmark R (1996) Computers in Railways V, WIT Press, chap Railroad capacity and traffic analysis using SIMON, pp 183–191

[4] Bertsimas D, Sim M (2004) The price of robustness. Operations Research 52:35–53

[5] Caprara A, Monaci M, Toth P, Guida L (2006) A lagrangian heuristic algorithm for a real-world train timetabling problem. Discrete Applied Mathematics 154:738–753

[6] Carey M, Carville S (2000) Exact heuristic measures of schedule reliability. Journal of Operational Research Society 51:666–682

[7] Carey M, Carville S (2004) esting schedule performance and reliability for train stations. International Transactions in Operational Research 11:382–394

[8] Cesta A, Oddi A, Smith S (1998) Profile based algorithms to solve multiple capacitated metric scheduling problems. In Proc of the 4th Int Conf on Artificial Intelligence Planning Systems AIPS-98

[9] D'Angelo, Di-Stefano G, G Navarra A, Pinotti C (2011) Recoverable robust timetables: An algorithmic approach on trees. IEEE Transations on Computers 60:433–446

[10] Davenport A, Gefflot C, Beck C (2001) Slack-based techniques for robust schedules. Proceedings of 6th European Conference on Planning

[11] Drummond M, Bresina J, Swanson K (1994) Justin-case scheduling. Proceedings of the 12th National Conference on Artificial Intelligence, pp 1098–1104

[12] Fischetti M, Salvagnin D, Zanette A (2009) Fast approaches to improve the robustness of a railway timetable. Transportation Science 43:321–335

[13] Ginsberg M, Parkes A, Roy A (1998) Supermodels and robustness. Proceedings of AAAI-98/IAAI-98 pp 334–339

[14] Goverde R (2005) Punctuality of railway operations and timetable stability analysis. Phd Thesis, Delft University of Technology, TRAIL Research School, Delft, Netherlands

[15] Goverde R (2007) Railway timetable stability analysis using max-plus system theory. Transportation Research Part B: Methodological 41(2):179 – 201

[16] Goverde R, Odijk M (2002) Computers in Railways VIII, WIT Press, chap Performance evaluation of network timetables using PETER., pp 731–740

[17] Hebrard E, Nich B, Walsh T (2004) Robust solutions for constraint satisfaction and optimization. Proc of the 16th European Conf on AI (ECAI 2004) pp 186–190

[18] Herroelen W, Leus R (2004) Robust and reactive project scheduling: a review and classification of procedures. International Journal of Production Research, 42(8):1599–1620

[19] Ho T, Tsang C, Ip K, Kwan K (2012) Train service timetabling in railway open markets by particle swarm optimisation. Expert Systems with Applications 39:861–868

[20] Hooghiemstra J, Tunisse M (1998) The use of simulation in the planning of the dutch railway services. In Proceedings of the 1998 Winter Simulation Conference

[21] Huisman T, Boucherie R (2001) Running times on railway sections with heterogeneous train traffic. Transportation Research Part B 35:271–292

[22] Jen E (2003) Stable or robust? what's the difference? Complexity, 8(3):12–18

[23] Kettner M, Sewcyk B, Eickmann C (2003) Integrating microscopic and macroscopic models for railway network evaluation. European Transport Conference, pp 18.–10.10., Straburg,

[24] Kroon L, Maroti G (2008) Robust train routing. Tech. rep., ARRIVAL Project

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

[26] Leon WS V and, Storer R (2004) Robustness measures and robust scheduling for job shops. IIE Transactions, 26:667–682

[27] Liebchen C, Schachtebeck M, Schöbel A, Stiller S, Prigge A (2010) Computing delay resistant railway timetables. Comput Oper Res 37:857–868

[28] Mattson L (2004) Train service reliability: A survey of methods for deriving relationship for train delays. http://usersduse/ jen/Seminarieuppsatser/Forsening-tag-Mattssonpdf

[29] Middelkoop D, Bouwman M (2001) Large scale train network simulations. In Proceedings of the 2001 Winter Simulation Conference pp 1042–1047

[30] Mohammad T Isaai MTHRA Aram Kanani (2011) Intelligent timetable evaluation using fuzzy ahp. Expert Systems with Applications 38:3718–3723

[31] Pachl J (2004) Railway operation and control. VTD Rail Publishing, Mountlake Terrace

[32] Policella N (2005) Scheduling with uncertainty: A proactive approach using partial order schedules. PhD Thesis, Universita degli Studi di Roma: La Sapienza

[33] Policella N, Smith S, Cesta A, Oddi A (2003) Steps toward computing flexible schedules. Proc of Online'03 Constraint Solving: Handling Change and Uncertainty

[34] Policella N, Cesta A, Oddi A, Smith S (2007) From precedence constraint posting to partial order schedules. a csp approach to robust scheduling. AI Communications 20:163–180

[35] Policella N, Cesta A, Oddi A, Smith S (2009) Solve-and-robustify. synthesizing partial order schedules by chaining. In Journal of Scheduling 12:299–314

[36] Saa R, Garcia A, Gomez C, Carretero J, Garcia-Carballeira F (2012) An ontology-driven decision support system for high-performance and cost-optimized design of complex railway portal frames. Expert Systems with Applications 39:8784–8792

[37] Sevaux M, Srensen K (2002) A genetic algorithm for robust schedules in a just-in-time environment. Technical Report LAMIH/SP-2003-1

[38] Tomii N (2005) Robustness indices for train rescheduling. 1st International Seminar on Railway Operations Modelling and Analysis

[39] UIC405-OR (1996) Links between railway infrastructure capacity and the quality of operations. International Union of Railways,

[40] UIC406 (2004) Uic leaflet 406, capacity. International Union of Railways,

[41] Vromans J, Dekker R, Kroon L (2006) Reliability and heterogeneity of railway services. European Journal of Operational Research 172:647–665

[42] Wahlborg M (1996) Computers in Railways V, WIT Press, chap Simulation models: important aids for Banverkets planning process., pp 175–181