Train platforming problem in busy and complex railway stations

Lijie Bai

N^o *d'ordre*: | 2 | 7 | 6 |

ÉCOLE CENTRALE DE LILLE

THÈSE

présentée en vue d'obtenir le grade de

DOCTEUR

en

Spécialité : Génie Industriel

par

Lijie BAI

DOCTORAT DELIVRE PAR L'ECOLE CENTRALE DE LILLE

Titre de la thèse :

Train platforming problem in busy and complex railway stations
Ordonnancement des trains dans une gare complexe et à forte densité de circulation

Soutenue le 28 Août 2015 devant le jury d'examen :

Président	M. Frédéric SEMET	Professeur, CRIStAL, École Centrale de Lille
Rapporteur	M. Saïd HANAFI	Professeur, LAMIH-UVHC, Université de Valenciennes
Rapporteur	M. Joaquin RODRIGUEZ	Directeur de Recherche, IFSTTAR
Rapporteur	M. Louis-Martin ROUSSEAU	Professeur, CIRRELT, École Polytechnique de Montréal
Examinateur	M. Frédéric SEMET	Professeur, CRIStAL, École Centrale de Lille
Examinateur	M. Ouajdi KORBAA	Professeur, ISITCom-Sousse-Tunisie
Invité	M. Martin PRIETO	Cadre au Pôle Pilotage SNCF - DIRECTION DE LA TRACTION
Directeur de thèse	M. Emmanuel CASTELAIN	Maître de conférences, HDR École Centrale de Lille
Co-directeur de thèse	M. Thomas BOURDEAUD'HUY	Maître de conférences, CRIStAL, École Centrale de Lille

Thèse préparée dans le Laboratoire de Modélisation et de Management des Organisations EA 4344
École Doctorale SPI 287 (EC Paris, EC Lille)
PRES Université lille Nord-de-France

I would like to dedicate this thesis to my loving parents, to my loving husband and to myself.

Acknowledgements

My first encounter with railway management goes back to my days struggling to complete my last year of French engineer program, option “Transport Systems & Logistics” at Ecole Centrale de lille. In this year, I was given a chance to understand the complex industry and transport systems from both of practical and academic disciplines. On one hand, many professional persons are invited from companies or national department of transportation to introduce the complex supply chain and transport systems in the real life and share their precious experiences. On the other hand, “Operations research” held by Emmanuel Castelain gave me an academic tool to translate complex systems into mathematical models, often easy to state but difficult to solve.

My interests in railway operations research started with the project “IMPACT ” during these last years of French engineer training at Ecole Centrale de Lille. I got this chance from Professors Emmanuel Castelain and Thomas Bourdeaud’huy. They heard about my strong curiosity on doing traffic optimization and proposed me to do a project on platforming trains in railway stations. I therefore spent three months to get familiar with railway operations and traffic management problems, and to collect and elaborate data for my report. This was mainly possible thanks to the effort done by the SNCF project liaison Martin Prieto as information provider.

At the end of project, the problem was formalized but still not solved properly. Well, I loved the research experience and I got really impressed by difficulty of the platforming trains problem. It was suddenly clear that I need to go further on this subject and find a practical solution. So I continued my studies at Ecole Centrale de Lille as a PhD student. Emmanuel Castelain has been my supervisor during the entire working period. His interests and trust on my research findings encouraged me to go further. His constructive criticism and rich experience guided me to explore the fascinating world of Operations Research. I also wish to express my gratefulness to Prof. Frederic Semet and Prof. Besoa Rabenasolo. They gave me very valuable suggestions when I was stuck by problems.

A special thank goes to Thomas Bourdeaud'huy. During development of the plaforming decision support tool, he helped me a lot to extend my knowledge in computer science.

On a regular basis, I had meetings with SNCF project liaison in order to direct my research to address problems of actual practical interest. His work has been really precious also when dealing with the data collection and elaboration for the test cases studied in this thesis.

I am grateful to the members of my evaluation committee, as I had the chance to improve my thesis with their helpful comments and suggestions. I would like to express my sincere gratitude to research director of IFSTTAR Joaquin Rodriguez, Prof. Saïd Hanafi and Prof. Louis-martin Rousseau who have kindly accepted the invitation to be the reviewers of my thesis. My heartfelt thanks go to Prof. Frederic Semet for his kind acceptance to be the president of my PhD committee.

I wish to express my special appreciation to Prof. Hélène CATSIAPIS, my French teacher. She has not only help my language development and vocabulary, but also given me a chance to understand charming French culture and history through the courses and trips, which have greatly enriched my experiences in France.

Of course, a big thank is for my kind colleagues, technical staff and secretaries of laboratory LM2O and laboratory CRIStAL during these years. A special thank goes to secretary of research sector Vanessa Fleury who helps me a lot with all my administrative procedures.

Finally, I would like to dedicate this thesis to my parents Mr. Guolin Bai and M. Ailing Yang and my husband Mr. Ziyao Dong for their love, confidence and support throughout my studies and life.

Lijie Bai

Abstract

This thesis focuses on the train platforming problem within busy and complex railway stations and aims to develop a computerized dispatching support tool for railway station dispatchers to generate a full-day conflict-free timetable. The management of rail traffic in stations requires careful scheduling to fit to the existing infrastructure, while avoiding conflicts between large numbers of trains and satisfying safety or business policy and objectives. An overview of railway scheduling and routing algorithms and existing computer-based systems is provided. However, the state-of-the-art in optimal train dispatching algorithms can handle only low traffic densities and a short time horizon within a reasonable amount of computation time.

Based on Operations Research techniques and professional railway expertise, we design a generalized mathematical model to formalize the train platforming problem including topology of railway station, trains' activities, dispatching constraints and objectives. As a large-scale problem, full-day platforming problem is decomposed into tractable sub-problems in time order by cumulative sliding window algorithm. Each sub-problem is solved by branch-and-bound algorithm implemented in CPLEX. To accelerate calculation process of sub-problems, tri-level optimization model is designed to provide a local optimal solution in a rather short time. This local optimum is provided to branch-and bound algorithm as an initial solution.

This system is able to verify the feasibility of tentative timetable given to railway station. Trains with unsolvable conflicts will return to their original activity managers with suggestions for the modification of arrival and departure times. Time deviations of commercial trains' activities are minimized to reduce the delay propagation within the whole railway network.

Résumé

Cette thèse porte sur l'ordonnancement des trains dans les gares complexes et à forte densité de circulation. L'objectif final est de réaliser un outil pour aider les managers de la gare à générer un tableau des horaires sans conflits pour une journée complète. La gestion des circulations ferroviaires dans la gare nécessite un ordonnancement précis pour s'adapter aux ressources limitées en évitant les conflits entre les trains tout en satisfaisant les objectifs et les politiques à la fois économique et de sécurité. Un aperçu des méthodes de planification et des systèmes automatisés est donné. Dans la littérature, les méthodes peuvent seulement résoudre les problèmes de ce type à faible densité de circulation et avec un horizon de temps court dans un temps de calcul raisonnable.

En s'inspirant des méthodes appliquées en Recherche Opérationnelle et des pratiques professionnelles, un modèle mathématique applicable à toutes les gares a été construit pour formaliser le problème de l'ordonnancement des trains tenant compte de la topologie de la gare, des activités des trains, des contraintes de planification et des objectifs. Comme c'est un problème de grande taille, l'ordonnancement des trains dans une journée est décomposé en sous-problèmes résolus séquentiellement par un algorithme à fenêtre glissante. Chaque sous-problème est résolu par le branch-and-bound de CPLEX. Afin d'accélérer le calcul des sous-problèmes, une méthode d'optimisation à 3 niveaux est proposée pour offrir une solution optimale locale dans un temps de calcul raisonnablement court. Cette solution est donnée à un branch-and-bound comme solution initiale.

Ce système consiste à vérifier la faisabilité des horaires donnés à la gare. Les trains à l'origine des conflits non résolus sont identifiés et les modifications d'horaires commerciaux proposées. Les détentes horaires des trains commerciaux sont minimisées pour diminuer la propagation des retards dans le réseau ferroviaire.

Table of contents

List of figures

List of tables

CHAPTER 1

Introduction

1.1 Railway operating practices

The French railway network is structured around a group of companies and bodies that cater to the various organisational requirements: administration, management, operation, renovation, safety, development, etc. As the network owner and manager, Réseau Ferré de France plays a pivotal role among these organisations.

The liberalisation of the railway market in terms of freight and passenger transportation has altered how the system is organised. To ensure a wider range of services, new organisations tasked with overseeing safety and the correct functioning of competition have emerged as part of a rationale of neutrality and transparency.

- Structure of the network

  The government defines the network's general orientations, makes decisions on major works, participates in the financing of projects and the renovation of the network, etc.

The regions are taking on a growing number of responsibilities in the area of public transport. On 1st January 2002, they became regional transport organisation authorities. They make a significant contribution on transport policies definition and network development financing, particularly under State/Region Strategic Plans (CPER).

- Network operation and management

  Réseau Ferré de France (RFF) plays a key role within the railway system. As the owner and manager of the French railway network, it decides what targets to apply in terms of traffic management and how the network is run and maintained. Its main commercial activity consists of selling slots, i.e. allotting time periods during which trains can transit from one point to another.

  The Department of Railway Circulation (DCF) has since 1 January 2010 been responsible for traffic and circulation management on behalf of Réseau Ferré de France. This independent entity, which forms part of SNCF, guarantees fair and completely transparent access to the network for all railway companies.

  Railway companies are responsible for the transportation of passengers and goods. They pay fees to Réseau Ferré de France in order to be able to run their trains on the network. Since the network was opened up to competition, new passenger or freight transportation companies have been given approval to operate on the French railway network. RFF, the network manager, provides them with the slots and infrastructures that allow them to operate, in the same way as the SNCF.

Seven million path reservation requests are received every year, the equivalent of 20,000 per day. It is Réseau Ferré de France's job to facilitate access to the network and to find the optimum balance between:

- Ensuring path profitability,
- Meeting the transport demand for each line (major lines, regional lines, freight),
- Arranging timetable windows for maintenance.

The tactical planning process of timetabling, as shown in Figure 1.1, already starts a year before a new annual timetable becomes operational. This *"tentative"* timetable of commercial movements (circulation with passengers or goods) within rail networks generated by RFF is delivered to railway stations. Different types of trains have different preparation times before

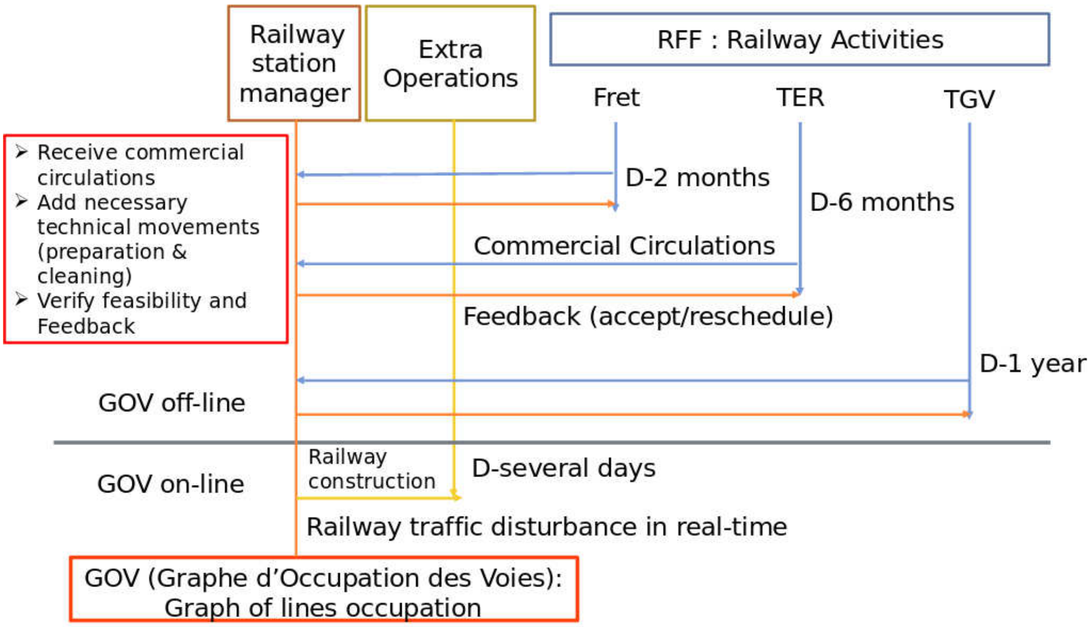


Fig. 1.1 The tactical planning process of timetable

their day of circulation. TGV is delivered to railway stations one year in advance, $D-1$ year. TER is delivered six months in advance, $D-6$ months. Preparation time of Freight is two months, $D-2$ months. As a rough capacity estimation of railway stations is used to produce the tentative timetable, traffic feasibility within railway station areas is not guaranteed. As a result, the tentative timetable including different types of trains (TGV, TER, Freight) arrives at railway station management department who arranges rail traffic within railway station areas. In addition, the timetable prescribes the infrastructure allocation in time and space for all regular passenger trains including extra traffic operations for short-term capacity requests which may be allocated a few days before operation, $D-$ several days. Verification of tentative timetable at level of railway stations is the problem studied in this dissertation. With addition of necessary technical movements, infeasible commercial movements are rejected and returned to RFF with a proper proposition of arrival or departure time. After information exchanges between stations and RFF, a feasible timetable is produced and called *off-line timetable* which is followed by real-time measures to manage disturbances. The train paths are designed at a precision of minutes and contain the routes through stations, platform track usage, and train sequences at open tracks. Rolling stock circulations and crew schedules are worked out in advance according to the timetable.

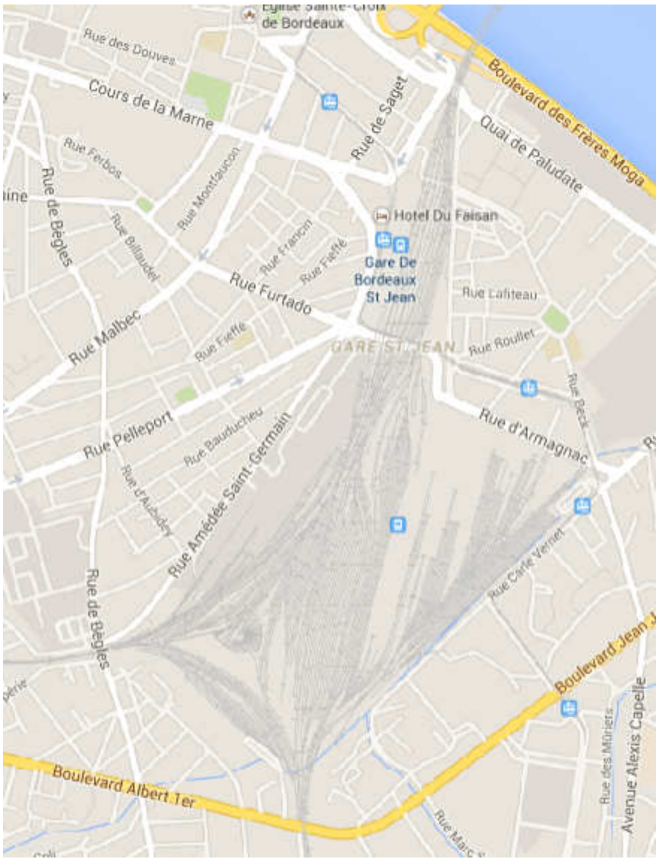


Fig. 1.2 Google Map of Railway station Bordeaux St Jean

Southern part of Bordeaux-Saint-Jean or Bordeaux-Midi, as a complex and busy railway station area, is studied here. To understand the complexity of this area, the Google map of Bordeaux St Jean is given in Figure 1.2. The massive paths criss-cross on southern part is our object of study. It is the southern terminus of the Paris–Bordeaux railway, and the western terminus of the Chemins de fer du Midi main line from Toulouse. The station is the main railway interchange in Aquitaine and links Bordeaux to Paris, Sète, Toulouse Matabiau and Spain.

Lines Occupation graph shown in Figure 1.3 is applied to arrange local rail traffic by railway station manager. At present, off-line timetabling processes is generated by hand, and planning simulation tool is used to represent the time-space timetable. Time window of one day (24 hours) is divided into 5 pages, and each page represents train circulations during 4 hours, e.g. 4-8h, 8-12h, 12-16h, 16-20h and 20-24h. No circulation is planned during 0-4h. Horizontal axis represents time span. Vertical axis represents the list of platforms in railway station. Trains are identified by number of coach. Positions of trains in chart

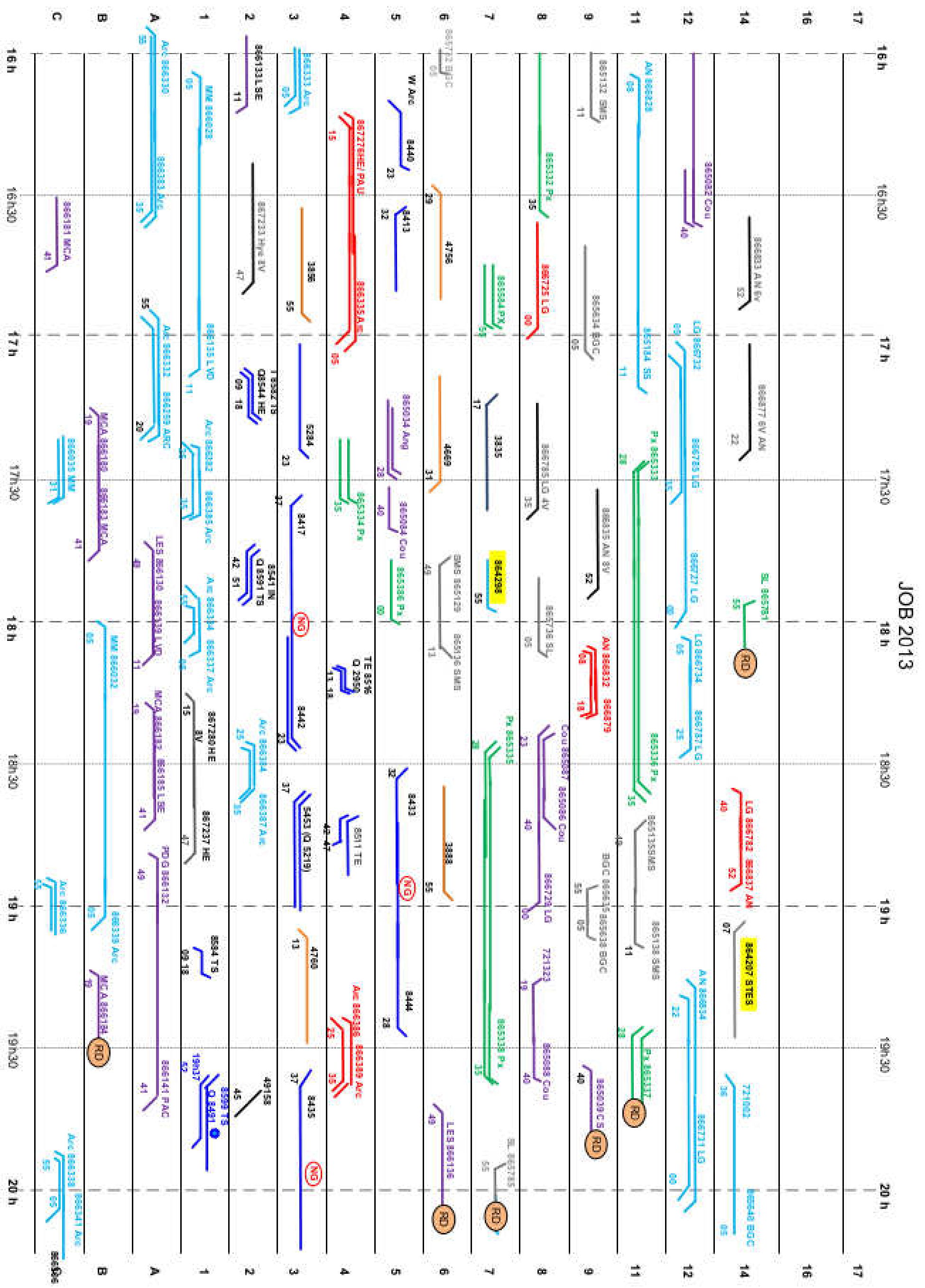


Fig. 1.3 Lines Occupation graph (Graphe d'Occupation des Voies GOV)

represent the chosen platform, departure time and arrival time. Standstill on platform of trains are represented by the length of a bar. Movement direction of trains are indicated by beginning and ending of bars. If a bar begins with "\", the train comes from North. If a bar begins with “/”, the train comes from South. If a bar ends with “\”, the train leaves for South. If a bar ends with “/”, the train leaves for North. Two trains coupled are represented by two bars in parallel. Conflicts on platforms can be eliminated directly by GOV, but conflicts on paths invisible on charts are eliminated by planners’ experience. To guarantee the safety of circulations, planners are used to suppose a cycle time within full-day timetable, for example one hour. Within one hour, paths and platforms are reserved by certain number of trains from different directions, for example 14 TGV, 5 TER and 2 Freight. In fact, the circulations are not cyclic. Maybe there are 18 TGV, 7 TER and 0 Freight within one hour. Instead of releasing the free resources, Freight still reserve their resources. In that case, the precious rail resources are wasted and not fully exploited. To meet the growing demand in rail traffic, railway networks are operated nearly at capacity margins. As bottleneck of networks, railway station demands eagerly a computer-based system to exploit the rail capacity.

1.2 Computer-based systems

The (off-line) design of rail network timetables is a complicated and recurrent problem, that typically requires many months. Stations are often the focal points of a railway network. Lines come together, trains are arriving and departing, passengers are boarding the trains, and last but not least, trains are parked, cleaned and maintained.

Stations can also be bottlenecks within an existing or future network. Therefore, timetable planners need to examine the track usage and platform occupation for any timetable they compile. Sophisticated decision support tools and macroscopic simulation tools are highly required in recent years to optimize the use of infrastructure capacity and to distribute suitable running time supplements and time margins that can absorb minor delays occurring in practice. There are two kinds of decision support tools corresponding to railway stations:

- Assessment of railway station capacity

- Dispatching trains within railway stations

1.2.1 Assessment of railway station capacity

The goal of capacity analysis is to determine the maximum number of trains that would be able to operate on a given railway infrastructure, during a specific time interval, given the operational conditions.

Numerous approaches and tools have been developed to address this problem; they are based on traffic patterns Forsgren (2003), single-track analytical models Petersen (1974), or algebraic approaches Egmond (1999). Several international companies are also working on similar computer-based systems:

• DEMIURGE designed by SNCF and Eurodecision (2004) is a software program designed to assist in making rail network capacity studies. This software can evaluate a network's capacity to absorb additional traffic, to locate bottlenecks, to assist in making decisions about infrastructure investments, to optimize current and future timetables, and to calculate the residual capacity of a timetable.

• CMS (AEA Technology Rail) provides a system to plan the effective utilization of the railway capacity. It offers an easy "what-if" scenario evaluation, automatic generation of timetables, simulation of operations to predict performance and identify remedies, identification of capacity available for sale, and usage forecasts based on improved timetables. However, CMS needs to be calibrated using updated punctuality data to ensure that its predictions are valid.

• RAILCAP (Stratec) measures how much of the available capacity is used by a given operation program in a straightforward way, and it offers a very detailed analysis of bottlenecks. However, it has one major disadvantage since the modelling requires a great deal of effort. RAILCAP requires detailed descriptions of the tracks, switches, crosses, signals and speed limits.

• VIRIATO (SMA and Partner) is mainly used for adapting infrastructure to future service concepts and coordinating several operators or products that share the same infrastructure. It allows the user to determine the amount of saturation of a specified line. It compresses a given timetable, and determines the saturation rate of a line or a part of a line as a percentage. This method leads to varying results for the same line, depending on the length of the section under consideration.

• CAPRES designed by Lucchini and Curchod (2001) is a model for the elaboration and saturation of timetable variants. Through the use of iterations, this model determines all available extra train paths, given all the constraints and interconnections between lines. A disadvantage of this model is that the traditional network and operational data have to be completed with the information about where, when and how the network capacity must be used.

• FASTTRACK II (Multimodal Applied Systems) is a computer-based train dispatching and meet-pass model that is capable of producing a feasible train dispatching plan for a user-selected corridor, given a set of proposed train schedules and a corridor's track configuration. It can be used to examine the feasibility of a set of proposed train schedules, test the impact of proposed changes in operating policies on train service, and measure both the theoretical and practical line capacity.

• MOM system designed by Barber et al. (2006) is a highly functional tool that helps railway managers to provide efficient and reactive management of railway infrastructures. The MOM system can generate optimized railway schedules both off-line and on-line (when disruptions occur). It also provides information on railway network capacity and on timetable robustness, helping managers to make better decisions. This module provides several analytical and empirical methods that can be used to obtain conclusions about the capacity of railway networks and that support the process of adapting the railway infrastructure to traffic needs. The MOM system project has been developed according to the requirements of the Spanish Administration of Railway Infrastructure, ADIF.

• AFAIG is a comprehensive software package designed by LITEP of École Polytechnique Fédérale de Lausanne for the planning of layout and operational plans of major railway passenger stations. AFAIG uses a database describing infrastructure, rolling stock, operation rules and timetables. In a conversational mode, AFAIG lets the scheduler place train movements, after which it calculates the occupation time of the successive sections for each itinerary, detects and measures the conflicts between movements, verifies that all the operational constraints are respected and takes care of tedious drafting tasks. Relieved of the operations that can be automated, the planner for a major railway station can devote his time effectively to the tasks of design, analysis and multi-criteria evaluation. AFAIG has been implemented in the timetable planning division of the main stations of the Swiss Federal Railway (SBB), in order to help planners to manage routes, platform tracks and connections according to the "Rail 2000" strategy.

• QuaiOPS is implemented in the timetable planning division of major stations of the French railway networks (RFF). Two screens are used to display separately occupation of platforms and local networks in stations. All trains are treated at the same time and allocated automatically a platform. Another software: Disco, is applied to verify the feasibility of timetable in stations. Incompatibility is marked in red. In order to generate a feasible timetable, timetable manager changes the resources planning firstly from the train with conflicts most concentrated.

Railway capacity is not static. It is extremely dependent on how it is used. The physical and dynamic variability of train characteristics makes capacity dependent on the particular mix of trains and the order in which they run on the line. Furthermore, it varies with changes in infrastructure and operating conditions. Finally, railway station capacity estimated need to be realized by a fine dispatching program with specific routing and scheduling decisions. For rail network within complex and busy railway stations, as Bordeaux St Jean shown in Figure 1.2, a proper decision is still not easy to be made.

1.2.2 Dispatching trains within railway stations

Barber et al. (2007) investigate and survey 21 automated systems for railway management. Main characteristics and functions of these tools are resumed in Figure 1.4. But most of tools focus on network scheduling and routing problems with a rough estimation of railway station capacity.

- Simulation: The tool provides the function to emulate and graphically display real train operations in order to generate simulation models of railway networks where finer analysis of the timetable can be assessed.
- Timetable Optimization: The tool provides optimization algorithms which schedules train movements and generate a timetable in accordance with an objective function, schedule priorities and network constraints.
- Timetable Manager: The tool provides the function to edit train timetables in graphic or tabulate way.
- Capacity Analysis: The tool can be used to assess railway capacity.
- Infrastructure Manager: The tool provides the function to model the existing infrastructure and to build up different infrastructure variants.

	Simulation	Timetable Optimization	Timetable Manager	Capacity Analysis	Infrastructure Manager	Evaluation Manager	On Line	Robustness Analysis	Station Manager
RAILSYS	✓	✗	✓	User-Deduct.	Very Detailed	✓	✗	User-Deduct.	✗
DONS	✓	✓	✗	User-Deduct.	✓	✓	✗	User-Deduct.	✓
PETER	✗	✗	✓	Very Limited	✓	✓	✓	✓	✗
ROMAN	✓	✗	✓	✗	✓	✓	✗	User-Deduct.	✗
VIRIATO	✗	✗	✓	Limited	✓	✓	✗	✗	✓
CAPRES	✗	✓	✗	✓	✓	✓	✗	✗	✓
FASTA	✓	✗	✓	✗	✓	✓	✗	User-Deduct.	✗
AFAIG	✗	✗	✗	✗	✓	✓	✗	✗	✓
DEMIURGE	✗	✓	✗	✓	✓	✓	✗	✗	✓
SISYFE	✓	✗	✓	✗	✓	✓	✗	User-Deduct.	✗
RAILCAP	✓	✗	✗	✓	Detailed	✓	✗	✗	✗
OPENTRACK	✓	✗	✓	User-Deduct.	✓	✓	✗	User-Deduct.	✗
OPENTIMETABLE	✗	✗	✓	✗	✗	✓	Future	User-Deduct.	✗
VISION	✓	✗	✓	User-Deduct.	✓	✓	✗	User-Deduct.	✗
CMS	✓	✓	✓	✓	✓	✓	✗	✓	✓
TPS	✓	✗	✓	✗	✓	✓	✗	User-Deduct.	✗
MULTIRAIL	✓	✓	✓	User-Deduct.	✓	✓	✗	User-Deduct.	✓
MOM-MALLAS	✗	✓	✓	✓	✓	✓	Limited	User-Deduct.	Limited

Fig. 1.4 Survey of automated systems for railway management: Barber et al. (2007)

- Evaluation Manager: The tool permits a graphical or tabular visualization of the results (timetables) and a more in-depth analysis of them.

- On Line: The tool carries out studies of the railway network in real-time.
- Robustness Analysis: The tool can help with the analysis of the stability of a timetable.
- Station Manager: The tool assists the planners in solving the problem of routing trains through a railway station.

Among these automated systems, the tool containing the most complete dispatching functions within railway station is DONS.

The project DONS (Design Of Network Schedules) was initiated by Railned (Railned has the task, amongst others, to advise the Dutch Ministry of Traffic with respect to the capacity of the Dutch railway infrastructure that will be necessary in the future) and Netherlands Railways. The aim of this project is to develop a DSS (Decision Support System), also called DONS, that will assist the planners of Railned and Netherlands Railways in generating timetables. DONS contains two complementary optimization modules which are linked together by a database module and a graphical user-system-interface. The two optimization modules correspond to the two steps of the timetable generation process.

• CADANS The first optimization module, called CADANS, assists the planners in generating a tentative timetable based on the constraints deduced from the rough layout of the railway network between the stations, the line system, and the connection requirements at the railway stations. The timetable determined by CADANS is cyclic with a cycle length of one hour. CADANS is being developed by Schrijver and Steenbeek (1994).

• STATIONS The second optimization module, called STATIONS, assists the planners in solving the problem of routing trains through a railway station. STATIONS considers the stations one by one. The output of STATIONS is a detailed assignment of trains to routes and platforms within the observed station. Such an assignment serves as a local feasibility check for the tentative timetable generated by CADANS. If not all trains can be routed through the station, then STATIONS also points at the blocking trains. STATIONS is being developed by Zwaneveld et al. (2001). STATIONS always considers one railway station at a time. The problem that is solved for this railway station can be stated as follows: Given the detailed layout of the involved railway station, and given the scheduled arrival and departure times of a set of trains, STATIONS aims at routing as many trains as possible through the station, taking into account the capacity of the station, the safety system, and several service requirements. The routing of the trains should minimize the number of shunting operations, and it should maximize the total preference for the platforms and routes. In the problem description, a hierarchy of objectives is included. The first objective is to find a feasible route for as many trains as possible. Since we need to comply with the overall timetable, basically all trains have to be routed. However, the problem has been formulated as a maximization problem, because STATIONS should point at the blocking trains if a solution for all trains can not be obtained. Furthermore, if all trains can be routed through the railway station, then the second objective is to minimize the number of shunting movements. A shunting movement is expensive, since personnel (a train driver and assisting personnel) must be allocated. Furthermore, a shunting movement also uses capacity of the railway station, because the routes towards and from the parking area need to be claimed by the safety system. The last objective is to maximize the preferences of the trains for certain platforms or routes. The preference of a train for a certain route mainly depends on the total number of switches in the route, and on the total number of switches in the non-preferred direction.

The processing principles of STATIONS are presented in Zwaneveld et al. (1996) and Zwaneveld et al. (2001). The deviations δ of original arrival and departure times are limited to ± 1 minute. Furthermore, initialisation and preprocessing step determine the admissible combinations of routing possibilities $(r, \delta; r', \delta') \in F_{t,t'}$ for combinations of trains t and t', considering the safety, (un-)coupling, connection requirements and the set of allowable routing possibilities for each train. The calculation of the set $F_{t,t'}$ requires a lot of calculation time $O(T^2 * R^2 * \delta^2)$ in a large and busy railway station which consists of R possible paths and is passed through by a group of T trains with permitted deviation δ. Hereafter, we consider a problem made of 250 trains per day, 310 possible paths and 60 minutes allowable deviation, which makes this preprocessing step unacceptable.

Existing dispatching systems are able to provide viable solutions only for small instances or cyclic timetable (one hour). For the complex and busy railway stations as Bordeaux-Saint-Jean, a routing and scheduling decision support tool is still needed, and both conflicts on platforms and paths must be taken into account in detail.

1.3 Research objectives

Based on the current requirements of railway station operational management and existing computer-based systems, as summarized above, the research objectives of this dissertation are as follows:

1. The main objective is to design, implement and evaluate an off-line timetabling tool for helping railway station manager to schedule technical movements within flexible interval, to reschedule commercial movements in need and to route all movements passing through the railway station area. The tool aims to handle large-scale problem: generation of full-day timetable (not cyclic) within short computation time.

2. An efficient model for railway traffic optimization is needed in order to solve the integrated problem containing three interacting sub-problems: scheduling, routing and platforming. In this dissertation, we call it "train platforming problem".

3. The development and implementation of fast and effective algorithm for checking timetable feasibility have to be addressed. To deal with the infeasible cases (especially in peak hours), blocking trains are sought out and cancelled to guarantee feasibility of timetable. Minimization of trains cancellation has to be achieved without permission of trains' delay.

4. A better arrangement of rail activities is to be achieved by reinsertion of trains cancelled with minimal trains' delay. Cancellation of trains is not the first choice in timetabling process. Proposition of proper revised timetable is required by RFF. This version of timetable includes all trains required, permits slight trains' delay and avoids large delays propagation.

5. Constructive algorithms for full-day train platforming problem in railway station are to be developed and implemented. The algorithms guarantee feasibility of timetable, propose revised timetable for infeasible cases and easily adapt different railway stations' requirements while respecting preference of platforms and safety regulations.

Clearly, the achievement of the first objective is highly related with the other four objectives. In fact, the development of an off-line full-day train platforming tool must contains suitable models and algorithms for scheduling, routing and planforming trains passing through railway stations.

1.4 Thesis contributions

An innovation contribution is presented in this thesis and realizes the combination of research objectives given in Section 1.3. Then, the main achievements are briefly introduced.

A decision support system for full-day (not cyclic) railway station traffic management (Objective 1) is developed to support generation of off-line timetable and to provide two version of resulting timetable:

Feasible timetable Blocking trains in infeasible cases are cancelled. The number of trains cancelled is minimized. Delays of commercial circulations are not permitted.

Revised timetable All train activities required are arranged in the revised timetable. In order to guarantee feasibility of timetable, delays of commercial circulations are permitted but minimized.

As platforms and switches are railway resources to be allocated, resulting timetables are displayed by time-space charts: time-platform chart and time-switch chart. It is easy to verify the conflict-free timetables and to carry out scheduling and routing solutions by railway station managers.

Train platforming problem is described as a continuous-time process and formalized as a mixed-integer linear programming model (Objective 2) in Bai et al. (2013), called decision model. The continuous-time model includes time constraints, platforms preferences, allocation of paths compatible to platform chosen, compatibility constraints of platforms and switches. It is also equipped with a cancellation processing to deal with infeasible cases and provide "Feasible timetable"(Objective 3) in Bai et al. (2014). Objective function is to minimize the number of trains cancelled.

Furthermore, in order to refine the "Feasible timetable", two linear programming models, separately called reinsertion model and refinement model, are implemented to reschedule commercial movements and reinsert trains cancelled while respecting feasibility of "Revised timetable"(Objective 4). Reinsertion model guarantees feasibility, and refinement model minimizes trains' delay.

As a framework of our decision system, a dynamic hybrid algorithm based on sliding window algorithm is developed to join all processes: initialization, preprocessing, resolution, reinsertion and refinement (Objective 5) in Bai et al. (2015). The problem of platforming trains becomes hard to solve when dealing with full-day timetable. In this case, a decomposition method is developed to enable the computation of effective dispatching solutions in a rather short computation time.

In this thesis, benchmark is established on real timetable of Railway station Bordeaux St Jean. For these real-world instances, computational experiments proves that our automated dispatching support system provides good quality timetable in terms of computation time, cancellation minimization and delay minimization.

1.5 Thesis outline

This section gives a short introduction to each chapter. Figure 1.5 describes the structure of this thesis.

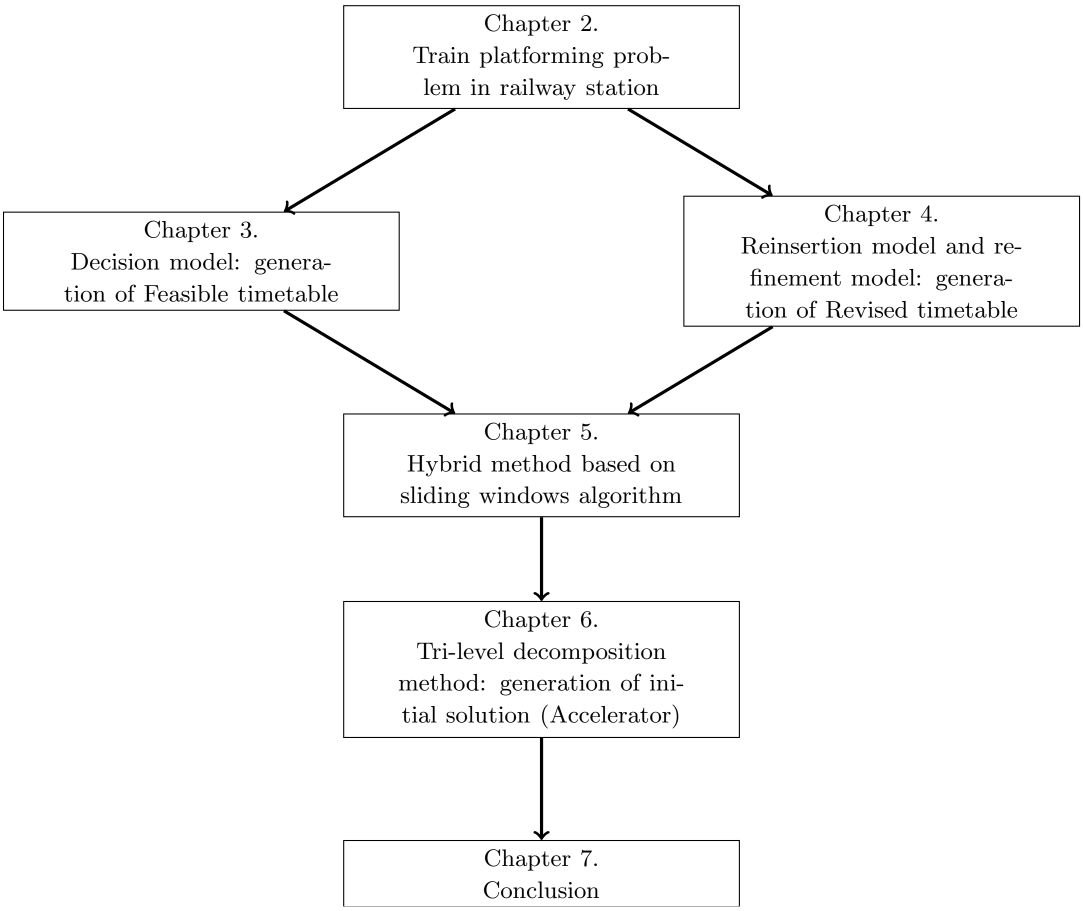


Fig. 1.5 ThesisStructure

Chapter 2 describes train platforming problem in context and in theory. Firstly, an overview of railway networks management is given to position our problem. Then, recent contributions on timetable analysis are described and various models for timetable design are compared and classified with regards to train scheduling and routing problems. An overview of stochastic models is presented. The current state-of-the-art limitations are also discussed. At last, we formalize the topology of railway station and develop the mathematical formula to describe constraints. Chapter 2 ends with the comparison between scheduling problem in a manufacturing production system and that in a railway station.

In Chapter 3, we develop decision model resulting with “Feasible timetable”. In the first part, a complete model is described in terms of parameters, variables and constraints. To reduce computational effort, the model is improved in two ways. The first method is to define time variables in continuous time domain. The second method is to probe potential conflicts between movements and between trains. In this way, the constraints are created only for the movements and trains with potential conflicts. The undesired constraints are cut off. Numerical experiments are presented to prove effects of these two methods. In the last part, cancellation processing is integrated into the improved model to deal with infeasible cases. Blocking trains are cancelled to guarantee feasibility of timetables. Cancellation of trains is minimized.

In Chapter 4, we develop reinsertion model and refinement model resulting in “Revised timetable”. A better arrangement of rail activities is achieved by reinsertion of trains cancelled with minimal train delays. Cancellation of trains is not the first choice in timetabling process. Proposition of proper revised timetable is required. This version of timetable includes all trains required, permits slight train delays and avoids large delays propagation. Objective function of reinsertion model is to minimize the number of trains cancelled. Objective function of the refinement model is minimization of train delays.

Chapter 5 introduces a hybrid method based on sliding window algorithm to organize all functions together and to generate a full-day timetable. At beginning of this chapter, we study real cases by resources occupation indicators. Degree of difficulty is evaluated and analysed by occupation indicators and flexible time interval L for technical movements. The value of L used in the complete algorithm is discussed. After the introduction of the complete algorithm, we present preprocessing of conflict-free on external lines and subgroups partition strategies in details. At last, the hybrid method is tested on real cases in the railway station Bordeaux-St-Jean.

In Chapter 6, we develop a tri-level decomposition method as an accelerator of decision model. Three linear programming models are separately introduced. Operating mechanism of tri-level is summarized and tested on real case. Improvement of calculation efficiency is assessed by comparison of hybrid method with and without tri-level decomposition method.

The main results obtained in this thesis are summarized in Chapter 7. Further research is also addressed in order to determine the next steps for the development of railway station on-line management.

CHAPTER 2

Train platforming problem in railway stations

The purpose of this chapter is threefold:

- to position our problem in railway networks management framework.
- to propose an innovative algorithm structure based on a suggestive literature review.
- to formalize train platforming problem.

The literature review firstly focuses on train platforming problems. Then we select some papers addressing railway network on-line rescheduling problems which are widely investigated by researchers. Based on limitations of current state-of-the-art and illuminating methods used on rescheduling problems, we propose an innovative algorithm structure. After orientation of our research, a formalization of platforming problem is proposed. The safety criterion of this formalization is examined in terms of headway. Complexity matters are considered through the comparison with manufacturing production systems which also provides an idea on linearisation of our constrained platforming problem.

2.1 Overview of railway scheduling and routing problems

Many countries have busy rail networks with highly complex patterns of train services that require careful scheduling to fit these to the existing infrastructure, while avoiding conflicts between large numbers of trains moving at different speeds within and between multi-platform stations on conflicting lines, while satisfying safety or business policy and objectives.

The traditional process to generate a timetable for a railway network is divided into several stages: Watson (2001). First, a tentative timetable is generated by train activities managers (national, regional, freight) based on the traffic frequencies, the volume of traffic, the rough layout of the railway network between the railway stations together with the desired lines and their connection requirements. Then, station operators need to check whether the tentative timetable is feasible within the railway station while satisfying capacity, safety and customer service. At the same time, schedules for the trains through the railway station are generated by including all the required technical operations such as carriage preparation, maintenance, etc. Most of the studies focus on the problem of railway network with a global point of view. Nevertheless, as a bottleneck problem, the routing and scheduling problem in large, busy, complex train stations is also a complex issue with respect to limited buffer of time and space.

However, the construction and coordination of train schedules and plans for many rail networks is a rather slow process in which conflicts of proposed train times, lines and platforms are found and resolved by hand. Even for a medium size rail network, this requires a large numbers of train schedulers or planners many months to complete, and makes it difficult or impossible to explore alternative schedules, plans, operating rules, objectives, etc.

2.2 General description of the problem

This paper studies a train routing and scheduling problem faced by railway station managers to generate a conflict-free timetable which consists of two sets of circulations. The first set is made of commercial circulations given by several administrative levels (national, regional, freight) over a large time horizon (typically one year before the effective realization of the production). The other set corresponds to technical circulations added by the railway station managers to prepare or repair trains. The destination or origin of these trains is then a special place called depot.

The problem of routing and scheduling trains through railway station that we address in this paper can be stated as follows. **Given the layout of a railway station, the arrival and departure times, as well as the arrival and departure directions of a number of trains, is it possible to schedule the technical circulations within the allowable deviations and to path these trains through the railway station?** This timetable must ensure that no pair of trains is conflicting over paths and platforms, while allowing the coupling and uncoupling of trains at a platform and respecting their preferences of platforms and the accessibility of complete path of trains. We call this problem the feasibility problem.

We start to describe this feasibility problem in detail. A railway station can be entered by a train from a number of external lines which connect with other railway stations, and it can be left through also the external lines. Each external line corresponds to a direction of travel. The railway network outside the external lines is not relevant for the feasibility problem. The entering and leaving external lines are given for each train.

A railway station consists of external lines, a large number of tracks sections and internal lines. An *inbound path* is a sequence of tracks linking an entering external line to an internal line next to a platform. In the opposite direction, an *outbound path* is a sequence of tracks linking an internal line to a leaving external line. A *complete path* is a combination of an inbound path, an internal line and an outbound path. There are often many different paths linking a given pair of external lines, and even several different paths using the same internal line.

The *arrival time* of a train is the time at which the train stops at the internal line next to the platform, after travelling along an inbound path. Similarly, the *departure time* of a train is the time at which the train starts to leave the railway station along an outbound path. As described at the beginning of this section, the arrival or departure time of trains are generated by administrative levels and railway station manager. A deviation interval L is permitted for the technical circulations depending on the direction and the reference time defined by railway station manager. Therefore, given departure and arrival times, we must use time flexibility to determine feasible paths for each of the trains passing through the railway station.

The routing problem is to assign each of the involved trains to a complete path including inbound path, platform and outbound path through the railway station. Thus, paths and platforms in the station are here the critical resources of the system. Clearly, the routing of one train depends on the routing of the others. Once a train arrives at the external line, it reserves an appropriate inbound path leading to an internal line. As the train runs along the inbound path, the tracks comprising the path are released sequentially. The safety rules dictate that any track section can be reserved by only one train at a time. In that case, another train cannot reserve the same track until it is released. In principle, the reservation duration of a track can be calculated by the length and speed of train. But the tightness of the resource distribution will easily lead to interruptions of movements or even collisions of trains by any small perturbation. To reduce the risk, all tracks in the inbound path are reserved until the train releases the path and stops at the internal line. This method guarantees that each train can travel along the reserved inbound path without interruption. Similarly, all tracks of an outbound path leading from the chosen internal line towards an external line are reserved until the train releases the whole path and leaves the railway station from the external line.

One internal line can be reserved by only one train at a time. Internal lines are reserved from the beginning of the entering movement until the beginning of the leaving movement. During standstill on internal lines next to platforms between these movements, a minimum customer service time must be given to passengers to board or take off trains, or to transfer between trains. In addition, preferences of internal lines and constraints on coupling and uncoupling of trains need also to be taken into account.

The scheduling problem is to adjust the timetable of technical movements to guarantee on-time arrivals and departures of all commercial movements. A conflict-free timetable with acceptable commercial movements and necessary technical movements is generated. Commercial movements with unsolvable conflicts will return to their original activity managers with suggestions for the modification of the arrival and leaving times.

2.3 Literature review

According to time sequence, timetabling process can be classified into two categories: off-line and on-line. Based on the management level, we have two levels: networks and railway stations. Railway traffic scheduling and routing problems are classified into four quadrants:

1. rail network on-line management (suggestive literature review)
2. rail network off-line management

3. railway station on-line management

4. railway station off-line management (our problem)

Among these four problems, most of researches focus on rail network on-line management. In this section, we present at first related works on railway station management. Then some related works concerning rail network on-line management are reviewed. The current state-of-the-art limitations are also discussed.

2.3.1 Train platforming problem in railway stations

To meet the growing demand in rail traffic, railway networks are operated nearly at capacity margins. However, only certain elements of the railway network are close to capacity limit. These elements can be described as critical resources. Railway stations are not only nodes in the rail network connecting lines but also a service center. When traffic on connecting lines is heavy, the busy and complex railway station is certainly to be a critical resource. In addition, due to a lack of sufficient space in the railway station, trains are not permitted to stop on the track between the internal and external lines during the set-up time S. Compared with scheduling problem on the railway network, the buffer time and space in the railway station is greatly limited. To enhance capacity, new tracks and points can be built. Space and investments are needed for such extensions. However, such solutions are unlikely to be feasible in the short or medium terms. Moreover, they are usually impossible to implement in urban areas. Thus, proper model and efficient algorithms are necessary to optimize the resource allocation in a railway station.

Carey (1994a) proposes a mixed integer program to find the paths of trains in a one-way track system. The numerical example provided in Carey's paper has 10 nodes, 28 links, and 10 train services and requires a significant amount of time to be solved. In another article, Carey (1994b) extends the model from one-way to two-way tracks system. The resulting model is also a mixed integer program, which is easier to solve than his earlier model, but this newer study does not provide testing results.

Kroon et al. (1997) consider computational complexity of the problem of routing trains through railway stations. They represent the safety rules using a given set $F_{t,t'}$ for each pair of trains t, t'. The set $F_{t,t'}$ contains the pairs of allowable path combinations (r, r') for trains t and t'. The problem is formulated as a node-packing problem and describes the relationship between the routing problem and the Fixed Interval Scheduling Problem. Based on the Satisfiability (SAT) problem, they deduce that the safety feasibility problem is NP-complete if each train has three or more routing possibilities. However, if each train has at most two routing possibilities, then the general feasibility problem can be solved in polynomial time. Furthermore, if the layout of the railway station is fixed, then the safety optimization problem can be solved by a dynamic programming approach in an amount of time that is polynomial in the number of trains. This result can be extended to the case of coupling and uncoupling of trains; some service considerations and a cyclic timetable have to be taken into account.

Zwaneveld et al. (1996) formulate the problem of constructing feasible routing trains through railway stations, with the given arrival and leaving times of trains in a cyclic timetable (one hour) and the detailed layout of the railway station, as an integer linear program, based on the Node Packing Problem (NPP). The deviations δ of original arrival and departure times are permitted. Furthermore, a solution procedure is proposed for the problem, based on a branch-and-cut approach. To reduce the size of the problem, they firstly determine the admissible combinations of routing possibilities $(r,\delta;r',\delta') \in F_{t,t'}$ for combinations of trains t and t', considering the safety, (un-)coupling, connection requirements and the set of allowable routing possibilities for each train. Secondly, the preprocessing step simplifies the problem by removing dominated nodes in the graph of node packing problem. Then, valid inequalities are added to tighten the problem. An initial solution is generated by heuristics method to accelerate the calculation. But the calculation of the set $F_{t,t'}$ requires a lot of calculation time $O(T^2 * R^2 * \delta^2)$ in a large and busy railway station which consists of R possible paths and is passed through by a group of T trains with permitted deviation δ. Hereafter, we consider a problem made of 250 trains per day, 310 possible paths and 60 minutes allowable deviation, which makes this preprocessing step unacceptable.

Zwaneveld et al. (2001) improves the model by including shunting decisions and preferences of trains for platforms and paths, and improves also the algorithm by extending the preprocessing techniques. If a train is to be shunted towards a parking area, the train is to have a standstill at its arrival platform and a different departure platform, and is to be at a platform of parking area between these two standstills. The shunting decision and allocation preferences are represented by the weight coefficient of allocation variables. The node packing of maximum weight represents a feasible routing of trains with the maximal preference and maximum number of trains scheduled. Three techniques separately called node-dominance, set-dominance and iterating set-dominance are proposed to reduce the size of problem. But the deviations of the arrival and departure times are not taken into account.

In the model, for each train that may be shunted, only the decision whether or not to shunt is considered. The detailed shunting movements are not taken into account explicitly.

Carey and Carville (2003) considers the problem of train planning or scheduling for large, busy, complex train stations. A scheduling heuristics analogous to those successfully adopted by train planners using manual methods is developed. The algorithm considers each train t separately. For each train, it considers each feasible platform and for each of these platforms it checks if there are any headway or platform conflicts between the train t and each train already scheduled temporarily or permanently at the station. If a conflict exists, the arrival and/or leaving times of train t are increased to a point where there is no longer a conflict. This constraint checking and time adjustment processes are repeated until a platform with the lowest costs or penalties is found. Then the algorithm proceeds to assign the next train in the list of non-fixed trains. Heuristic techniques are designed according to train planners' objectives, and take account of a weighted combination of costs and preference trade-off. The model and algorithms are tested in a typical station that exhibits most of the complexities found in practice. With the progressive improvement of the search rules, the computing times fell from several hours to a few seconds, depending on the version of the algorithm and the scheduling problem, but the insolvable conflicts are removed by hand before the heuristics methods. This method based on the scheduling experience is not suitable for other railway stations.

Cardillo (1998) uses a graph colouring formulation and an efficient heuristic called Conflict-Direct Backtracking to solve the feasibility problem in short calculation time. Between a platform and a line, only a single route variant is considered. Additionally, a list of incompatible routes is pre-calculated. The method is tested on 6 real stations. Problems in 5 stations are solved in less than one second, and problems in the other station are solved in 115 seconds.

Billionnet (2003) uses integer programming to solve the same problem considered in Cardillo (1998) with various goal functions. This model is tested on 20 randomly generated stations and train sets. Solving time on these instances are from below one second to 80s and one case had no solution in 1200s. With an alternative model formulation and the addition of clique cuts, solving time is reduced to 0.01-0.03s to solve the real station of *Abatone* with 5 platforms and 41 trains.

Caprara (2010) and Caprara and Galli (2011a) treat train platforming problem considered for the case of multiple routes where platform times can vary in a discrete interval. They minimize a quadratic function involving deviation from preferred platforms and deviation from platform times. The platforms set includes two parts: regular platforms corresponding to platforms that one foresees to use and dummy platforms corresponding to platforms that one would like not to use but that may be necessary to find a feasible solution. They linearise the objective function and use clique inequalities which reduce solver times. Thanks to the dummy platforms, the system always returns solutions and prefers to not platform a train in order to reduce time deviation for the platformed trains.

Pellegrini et al. (2014) proposes a mixed-integer linear programming formulation to tackle the real-time train platforming problem. The local networks within railway stations are represented with fine granularity to reduce the size of the search space. The impact of the granularity finesse is evaluated by randomly generated instances representing traffic in *Triangle of Gagny* and real instances in *Lille-Flandres* station under multiple perturbation scenarios. Through these experiments, the negative impact of a rough granularity on the delay is remarkable and statistically significant.

Due to the complexity and size of problem, the trains routing and scheduling problem during one day with cancellation processing in the railway station is not yet properly solved. An effective model is still needed to describe a suitable resources allocation strategy. According to the information we have, train platforming problem is mainly solved by branch-and-bound and heuristic algorithms.

Then we review some works on on-line rescheduling problem, and we study two meta-heuristic algorithms: genetic and tabu algorithm which are not applied in train platforming problem.

2.3.2 Railway networks on-line rescheduling problem

Compared with off-line platforming problem, on-line rescheduling problem has been widely investigated by researchers. Various heuristic and meta-heuristic algorithms are designed and applied on rescheduling problem. Although solution requirement of on-line rescheduling problem is different from that of off-line platforming problem, design ideas of algorithms can be referred to explore our own proper method.

A railway network consists of track segments and signals. Signals allow to regulate traffic in the network by enforcing speed restriction to running trains. The track segment between two signals is called block section, and can host at most one train at a time. Traffic regulations also impose a minimum distance separation among the trains, which translates into a setup time between the exit of a train from a block section and the entrance of the subsequent train in the same block section.

Regulation of railway traffic aims at ensuring safe, seamless and as much as possible punctual and energy-efficient train operations. Due to the strict time limits for computing a new timetable in the presence of disturbances, train dispatchers usually perform manually only a few modifications, while the efficiency of the chosen measures is often unknown. Some computerized dispatching support systems have been developed, so far, which can provide fairly good solutions for small instances and simple perturbations. However, this kind of systems cannot deal with heavy disturbances in larger networks as the actual train delay propagation is simply extrapolated and does insufficiently take into account the train dynamics and signalling constraints. Therefore, extensive control actions are necessary to obtain globally feasible solutions.

Some researchers take the rerouting, cancellation and other complicated strategies into consideration. Jacobs (2004) has developed a detailed model based on the identification of possible route conflicts with high accuracy using the blocking time theory. The objective function is minimization of additional running time. In presence of disturbances, infeasible train routes are detected, and conflicts are solved locally based on train priorities.

Norio1 and Yoshiaki (2005) regard train rescheduling problem as a constraint optimization problem. Passengers' dissatisfaction is used as objective criterion. Then an efficient algorithm combining PERT and meta-heuristics has been developed. Numerical experiments show that it works quite fast and it supports versatile methods of rescheduling including cancellation, change of train-set operation schedule, change of tracks etc..

Rodriguez (2007) focuses on the real-time CDR problem through junctions and proposes a constraint programming formulation for the combined routing and sequencing problem. His results show that a truncated branch and bound algorithm can find satisfactory solutions for a junction within computation time compatible with real-time purposes.

D'Ariano (2008) has developed a real-time dispatching system, called ROMA (Railway traffic Optimization by Means of Alternative graphs), to automatically recover disturbances. ROMA is able to automatically control traffic, evaluating the detailed effects of train reordering D'Ariano et al. (2007a) and local rerouting D'Ariano et al. (2006b) actions, while taking into account minimum distance headways between consecutive trains and the corresponding variability of train dynamics (D'Ariano and Albrecht (2006a); D'Ariano et al. (2007b); D'Ariano et al. (2008)). In order to handle large time horizons within a linear increase of computation time, the temporal decomposition approach which decompose a long time horizon into tractable intervals to be solved in cascade with the objective of improving punctuality has been proposed D'Ariano and Pranzo (2009).

Corman et al. (2010) develop two new routing neighborhoods of different size in order to search for more effective routings, and study their structural properties in a tabu search scheme. Three different methods: a lower bound and two upper bounds on the objective function for each neighbour are tested on an extensive campaign of experiments based on practical size instances referring to the Dutch dispatching area between Utrecht and Den Bosch. The instances include timetable disturbances, passenger connections, multiple delayed trains and heavy network disruptions. Compared with those obtained by the branch and bound algorithm of D'Ariano et al. (2007) and by the local search algorithm of D'Ariano et al. (2008), the new algorithms improve significantly the performance of the previous version of ROMA both in terms of solution quality and computation time. For small instances, the new algorithms allow to close the optimality gap. For large instances, the new algorithms achieve significantly better results with respect to the previous version of ROMA within remarkably reduced computation times.

To speed up trains rescheduling computation time, some decomposition or multi-layer methods are adopted. D'Ariano and Pranzo (2009) have proposed the temporal decomposition approach that decompose a long time horizon into tractable intervals. Wegele and Schneider (2005) have presented an optimization method for fast construction of time tables which can be used for dispatching or long term operation planning. This method contains two steps: constructing and iterative improving. Constructing step uses branch-and-bound method to construct the first solution taking into account only restricted amount of possible decisions. Improving step adopts the genetic algorithm as an iterative improving method. These several levels optimization method can help reduce the calculation effort.

Furthermore, Tornquist and Persson (2005) address the problem of solving conflicts in railway traffic due to disturbances. The problem is formulated as a problem of re-scheduling meets and overtakes of trains and has been dealt with in a two-level process. The upper level handles the order of meets and overtakes of trains on the track sections while the lower level determines the start and end times for each train and the sections it will occupy.

Sahin (1999) deals with analysing dispatcher's decision process in inter-train conflict resolutions and developing a heuristic algorithm for re-scheduling trains by modifying existing meet/pass plans in conflicting situations in a single-track railway. A systems approach is used in construction of the heuristic algorithm, which is based on inter-train conflict management.

Adenso-Diaza et al. (1999) investigate the train rescheduling problem by optimization and simulation in order to obtain an exact or approximate solution. The model aims at maximizing the number of passenger transported and is solved by heuristic procedure based on backtracking algorithm. The model and DSS is implemented in Asturias of the Spanish National Railway Company in 1998.

Higgins et al. (1996) propose a model and a solution method for the dispatching of trains on a single-track line. Their model mainly addresses the operational problem of dispatching trains in real time but can also serve at the strategic level to evaluate the impacts of timetable or infrastructure changes on train arrival times and train delays. The formulation is a non-linear mixed integer program that incorporates lower and upper limits on train velocities for each train on each segment. The objective function only seeks to minimize a combination of total train tardiness and fuel consumption.

Based on the literature review of on-line rescheduling and rerouting problem, there are two important ideas suitable to be applied on our full-day off-line platforming problem:

- Acceleration of calculation: addition of lower and upper solution limits, identification of possible route conflicts.
- Simplification of large-scale problem: decomposition or multi-layer methods applied by D'Ariano and Pranzo (2009), Tornquist and Persson (2005) and Wegele and Schneider (2005).

In decision model described in Section 3, probes of potential conflicts between pair of trains and between pair of movements are designed to cut off the undesired constraints, details can be found in Section 3.4.2. Furthermore, to reduce calculation efforts required by decision model, an accelerator is designed as a tri-level optimization model to provide an initial solution for decision model. To deal with the large-scale problem, we apply also decomposition methods to divide all trains of a day into small sub-groups to be solved in cascade.

When the problem is properly formalized and constructed, we need to find a suitable algorithm to solve it. As exact and heuristic methods have been widely used, we study two meta-heuristic algorithms in following sections and try to analyse their advantages and disadvantages for our platforming problem.

2.3.2.1 Genetic algorithms

Genetic algorithms, which were developed by Holland (1975), are randomized search algorithms based on natural selection and natural genetics. They merge "the survival of the fittest" rule applied to the structures in form of series and a structural but randomized knowledge exchange method to eventually compose a search algorithm. GAs generate a set of different solutions instead of finding a single solution for problems. They form a new set of artificial organisms (series) in every generation and calculate their fitness for survivability. They explore new search points efficiently, using the available knowledge in a generation to increase performance (Goldberg (1989)). In this way, many points in the search space can be evaluated, and possibility to reach the optimal solution is increased.

GAs mimic the natural evolutionary process to solve problems using computers. Contrary to other optimization methods, which generate a single structure for the solution, they generate a set of composed structures. A set representing many potential solutions for the problem at hand is called "generation" in the GA terminology. A generation is composed of a numerical series each of which is named as a "vector", "individual", or "chromosome". Each component of an individual is a "gene". Individuals in a generation are determined by the operators of the GA. Contrary to other optimization methods, GAs are efficient in solving problems having huge search space. They do not guarantee finding an optimal solution for the problem, but they could provide acceptable solutions in reasonable times. GAs are useful to search solutions for the problems for which no efficient solution method exists. Hence they are not suitable for problem types where dedicated effective and efficient techniques exist to reach the optimal solution. GAs are suitable if:

- Search space is huge and complex.
- Reaching the optimal solution in the limited search space is difficult with the available knowledge.
- The problem cannot easily be formalized by a mathematical model.

- The desired solution cannot be achieved using traditional optimization algorithm.

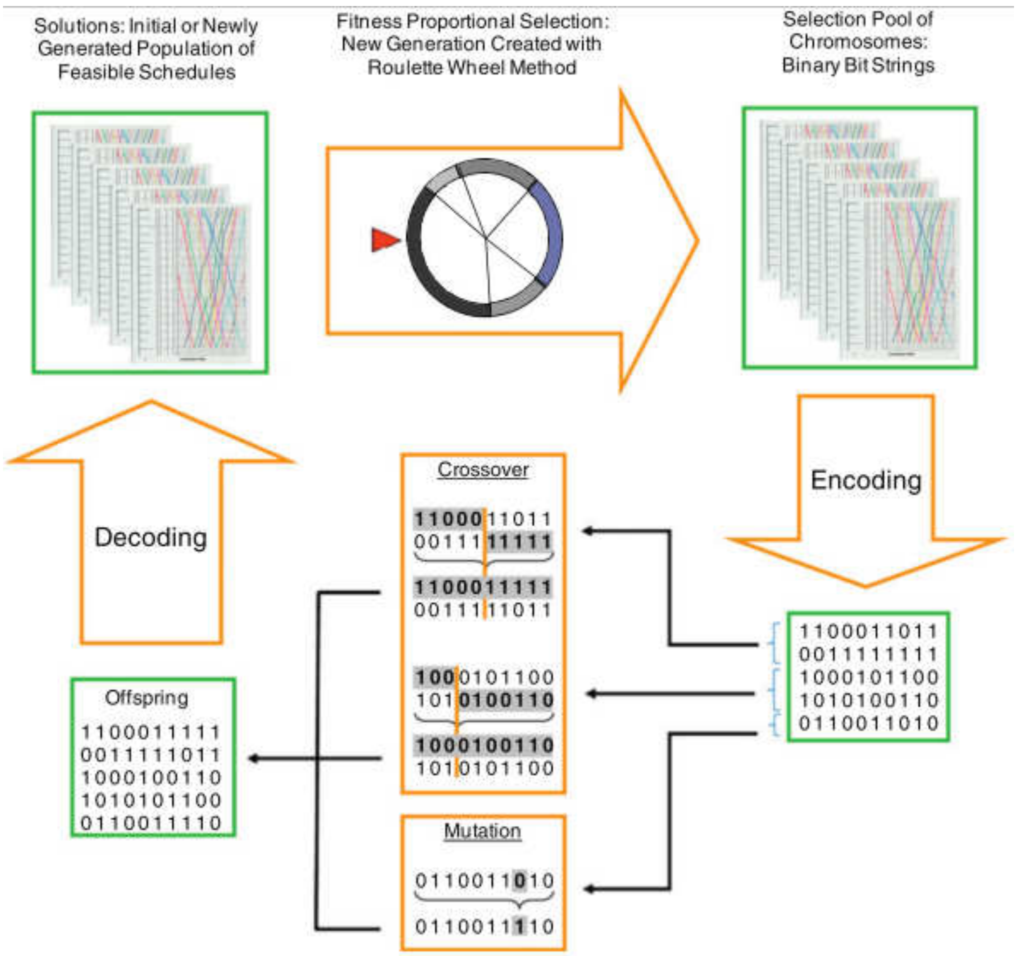


Fig. 2.1 Genetic algorithm for train re-scheduling

Dündar and Sahin (2013) introduce the steps of a binary GA for train re-scheduling shown in Figure 2.1 and summarize them as follows:

1. Form an initial generation composed of n randomly generated individuals; i.e., feasible schedules.

2. Choose individuals from the current population through proportional selection.

3. Encode each feasible schedule in new generation to represent them in binary bit strings (chromosomes), each of which has a length of l genes.

4. Group individuals (parents) in pairs by applying crossover, and create new individuals (offspring).
5. Apply mutation to each individual with a very small probability.
6. Decode each offspring (individual) and calculate its fitness value.
7. Replace the old generation with the newly formed one.
8. Go back to Step 2 unless a termination criterion has been reached.

One of the pioneering works based on GAs applied to the railway traffic control problem was conducted by Salim and Cai (1995). In the GA developed, genes are coded in the binary system and the size of chromosomes is $n*m$, where n is the number of stations (meeting points) and m is the number of trains. The value of each gene represents the state of a train in a station; 0 means stopping and 1 passing. The authors reported that for a problem instance of 12 stations and 9 trains, a feasible solution was reached in 1.5 h.

Higgins et al. (1997) used several meta-heuristic techniques to minimize the total weighted delay of trains in a single-track railway line: a local search heuristic with an improved neighbourhood structure, genetic algorithms, tabu search and two hybrid algorithms. In the GA modelling approach of this study, each gene consists of three data, namely, the train delayed, the train confirmed, and the track segment where the conflict occurs. This representation method may give a number of infeasible solutions. Both GAs and hybrid algorithms are unable to provide feasible solutions in short times.

In another GA developed by Ping et al. (2001), the objective is to minimize the total conflict resolution delay. The chromosomes are coded to represent train departure orders at stations; and therefore, it is solved as a sequencing problem.

In the study of Wegele and Schneider (2005), the fitness function of the GA developed includes signalling and switching delays, and penalties due to missing transfers. Each chromosome consists of 17,000 genes in a problem instance with a size of 12 stations and 260 trains. Because this representation method may cause congestion in the network, the authors present another coding scheme, in which the chromosomes represent train orders, with which congestions are overcome more easily. This method can decrease delays by 11.2% in 3 min with respect to the previous method.

Chang and Chung (2005) developed another GA using a matrix representation, where trains are in the rows and stations in the columns. The sum of a set of defined constraints constitutes the objective function. It takes about 20 min to obtain a feasible schedule using this method.

Tormos et al. (2008) used a chromosome representation in their GA consisting of an activity list of a solution. This solution representation has been widely used in project scheduling. The solution is encoded as a precedence feasible list of pairs. In the implementation, the new trains are scheduled following the order established by the list. Each individual in the population is represented by an array with as many positions as pairs existing in the railway scheduling problem considered. Numerical tests show that the new modelling technique has a potential for obtaining better solutions for scheduling new trains in a fixed schedule.

Arenas et al. (2014) addresses the problem of generating periodic timetables. In railway operations, a timetable is established to determine the departure and arrival times for the trains or other rolling stock at the different stations or relevant points inside the rail network or a subset of this network. The elaboration of this timetable is done to respond to the commercial requirements for both passenger and freight traffic, but it must also respect a set of security and capacity constraints associated with the railway network, rolling stock and legislation. Combining these requirements and constraints, as well as the important number of trains and schedules to plan, makes the preparation of a feasible timetable a complex and time-consuming process, that normally takes several months to be completed. They present a constraint-based model and propose a genetic algorithm, allowing a rapid generation of feasible periodic timetables.

If we evaluate genetic algorithm based on the requirement of our platforming problem, we may get several disadvantages:

- There is no absolute assurance that a genetic algorithm will find a global optimum. It happens very often when the populations have a lot of subjects. Different from on-line scheduling problem, minimization of trains cancellation in platforming problem is an important criterion of solution quality. A global optimum is highly required in terms of feasibility.

- Like other artificial intelligence techniques, the genetic algorithm cannot assure constant optimisation response times. Even more, the difference between the shortest and the longest optimisation response time is much larger than with conventional gradient methods. This unfortunate genetic algorithm property limits the genetic algorithms' use in real time applications.

- Genetic algorithm applications in controls which are performed in real time are limited because of random solutions and convergence, in other words this means that the entire population is improving, but this could not be said for an individual within this population. Therefore, it is unreasonable to use genetic algorithms for on-line controls in real systems without testing them first on a simulation model.

- A large size of binary chromosome representation is required by platforming problem. As shown in Figure 2.3, there are 42 meeting points including 17 switches, 15 platforms, 10 entrance lines. Besides order of trains, representation of resources allocation for one train needs at least 42 genes. Full-day timetable contains around 250 trains.

- Different from on-line rescheduling problem, there is no feasible original timetable and conflicts easily identified in off-line platforming problem. In that case, search space is more complex, and evaluation criterion is crucial to guarantee improvement of population. Certain optimisation problems cannot be solved by means of genetic algorithms. This occurs due to poorly known fitness functions which generate bad chromosome blocks in spite of the fact that only good chromosome blocks cross-over.

2.3.2.2 Tabu search algorithm

On-line rescheduling and rerouting problem is also a train conflict detection and resolution (CDR) problem. Conflicts are eliminated by adjusting dwell times, train speed profile, train order and routes. The primary goal of reordering strategies is to reduce delay propagation, while the combined adjustment of train orders and routes allows to thoroughly reorganize the use of available resources in order to minimize train delays. For example, manual train rerouting is commonly practised at the Swiss Federal Railways to control traffic in heavily used areas (Lüthi et al. (2007)).

Since the combinatorial structure of the CDR problem is similar to that of the job shop scheduling problem with routing flexibility, we focus on the tabu search approach that achieved very good results with the latter problem (Mastrolilli and Gambardella (2000)).

The tabu search (TS) is a deterministic meta-heuristic based on local search (Glover (1986)), which makes extensive use of memory for guiding the search. Basic ingredients of a tabu search are the concepts of move and tabu list, which restrict the set of solutions to explore. From the incumbent solution, non-tabu moves define a set of solutions, called the neighbourhood of the incumbent solution. At each step, the best solution in this set is chosen as the new incumbent solution. Then, some attributes of the former incumbent are stored in a tabu list (TL), used by the algorithm to avoid being trapped in local optima and to avoid re-visiting the same solution. The moves in the tabu list are forbidden as long as these are in the list, unless an aspiration criterion is satisfied. The tabu list length can remain constant or be dynamically modified during the search. We notice that, despite their similarity, there are significant differences between the CDR problem and the job shop scheduling problem, such as the absence of inter-machine buffers in the CDR problem, called no-store or blocking constraint (Hall and Sriskandarajah (1996)).

As a result, most of the properties that are used in the job shop scheduling problem to design effective neighbourhood structures do not hold for the CDR problem. Specifically, computing the value of the objective function after a local change, either train rerouting or reordering, may require a significant amount of time. Even the feasibility of a solution after a local change cannot be ensured as it occurs, e.g., in the job shop scheduling problem when reordering two consecutive operations laying on a critical path (Balas (1969)).

2.3.3 Conclusion of literature review: our strategies

In the previous sections, we have firstly introduced several related works on train platforming problem. Due to complexity of the large-scale problem, the full-day train platforming problem with cancellation processing in railway stations is not yet properly solved. An effective and efficient model is still needed to describe a suitable resources allocation strategy.

Then, in order to start with a well-directed research, we have studied some related works on on-line rescheduling problem which is wildly investigated by researchers. Two suggestive methods are borrowed to solve our platforming problem:

1. An upper or lower bound of solutions can accelerate the calculation process of branch-and-bound. Because the search tree will be cut by the bound. In this way, the search space is greatly reduced.

2. A large-scale scheduling problem can be simplified by chronological and multi-level decomposition methods.

In our thesis, these two ideas are our main research directions.

Finally, two meta-heuristic algorithms applied on on-line rescheduling problem are presented in details. Further studies on chromosome representation and solution evaluation criterion is still needed to apply Genetic Algorithms in railway management. If we use Tabu search, computing the value of the objective function after a local change, either train rerouting or reordering, may require a significant amount of time. Even the feasibility of a solution after a local change cannot be ensured as it occurs. Considering unstable optimisation response time and insurance of global optimum of these two meta-heuristic algorithms, in this thesis, we do not choose meta-heuristic algorithms and apply a hybrid algorithm combining branch-and-bound and heuristic algorithms.

Our solving strategies are described as a hybrid method based on sliding window algorithm in Figure 2.2.

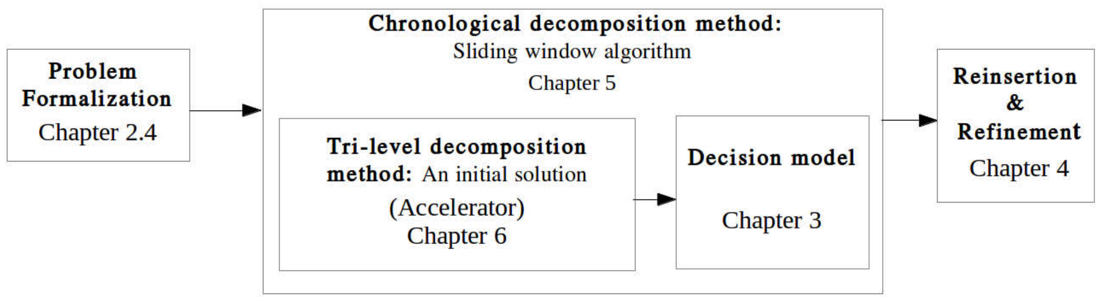


Fig. 2.2 Our solving strategies

The set of trains in full-day timetable is divided into small sub-problems to optimise in time sequence by sliding-window algorithm (chronological decomposition method) as described in Chapter 5. Sub-problems are formalized by Decision model as described in Chapter 3 and solved by branch-and-bound supported by CPLEX. To accelerate the calculation of sub-problems, we apply tri-level decomposition method, as described in Chapter 6, to provide an initial solution to start the brand-and-bound calculation process of sub-problems. The feasible timetable solved by decision model is refined by reinsertion model and refinement model described in Chapter 4. Concentration of potential conflicts is applied in decision model, reinsertion model and refinement model to reduce calculation efforts.

2.4 Problem formalization

In this section, we formalize the whole problem in parts, the railway station layout and the trains' activities. The section 2.4.1 carries out a reasonable formalization of railway station layout which greatly helps to reduce the size of problem. Instead of the complete path definition, the allocation resources are defined as three parts: inbound path, internal line and outbound path. Then the elimination of detour paths and superfluous switches avoids the useless paths. At last, we assess the improvements by comparing problem sizes. Trains' activities are formalized in section 2.4.2. Except the scheduling and routing notation, we analyse also coupling and uncoupling operations, compatibility of resources and preference list of internal lines.

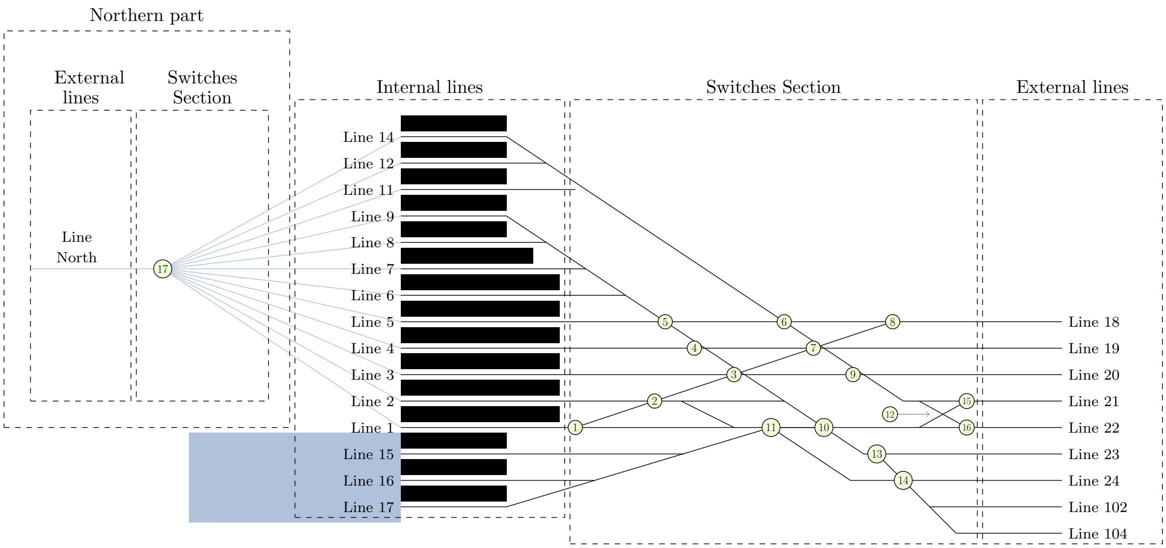


Fig. 2.3 Southern part of the railway station

2.4.1 Railway station layout

To produce an integer linear program to solve platforming problem, complete paths may be considered. However, if there are many inbound paths and/or outbound paths connecting certain external line with certain internal line, then it is worthwhile to replace the complete paths by combinations of internal line, inbound and outbound paths.

Indeed, suppose for train t there are n_1 inbound paths from the entering external line to each internal line, and n_2 outbound paths from each internal line to the leaving external line.

Then the number of complete paths for train t from one external line l_e to another external line $l_{e'}$ visiting on of the internal line equals $n_1 * n_2 * L^i$, where L^i denotes the number of internal lines. Hence in the model the number of variables $X_{t,r}$ for train t also equals to $n_1 * n_2 * L^i$. Kroon et al. (1997) define the complete path as a combination of inbound and outbound paths, where each path is associated to its corresponding internal line. Thus, $n_1 * L^i$ variables are required to choose an inbound path for train t, and $n_2 * L^i$ variables are required to choose an outbound path for train t. Thus the total number of variables required for train t is $(n_1 + n_2) * L^i$.

Our strategy is to make a distinction among the internal line, the inbound paths and the outbound paths. Then n_1 variables are required to choose an inbound path for train t, L^i variables are required to choose an internal line for train t, and n_2 variables are required to choose an outbound path for train t. Then the total number of variables required for train t is $n_1 + L^i + n_2$, which may be significantly less than $n_1 * n_2 * L^i$, even greatly less than $(n1 + n2) * L^i$.

To describe accurately the railway station, we need to firstly choose the elementary resource of a path between the track and the switch which are separately represented by letters and numbers in Figure 2.4 and in Table 2.1. If two trains are on the conflicting paths, there are two possible cases shown in Figure 2.4.

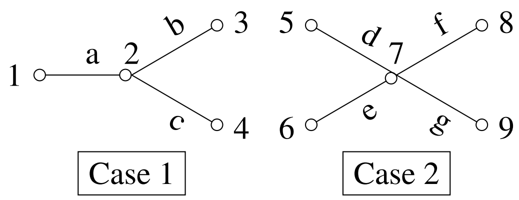


Fig. 2.4 Two cases of conflicting paths

The two pair of conflicting paths in the two cases above can be represented in the table as below.

Elementary resources	Tracks	Switches
Case 1	a,b	1,2,3
	a,c	1,2,4
Case 2	d,g	5,7,9
	e,f	6,7,8

Table 2.1 Comparison between two cases of conflicting paths

If a path is represented by a list of tracks, the two paths in the first case have the track *a* in common, but we cannot see the conflict between the two paths in the second case. If the paths are described as a set of switches, the two paths in the first case have the switches 1 and 2 in common and the two paths in the second case share the switch 7. So we describe the path as a set of switches instead of tracks to detect the routing conflits.

The layout of the railway station is given in Figure 2.3. As the southern part of the railway station receives always the overload of trains' activities, we focus only on the southern part to reduce the size of the problem and retain the preference of internal lines to adapt to the trains' activities on the northern part. Lines 15, 16 and 17 are not accessible to the northern part. In our case study, we neglect the northern part, and model it as a single switch. The layout of the northern part is then replaced by the external line *Line north* and one path including *Switch* 17 which can be occupied by more than one train. Conflicts on the northern part are not taken into account. Note that our model could be used to model the full complexity of the northern part if needed.

Definition 1 (Railway station)**.** A railway station $\mathbb{R} = (\mathbb{S}, \mathbb{L}, \mathbb{P})$ is defined by a set of lines $\mathbb{L}$ on which trains follow some paths in a set $\mathbb{P}$, defined using switches in the set $\mathbb{S}$.

Switches (s_k). The set $\mathbb{S} = \{s_1, s_2, \dots, s_{\mathsf{S}}\} = \{s_k\}_{k \in [\![1,\mathsf{S}]\!]}$ designates a set of switches. The cardinal number of $\mathbb{S}$ is denoted as S.

Lines (l_f). The set of lines is defined by $\mathbb{L} = \{l_1, l_2, \dots, l_{\mathsf{L}}\} = \{l_f\}_{f \in [\![1,\mathsf{L}]\!]}$. L denotes the cardinal of the lines set $\mathbb{L}$. We make a distinction between *internal* and *external* lines. External lines located at the entrance of the railway station are denoted by the set $\mathbb{L}^e$. Internal lines are denoted by the set $\mathbb{L}^i$. Internal and external lines can be connected together using the set of switches, through a small railway network inside the railway station. Every line $l \in \mathbb{L}$ is connected to a unique *"entrance"* switch denoted as $\zeta(l) \in \mathbb{S}$, while a switch may be connected to multiple lines.

Paths (p_c). The set of paths is denoted by $\mathbb{P} = \{p_1, p_2, \ldots, p_{\mathsf{P}}\} = \{p_c\}_{c \in [\![1,\mathsf{P}]\!]}$. P denotes the cardinal number of $\mathbb{P}$. A path $p \in \mathbb{P}$ consists of a set of ordered switches $\mathbb{S}^p = [s_1^p, s_2^p, \ldots, s_{\mathsf{S}^p}^p] = \{s_k^p\}_{k \in [\![1,\mathsf{S}^p]\!]}$ with the cardinal number S^p. Switches of a path are always described from railway station to the outside. For each path, we consider two special switches s_1^p (*internal switch*) and $s_{\mathsf{S}^p}^p$ (*external switch*). The set $\mathbb{P}$ reflects the topology of the railway station, and some sequences of switches are not valid paths.

The subset of paths that connect the internal line $l_i \in \mathbb{L}^i$ and the external line $l_e \in \mathbb{L}^e$ is denoted by $\mathbb{P}^{(l_i,l_e)} = \{p_c\}_{c \in [\![1,\mathsf{P}^{(l_i,l_e)}]\!]}$. The subset of internal lines l_i reachable from an external line $l_e \in \mathbb{L}^e$ is denoted by $\mathbb{L}^i_{l_e}$.

Then, we minimize the number of paths and switches in the railway station without losing feasible solutions. Firstly we minimize the number of switches without the elimination of useful paths. Only the switch corresponding to the internal or external line, or corresponding to a cross-over of paths can be considered as an elementary resource. Moreover, some switches shown in figure 2.5 can be grouped in one section as one switch.

A train cannot pass along a path with an acute angle, for example, the paths [3,2,4,6],

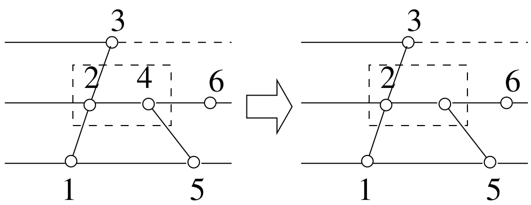


Fig. 2.5 Superfluous switches

[1,5,4,6] are infeasible. There are 7 paths in Figure 2.5, such as [1,2,3], [1,5], [1,2,4,6], [1,2,4,5], [2,4,6], [2,4,5], [2,3]. All paths containing the switch s_4 pass also the switch s_2. Furthermore, any pair of conflicting paths sharing the switch s_4 contains also the switch s_2. So the switch s_4 can be combined with the switch s_2. The paths are simplified as [1,2,3], [1,5], [1,2,6], [1,2,5], [2,6], [2, 5], [2,3]. On the other hand, the path [1,2,5], which is useless compared to the path [1,5], can be called a detour path and be neglected.

Definition 2 (Superfluous switches)**.** If all the paths $p \in \mathbb{P}$ including the switch s_1 contain also the switch s_2 in common, the conflict on the switch s_1 can be represented by s_2. Then switch s_1 is a superfluous switch.

$$\begin{aligned} & s_1 \in \mathbb{S} \text{ is a superfluous switch.} \\ & \Leftrightarrow \exists s_2 \in \mathbb{S} \text{ s.t. } \forall p \in \mathbb{P}, s_1 \in p \Rightarrow s_2 \in p \end{aligned}$$

Definition 3 (Detour paths)**.** Consider two paths p_1 and p_2 have the same first and last switches. If path p_2 contains only a subset of switches of path p_1, path p_1 is a detour path.

$$\begin{aligned} & p_1 \in \mathbb{P} \text{ is a detour path.} \\ \Leftrightarrow\ & \exists p_2 \in \mathbb{P} \text{ s.t. } s^{p_1}_{\mathsf{S}^{p_1}} = s^{p_2}_{\mathsf{S}^{p_2}}, s^{p_1}_1 = s^{p_2}_1 \text{ and } p_2 \subset p_1 \end{aligned}$$

To estimate the reduction of problem size, we compare the number of variables needed to describe all routing possibilities within the railway station shown in Figure 2.3, around 250 trains per day.

Formalization mode	With detour paths		Without detour paths	
	1 Train	1 Day	1 Train	1 Day
Complete paths	2377	594250	2015	503750
Inbound path + Outbound path	330	82500	310	77500
Inbound path+ Internal line+ Outbound path	169	42250	159	39750

Table 2.2 Variables number under three formalization modes

In Table 2.2, we show the number of variables needed for each formalization, for a typical train made of two movements (entering and leaving movements). Such variables correspond to $X^{PM}_{p,m}$ and $X^{LiT}_{l,t}$ that will be used in the MILP model of section 3. From Table 2.2, we can find that the mode "Inbound path+Internal line+Outbound path" needs only 7.5% of the number of variables required by the mode "Complete paths" which is equivalent to 51.2% of the number of variables required by the mode "Inbound path+Outbound path". In the example, there are 10 detour paths. Without the detour paths, 6% of the number of variables, separately equivalent to 5000 and 2500 variables for the one-day timetable, are eliminated in the mode "Inbound path+Outbound path" and "Inbound path+Internal line+Outbound path". In the mode "Complete paths", 15.2% of the number of variables is removed by the elimination of detour paths. So we can see that our routing formalization strategy helps greatly to reduce the size of the problem.

2.4.2 Trains' activities

The traffic in the railway station is defined by a set of trains $\mathbb{T} = \{t_1, t_2, \ldots, t_\mathsf{T}\} = \{t_i\}_{i \in [\![1,\mathsf{T}]\!]}$ with the cardinal number T. Every train $t \in \mathbb{T}$ consists of a set of ordered movements $\mathbb{M}^t = [m^t_1, m^t_2, \ldots, m^t_{\mathsf{M}^\mathsf{t}}] \subset \mathbb{M}$ with the cardinal number M^t. The index of the movement represents the chronological order, for example m^t_1 occurs before m^t_2. Four types of movements are defined depending on their commercial or technical nature , and their direction (entering or leaving the railway station). The full circle $\bigcirc$ is a mnemotechnic way to denote the railway station side. Trains are denoted by an arrow $\rightarrow$. When a train enters the railway station from an external line, this action can be represented by $\rightarrow\!\!\!\bigcirc$. When a train stops on an internal line after passing through the switches section, this action can be represented by $\bigcirc\!\!\!\rightarrow$. In the following paragraphs, the technical movements are denoted by a semi-arrow $\rightharpoonup$; the commercial movements are denoted by a full arrow $\hookrightarrow$; a train leaving the railway station is denoted by $\bigcirc\!\!\!\rightarrow$; a train entering the railway station is denoted by $\rightarrow\!\!\!\bigcirc$ (the full circle $\bigcirc$ being a mnemotechnic way to denote the railway station side). We divide thus the set of movements $\mathbb{M}$ into four subsets such that: $\mathbb{M} = \mathbb{M}^{\bigcirc\!\!\hookrightarrow} \bigcup \mathbb{M}^{\hookrightarrow\!\!\bigcirc} \bigcup \mathbb{M}^{\bigcirc\!\!\rightharpoonup} \bigcup \mathbb{M}^{\rightharpoonup\!\!\bigcirc}$.

2.4.2.1 Scheduling notations

In this section, we present scheduling notations in three aspects. Firstly, trains scheduling notations are described in terms of movements. Then, scheduling notations for the four subsets of movements are separately expressed in three cases: initial station, transfer station and arrival station. At last, potential scheduling time intervals for trains are formulated to detect the potential conflicts.

Trains scheduling notations A train t at least consists of two movements $[m^t_1, m^t_2]$, one entering movement from the external line to the internal line and one leaving movement in the opposite direction passing along the outbound path. The reference arrival and departure times of a train separately correspond to the ending time of entering movement and the beginning time of leaving movement. For the train shown in figure 2.6, $\beta_{m_1}^{\mathrm{ref}}$ is given as the arrival time of the train and $\alpha_{m_2}^{\mathrm{ref}}$ is given as the departure time of the train. The order of movements ensures $\beta_{m_1}^{\mathrm{ref}} \leq \alpha_{m_2}^{\mathrm{ref}}$. The actual time interval of the movement m is defined by $[\alpha_m, \beta_m[$ with $\alpha_m, \beta_m \in \mathbb{N}$ and $\alpha_m < \beta_m$. The time duration used for a movement passing through railway station is defined by $S = \beta_m - \alpha_m$. In our case study, the length of this time interval is fixed to $S = 5$ minutes. The train occupies the allocated internal line during the interval $[A_t, B_t[$, such that:

$$
\begin{aligned}
A_t &= \alpha_{m_1^t} && (2.1)\\
B_t &= \alpha_{m^t_{\mathrm{M}^t}} = \beta_{m^t_{\mathrm{M}^t}} - S && (2.2)
\end{aligned}
$$

Obviously, every movement of the train occurs during the time interval $[A_t, B_t + S[$:

$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t, [\alpha_m, \beta_m[\subset [A_t, B_t + S[\tag{2.3}$$

Fig. 2.6 Trains' activities

Movements scheduling notations We discuss the movements in three cases.

- If the railway station is the initial station of the train, only the departure time $\alpha_{m_2}^{\mathrm{ref}}$ is given. According to the customer service time on the platform, the railway station planners add the reference arrival time $\beta_{m_1}^{\mathrm{ref}}$. The passengers have not yet boarded the train during the entering movement. So the train needs not to be strictly punctual. Instead, the arrival time can be adjusted within a certain time interval L before the reference arrival time to ensure the service time on the platform.

  The entering movement without passengers is called *technical entering movement* $m \in \mathbb{M}^{\ominus}$.

$$\forall m \in \mathbb{M}^{\ominus}, \exists \beta_m^{\mathrm{ref}} \in \mathbb{N} \qquad \text{s.t. } \beta_m^{\mathrm{ref}} - L \leq \beta_m \leq \beta_m^{\mathrm{ref}} \tag{2.4}$$

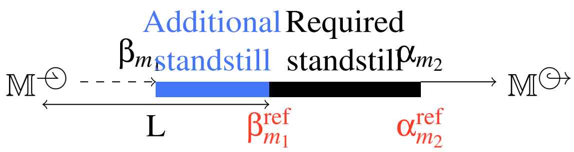


Fig. 2.7 Technical entering movement and commercial leaving movement

The leaving movement with passengers is called *commercial leaving movement* $m \in \mathbb{M}^{\ominus\rightarrow}$.

$$\forall m \in \mathbb{M}^{\ominus\rightarrow}, \alpha_m = \alpha_m^{\text{ref}} \tag{2.5}$$

- If the railway station is the arrival station of the train, only the arrival time $\beta_{m_1}^{\text{ref}}$ is given. According to the customer service time on the platform, railway station planners set the reference departure time $\alpha_{m_2}^{\text{ref}}$. Passengers have already got off the train during the leaving movement. So the train needs not to be strictly punctual. Instead, the departure time can be adjusted within a certain time interval L after the reference departure time to ensure the service time on the platform.

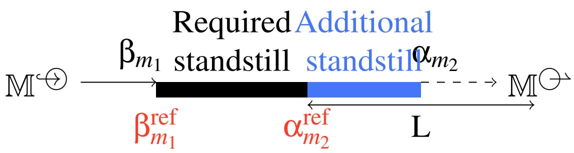


Fig. 2.8 Commercial entering movement and technical leaving movement

The leaving movement without passengers is a *technical leaving movement* $m \in \mathbb{M}^{\ominus\rightarrow}$.

$$\forall m \in \mathbb{M}^{\ominus\rightarrow}, \exists \alpha_m^{\text{ref}} \in \mathbb{N} \qquad \text{s.t. } \alpha_m^{\text{ref}} + L \geq \alpha_m \geq \alpha_m^{\text{ref}} \tag{2.6}$$

The entering movement with passengers is called *commercial entering movement* $m \in \mathbb{M}^{\rightarrow\oplus}$.

$$\forall m \in \mathbb{M}^{\rightarrow\oplus}, \beta_m = \beta_m^{\text{ref}} \tag{2.7}$$

- If the railway station is a transfer station of the train, the departure and arrival times are both given and invariable, as shown in the equations (2.7) and (2.5).

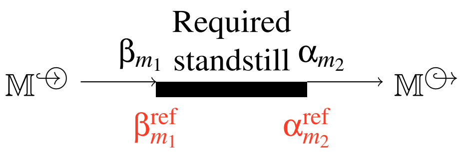


Fig. 2.9 Commercial entering movement and commercial leaving movement

Potential scheduling time interval The rescheduling problem in real-time faced by railway station managers is called conflict detection and resolution (CDR) problem. Given a timetable, the traffic disruptions occurred and the real-time position of the trains, the CDR problem consists of rescheduling trains such that the deviation from the timetable is minimized and the new schedule is compatible with the actual train positions. Different from the rescheduling problem in real-time, the scheduling and routing problem off-line cannot be simply described as CDR problem. As the timetable is given without allocated paths and internal lines, the routing processing becomes an underlying triggering factor of the traffic conflicts. So the detection of conflicts is executed in the dynamic state.

Instead, the potential conflicts can be detected according to the flexible interval L before the routing processing. As mentioned above, there are four subsets of movements $\mathbb{M} = \mathbb{M}^{\ominus\rightarrow} \bigcup \mathbb{M}^{\rightarrow\ominus} \bigcup \mathbb{M}^{\ominus\rightharpoonup} \bigcup \mathbb{M}^{\rightharpoonup\ominus}$ with adjustable time interval of technical movements indicated in equations (2.4)-(2.6). The earliest starting time of movement m is represented by $\alpha_m{}^{Early}$. The latest arrival time of movement m is defined as $\beta_m{}^{Late}$.The potential scheduling time interval of trains can be resumed in Table 2.3.

Type of movement	$\alpha_m{}^{Early}$	$\beta_m{}^{Late}$
$\mathbb{M}^{\ominus\rightarrow}$	α^{ref}	β^{ref}
$\mathbb{M}^{\rightarrow\ominus}$	α^{ref}	β^{ref}
$\mathbb{M}^{\ominus\rightharpoonup}$	α^{ref}	$\beta^{ref}+L$
$\mathbb{M}^{\rightharpoonup\ominus}$	$\alpha^{ref}-L$	β^{ref}

Table 2.3 The potential scheduling time interval of movements

The earliest entering time of train t is the earliest starting time of the first movement of the train m_1^t, shown as $A_t{}^{Early} = \alpha_{m_1^t}{}^{Early}$. The latest leaving time of train t is the latest arrival time of the last movement $m_{\mathbf{M_t}}^t$, shown as $B_t{}^{Late} = \beta_{m_{\mathbf{M_t}}^t}{}^{Late} - S$. So the potential scheduling time interval for train t is $[A_t{}^{Early}, B_t{}^{Late}[$.

The increasing of L helps to avoid the conflicts on the paths but intensifies the conflicts on internal lines. The flexible interval L is discussed in details in section 5.2. In our problem, L proposed by the railway station is 60 minutes. Therefore we will use the given departure and arrival time, taking full advantage of the time flexibility, and determine feasible paths for each of the trains passing through the railway station.

2.4.2.2 Routing Notations

As described in section 2.2, a train passing through the railway station is given a pair of external lines and need to be assigned to a complete path including inbound path, internal line and outbound path. As explained in section 2.4.1, we assign separately the three parts of the complete path to the activities of trains. Moreover, the internal line is occupied during the whole time interval of the train $[A_t, B_t[$. The external line and path are reserved during the time interval of the movement $[\alpha_m, \beta_m[$. So the internal line is assigned to the train; the external line and paths are assigned to the movements of the train.

For a train t consisting of the movements set $[m_1^t, m_2^t]$, the pair of external lines given to the train t are represented by $\{l^e_{m_1^t}, l^e_{m_2^t}\}$, and we have $l^e_m \in \mathbb{L}^e$. The train enters the railway station from the external line $l^e_{m_1^t}$ and leaves by the external line $l^e_{m_2^t}$.

The internal line allocated to train t is denoted by $\lambda_t \in \mathbb{L}^i$. All movements of the train must arrive at or depart from the same internal line λ_t.

The inbound path allocated to the train t is denoted by $p_{m_1^t} \in \mathbb{P}$. As switches of a path are described from internal lines to external lines, the first switch of the inbound path connects with the internal line of train λ_t. The last switch connects with the given external line $l^e_{m_1^t}$. Similarly, the outbound path of the train t is denoted by $p_{m_2^t} \in \mathbb{P}$ linking with the internal line λ_t and the external line $l^e_{m_2^t}$. We have obviously:

$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t,$$

$$s_1^{p_m} = \zeta(\lambda_t) \tag{2.8}$$

$$s_{\mathsf{Sp_m}}^{p_m} = \zeta(l^e_m) \tag{2.9}$$

which restricts the number of possible paths for the movement m.

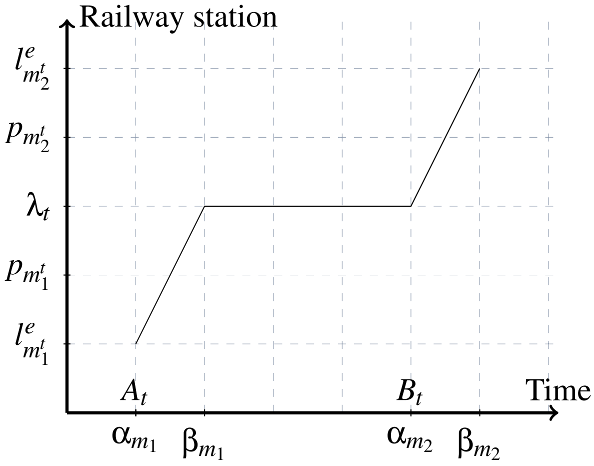


Fig. 2.10 Trains routing graph in the railway station

The actual trains' activities can be described by Figure 2.10 in terms of time and space. The horizontal axis represents the real time. The vertical axis represents the railway station resources allocated to trains' activities. Logically, the paths $p_{m^t_1}$ and $p_{m^t_2}$ in the chart represent a space interval linking the external lines and internal lines, separately $(l^e_{m^t_1}, \lambda_t)$ and $(\lambda_t, l^e_{m^t_2})$. The train enters from the external line $l^e_{m^t_1}$ at time α_{m_1} and passes along the inbound path $p_{m^t_1}$ to arrive at the internal line λ_t at time β_{m_1}. After the services at the platform during the time interval $[\beta_{m_1}, \alpha_{m_2}[$, the train departs from the internal line λ_t, passes along the outbound path $p_{m^t_2}$ and leaves the railway station from the external line $l^e_{m^t_2}$ at time β_{m_2}.

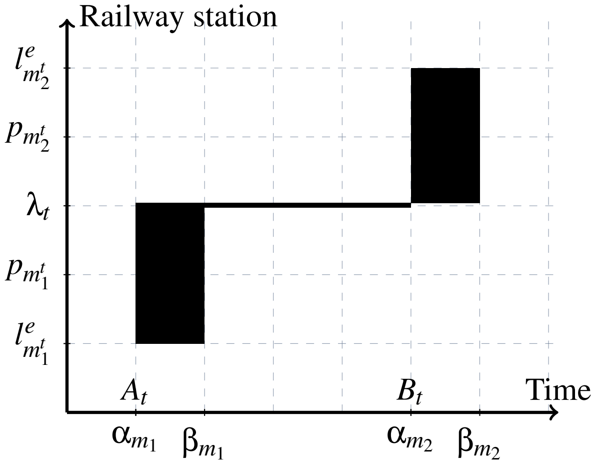


Fig. 2.11 Occupation of routing resources

To ensure the security and continuity of movements, the train t reserves the entering external line $l^e_{m^t_1}$, inbound path $p_{m^t_1}$ and internal line λ_t during the whole entering movement $[\alpha_{m_1}, \beta_{m_1}[$. Similarly, the internal line λ_t, outbound path $p_{m^t_2}$ and leaving external line $l^e_{m^t_2}$ are reserved by the train t during the whole leaving movement $[\alpha_{m_2}, \beta_{m_2}[$. The allocated internal line is not only reserved during the entering and leaving movements, but also reserved during the service on the platform. Totally, the internal line is reserved during $[A_t, B_t[$. So the occupation of routing resources is shown in Figure 2.11.

2.4.2.3 Coupling and uncoupling

If two trains have to be coupled at the railway station, the first train called *"leading train"* stops at the internal line, while the other train called *"following train"* has to be coupled onto the leading train. So the coupling of trains consists of two entering movements, one leaving movement and the standstill on the internal line from the ending of the first entering movement until the beginning of the leaving movement. The two entering movements are separately assigned to two inbound paths leading to the same internal line on which the coupling operation is carried out. According to the position of the two trains, the order of the two entering movements must be verified by the following constraints.

$$\forall m \in \mathbb{M}^t, \text{s.t. } \mathbb{M}^t = [m^t_1, m^t_2, m^t_3],$$

$$\beta_{m_1} \leq \alpha_{m_2} \tag{2.10}$$

$$\beta_{m_2} \leq \alpha_{m_3} \tag{2.11}$$

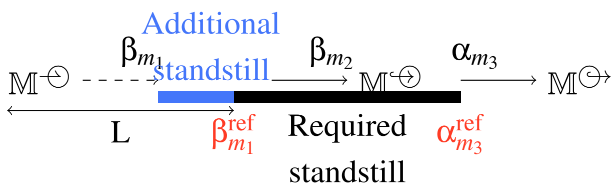


Fig. 2.12 Coupling operation

A similar situation occurs if a train has to be uncoupled into two parts. The uncoupling of trains consists of one entering movement, two leaving movements and the standstill on the internal line from the ending of the entering movement until the beginning of the second leaving movement. The two leaving movements are separately assigned to two outbound paths starting from the same internal line on which the uncoupling operation is carried out. According to the position of two trains, the order of two leaving movements must be assured.

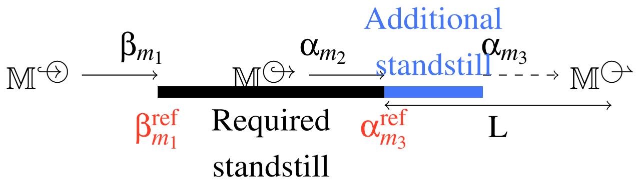


Fig. 2.13 Uncoupling operation

2.4.3 Compatibility of resources

The compatibilities of three element resources (external line, switch and internal line) are considered in the railway station. Each resource cannot be shared by more than one train during the same time interval. With the given external lines allocated to trains and the fixed reference time of commercial movements, we make the hypothesis that there is no conflict on the external lines between commercial movements. This hypothesis is guaranteed by the preprocessing step in the hybrid method explained in section 5.3.1.

Conflicts between trains on internal lines are avoided by the following constraints, expressing that one internal line cannot be occupied by two trains during the same time interval:

$$\forall t, t' \in \mathbb{T} \text{ s.t. } \lambda_t = \lambda_{t'},$$
$$[A_t, B_t[\cap [A_{t'}, B_{t'}[= \varnothing \quad (2.12)$$

Conflicts between movements on switches are eliminated in the same way: two movements using paths containing a common switch cannot be scheduled during the same time interval:

$$\forall s \in \mathbb{S}, \forall m, m' \in \mathbb{M} \text{ s.t. } s \in \mathbb{S}^{p_m} \cap \mathbb{S}^{p_{m'}}$$
$$[\alpha_m, \beta_m[\cap [\alpha_{m'}, \beta_{m'}[= \varnothing \quad (2.13)$$

2.4.4 Preference list of internal lines $\mathbb{L}_t^{Pref}$

To solve the platforming problem, many factors need to be taken into account, such as customer services and railway station usual practices. The "preference list of internal lines" is proposed to meet these requirements for each train.

Length of train $\mathbb{L}_{L_t}^{Comp}$ The length of trains must be taken into account as a criterion to choose an internal line. Three kinds of train lengths are defined as $L_t \in \{short, medium, long\}$, as well as three kinds of internal lines' lengths. For a train t of length L_t, the compatible list of internal lines $\mathbb{L}_{L_t}^{Comp} = \{l_1^i, l_2^i, ..., l_{\mathsf{L_{L_t}}}^i\}$ with cardinal number L_{L_t} is computed depending on the length of internal lines in an obvious way. So we always have $\mathbb{L}_{Long}^{Comp} \subset \mathbb{L}_{Medium}^{Comp} \subset \mathbb{L}_{Short}^{Comp}$.

Direction of train $\mathbb{L}_t^{PrefD}$ Service considerations towards the passengers indicate that trains going towards a given direction all leave from the same group of internal lines. The trains' direction is defined by the direction of its commercial movements which enforce the direction of corresponding technical movements. Movements are divided into 18 directions depending on the nature of trains (TGV or TER), their origin and destination railway stations. We denote by D_m the direction of the movement m. For the movement m in the direction D_m, the direction preference list of internal lines is defined as an ordered set $\mathbb{L}_{D_m}^{Pref} = [l_1^i, l_2^i, ..., l_{\mathsf{L_{D_m}}}^i]$ with the cardinal number L_{D_m}. The allocation priority of l_i with index i is given by railway station manager and formulated as $VP_{m,l_i} = L_{D_m} - i + 1$. The internal line l_1^i has the highest allocation priority $VP_{m,l_1^i} = L_{D_m}$ in the set.

If the train t contains more than one commercial movement, the train's direction preference list of internal lines is the intersection of all movements' direction preference lists.

$$\forall t \in \mathbb{T}, \mathbb{L}_t^{PrefD} = \cap_{m \in \mathbb{M}^t} \mathbb{L}_{D_m}^{Pref} \tag{2.14}$$

The priority order of internal lines for the train t in the new combined set $\mathbb{L}_t^{PrefD}$ is measured by its priority value for the movement of the train $\sum_{m \in \mathbb{M}^t} VP_{m,l_i^i}$. For example, the train t contains two commercial movements $[m_1^t, m_2^t]$ which have separately the direction preference set of internal lines $\mathbb{L}_{D_{m_1^t}}^{Pref} = [7,1,2,3,4,5,6,14,12]$ and $\mathbb{L}_{D_{m_2^t}}^{Pref} =$

$[15,16,17,1,2,3,4,5,6,7,8]$. After the calculation of the intersection and the combined preference value shown in Figure 2.14, the trains' direction preference list $\mathbb{L}_t^{PrefD}$ is $[1,2,3,7,4,5,6]$.

$\mathbb{L}_{D_{m_1^t}}^{Pref}$

Internal line	Priority value
7	9
1	8
2	7
3	6
4	5
5	4
6	3
14	2
12	1

$+$

$\mathbb{L}_{D_{m_2^t}}^{Pref}$

Internal line	Priority value
15	11
16	10
17	9
1	8
2	7
3	6
4	5
5	4
6	3
7	2
8	1

$=$

$\mathbb{L}_t^{PrefD}$

Internal line	Combined priority
1	16
2	14
3	12
7	11
4	10
5	8
6	6

Fig. 2.14 Trains' direction preference set of internal lines

Combined preference list $\mathbb{L}_t^{Pref}$ The preference list of internal lines for a train t is defined as an ordered set $\mathbb{L}_t^{Pref} = [l_1^i, l_2^i, \ldots, l_{\mathrm{L}_t^{\mathrm{Pref}}}^i]$ with the cardinal number L_t^{Pref} depending on the length of trains L_t and the directions of all relevant movements $D_{m \in \mathbb{M}^t}$. The order of internal lines in the set $\mathbb{L}_t^{Pref}$ is the same as the order in the set $\mathbb{L}_t^{PrefD}$ as shown in Figure 2.15.

$$\begin{aligned} \forall t \in \mathbb{T}, \mathbb{L}_t^{Pref} &= \cap_{m \in \mathbb{M}^t} \mathbb{L}_{D_m}^{Pref} \cap \mathbb{L}_{L_t}^{Comp} \\ &= \mathbb{L}_t^{PrefD} \cap \mathbb{L}_{L_t}^{Comp} \end{aligned} \tag{2.15}$$

$\mathbb{L}_t^{PrefD}$

Internal line	Combined priority
1	16
2	14
3	12
7	11
4	10
5	8
6	6

$+$

$\mathbb{L}_{L_t}^{Comp}$

Internal line
1
2
3
4
5
6

$=$

$\mathbb{L}_t^{Pref}$

Internal line	Priority value
1	6
2	5
3	4
4	3
5	2
6	1

Fig. 2.15 Trains' preference list of internal lines

2.4.5 Safety analysis: Headway

In railway network, headway is one of the most important safety evaluation criteria. In this section, we analyse the headway between trains on conflicting resources (switches, internal lines) and examine the safety of the representation scheme proposed in the previous sections.

In order to avoid collisions, headway is a measurement of the distance or time between vehicles in a transit system. It is most commonly measured as the distance from the tip of one vehicle to the tip of the next one behind it, expressed as the time it will take for the trailing vehicle to cover that distance.

Trains take a very long time to stop, covering long stretches of ground in the process. The amount of ground covered is often much longer than the range of the driver's vision. If a train is stopping on the tracks in front, the train behind it will probably see it far too late to avoid a collision. To have visual contact as method to avoid collision is done only at low speeds like 40 km/h. A key safety factor of train operations is to space the trains out by at least this distance, the *"brick-wall stop"* criterion: in order to signal the trains in time to allow them to stop, the railways used to place workmen on the lines who timed the passing of a train, and then signalled any following trains if a certain elapsed time had not passed. This is why train headways are normally measured as tip-to-tip times, because the clock was reset as the engine passed the workman.

When remote signalling systems were invented, workmen were replaced with signal towers at given locations along the track. This broke the track into a series of "blocks" between the towers. Trains were not allowed to enter a block until the signal said it was clear, thereby guaranteeing a minimum of one block's headway between the trains.

Based on practical railway management rules, the minimum safe headway measured tip-to-tail is defined by the braking performance:

$$T_{min} = t_r + \frac{kV}{2}\left(\frac{1}{a_f} - \frac{1}{a_l}\right) \tag{2.16}$$

where:

- T_{min} is the minimum safe headway, in seconds
- V is the speed of the vehicles
- t_r is the reaction time, the maximum time it takes for a following vehicle to detect a malfunction in the leader, and to fully apply emergency brakes.
- a_f is the maximum braking deceleration of the follower.
- a_l is the maximum braking deceleration of the leader. For brick-wall considerations, a_l is infinite and this consideration is eliminated.
- k is an arbitrary safety factor, greater than or equal to 1.

The tip-to-tip headway is simply the tip-to-tail headway plus the length of the vehicle, expressed in time:

$$T_{tot} = \frac{L}{V} + t_r + \frac{kV}{2}\left(\frac{1}{a_f} - \frac{1}{a_l}\right) \tag{2.17}$$

where:

- T_{tot} is the time for vehicle and headway to pass a point
- L is the vehicle length

In railway stations, headway time is defined to make sure trains pass safely through the local network without intermediate stops. As railway paths criss-cross in the dense local network around the station, intermediate stops of train may block large area in station and cause expansion of delay propagation. A train accomplishes an entering or leaving movement within 4 minutes in Bordeaux St Jean station. A minimum headway in time interval is required for two trains on conflicting paths, for safety and signaling reasons. In our problem, the duration of a movement between the external line and the internal line is proposed, by the railway station, as $S = 5$ minutes including 1-2 minutes buffer time.

In this context, headway time on switches and internal lines is discussed in following paragraphs. As the relative directions of trains impact on headway time, we analyse headway separately between two trains in the same direction and in opposite directions.

Headway time on Switches Firstly, we analyse the headway time between two trains running sequentially on conflicting path without spare time in the same direction, for example from internal line to external line. The leaving movements m_1 and m_2 pass along the same path $p = [s_1^p, s_2^p, s_3^p, s_4^p, s_5^p]$. The headway time on the common switch is S as shown in Figure (2.16).

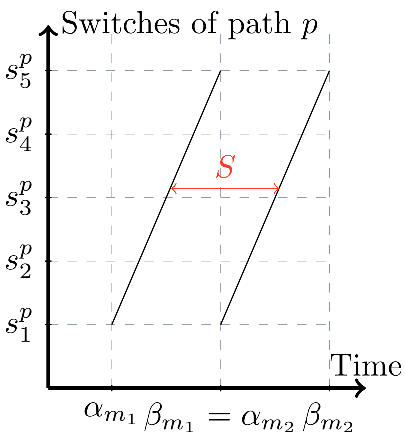


Fig. 2.16 Headway time between two trains on conflicting path in the same direction

On the other hand, two trains running sequentially on conflicting path without spare time in the opposite direction have less headway time. The leaving movement m_1 of the train t_1 is executed before the entering movement m_2 of the train t_2. The headway time on the common switch depending on the position of the switch in the path is between 0 and $2*S$ as shown in

Figure (2.17). The headway time is $2*S$ on the first switch s_1^p linking with the internal line. The headway time is 0 on the last switch s_5^p linking with the external line.

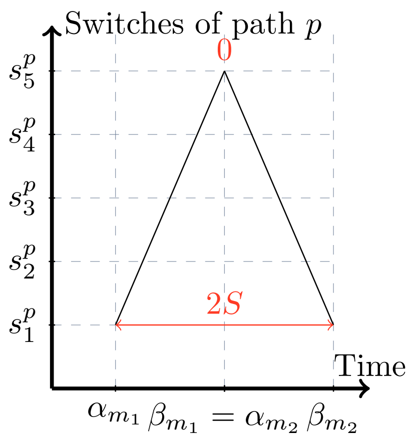


Fig. 2.17 Headway time between two trains on conflicting path in opposite directions

Headway time on internal Lines In this section, we analyse headway time between trains using the same internal line. As the internal line and the corresponding path are chosen, the internal line will be reserved from the train entering the railway station until departure of leaving movement. But the actual dwell time of train on the internal line exclude the movement time interval S. If $p_{m_2^t}$ and $p_{m_1^{t'}}$ don't have common switches, m_2^t and $m_1^{t'}$ can pass the railway station in parallel. As a result, the headway time on the internal line is S as shown in Figure (2.18).

If $p_{m_2^t}$ and $p_{m_1^{t'}}$ contain conflicting switches, m_2^t and $m_1^{t'}$ must pass the railway station sequentially. So the headway time on the internal line is $2*S$ as shown in Figure (2.19).

Based on the analysis of headway on switches and internal lines, we can see that no time intersection exists between trains sharing the same rail resources. **So the formalization of problem ensures that trains can circulate safely through stations' local networks without midway stops.**

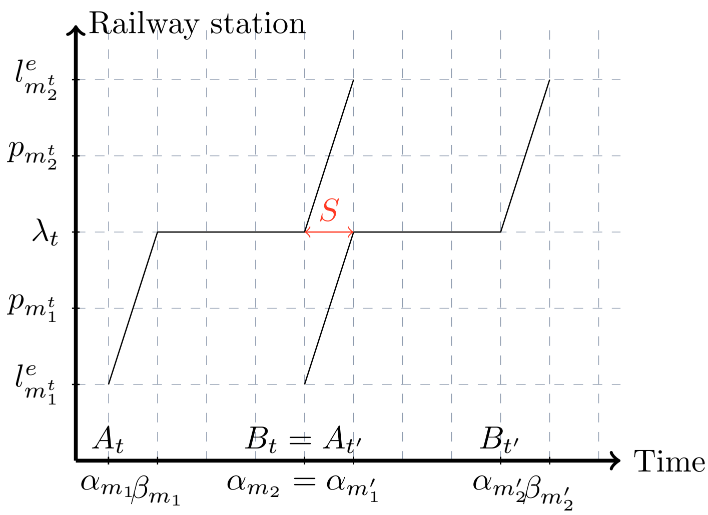


Fig. 2.18 Two trains on the same internal line in the same direction

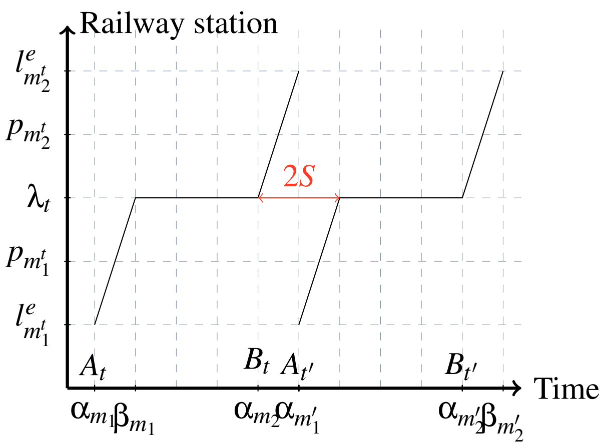


Fig. 2.19 Two trains on the same internal line in the opposite directions

2.4.6 Analogy Between Manufacturing Production Systems and Railway Stations

In this section, we study train platforming problems in railway stations in the form of job-shop problems. A great deal of research has been focused on solving the job-shop problem, over the last fifty years, resulting in a wide variety of approaches. So we try to trace an analogy between train platforming problem and job-shop scheduling problem. The analogy may help us to understand the complexity of our problem.

The scheduling of flow shops with multiple parallel machines per stage, usually referred to as the Hybrid Flow Shop (HFS), is a complex combinatorial problem encountered in many real world applications. HFS are common manufacturing environments in which a set of n jobs are to be processed in a series of m stages. There are a number of variants, all of which have most of the following characteristics in common:

- The number of stages m is at least 2.
- Each stage has at least one machine.
- All jobs are processed following the same production flow: stage 1, stage 2,...,stage m.

The problem is to find a schedule which optimizes a given objective function. The HFS problem is, in most cases, NP-hard. For instance, HFS restricted to two processing stages, even in the case when one stage contains two machines and the other one a single machine, is NP-hard, after the results of Gupta (1988).

To learn about the problem from a global perspective, we can describe train platforming problem as 5-stages Hybrid Flow Shop scheduling problem as shown in Figure (2.20), and the allocation of resource in each stage is highly related with each other.

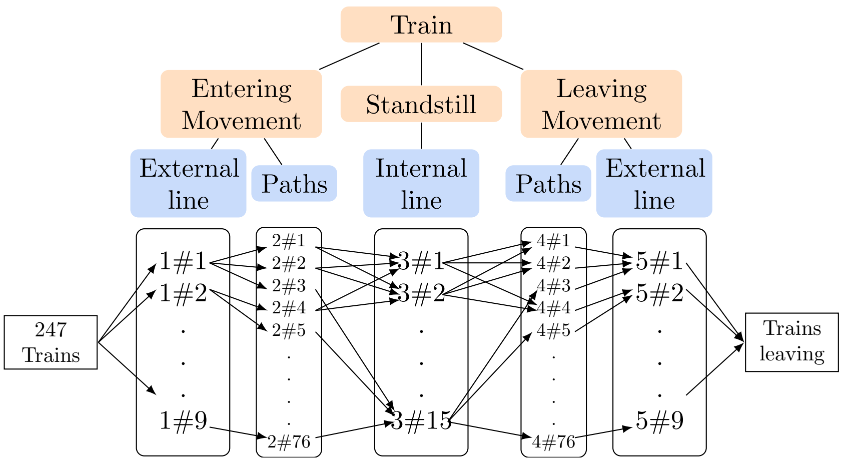


Fig. 2.20 Analogy Flow-Shop/Railway station

Here we take Bordeaux St-Jean station as an example. The first stage contains 10 external lines as machines which are predefined for each job. On the second stage, the parallel machines are 76 different paths consisting of 17 switches. The chosen path for each job has to be accessible by its predefined external line. 15 internal lines are considered as the parallel machines on the third stage. The selected internal line for each job has to be accessible by its path chosen on the second stage. The fourth stage contains the same machines as the second stage. The path chosen on the fourth stage allows the connection with the selected internal line on the third stage and the connection with the predefined external line on the last stage. The last stage contains the same machines as the first stage. The external lines on the fifth stage are also predefined for each job.

Trains passing through the railway station are defined as jobs. Every train consists of two or three movements, at least one movement entering and one movement leaving the railway station. For trains including three movements, a separation or combination of trains need to be added as an extra operation on the third stage. So the operation for special trains is different. Otherwise, processing on the first and second stages will be finished within 5 minutes; processing on the fourth and fifth stage will be also finished within 5 minutes; the setup time on the third stage depends on the finish time of second stage and the start time of fourth stage for each train. So the setup time on the third stage is variable.

Finally, one must notice that, despite their similarity, there are significant differences between platforming problem and 5-stages Hybrid Flow Shop scheduling problem, such as the absence of inter-machine buffers in the platforming problem, called no-store or blocking constraint (Hall and Sriskandarajah (1996)).

Our train platforming problem can be identified as a 5-stages no-wait Hybrid Flow Shop scheduling problem with no-identical jobs and no-identical parallel machines. The term no-identical job means that every train has a target arrival or departure time. So train scheduling issues can not be considered as trains ordering problems. The term no-identical parallel machine means that choice of internal lines and paths depends on trains' preferences, lengths and connections between paths and internal lines.

Based on the knowledge of Hybrid Flowshop, our problem is at least an NP-complete problem, when we try to find a feasible solution. Since the feasibility is not evident in some difficult cases, the minimization of infeasible trains will come up to be our objective. In fact, our problem is a multi-objectives problem with another objective: minimization of commercial delays. In this case, our problem becomes a NP-hard problem. Furthermore, to find a global optimum, we need to solve the routing and scheduling problem at the same time. As a bi-dimensional problem. we cannot solve independently the routing and scheduling problems as we solve flow-shop problems.

However, as a field widely investigated by researchers, the MIP format applied in flow-shop scheduling problems inspires us to design a MILP model for our platforming problem, as described in the next chapter.

2.5 Conclusion

In this section, three objectives are achieved:

1. Our problem is positioned in context and in theory. Train platforming problem is a complement sub-problem in railway networks management. We concentrate on dispatching dense rail transport in complex local rail networks, typically in interchange central stations. We describe the train platforming problem as a bi-dimensional feasibility problem including scheduling and routing. The timetable to be generated must ensure that no pair of trains is conflicting over paths and platforms, while allowing the coupling and uncoupling of trains at a platform and respecting their preferences of platforms and the accessibility of complete path of trains.

2. Our researches are oriented based on a suggestive literature review. The literature review specially focusing on train platforming problems is summarized and ended by the discussions on current state-of-the-art limitations. Then we provide a literature review about on-line rescheduling problem which has been widely investigated by researchers. On one hand, the similarities between on-line rescheduling problem and off-line platforming problem inspire us to borrow two ideas: accumulative methods and problem size reduction techniques. On the other hand, an innovative algorithm structure is proposed in Section 2.3.3 based on the particularities of our train platforming problem. The detailed explanation of algorithm structure can be found in Section 5.

3. Our train platforming problem is formalized in four aspects: station layout, trains' activities, resource compatibility constraints and internal lines preferences. In order to validate this formalization, safety criteria are evaluated in terms of headway time. At last, our problem is presented in the form of flow-shop variant: 5-stages no-wait Hybrid Flow Shop scheduling problem. This comparison helps us to understand problem characteristics and complexity matters. Furthermore, the MIP format applied in flow-shop scheduling problems can be borrowed to propose a MIP model in the next chapter.

CHAPTER 3

Decision model: generation of Feasible timetable

In the previous section, we have described our train platforming problem in the form of a **5-stages no-wait Hybrid Flow Shop** scheduling problem. In that case, we can borrow some resolution methods from corresponding literature. In this section, we propose a mixed integer linear programming formulation based on works on flow-shop scheduling problems.

As introduced by Jain and Meeran (1998), it has been recognised by many researchers that scheduling problems can be solved optimally using mathematical programming techniques. One of the most common forms of mathematical formulation for job shop scheduling problem is the mixed integer linear programming (MIP) format of Manne (1960) which is highlighted below. The MIP format is simply that of a linear program with a set of linear constraints and a single linear objective function, but with the additional restriction that some of the decision variables (y_{ipk}) are integers:

Minimise C_{max} subject to:

$$\text{starting time} \quad \{i,p\} \in Jobs, \{k,h\} \in Machines \quad t_{ik} \geq 0, \tag{3.1}$$

$$\text{precedence constraint} \quad \text{if } O_{ih} \text{ precedes } O_{ik} \quad t_{ik} - t_{ih} \geq \tau_{ik} \tag{3.2}$$

$$\text{disjunctive constraint} \quad \text{if } O_{ik} \text{ precedes } O_{pk}, \; y_{ipk} = 1$$

$$t_{pk} - t_{ik} + R \cdot (1 - y_{ipk}) \geq \tau_{ik} \tag{3.3}$$

$$\text{otherwise, } y_{ipk} = 0$$

$$t_{ik} - t_{pk} + R \cdot y_{ipk} \geq \tau_{pk} \tag{3.4}$$

$$\text{where } R > \left(\sum_{i=1}^{n} \sum_{k=1}^{m} \tau_{ik} - min(\tau_{ik})\right)$$

Here the integer variables are binary and are used to implement disjunctive constraints. R is a large number and Jain and Meeran (1991) indicate that in order for the feasible region to be properly defined R has to be greater than the sum of all but the smallest of the processing times.

Based on this MIP format, we propose a MILP, called *decision model*, for our train platforming problem which consists of scheduling and routing as a bidimensional problem. In this section, the decision model is described in three steps: parameters, variables and constraints. Some reasonable improvements of model are proposed and realized at the end of this section.

3.1 Parameters

- R is a sufficiently big constant.
- L is the adjustable time interval for technical movements.
- α_m^{ref} is the reference starting time of the movement m.
- β_m^{ref} is the reference ending time of the movement m.
- S is the time duration used for a movement passing through the railway station. In our context, $S = 5$ minutes.

- $Y^P_{p,p'}$ identifies a pair of conflicting paths. If $p \cap p' \neq \varnothing$, $Y^P_{p,p'} = 1$. Otherwise, $Y^P_{p,p'} = 0$.

- $C^{refM}_{m,m'}$ probes the potential conflicts between two movements m and m'.
If $[\alpha_m{}^{Early}, \beta_m{}^{Late}) \cap [\alpha_{m'}{}^{Early}, \beta_{m'}{}^{Late}) \neq \varnothing, C^{refM}_{m,m'} = 1$. Otherwise, $C^{refM}_{m,m'} = 0$.

- $C^{refT}_{t,t'}$ probes the potential conflicts between two trains t and t'. If $[A_t{}^{Early}, B_t{}^{Late}) \cap [A_{t'}{}^{Early}, B_{t'}{}^{Late}) \neq \varnothing$, $C^{refT}_{t,t'} = 1$. Otherwise $C^{refT}_{t,t'} = 0$.

3.2 Variables

In the practical situation, the arrival and leaving times of trains are measured in minutes. The scheduling decision variables are thus defined as integers with units of minutes, characterizing a discrete-time scheduling problem.

- α_m is the actual starting time of the movement m.
- β_m is the actual ending time of the movement m, $\alpha_m + S = \beta_m$.
- A_t is the starting time of occupation of the internal lines by the train t.
- B_t is the ending time of occupation of the internal lines by the train t.

The routing decision variables are defined as binary variables.

- $X^{L^iT}_{l,t}$ identifies the internal lines allocated to the train t. If $\lambda_t = l$, $X^{L^iT}_{l,t} = 1$. Otherwise, $X^{L^iT}_{l,t} = 0$.

- $X^{PM}_{p,m}$ identifies the path allocated to movement m. If $p = p_m$, $X^{PM}_{p,m} = 1$. Otherwise, $X^{PM}_{p,m} = 0$.

- $X^{OrderT}_{t,t'}$ identifies the time order of two trains using the same line. If t circulates before t', $X^{OrderT}_{t,t'} = 1$. Otherwise, $X^{OrderT}_{t,t'} = 0$.

- $X^{OrderM}_{m,m'}$ identifies the time order of two movements using two conflicting paths. If m circulates before m', $X^{OrderM}_{m,m'} = 1$. Otherwise, $X^{OrderM}_{m,m'} = 0$.

- $X_m^{CancelM}$ identifies the cancellation of the movement m. If m is cancelled, $X_m^{CancelM} = 1$. Otherwise, $X_m^{CancelM} = 0$.

- $X_t^{CancelT}$ identifies the cancellation of the train t. If t is cancelled, $X_t^{CancelT} = 1$. Otherwise, $X_t^{CancelT} = 0$.

3.3 Constraints

Except the time constraints (2.1)-(2.7), we express the allocation constraints with the parameters and variables defined above.

Time interval of a train. In the railway station, two kinds of resources are considered: switches and internal lines. The switches are occupied during the time of movements. The lines are occupied by the train t during $[A_t, B_t]$. According to equation (2.3), the time interval of a train covers all the movements of the train, which can be formulated in a classical way as below:

$$A_t \quad = \alpha_{m_1^t} \tag{3.5}$$

$$B_t \quad = \beta_{m_{\mathrm{M}^t}^t} \tag{3.6}$$

Time constraints. To satisfy the passengers' demand, we cannot change the time of commercial movements. In contrast, the time interval of technical movements is permitted to move within an hour. The constraints (2.7) to (2.5) are expressed as follows:

$$\forall m \in \mathbb{M}^{\rightarrow\!\bigcirc} \quad \beta_m^{ref} - L \leq \beta_m \leq \beta_m^{ref} \tag{3.7}$$

$$\forall m \in \mathbb{M}^{\bigcirc\!\rightarrow} \quad \alpha_m^{ref} + L \geq \alpha_m \geq \alpha_m^{ref} \tag{3.8}$$

$$\forall m \in \mathbb{M}^{\hookrightarrow\!\bigcirc} \quad \beta_m = \beta_m^{ref} \tag{3.9}$$

$$\forall m \in \mathbb{M}^{\bigcirc\!\hookrightarrow} \quad \alpha_m = \alpha_m^{ref} \tag{3.10}$$

Preference of internal lines. Considering length, origin and destination of trains, we choose only one internal line from the preference list $\mathbb{L}_t^{Pref}$ for the train t. The order of the internal lines in the preference list is followed in the calculation process.

$$\forall t \in \mathbb{T}, \sum_{l_i \in \mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} = 1 \tag{3.11}$$

Allocation of paths. For all movements, one path must be allocated, and the chosen path must connect the given external line and the line allocated in the railway station as described in equations (2.8) and (2.9). All movements of a train share a single line in the railway station. If the movement passes on the external line l_e, we allocate an accessible internal line in the railway station $l_i \in \mathbb{L}_{l_e}^i$ and allocate a path $p \in \mathbb{P}^{(l_i,l_e)}$ connecting internal line l_i and external line l_e.

The relation between the allocation of paths and the allocation of internal lines is represented by the following constraints:

$$\forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \qquad \sum_{p \in \mathbb{P}^{l_e}} X_{p,m}^{PM} = 1 \tag{3.12}$$

$$\forall t \in \mathbb{T}, \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \cap \mathbb{M}^t, \forall l_i \in \mathbb{L}_{l_e}^i \qquad \sum_{p \in \mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} \geq X_{l_i,t}^{L^iT} \tag{3.13}$$

When a path $p \in \mathbb{P}^{(l_i,l_e)}$ is chosen for a movement $m \in \mathbb{M}^t$, we get $\sum_{p \in \mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} = 1$, but we cannot decide if the internal line l_i is chosen or not (i.e. if $X_{l_i,t}^{L^iT} = 0$ or 1). Conversely, when l_i is chosen, i.e. $X_{l_i,t}^{L^iT} = 1$, we get $\sum_{p \in \mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} = 1$.

Compatibility of lines. The constraints of occupation of lines (2.12) indicate that two trains cannot occupy the same line at the same time. This rule is expressed as follows:

$$\forall t, t' \in \mathbb{T}, t \neq t', \forall l \in \mathbb{L}^i, \quad B_t \leq A_{t'} + R \cdot (3 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT} - X_{t,t'}^{OrderT}) \tag{3.14}$$

$$\forall t, t' \in \mathbb{T}, t \neq t', \qquad X_{t,t'}^{OrderT} + X_{t',t}^{OrderT} = 1 \tag{3.15}$$

The constraint (3.14) indicates that if two trains t and t' are allocated to the same line l in the railway station and if the train t circulates before t', then the term $3 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT} - X_{t,t'}^{OrderT} = 0$. We have then $B_t \leq A_{t'}$. Otherwise this term is larger than zero, and the constraint (3.14) is relaxed. It is a classical way to linearise the constraints.

Compatibility of switches. The constraint of occupation of switches (2.13) indicates that two movements m and m' cannot pass the same switches at the same time. If two movements are allocated to two paths which have switches conflicts and the movement

m circulates before m', they must verify $\alpha_m + S = \beta_m \leq \alpha_{m'}$. Such constraint is enforced as above:

$$\forall m, m' \in \mathbb{M}, m \neq m', \forall p, p' \in \mathbb{P}, p \neq p', \text{ s.t. } Y^P_{p,p'} = 1,$$
$$\beta_m \leq \alpha_{m'} + R \cdot (3 - X^{PM}_{p,m} - X^{PM}_{p',m'} - X^{OrderM}_{m,m'}) \tag{3.16}$$
$$\forall m, m' \in \mathbb{M}, m \neq m', \qquad X^{OrderM}_{m,m'} + X^{OrderM}_{m',m} = 1 \tag{3.17}$$

Objective function. The objective we focus on is to minimize the lines' occupancy rate, which can be expressed as follows:

$$\min \sum_{t \in \mathbb{T}} B_t - A_t \tag{3.18}$$

3.4 Improvement of the mathematical model

3.4.1 Continuous-time model

The first major issue for scheduling problems concerns the time representation. Based on two different ways for time representation, all existing scheduling formulations can be classified into two main categories: discrete-time models and continuous-time models. An extensive discussion of discrete-time versus continuous time models for process scheduling is referred in Floudas and Lin (2004).

The time horizon of discrete-time scheduling formulations is divided into a number of time intervals of uniform durations, and events such as the arrival and leaving of trains are associated with the boundaries of these time intervals. To achieve a suitable approximation of trains scheduling problem, the duration of the time intervals is defined to one minute, in accordance with the train timetable showed to passengers. The division of the full time horizon into small length time interval leads to very large combinatorial problems of intractable size, especially for real-world problems.

Due to the aforementioned drawbacks of the discrete-time approach, research efforts in modeling scheduling problems relied on the continuous-time approach in the past decade. In these models, events can be potentially associated with any point in the continuous domain of time. Because of the possibility of eliminating a major fraction of the inactive event-time interval assignments using the continuous-time approach, the mathematical programming problems are usually of much smaller size and require less computational efforts for their resolution. However, because of the variable nature of the timings of the events, it becomes more challenging to model the scheduling process and the resulting mathematical models may exhibit more complicated structures compared to their discrete-time counterparts.

In our train scheduling problem, we want to use the continuous-time model with smaller computational size. But if we get a rational number as the leaving time of train (for example 234.173 minutes), that will be meaningless. Fortunately, based on the studies of the mathematical model, we can prove that all scheduling variables are integer-valued.

We divide the whole problem into two separate parts: routing problem and scheduling problem. We focus on the scheduling problem and suppose the routing issue is known. The scheduling corresponding equations (2.1, 2.2, 3.14 and 3.16) are formulated as equation (3.19), and equations (3.7) to (3.10) can be rewritten as form of equation (3.20):

$$A \cdot x \leq b \tag{3.19}$$

$$c \geq x \leq d \tag{3.20}$$

So (2.1) and (2.2) are rewritten in the form of (3.19) as (3.21, 3.22, 3.23) and (3.24). Resources compatibility constraints (3.14) and (3.16) are rewritten separately as (3.25) and (3.26).

$$\forall t \in \mathbb{T}, \quad A_t - \alpha_{m_1^t} \leq 0 \tag{3.21}$$

$$\forall t \in \mathbb{T}, \quad \alpha_{m_1^t} - A_t \leq 0 \tag{3.22}$$

$$\forall t \in \mathbb{T}, \quad B_t - \beta_{m_{\mathsf{Mt}}^t} \leq 0 \tag{3.23}$$

$$\forall t \in \mathbb{T}, \quad \beta_{m_{\mathsf{Mt}}^t} - B_t \leq 0 \tag{3.24}$$

$$\forall t, t' \in \mathbb{T}, t \neq t', \forall l \in \mathbb{L}^i, \quad B_t - A_{t'} \leq R \cdot (3 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT} - X_{t,t'}^{OrderT}) \tag{3.25}$$

$$\forall m, m' \in \mathbb{M}, m \neq m', \forall p, p' \in \mathbb{P}, \quad p \neq p' \quad \text{, s.t. } Y_{p,p'}^{P} = 1,$$

$$\beta_m - \alpha_{m'} \leq R \cdot (3 - X_{p,m}^{PM} - X_{p',m'}^{PM} - X_{m,m'}^{OrderM}) \tag{3.26}$$

Theorem Hoffman and Kruskal (1956). Let A be an integral $m \cdot n$ matrix, the polyhedron $P(A,b) = \{x : x \geq 0, A \cdot x \leq b\}$ is integral for all integral vectors $b \in \mathbb{Z}^m$ if and only if A is totally unimodular.

Theorem If A is totally unimodular then A^T also totally unimodular.

Based on the theorems above, if the matrix A or A^T generated from (3.21)-(3.26) is proven as totally unimodular matrix, scheduling variables can be solved as integers even if they are defined in continuous-time domain. Our matrix $A_{platforming}$ can be represented as below:

$$
A_{platforming} = \begin{array}{cc}
 & \begin{matrix} \alpha_{m_1} & \alpha_{m_2} & \dots & \beta_{m_1} & \beta_{m_2} & \dots & A_{t_1} & A_{t_2} & \dots & B_{t_1} & B_{t_2} & \dots \end{matrix} \\
\begin{matrix} (3.21) \\ (3.22) \\ \vdots \\ (3.23) \\ (3.24) \\ \vdots \\ (3.25) \\ (3.25) \\ \vdots \\ (3.26) \\ (3.26) \\ \vdots \end{matrix} &
\begin{bmatrix}
-1 & 0 & \dots & 0 & 0 & \dots & 1 & 0 & \dots & 0 & 0 & \dots \\
1 & 0 & \dots & 0 & 0 & \dots & -1 & 0 & \dots & 0 & 0 & \dots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\
0 & 0 & \dots & -1 & 0 & \dots & 0 & 0 & \dots & 1 & 0 & \dots \\
0 & 0 & \dots & 1 & 0 & \dots & 0 & 0 & \dots & -1 & 0 & \dots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\
0 & 0 & \dots & 0 & 0 & \dots & -1 & 0 & \dots & 0 & 1 & \dots \\
0 & 0 & \dots & 0 & 0 & \dots & 0 & -1 & \dots & 1 & 0 & \dots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\
0 & -1 & \dots & 1 & 0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots \\
-1 & 0 & \dots & 0 & 1 & \dots & 0 & 0 & \dots & 0 & 0 & \dots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots
\end{bmatrix}
\end{array}
$$

The first row of matrix $A_{platforming}$ includes all scheduling variables. The first column represents the indices of correspondent equations. $A_{platforming}$ is an $m \cdot n$ matrix of $\{0, 1, -1\}$, and b, c and d are positive integer m-vectors. Our goal is to prove that every vertex solution, the n-vector x, is integral. In our problem, x represents a vector including all scheduling variables.

Theorem Heller and Tompkins (1956). A matrix A is totally unimodular if

- each entry is 0,1 or -1;
- each column contains at most two non-zeroes;

- the set N of row indices of A can be partitioned into $N_1 \cup N_2$, so that in each column j with two non-zeroes we have $\sum_{i \in \mathbb{N}_1} a_{i,j} = \sum_{i \in \mathbb{N}_2} a_{i,j}$.

We can see that each row i of $A_{platforming}$ contains two non-zeroes (1 and -1), so each column of transpose matrix $A^T_{platforming}$ contains at most two non-zeroes. The set N of column indices of $A_{platforming}$ can be partitioned into $N_1 \cup N_2$. N_1 contains all columns, and $N_2 = \emptyset$. So the transpose matrix A^T accords with $\sum_{i \in \mathbb{N}_1} a_{i,j} = \sum_{i \in \mathbb{N}_2} a_{i,j} = 0$. i denotes row indices of A^T (column indices of A: scheduling variables). j denotes column indices of A^T (row indices of A: equations). Based on the theorem proposed by Heller and Tompkins (1956), we can say that our matrix $A^T_{platforming}$ is unimodular. According to the theorem, if A is totally unimodular then A^T also totally unimodular. So $A_{platforming}$ is also a totally unimodular matrix. Based on Hoffman and Kruskal's theorem, every vertex solution, the n-vector x, is integral. This kind of scheduling sub-problems is called **Network Linear Programs** (Guéret et al. (2000)).

Network Optimization is a special type of linear programming model. **They can be solved very quickly by Simplex method.** Problems whose linear program would have 1000 rows and 30,000 columns can be solved in a matter of seconds. **Moreover, they have naturally integer solutions.** By recognizing that a problem can be formulated as a network program, it is possible to solve special types of integer programs without resorting to the ineffective and time consuming integer programming algorithms. In Chapter 6, we use the same idea (Network Optimization) to design a tri-level optimization model to provide a local optimum in rather a short time (several seconds). Of course, these advantages come with a drawback: network models cannot formulate the wide range of models as linear and integer programs can. In our problem, routing sub-problem is not a network optimization problem. In the next section, we try to cut the useless routing constraints to reduce the problem size.

We propose to solve the train scheduling problem by the continuous-time approach. The scheduling decision variables α_j, A_t and B_t are defined in the continuous-time domain. In that case, the continuous-time model not only satisfies our computational requirement of the integer scheduling decision variables, but also require less calculation efforts.

3.4.2 Reduction of model

To improve the calculation performance, we seek to reduce the number of constraints. We design an indicator as the probe of potential conflicts between movements $C^{refM}_{m,m'}$ and between trains $C^{refT}_{t,t'}$. In this way, the constraints are created only for the movements and trains with potential conflicts. The undesired constraints are cut off. Four additional parameters need to be created as below.

- ${\alpha_m}^{Early}$ is the earliest starting time of the movement m.
- ${\beta_m}^{Late}$ is the latest arrival time of the movement m.
- ${A_t}^{Early} = \min_{m \in \mathbb{M}^t} {\alpha_m}^{Early} = {\alpha_{m^t_1}}^{Early}$
- ${B_t}^{Late} = \max_{m \in \mathbb{M}^t} {\alpha_m}^{Late} = {\beta_{m^t_{\mathbf{M^t}}}}^{Late} - S$

The possible time interval of technical movements is $[{\alpha_m}^{Early}, {\beta_m}^{Late}[$. The possible time interval of trains is $[{A_t}^{Early}, {B_t}^{Late}[$. In this case, for all $m \in \mathbb{M}^{t\circlearrowright}$, we have $[{\alpha_m}^{Early}, {\beta_m}^{Late}[= [{\alpha_m}^{ref}, {\beta_m}^{ref} + L[$. For all $m \in \mathbb{M}^{t\circlearrowleft}$, we have $[{\alpha_m}^{Early}, {\beta_m}^{Late}[= [{\alpha_m}^{ref} - L, {\beta_m}^{ref}[$.

If $[{\alpha_m}^{Early}, {\beta_m}^{Late}] \cap [{\alpha_{m'}}^{Early}, {\beta_{m'}}^{Late}] \neq \varnothing$, then we can identify the potential time conflict of two movements by $C^{refM}_{m,m'} = 1$; Otherwise $C^{refM}_{m,m'} = 0$.
If $[{A_t}^{Early}, {B_t}^{Late}[\cap [{A_{t'}}^{Early}, {B_{t'}}^{Late}[\neq \varnothing$, then we can identify the potential time conflict of two trains $C^{refT}_{t,t'} = 1$; Otherwise $C^{refT}_{t,t'} = 0$.
The equations (3.14) to (3.17) are rewritten as follows:

$$\forall t,t' \in \mathbb{T}, t \neq t', \forall l_f \in \mathbb{L}^i, C^{refT}_{t,t'} = 1,\ \ B_t \leq A_{t'} + R \cdot (3 - X^{L^iT}_{f,t} - X^{L^iT}_{f,t'} - X^{OrderT}_{t,t'}) \tag{3.27}$$

$$\forall t,t' \in \mathbb{T}, t \neq t', C^{refT}_{t,t'} = 1, \qquad X^{OrderT}_{t,t'} + X^{OrderT}_{t',t} = 1 \tag{3.28}$$

$$\forall m,m' \in \mathbb{M}, m \neq m', \forall p,p' \in \mathbb{P}, p \neq p', C^{refM}_{m,m'} = 1, Y^P_{p,p'} = 1,$$
$$\alpha_m + S \leq \alpha_{m'} + R \cdot (3 - X^{PM}_{p,m} - X^{PM}_{p',m'} - X^{OrderM}_{m,m'}) \tag{3.29}$$

$$\forall m,m' \in \mathbb{M}, m \neq m', C^{refM}_{m,m'} = 1, \qquad X^{OrderM}_{m,m'} + X^{OrderM}_{m',m} = 1 \tag{3.30}$$

Based on the numerical experiments in section 3.4.3, we find that the number of constraints decreases considerably.

3.4.3 Numerical experiments

The following software was used for our computational study:

- Model construction environment: AMPL.
- Solver for integer programs: IBM ILOG CPLEX Optimizer Version 12.5. (Calculation stops at the first feasible solution.)

The machine on which the computations were executed has the following hardware characteristics:

- Architecture: x86-64.
- Processors: Intel(R) Core(TM) i5-2520M CPU at 2.5GHz.
- Memory (RAM): 8GB.

We compare the original model and the improved model using a railway station with 18 switches, 17 internal lines and 10 external lines. Once the variables and constraints are sent to the solver, the problem is adjusted by CPLEX presolve which eliminates redundant constraints and variables. With Cplex options, we can choose to solve the optimal solution or the first feasible solution (solutionlim=1). Problems of small size generally can be solved to optimality within acceptable calculation time. For problems of larger size, we only get the first feasible solution. In our complex and busy railway station, there are 247 trains and 504 movements per day. In the rush hours, there are up to 3 trains running at the same time. The whole problem is divided into small size problems in chronological order. So we have 50 groups of 5 trains, 25 groups of 10 trains, 16 groups of 15 trains, 8 groups of 30 trains. The adjustable time interval here is 10 minutes, $L = 10$. We solve the first feasible solution of all problems. The draft timetable we have includes the parameters of commercial movements and technical movements without any feasibility checking. So there are conflicting movements which occupy the same external line at the same time interval. The first step is to remove all existing conflicts between commercial movements which can not be solved in our model. The sum of commercial movements' modifications is minimized. The second step is to update the technical movements to adapt the modified commercial movements. The third step is to input the modified data into our solver.

We try to solve the problems with three different models that are described in Section 3 and Section 4: discrete-time model, continuous-time model and reduced continuous-time model. The results are separately presented in Table 3.1, Table 3.2 and Table 3.3. For each numerical experiment, we describe the number of trains, the number of technical movements and commercial movements. After the calculation process, we note the length of time interval covered by trains. As we said, the problem will be firstly adjusted by CPLEX presolve. So we have less variables and constraints after presolve. Here, CPLEX stops at the first feasible solution.

The whole numerical experiments consist of 6 different sizes problems. Then we analyse the results in groups of the same size. In each group, the complexity of the problem is different. The conflicts can not be all solved. We count the number of problems solved in a given solve time, the number of problems settled as infeasible problems and the number of problems unsolved in the acceptable duration. We rank the problems in the order of solve time. The problem solved in the minimum or the maximum solve time is presented in the tables. The average solve time is computed from the solve informations of all groups.

Compared with the discrete-time model in Table 3.1, the continuous-time model has the same amount of variables and constraints, but the solve time decreases by 17.5% on average. The discrete-time model has 9 problems unsolved, and the continuous-time model has 5 problems unsolved. The solutions are all integral as we have proven in Section 3.4.1. So the improvements of continuous-time model are qualified.

Compared with the complete model in Table 3.1, the compressed continuous-time model cuts 22.1% variables and 66.2% constraints on average. The solve time decreases by 45.7% compared with the discrete-time model and decreases by 30.6% compared with the continuous-time model. In the groups of 25 trains, the continuous-time model solves 8 problems, but the reduced continuous-time model solves 6 problems. Some redundant constraints help to cut useless research branches and speed up the calculation. In the groups of 30 trains, the continuous-time model solves 4 problems, but the reduced continuous-time model solves 5 problems. The reduction of constraints plays a role in the acceleration of calculation. From the figures, we can see that the competition of calculation performance exists between the compressed continuous-time model and the continuous-time model.

From the three Tables 3.1, 3.2 and 3.3, we find infeasible cases even in small-size problems of 15 trains. That means unsolvable conflicts exist between movements or trains. One of reasons for unsolvable conflict may be the limited flexible time interval of technical movements. The flexible time interval for technical movements $L = 10$ minutes in (3.7) and (3.8) is too tight to ensure the existence of solution. In most of the cases, the group of trains can be solved except during the periods of heavy traffic : 8:00-10:00, 11:00-12:00 and around 18:00 (see Figures 3.1 to 3.4). When the value of L is increased, we can find an optimal solution but the solving time can also be greatly increased, because L represents also the time dimension of solution search space (bi-dimensional routing-scheduling). Further experimentations are necessary so that the value of L is adjusted in order to get the best trade-off between solution feasibility and solving time.

Based on the numerical experiments, there are two difficulties found in our problem: **Infeasible cases** and **No result cases**. As the size of problem increases, the proportion of "Infeasible" and "No result" cases is up to 62.5% in problems of 30 trains. These two difficult cases mainly come from the lack of well-directed conflicts. **The effectiveness of the model improved are not good enough to solve our full-day platforming problem. In the next section, we propose a cancellation processing which focuses on the cancellation of infeasible trains and generates at least a feasible timetable instead of an infeasibility conclusion. Furthermore, The flexible time interval for technical movements is discussed in details in Section 5.2. A proper L can effectively improve the efficiency of calculation while ensuring the feasibility of timetable.**

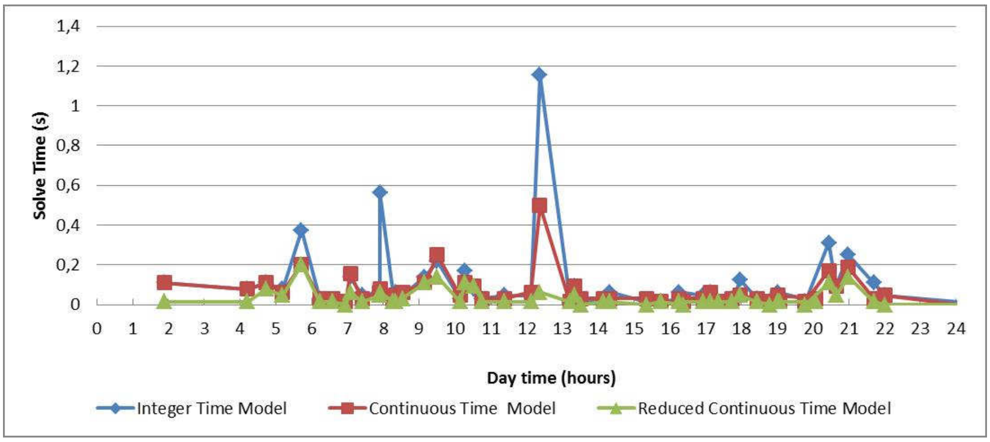


Fig. 3.1 5 trains benchmark

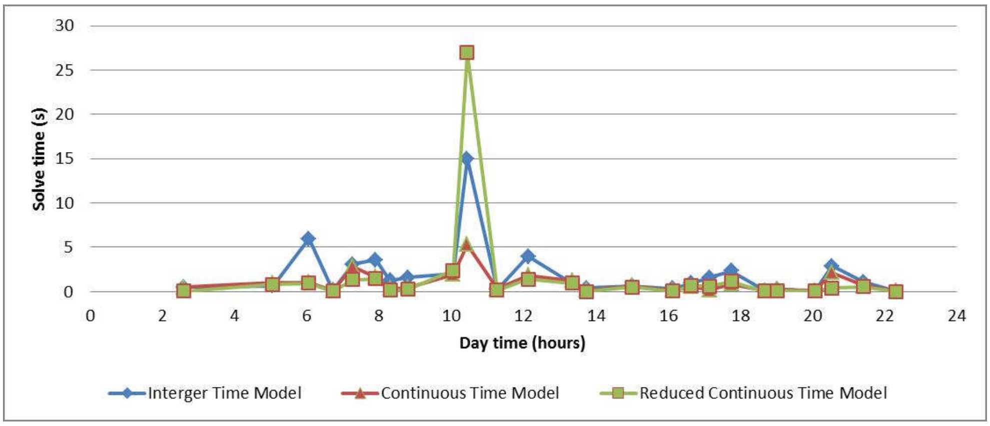


Fig. 3.2 10 trains benchmark

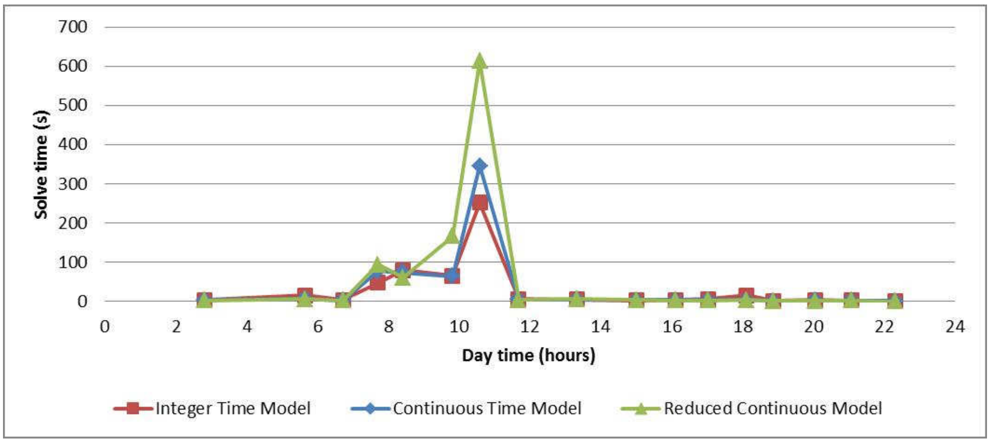


Fig. 3.3 15 trains benchmark

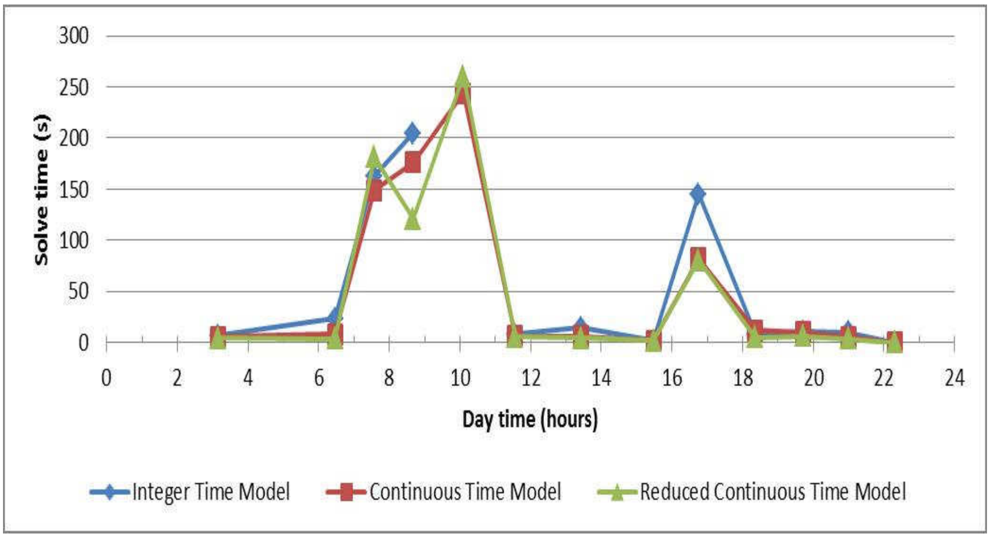


Fig. 3.4 20 trains benchmark

	Trains	Movements		Time	Before presolve		After presolve		First solution		Number of groups		
	per group	tech.	comm.	interval	Variables	Constraints	Variables	Constraints	Solution	Solve time	solved	infeasible	no result
Min	5	2	8	77	2140	4624	318	1913	156	0,02			
Average	5	3	7	77,6	2175	15199	391	7772	155	0,11	50	0	0
Max	5	5	5	51	2140	24684	455	14178	118	1,15			
Min	10	3	17	122,0	4530	28666	754	13457	365	0,08			
Average	10	7	13	105,7	4598	59962	929	32104	312	2,07	24	0	0
Max	10	9	12	64,0	4764	89769	1030	49525	254	14,98			
Min	15	7	23	94,0	7170	85283	1429	45439	524	1,15			
Average	15	10	20	140,3	7268	129144	1604	68617	485	9,80	13	3	0
Max	15	12	18	137,0	7170	194985	1748	113906	475	64,88			
Min	20	13	31	214,0	11168	150755	2318	74511	815	2,03			
Average	20	13	27	176,8	10233	215515	2391	112708	650	10,52	8	3	1
Max	20	13	27	140,0	10060	184949	2300	101232	646	23,59			
Min	25	17	33	213,0	13200	362041	3015	183220	769	9,02			
Average	25	17	33	220,4	13377	345295	3300	178056	811	19,93	5	1	3
Max	25	21	31	243,0	13790	265689	3015	142350	902	29,58			
Min	30	18	42	188,0	16590	475438	4348	240245	980	12,36			
Average	30	23	38	269,0	16800	533594	4490	272449	978	50,97	3	0	5
Max	30	23	39	243,0	17220	357778	3962	193601	1090	73,88			

Table 3.1 Discrete-time model

	Trains	Movements		Time	Before presolve		After presolve		First solution		Number of groups		
	per group	tech.	comm.	interval	Variables	Constraints	Variables	Constraints	Solution	Solve time	solved	infeasible	no result
Min	5	2	8	77	2140	4624	318	1913	156	0,02			
Average	5	3	7	77	2175	15199	391	7772	154	0,07	50	0	0
Max	5	5	5	51	2140	24684	455	14178	118	0,50			
Min	10	3	17	103	4530	28666	754	13457	300	0,09			
Average	10	7	13	106	4598	59962	929	32104	313	1,03	24	0	0
Max	10	9	12	64	4764	89769	1030	49525	254	5,30			
Min	15	7	23	94	7170	85283	1429	45439	528	1,05			
Average	15	10	20	140	7268	129144	1604	68617	487	7,63	13	3	0
Max	15	12	18	137	7170	194985	1748	113906	475	63,45			
Min	20	13	31	214	11168	150755	2318	74511	827	1,83			
Average	20	8	33	176	10233	215515	2391	112708	648	7,03	8	4	0
Max	20	13	27	137	10060	184949	2300	101232	617	11,79			
Min	25	14	36	149	13200	362041	3015	183220	844	8,92			
Average	25	14	38	221	13377	345295	3300	178056	823	10,61	5	3	1
Max	25	21	31	163	13790	265689	3015	142350	889	15,16			
Min	30	18	42	188	16590	475438	4348	240245	992	11,67			
Average	30	23	38	269	16800	533594	4490	272449	973	24,94	3	1	4
Max	30	23	39	376	17220	357778	3962	193601	858	35,24			

Table 3.2 The continuous-time model.

	Trains	Movements		Time	Before presolve		After presolve		First solution		Number of groups		
	per group	tech.	comm.	interval	Variables	Constraints	Variables	Constraints	Solution	Solve time	solved	infeasible	no result
Min	5	2	8	77	2130	2812	304	1013	156	0,00			
Average	5	3	4	76,82	2165	5651	350	3265	154	0,04	50	0	0
Max	5	5	5	45	2130	18610	422	13545	140	0,20			
Min	10	9	11	102	4510	9048	771	6472	256	0,03			
Average	10	7	13	106,08	4578	18413	753	12649	314	1,71	24	0	0
Max	10	9	12	112	4743	31484	845	23404	254	26,96			
Min	15	7	23	94	7140	19883	1100	13133	543	0,94			
Average	15	10	20	140,31	7238	31139	1164	22499	489	15,13	13	3	0
Max	15	12	18	137	7140	73169	1348	57475	473	166,27			
Min	20	13	31	214	11124	30407	1490	18974	841	1,93			
Average	20	13	27	176,50	10192	42426	1573	30343	657	4,38	8	4	0
Max	20	9	31	163	10020	41532	1560	29595	775	6,30			
Min	25	21	31	243	13738	45417	1905	33419	914	4,93			
Average	25	17	33	220,40	13326	52886	1995	38258	818	10,33	5	1	3
Max	25	11	40	148	44199	13443	2041	30294	832	24,84			
Min	30	18	42	188	16530	73634	2386	53136	994	10,20			
Average	30	23	38	269	16739	74726	2407	55610	998	11,43	3	2	3
Max	30	27	33	376	16530	88703	2503	66487	877	12,76			

Table 3.3 The reduced continuous-time model.

3.5 Cancellation processing

Cancellation processing. Based on the numerical expriments in previous section, we find that the existence of infeasibility is a great difficulty to overcome. To ensure the feasibility, we propose to study a problem allowing train cancellations. Such way of modelling guarantees the existence of solutions which can be found within acceptable computation time. This model will be used in the first stage of the complete algorithm presented in section 5. Two additional binary variables are added in the model to represent separately the cancellation of trains $X_t^{CancelT}$ and movements $X_m^{CancelM}$. The cancellation of trains means the release of internal line. The path can be released by the cancellation of movements.
If the train t is cancelled, $X_t^{CancelT} = 1$. Otherwise, $X_t^{CancelT} = 0$.
If the movement m is cancelled, $X_m^{CancelM} = 1$. Otherwise, $X_m^{CancelM} = 0$.

$$\forall t,t' \in \mathbb{T}, t \neq t', \forall l \in \mathbb{L}^i, \text{ s.t. } C_{t,t'}^{refT} = 1,$$
$$B_t \leq A_{t'} + R \cdot (3 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT} - X_{t,t'}^{OrderT} + X_t^{CancelT} + X_{t'}^{CancelT}) \tag{3.31}$$

If the train t or t' is cancelled, we have $2 \geq X_t^{CancelT} + X_{t'}^{CancelT} \geq 1$. So $5 \geq 3 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT} - X_{t,t'}^{OrderT} + X_t^{CancelT} + X_{t'}^{CancelT} \geq 1$. With a big enough constant R, the constraint (3.31) is relaxed.

$$\forall m,m' \in \mathbb{M}, m \neq m', \forall p,p' \in \mathbb{P}, p \neq p', \text{ s.t. } C_{m,m'}^{refM} = 1 \text{ and } Y_{p,p'}^{P} = 1,$$
$$\beta_m \leq \alpha_{m'} + R \cdot (3 - X_{p,m}^{PM} - X_{p',m'}^{PM} - X_{m,m'}^{OrderM} + X_m^{CancelM} + X_{m'}^{CancelM}) \tag{3.32}$$

If the movement m or m' is cancelled, we have $2 \geq X_m^{CancelM} + X_{m'}^{CancelM} \geq 1$. So $5 \geq 3 - X_{p,m}^{PM} - X_{p',m'}^{PM} - X_{m,m'}^{OrderM} + X_m^{CancelM} + X_{m'}^{CancelM} \geq 1$. With a big enough constant R, The constraint (3.32) is relaxed. So the constraints (3.14) and (3.16) are separately replaced by the constraints (3.31) and (3.32).

Objective function. The objective we focus on is to minimize the train cancellation. If the train t is cancelled, all movements of the train are cancelled as well. If one movement of the train is cancelled, the train is also cancelled.

$$\text{Minimize } \sum_{t \in \mathbb{T}} X_t^{CancelT} \tag{3.33}$$
$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t, \qquad X_m^{CancelM} = X_t^{CancelT} \tag{3.34}$$

This section ends with the full mathematical programming model formalized as below. In next section, we present the complete algorithm based on this model to solve the problem of platforming trains with cancellation processing in one-day timetable through a railway station.

$$\text{Minimize } \sum_{t\in\mathbb{T}} X_t^{CancelT} \tag{3.35}$$

$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t, \qquad X_m^{CancelM} = X_t^{CancelT} \tag{3.36}$$

$$
\begin{array}{llllrcll}
 & \forall m \in \mathbb{M} & & & \beta_m - \alpha_m & = & S & (3.37)\\
\forall t \in \mathbb{T} & & & & A_t - \alpha_{m_1^t} & = & 0 & (3.38)\\
\forall t \in \mathbb{T} & & & & B_t - \beta_{m^t_{\mathsf{M^t}}} & = & 0 & (3.39)\\
\forall t \in \mathbb{T} & & & \forall i \in \{1,...,\mathsf{M^t}-1\} & \beta_{m_i^t} - \alpha_{m_{i+1}^t} & <= & 0 & (3.40)\\
 & \forall m \in \mathbb{M}^{\leftarrow\ominus}, & & & \beta_m & \geq & \beta_m^{ref} - L & (3.41)\\
 & \forall m \in \mathbb{M}^{\leftarrow\ominus}, & & & \beta_m & \leq & \beta_m^{ref} & (3.42)\\
 & \forall m \in \mathbb{M}^{\ominus\rightarrow}, & & & \alpha_m & \geq & \alpha_m^{ref} & (3.43)\\
 & \forall m \in \mathbb{M}^{\ominus\rightarrow}, & & & \alpha_m & \leq & \alpha_m^{ref} + L & (3.44)\\
 & \forall m \in \mathbb{M}^{\hookrightarrow\ominus}, & & & \beta_m & = & \beta_m^{ref} & (3.45)\\
 & \forall m \in \mathbb{M}^{\ominus\hookrightarrow}, & & & \alpha_m & = & \alpha_m^{ref} & (3.46)\\
\forall t \in \mathbb{T} & & & & \sum_{l_i \in \mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} & = & 1 & (3.47)\\
\forall m \in \mathbb{M}^{l_e}, & \forall l_e \in \mathbb{L}^e & & \text{s.t. } Y_{l_e,m}^{L^eM} = 1 & \sum_{p\in\mathbb{P}^{l_e}} X_{p,m}^{PM} & = & 1 & (3.48)\\
\forall t \in \mathbb{T}, & \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \cap \mathbb{M}^t, \forall l_i \in \mathbb{L}_{l_e}^i, & & & \sum_{p\in\mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} - X_{l_i,t}^{L^iT} & \geq & 0 & (3.49)\\
\forall t \neq t' \in \mathbb{T}, & \forall l \in \mathbb{L}^i & & \text{s.t. } C_{t,t'}^{refT} = 1, & & & & \\
 & B_t - A_{t'} + R & \cdot (X_{l,t}^{L^iT} + X_{l,t'}^{L^iT} + X_{t,t'}^{OrderT} & - X_t^{CancelT} - X_{t'}^{CancelT}) & & \leq & 3\cdot R & (3.50)\\
\forall t \neq t' \in \mathbb{T}, & & & & X_{t,t'}^{OrderT} + X_{t',t}^{OrderT} & = & 1 & (3.51)\\
\forall m \neq m' \in \mathbb{M}, & \forall p \neq p' \in \mathbb{P} \text{ s.t. } C_{m,m'}^{refM} = 1,\ Y_{p,p'}^{P} = 1, & & & & & & \\
 & \beta_m - \alpha_{m'} + R \cdot (X_{p,m}^{PM} + X_{p',m'}^{PM} + X_{m,m'}^{OrderM} & - X_m^{CancelM} - X_{m'}^{CancelM}) & & & \leq & 3\cdot R & (3.52)\\
 & \forall m \neq m' \in \mathbb{M}, & & & X_{m,m'}^{OrderM} + X_{m',m}^{OrderM} & = & 1 & (3.53)\\
\forall t \in \mathbb{T}, & & & & A_t & \in & [\![0,1440]\!] & (3.54)\\
\forall t \in \mathbb{T}, & & & & B_t & \in & [\![0,1440+L]\!] & (3.55)\\
\forall t \in \mathbb{T}, & \forall m \in \mathbb{M}^t, & & & \alpha_m & \in & [\![0,1440]\!] & (3.56)\\
\forall t \in \mathbb{T}, & \forall m \in \mathbb{M}^t, & & & \beta_m & \in & [\![0,1440+L]\!] & (3.57)\\
\forall t \in \mathbb{T}, & \forall l \in \mathbb{L}, & & & X_{l,t}^{L^iT} & \in & \{0,1\} & (3.58)\\
 & \forall m \in \mathbb{M}, & & \forall p \in \mathbb{P}, & X_{p,m}^{PM} & \in & \{0,1\} & (3.59)\\
\forall t \neq t' \in \mathbb{T}, & & & & X_{t,t'}^{OrderT} & \in & \{0,1\} & (3.60)\\
 & \forall m \neq m' \in \mathbb{M}, & & & X_{m,m'}^{OrderM} & \in & \{0,1\} & (3.61)\\
\forall t \in \mathbb{T}, & & & & X_t^{CancelT} & \in & \{0,1\} & (3.62)\\
 & \forall m \in \mathbb{M}, & & & X_m^{CancelM} & \in & \{0,1\} & (3.63)
\end{array}
$$

Fig. 3.5 Complete Mathematical Model

3.6 Conclusion

In this section, we succeed to establish the decision model which aims to generate a "Feasible timetable" with permission of trains cancellation. In the first part, an Integer Linear Programming model is described to present all basic constraints in terms of parameters, variables and constraints. To reduce computation effort, the model is improved in two ways. The first method is to define time variables in continuous time domain. The second method is to probe potential conflicts between movements and between trains. In this way, the constraints are created only for the movements and trains with potential conflicts. The useless constraints are cut off. Improved performance achieved by these two methods is proven by numerical experiments. But we find another difficulty that infeasible cases appear even in small size problems. In the last part, cancellation processing is integrated into the improved model to deal with infeasible cases. Blocking trains are cancelled to guarantee feasibility of timetables. Cancellation of trains is minimized. Decision model is integrated in the hybrid method based on a sliding window algorithm explained in Chapter 5. The performance of decision model with cancellation processing is tested on real cases in Section 5.5.

CHAPTER 4

Reinsertion model and refinement model: generation of Revised timetable

For the moment, blocking trains in infeasible timetable are found and cancelled by decision model. But cancellation of trains is not the first choice in timetabling process. A better arrangement of rail activities is achieved by reinsertion of trains cancelled with minimal train delays. Proposition of proper revised timetable is required. This version of timetable includes all trains required, permits slight train delays and avoids large delays propagation. Revised timetable is generated by two models: reinsertion model and refinement model. Objective function of reinsertion model is minimization of trains cancelled. Objective function of refinement model is minimization of train delays.

4.1 Reinsertion model

Reinsertion of trains cancelled by decision model is similar to on-line rescheduling process. It also can be represented as train conflicts detection and resolution problem (CDR). Trains to be reinserted act as traffic disturbances to be absorbed by “Feasible timetable”. Due to the interaction between trains, reinsertion of trains may cause train delay propagation as knock-on delays to other trains within the railway station dispatching areas. Reinsertion of trains cancelled can be achieved by rerouting and rescheduling trains within relaxed feasible time interval for technical and commercial movements.

A recent stream of research on CDR focuses on detailed formulations based on the alternative graph of Mascis and Pacciarelli (2002). The first alternative graph formulation of the train scheduling problem with fixed routes was developed within the European project COMBINE (Mascis et al., 2008). Mazzarello and Ottaviani (2007) report on the practical implementation of the COMBINE system, using simple routing and sequencing algorithms, on a pilot site in The Netherlands. Flamini and Pacciarelli (2008) address the problem of routing trains through an underground rail terminus and develop a heuristic algorithm for a bi-criteria version of the problem in which earliness/tardiness and train headways have to be optimized. D’Ariano et al. (2007) propose a branch and bound algorithm for the CDR problem with fixed routing. Their computational experiments, carried on the Dutch railway bottleneck around Schiphol International airport and for multiple delayed trains, show that optimal or near-optimal solutions can be found within a short computation time. In a follow-up paper (D’Ariano et al., 2008), the traffic management system ROMA (Railway traffic Optimization by Means of Alternative graphs) is described. In ROMA, this branch and bound algorithm is incorporated in a local search framework such that train routes are changed when better solutions can be achieved. Computational tests, carried on the Dutch dispatching area between Utrecht and Den Bosch, include instances with multiple delayed trains and different blocked tracks in the network. The results show that significant delays reduction is achieved by rerouting and rescheduling train movements, even though the benefit is mainly due to the sequencing optimization rather than to rerouting, particularly when dealing with heavy disruptions in the network.

An accurate detection of conflicts can sharply narrow the solution search space and further decrease calculation effort and computation time. Alternative graph represents trains’ conflicting relationship on routes at every minutes of scheduling time horizon. This method enable an accurate detection of conflicts, but it also costs long time to construct this graph.

Different from alternative graph method, we propose an innovative conception, *conflict degree*, to define our feasible solution search space and efficiently limit train delay propagation. *Conflict degree* evaluates the conflicting relationship in terms of potential scheduling time interval described in Section 2.4.2.1. In this conception framework, scheduling time horizon is considered as continuous-time. Trains involving relevant conflict degree are relaxed to be rescheduled and rerouted in order to absorb the trains reinserted and guarantee the feasibility at the same time. Then a detailed description of this conception is given in the following paragraphs.

Reinsertion problem is solved following three steps:

1. Define potentially flexible time interval for technical movements and commercial movements.

2. Anticipate negative effect of traffic influence caused by reinsertion: limitation of train delay propagation

3. Reinsert trains cancelled within flexible time interval.

4.1.1 Flexible time interval

Potentially flexible time intervals are determined by railway station managers and highly impact the time search space of solutions. As described in Section 2.4.2.1, potential scheduling time intervals of trains are generated with flexible interval L for technical movements. To absorb trains cancellation, the acceptable deviation upper bound F for commercial movements is permitted. Considering feasibility of the deviation operations, the reference departure or arrival time of commercial movements can only be postponed as shown in equations (4.1)-(4.4). The equation (4.5) is added to ensure the required standstill on the internal lines.

$$\forall m \in \mathbb{M}^{\to\ominus}, \qquad \beta_m^{\mathrm{ref}} - L \leq \beta_m \leq \beta_m^{\mathrm{ref}} + F \tag{4.1}$$

$$\forall m \in \mathbb{M}^{\ominus\to}, \qquad \alpha_m^{\mathrm{ref}} \leq \alpha_m \leq \alpha_m^{\mathrm{ref}} + F \tag{4.2}$$

$$\forall m \in \mathbb{M}^{\ominus\rightharpoonup}, \qquad \alpha_m^{\mathrm{ref}} \leq \alpha_m \leq \alpha_m^{\mathrm{ref}} + L + F \tag{4.3}$$

$$\forall m \in \mathbb{M}^{\hookrightarrow\ominus}, \qquad \beta_m^{\mathrm{ref}} \leq \beta_m \leq \beta_m^{\mathrm{ref}} + F \tag{4.4}$$

$$\forall t \in \mathbb{T}, \qquad \beta_{m^t_{\mathsf{M}_t}}^{\mathrm{ref}} - \alpha_{m^t_1}^{\mathrm{ref}} \leq B_t - A_t \tag{4.5}$$

4.1.2 Relaxation of trains

In this section, we propose a procedure to identify *influenced trains*, regarded as another dimension of search space, which will be relaxed to absorb trains cancellation. The first two steps determinate time-train two-dimensional search space. If we under-estimate the size of this search space, less trains will be relaxed and rescheduled. Less calculation efforts are needed, but the relaxed trains may not be able to release enough resources to reinsert the train. If we overvalue the seach space, more trains will be relaxed and rescheduled. The train is probably reinserted, but more calculation efforts are required. Here we define two *conflict degrees* to properly handle the group of trains to be relaxed. *Conflict degree 1* represents direct conflict between the train to be reinserted and trains in "Feasible timetable". If the trains group of conflict degree 1 cannot absorb the disturbance, we will generate another group of trains with *conflict degree 2*.

Definition 4 (Conflict degree 1)**.** Consider one train t to be reinsert to a feasible timetable. If actual scheduling time interval $[A_{t'}, B_{t'}[$ of a train t' has an intersection time interval with $[A_t{}^{Early}, B_t{}^{Late}[$, the train t' potentially conflicts with the train t. The conflict degree of t' is 1 which means degree of correlation.

$$[A_{t'}, B_{t'}[\cap[A_t{}^{Early}, B_t{}^{Late}[\quad \neq \quad \emptyset \tag{4.6}$$

We denote the group of trains in conflict degree 1 with the train t by $\mathbb{T}_t^{Conflict1}$.

Definition 5 (Conflict degree 2)**.** Consider one train t to be reinsert to a feasible timetable. The train t' conflicts with the train t in degree 1. If actual scheduling time interval $[A_{t''}, B_{t''}[$ of a train t'' has an intersection time interval with $[A_{t'}{}^{Early}, B_{t,}{}^{Late}[$, the train t', potentially conflicts with the train t'. The conflict degree of t'' is 2 which means indirect conflict.

$$[A_{t''}, B_{t''}[\cap[A_{t'}{}^{Early}, B_{t'}{}^{Late}[\neq \emptyset \tag{4.7}$$

We denote the group of trains in conflict degree 2 with the train t corresponding to train t' by $\mathbb{T}_{t,t'}^{Conflict2}$.

4.1.3 Reinsertion of trains

In this case, trains in a feasible timetable can be classified into two groups: trains relaxed $\mathbb{T}^{Relaxed}$ and trains fixed $\mathbb{T}^{Fixed}$ as valid constraints. Trains fixed keep the same scheduling time and resources allocation in "Feasible timetable". Trains relaxed are rescheduled and rerouted to reinsert the trains cancelled in $\mathbb{T}^{Cancelled}$. All trains considered consist of three parts $\mathbb{T} = \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled} \cup \mathbb{T}^{Fixed}$.

Rescheduling and rerouting principles of reinsertion model are the same as in decision model. In addition, compatibility of resources is not only evaluated among trains in $\mathbb{T}^{Relaxed}$, but also need to be verified between trains from $\mathbb{T}^{Relaxed}$ and $\mathbb{T}^{Fixed}$. In other words, trains fixed act as valid constraints while rescheduling and rerouting the relaxed trains and the train to be reinserted.

To reinsert the train $t \in \mathbb{T}^{Cancelled}$, we generate the group of trains with conflict degree 1 as relaxed trains group $\mathbb{T}^{Relaxed}$. Then the rest of trains in Feasible timetable form a fixed trains group $\mathbb{T}^{Fixed}$. Potential scheduling time intervals are represented in Table 4.1.

$t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, m \in \mathbb{M}^t$	$\alpha_m{}^{Early}$	$\beta_m{}^{Late}$
$\mathbb{M}^{\ominus\!\rightarrow}$	α^{ref}	$\alpha^{ref} + S + F$
$\mathbb{M}^{\hookrightarrow\!\circ}$	$\beta^{ref} - S$	$\beta^{ref} + F$
$\mathbb{M}^{\ominus\!\rightharpoonup}$	α^{ref}	$\alpha^{ref} + S + L$
$\mathbb{M}^{\rightharpoonup\!\circ}$	$\beta^{ref} - S - L$	$\beta^{ref} + F$

Table 4.1 Potential scheduling time interval for movements to be rescheduled

The earliest entering time of the train t is the earliest starting time of the first movement of the train m_1^t, shown as $A_t{}^{Early} = \alpha_{m_1^t}{}^{Early}$. The latest leaving time of the train t is the latest arrival time of the last movement $m^t_{\mathbf{M_t}}$, shown as $B_t{}^{Late} = \beta_{m^t_{\mathbf{M_t}}}{}^{Late} - S$. So the potential scheduling time interval for the train t is $[A_t{}^{Early}, B_t{}^{Late}[$.

The complete reinsertion model is given below.

4.1.3.1 Parameters

- R is a sufficiently big constant.

- L is the adjustable time interval of the technical movements. In our context, $L = 60$.
- α_m^{ref} is the reference starting time of the movement m.
- β_m^{ref} is the reference ending time of the movement m.
- $X_{p,m}^{PMref}$ is the path allocation decision for the movement m in Feasible timetable. If the movement m is allocated to the path p, $X_{p,m}^{PMref} = 1$. Otherwise $X_{p,m}^{PMref} = 0$.
- $X_{l_i,t}^{L^iTref}$ is the internal line allocation decision for the train t in Feasible timetable. If the train t is allocated to the internal line l_i, $X_{l_i,t}^{L^iTref} = 1$. Otherwise $X_{l_i,t}^{L^iTref} = 0$.
- S is the time allocated to a movement. In our context, $S = 5$ minutes.
- $Y_{p,p'}^{P}$ identifies the pair of conflicting paths. If $p \cap p' \neq \varnothing$, $Y_{p,p'}^{P} = 1$. Otherwise $Y_{p,p'}^{P} = 0$.
- $C_{m,m'}^{refM}$ probes the potential conflicts between two movements m and m'.
  If $[\alpha_m{}^{Early}, \beta_m{}^{Late}) \cap [\alpha_{m'}{}^{Early}, \beta_{m'}{}^{Late}) \neq \varnothing, C_{m,m'}^{refM} = 1$. Otherwise, $C_{m,m'}^{refM} = 0$.
- $C_{t,t'}^{refT}$ probes the potential conflicts between two trains t and t'.
  If $[A_t{}^{Early}, B_t{}^{Late}) \cap [A_{t'}{}^{Early}, B_{t'}{}^{Late}) \neq \varnothing$, $C_{t,t'}^{refT} = 1$. Otherwise $C_{t,t'}^{refT} = 0$.

4.1.3.2 Variables

In the practical situation, the arrival and leaving times of trains are measured in minutes. The scheduling decision variables are thus defined as integers with units of minutes, characterizing a discrete-time sheduling problem.

- α_m is the actual starting time of the movement m.
- β_m is the actual ending time of the movement m, $\alpha_m + S = \beta_m$.
- A_t is the starting time of occupation of the internal lines by the train t.
- B_t is the ending time of occupation of the internal lines by the train t.

The routing decision variables are defined as binary variables.

- $X_{l,t}^{L^iT}$ identifies the internal lines allocated to the train t. If the train t is allocated to the internal line l, $X_{l,t}^{L^iT} = 1$. Otherwise 0.
- $X_{p,m}^{PM}$ identifies the path allocated to the movement m. If the path p is allocated to the movement m, $X_{p,m}^{PM} = 1$. Otherwise 0.
- $X_{t,t'}^{OrderT}$ identifies the time order of two trains using the same line. If t circulates before t', $X_{t,t'}^{OrderT} = 1$. Otherwise 0.
- $X_{m,m'}^{OrderM}$ identifies the time order of two movements using two conflicting paths. If m circulates before m', $X_{m,m'}^{OrderM} = 1$. Otherwise 0.
- $X_{m}^{CancelM}$ identifies the cancellation of the movement m. If m is cancelled, $X_{m}^{CancelM} = 1$. Otherwise 0.
- $X_{t}^{CancelT}$ identifies the cancellation of the train t. If t is cancelled, $X_{t}^{CancelT} = 1$. Otherwise 0.

4.1.3.3 Constraints

Time constraints Potential scheduling time intervals are defined for trains to be rescheduled $t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}$ as follows:

$$
\begin{aligned}
&\forall m \in \mathbb{M}^t \cap \mathbb{M}^{\rightarrow\circlearrowright}, && \beta_m^{\text{ref}} - L && \leq \beta_m \leq \beta_m^{\text{ref}} + F && (4.8)\\
&\forall m \in \mathbb{M}^t \cap \mathbb{M}^{\circlearrowleft\rightarrow}, && \alpha_m^{\text{ref}} && \leq \alpha_m \leq \alpha_m^{\text{ref}} + F && (4.9)\\
&\forall m \in \mathbb{M}^t \cap \mathbb{M}^{\circlearrowleft\rightharpoondown}, && \alpha_m^{\text{ref}} && \leq \alpha_m \leq \alpha_m^{\text{ref}} + L + F && (4.10)\\
&\forall m \in \mathbb{M}^t \cap \mathbb{M}^{\hookrightarrow\circlearrowright}, && \beta_m^{\text{ref}} && \leq \beta_m \leq \beta_m^{\text{ref}} + F && (4.11)\\
&\forall t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, && \beta_{m^t_{\mathsf{M}_t}}^{\text{ref}} - \alpha_{m^t_1}^{\text{ref}} && \leq B_t - A_t && (4.12)
\end{aligned}
$$

For trains fixed as valid constraints $\mathbb{T}^{Fixed}$, departure and arrival times remain the same as in feasible timetable.

$$
\begin{aligned}
&\forall t \in \mathbb{T}^{Fixed}, m \in \mathbb{M}^t, && \beta_m^{\text{ref}} = \beta_m && (4.13)\\
&\forall t \in \mathbb{T}^{Fixed}, m \in \mathbb{M}^t, && \alpha_m^{\text{ref}} = \alpha_m && (4.14)
\end{aligned}
$$

Resources allocation For trains $\mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}$ to be rerouted, routing principles are the same as in Decision model. Preference of internal lines is guaranteed by equation 4.15. A path connecting external line and internal line chosen is allocated to movements to be rescheduled, shown in equations 4.16 and 4.17.

$$\forall t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, \sum_{l_i \in \mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} = 1 \tag{4.15}$$

$$\forall t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^t \cap \mathbb{M}^{l_e}$$
$$\sum_{p \in \mathbb{P}^{l_e}} X_{p,m}^{PM} = 1 \tag{4.16}$$
$$\forall t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \cap \mathbb{M}^t, \forall l_i \in \mathbb{L}_{l_e}^i$$
$$\sum_{p \in \mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} \geq X_{l_i,t}^{L^iT} \tag{4.17}$$

For trains fixed as valid constraints $\mathbb{T}^{Fixed}$, routing decisions remain the same as in feasible timetable.

$$\forall t \in \mathbb{T}^{Fixed}, m \in \mathbb{M}^t, \forall p \in \mathbb{P} \qquad X_{p,m}^{PMref} = X_{p,m}^{PM} \tag{4.18}$$
$$\forall t \in \mathbb{T}^{Fixed}, \forall l_i \in \mathbb{L}^i \qquad X_{l_i,t}^{L^iTref} = X_{l_i,t}^{L^iT} \tag{4.19}$$

Compatibility of resources One path or internal line cannot be occupied by two movements or trains. This constraint need to be verified not only on pair of trains to be rescheduled by equations (4.20) and (4.21), but also between trains to be rescheduled and trains fixed by equations (4.23) and (4.22). Compatibility of resources on pair of trains fixed is guaranteed by Decision model, so it is not necessary to verify it here again. Principles used to verify resources compatibility are the same as applied in Decision model, found in Section 3.5.

$$\forall t, t' \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, t \neq t', \forall l \in \mathbb{L}^i, \text{ s.t. } C_{t,t'}^{refT} = 1,$$
$$B_t \leq A'_t + R \cdot (3 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT} - X_{t,t'}^{OrderT} + X_t^{CancelT} + X_{t'}^{CancelT}) \tag{4.20}$$

$$\forall t,t' \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, m \in \mathbb{M}^t, m' \in \mathbb{M}^{t'}, m \neq m',$$
$$\forall p,p' \in \mathbb{P}, p \neq p', \text{ s.t. } C^{refM}_{m,m'} = 1 \text{ and } Y^P_{p,p'} = 1,$$
$$\beta_m \leq \alpha_{m'} + R \cdot (3 - X^{PM}_{p,m} - X^{PM}_{p',m'} - X^{OrderM}_{m,m'} + X^{CancelM}_{m} + X^{CancelM}_{m'}) \tag{4.21}$$

$$\forall t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, \forall t' \in \mathbb{T}^{Fixed} \text{ or } \forall t' \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, \forall t \in \mathbb{T}^{Fixed}$$
$$\forall l \in \mathbb{L}^i, \text{ s.t. } C^{refT}_{t,t'} = 1,$$
$$B_t \leq A'_t + R \cdot (3 - X^{L^iT}_{l,t} - X^{L^iT}_{l,t'} - X^{OrderT}_{t,t'} + X^{CancelT}_{t} + X^{CancelT}_{t'}) \tag{4.22}$$

$$\forall t \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, \forall t' \in \mathbb{T}^{Fixed} \text{ or } \forall t' \in \mathbb{T}^{Relaxed} \cup \mathbb{T}^{Cancelled}, \forall t \in \mathbb{T}^{Fixed}$$
$$m \in \mathbb{M}^t, m' \in \mathbb{M}^{t'}, m \neq m',$$
$$\forall p,p' \in \mathbb{P}, p \neq p', \text{ s.t. } C^{refM}_{m,m'} = 1 \text{ and } Y^P_{p,p'} = 1,$$
$$\beta_m \leq \alpha_{m'} + R \cdot (3 - X^{PM}_{p,m} - X^{PM}_{p',m'} - X^{OrderM}_{m,m'} + X^{CancelM}_{m} + X^{CancelM}_{m'}) \tag{4.23}$$

The order between two trains or two movements is single and guaranteed by following equations.

$$\forall t,t' \in \mathbb{T}, t \neq t', \qquad X^{OrderT}_{t,t'} + X^{OrderT}_{t',t} = 1 \tag{4.24}$$
$$\forall m,m' \in \mathbb{M}, m \neq m', \qquad X^{OrderM}_{m,m'} + X^{OrderM}_{m',m} = 1 \tag{4.25}$$

Cancellation decisions on trains fixed remain the same as in Feasible timetable, shown in equations 4.26 and 4.27.

$$\forall t \in \mathbb{T}^{Fixed} \qquad X^{CancelT}_{t} = 0 \tag{4.26}$$
$$\forall t \in \mathbb{T}^{Fixed}, m \in \mathbb{M}^t \qquad X^{CancelM}_{m} = 0 \tag{4.27}$$

Objective function The objective we focus on is to minimize the train cancellation. If the train t is cancelled, all movements of the train are cancelled as well. If one movement of the train is cancelled, the train is also cancelled.

$$\text{Minimize} \sum_{t \in \mathbb{T}} X^{CancelT}_{t} \tag{4.28}$$
$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t, \qquad X^{CancelM}_{m} = X^{CancelT}_{t} \tag{4.29}$$

To avoid extra trains cancellation, the feasible timetable obtained from Decision model is given as an initial solution to start the calculation process of reinsertion model. Trains are reinserted into the feasible timetable one by one. If the reinsertion is completed, the number of trains cancelled decreases. If the relaxation of trains with conflict degree 1 cannot absorb all trains cancellation, we need to increase deviation upper bound F or relax trains with conflict degree 2 of blocking trains. As a result, more flexibility is given to the blocking trains.

4.2 Refinement model

Reinsertion model has already guaranteed feasibility of "Revised timetable". After rerouting and rescheduling trains cancelled in Decision model and their neighbour trains with conflict degree 1 and 2, all trains are included in "Revised timetable". However, reinsertion model only focus on absorbing trains cancellation. In order to reduce calculation efforts, time deviations of commercial movements caused by reinsertion are limited by an upper bound F but not considered in the objective function. Superfluous time deviations may cause delay propagation in the whole railway networks. To decrease negative effects of reinsertion, further refinement of timetable is operated by refinement model. The objective function of refinement model is the minimization of time deviations of commercial movements.

First of all, we find all commercial movements with time deviations by comparing reference time α_m^{ref} and scheduling time revised by reinsertion model $\alpha_m^{revised}$. If the revised time of a commercial movement is different from its reference time $\alpha_m^{ref} = \alpha_m^{revised}$, the train containing this commercial movement is to be rescheduled and rerouted. These trains form the group of trains relaxed $\mathbb{T}^{Relaxed}$. If all commercial movements of a train still keep their reference times, the train is fixed with the same scheduling and routing decisions obtained in reinsertion model. This kind of trains form the group of trains fixed $\mathbb{T}^{Fixed}$. Potential scheduling time intervals are the same as used in reinsertion model, shown in Table 4.1.

4.2.1 Parameters

- R is a sufficiently big constant.
- L is the adjustable time interval of the technical movements. In our context, $L = 60$.

- S is the time allocated to a movement. In our context, $S = 5$ minutes.
- α_m^{ref} is the reference starting time of the movement m.
- β_m^{ref} is the reference ending time of the movement m.
- $\alpha_m^{revised}$ is the starting time of the movement m revised by reinsertion model.
- $\beta_m^{revised}$ is the ending time of the movement m revised by reinsertion model, $\alpha_m + S = \beta_m$.
- C_t^{dev} probes trains containing commercial movements with time deviations. If $\exists m \in \mathbb{M}^t \cap \mathbb{M}^{Commercial}, \alpha_m^{ref} \neq \alpha_m^{revised}$, $C_t^{dev} = 1$. Otherwise 0.
- $X_{l,t}^{L^iTRevised}$ identifies the internal line allocated to the train t by the reinsertion model. If the internal line l is allocated to the train t by the reinsertion model, $X_{l,t}^{L^iTRevised} = 1$. Otherwise 0.
- $X_{p,m}^{PMRevised}$ identifies the path allocated to the movement m by the reinsertion model. If the path p is allocated to the movement m by the reinsertion model, $X_{p,m}^{PMRevised} = 1$. Otherwise 0.
- $Y_{p,p'}^{P}$ identifies the pair of conflicting paths. $Y_{p,p'}^{P} = \delta(p \cap p' \neq \varnothing)$.
- $C_{m,m'}^{refM}$ probes the potential conflicts between two movements m and m'.
If $[{\alpha_m}^{Early}, {\beta_m}^{Late}) \cap [{\alpha_{m'}}^{Early}, {\beta_{m'}}^{Late}) \neq \varnothing, C_{m,m'}^{refM} = 1$. Otherwise, $C_{m,m'}^{refM} = 0$.
- $C_{t,t'}^{refT}$ probes the potential conflicts between two trains t and t'. If $[{A_t}^{Early}, {B_t}^{Late}) \cap [{A_{t'}}^{Early}, {B_{t'}}^{Late}) \neq \varnothing$, $C_{t,t'}^{refT} = 1$. Otherwise $C_{t,t'}^{refT} = 0$.
- $\mathbb{T}^{Relaxed}$ is the group of trains with commercial time deviations $C_t^{dev} = 1$.
- $\mathbb{T}^{Fixed}$ is the group of trains without commercial time deviations $C_t^{dev} = 0$.

4.2.2 Variables

In the practical situation, the arrival and leaving times of trains are measured in minutes. The scheduling decision variables are thus defined as integers with units of minutes, characterizing a discrete-time sheduling problem.

- α_m is the actual starting time of the movement m.
- β_m is the actual ending time of the movement m, $\alpha_m + S = \beta_m$.
- A_t is the starting time of occupation of the internal lines by the train t.
- B_t is the ending time of occupation of the internal lines by the train t.

The routing decision variables are defined as binary variables.

- $X^{L^iT}_{l,t}$ identifies the internal lines allocated to the train t. If the train t is allocated to the internal line l, $X^{L^iT}_{l,t} = 1$. Otherwise 0.
- $X^{PM}_{p,m}$ identifies the path allocated to the movement m. If the path p is allocated to the movement m, $X^{PM}_{p,m} = 1$. Otherwise 0.
- $X^{OrderT}_{t,t'}$ identifies the time order of two trains using the same line. If t circulates before t', $X^{OrderT}_{t,t'} = 1$. Otherwise 0.
- $X^{OrderM}_{m,m'}$ identifies the time order of two movements using two conflicting paths. If m circulates before m', $X^{OrderM}_{m,m'} = 1$. Otherwise 0.

4.2.3 Constraints

Time constraints Potential scheduling time intervals are defined for trains to be rescheduled $t \in \mathbb{T}^{Relaxed}$ as follows:

$$\forall m \in \mathbb{M}^t \cap \mathbb{M}^{-\!\ominus}, \qquad \beta^{\text{ref}}_m - L \leq \beta_m \leq \beta^{\text{ref}}_m + F \tag{4.30}$$
$$\forall m \in \mathbb{M}^t \cap \mathbb{M}^{\ominus\!\rightarrow}, \qquad \alpha^{\text{ref}}_m \leq \alpha_m \leq \alpha^{\text{ref}}_m + F \tag{4.31}$$
$$\forall m \in \mathbb{M}^t \cap \mathbb{M}^{\ominus\!\rightharpoondown}, \qquad \alpha^{\text{ref}}_m \leq \alpha_m \leq \alpha^{\text{ref}}_m + L + F \tag{4.32}$$
$$\forall m \in \mathbb{M}^t \cap \mathbb{M}^{\hookrightarrow\!\ominus}, \qquad \beta^{\text{ref}}_m \leq \beta_m \leq \beta^{\text{ref}}_m + F \tag{4.33}$$
$$\forall t \in \mathbb{T}^{Relaxed}, \qquad \beta^{\text{ref}}_{m^t_{\mathrm{M}_t}} - \alpha^{\text{ref}}_{m^t_1} \leq B_t - A_t \tag{4.34}$$

For trains fixed as valid constraints $\mathbb{T}^{Fixed}$, departure and arrival times are given by the reinsertion model.

$$\forall t \in \mathbb{T}^{Fixed}, m \in \mathbb{M}^t, \qquad \beta_m^{revised} = \beta_m \tag{4.35}$$

$$\forall t \in \mathbb{T}^{Fixed}, m \in \mathbb{M}^t, \qquad \alpha_m^{revised} = \alpha_m \tag{4.36}$$

Resources allocation For trains $\mathbb{T}^{Relaxed}$ to be rerouted, routing principles are the same as in Decision model. Preference of internal lines is guaranteed by equation (4.37). A path connecting external line and internal line chosen is allocated to movements to be rescheduled, shown in equations (4.38) and (4.39).

$$\forall t \in \mathbb{T}^{Relaxed}, \sum_{l_i \in \mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} = 1 \tag{4.37}$$

$$\forall t \in \mathbb{T}^{Relaxed}, \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^t \cap \mathbb{M}^{l_e}$$
$$\sum_{p \in \mathbb{P}^{l_e}} X_{p,m}^{PM} = 1 \tag{4.38}$$

$$\forall t \in \mathbb{T}^{Relaxed}, \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \cap \mathbb{M}^t, \forall l_i \in \mathbb{L}_{l_e}^i$$
$$\sum_{p \in \mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} \geq X_{l_i,t}^{L^iT} \tag{4.39}$$

For trains fixed as valid constraints $\mathbb{T}^{Fixed}$, routing decisions are given by the reinsertion model.

$$\forall t \in \mathbb{T}^{Fixed}, m \in \mathbb{M}^t, \forall p \in \mathbb{P} \qquad X_{p,m}^{PMRevised} = X_{p,m}^{PM} \tag{4.40}$$

$$\forall t \in \mathbb{T}^{Fixed}, \forall l_i \in \mathbb{L}^i \qquad X_{l_i,t}^{L^iTRevised} = X_{l_i,t}^{L^iT} \tag{4.41}$$

Compatibility of resources One path or internal line cannot be occupied by two movements or trains. This constraint need to be verified not only on pairs of trains to be rescheduled by equations (4.42) and (4.21), but also between trains to be rescheduled and trains fixed by equations (4.45) and (4.44). Compatibility of resources on pairs of trains fixed is guaranteed by Reinsertion model, so it is not necessary to verify it here again.

Principles used to verify resources compatibility are the same as applied in Decision model, found in Section 3.5.

$$\begin{aligned} &\forall t, t' \in \mathbb{T}^{Relaxed}, t \neq t', \forall l \in \mathbb{L}^i, \text{ s.t. } C^{refT}_{t,t'} = 1, \\ &B_t \leq A'_t + R \cdot (3 - X^{L^iT}_{l,t} - X^{L^iT}_{l,t'} - X^{OrderT}_{t,t'}) \end{aligned} \tag{4.42}$$

$$\begin{aligned} &\forall t, t' \in \mathbb{T}^{Relaxed}, m \in \mathbb{M}^t, m' \in \mathbb{M}^{t'}, m \neq m', \\ &\forall p, p' \in \mathbb{P}, p \neq p', \text{ s.t. } C^{refM}_{m,m'} = 1 \text{ and } Y^P_{p,p'} = 1, \\ &\beta_m \leq \alpha_{m'} + R \cdot (3 - X^{PM}_{p,m} - X^{PM}_{p',m'} - X^{OrderM}_{m,m'}) \end{aligned} \tag{4.43}$$

$$\begin{aligned} &\forall t \in \mathbb{T}^{Relaxed}, \forall t' \in \mathbb{T}^{Fixed} \text{ or } \forall t' \in \mathbb{T}^{Relaxed}, \forall t \in \mathbb{T}^{Fixed} \\ &\forall l \in \mathbb{L}^i, \text{ s.t. } C^{refT}_{t,t'} = 1, \\ &B_t \leq A'_t + R \cdot (3 - X^{L^iT}_{l,t} - X^{L^iT}_{l,t'} - X^{OrderT}_{t,t'}) \end{aligned} \tag{4.44}$$

$$\begin{aligned} &\forall t \in \mathbb{T}^{Relaxed}, \forall t' \in \mathbb{T}^{Fixed} \text{ or } \forall t' \in \mathbb{T}^{Relaxed}, \forall t \in \mathbb{T}^{Fixed} \\ &m \in \mathbb{M}^t, m' \in \mathbb{M}^{t'}, m \neq m', \\ &\forall p, p' \in \mathbb{P}, p \neq p', \text{ s.t. } C^{refM}_{m,m'} = 1 \text{ and } Y^P_{p,p'} = 1, \\ &\beta_m \leq \alpha_{m'} + R \cdot (3 - X^{PM}_{p,m} - X^{PM}_{p',m'} - X^{OrderM}_{m,m'}) \end{aligned} \tag{4.45}$$

The order between two trains or two movements is single and guaranteed by following equations.

$$\forall t, t' \in \mathbb{T}, t \neq t', \qquad X^{OrderT}_{t,t'} + X^{OrderT}_{t',t} = 1 \tag{4.46}$$

$$\forall m, m' \in \mathbb{M}, m \neq m', \qquad X^{OrderM}_{m,m'} + X^{OrderM}_{m',m} = 1 \tag{4.47}$$

Objective function The objective we focus on is to minimize total time deviation of commercial movements. Considering feasibility of the deviation operations, the departure or arrival time of commercial movements can only be postponed. So we always have $\alpha_m \geq \alpha^{ref}_m$. Time deviations considered here only relate to commercial movements of relaxed trains, because time deviations of technical movements do not influence the traffic on railway networks.

$$\text{Minimize} \sum_{t \in \mathbb{T}^{Relaxed}, m \in \mathbb{M}^t \cap \mathbb{M}^{Commercial}} \alpha_m - \alpha_m^{ref} \tag{4.48}$$

For now, the revised timetable is generated by two steps: reinsertion and refinement. All trains are included in revised timetable, and infeasibility of timetable is absorbed by commercial time deviations. Minimization of trains cancellation in reinsertion model aims to reinsert trains cancelled in Decision model within allowable commercial time deviation upper bound F. Two conflict degrees are designed to balance the flexible search space and efficiency of calculation. Further refinement of timetable is operated by Refinement model. Trains with commercial time deviations are rescheduled and rerouted. The calculation ends with the minimization of commercial time deviations which may cause delays propagation in railway networks.

4.3 Conclusion

In this chapter, we generate the "Revised timetable" with minimal train cancellation and minimal commercial delays. To reduce computation time, these two objectives are achieved separately by two models: reinsertion model and refinement model, instead of one multi-objective optimization model.

Cancellation of trains is not the first choice in timetabling process. A better arrangement of all trains' activities is achieved by reinserting trains cancelled solved in Decision model. An upper bound of delays F is permitted for commercial movements to absorb cancellations in reinsertion model. At the same time, commercial delays may trigger delay propagation in the whole rail networks. So refinement model takes the routing decision solved in reinsertion model and reschedules all movements to minimize total commercial delays.

This version of timetable includes all trains required, permits slight train delays and limits delays propagation in rail networks. Objective function of reinsertion model is minimization of trains cancelled. Objective function of refinement model is minimization of train delays. These two models are integrated in the hybrid method based on sliding window algorithm explained in Chapter 5. The performance of reinsertion and refinement model is tested on real cases in Section 5.5.

CHAPTER 5

Hybrid method based on sliding window algorithm

We have already formulated train platforming problem, designed the decision model to generate "Feasible timetable" with minimal trains cancellation and designed the reinsertion model and refinement model to generate "Revised timetable" with maximal trains reinsertion absorbed by minimal commercial delays.

In this chapter, we need to handle the full-day timetable. To reduce the calculation efforts required, we propose a hybrid method based on a sliding window algorithm. This method integrates five functional modules: initialization, preprocessing, resolution, reinsertion and refinement. Before the description of the complete algorithm, we start by considering the real case study which provide some insight into the problem. Performance of hybrid method is evaluated by solving a real full-day timetable in Bordeaux-St-Jean station.

5.1 Real case studies

Data In order to keep the presentation clear, an example of train data is given below:

Train t	Length L_t	Movement					
		m^t	l^e_m	Type	α^{ref}_m	β^{ref}_m	D_m
215	Medium	333	18	$\mathbb{M}^{\ominus}$		16:20	6
		334	23	$\mathbb{M}^{\ominus}$	16:35		6

Table 5.1 Example of train data

The medium length train t_{215} enters the railway station from the external line 18 and leaves by the external line 23. The train arrives at the internal line at 16:20 and departs from the same internal line at 16:35. Dates are expressed in minutes from the day before midnight 0:01 to the current day midnight 0:00, so the full-day time horizon is $[1,1440]$. For example, 16:20 is expressed as 980. Considering the movement duration $S = 5min$, the technical entering movement m_{333} is executed during $[975,980)$. The commercial leaving movement m_{334} is executed during $[995,1000)$. The train occupies the relevant internal line during $[975,1000)$. The compatible list of internal lines for the trains of medium length is $\{1,2,3,4,5,6,7\}$. The preference list $\mathbb{L}_6^{Pref}$ is $[15,16,17,1,2,3,4,5,6,7,8,9]$. Based on the preference rules of the internal lines described in section 2.4.4, the preference list of internal lines of the train t_{215} is $[1,2,3,4,5,6,7]$.

Occupancy of long internal lines Considering the full case study, we compute the number of trains circulating every minute to compare the capacity of railway station to the activities in the tentative timetable. The long trains are summed up in Figure 5.1. The discrete-time on one day is represented on the horizontal axis, and the sum of trains occupying the internal line is observed per minute. The capacity for the long trains is limited by 6 long internal lines. Without considering the preference list of internal lines, we can find that the capacity of long internal lines can meet train activity. However, the maximum capacity is reached within time intervals $[616,619)$ and $[621,627)$. The resources occupancy rate μ can be measured in space-time as below:

$$\mu^{Long} = \frac{\sum_{t\in\mathbb{T}^{Long}}(B_t - A_t)}{1440 * \text{Internal line Capacity}^{Long}} \tag{5.1}$$

The occupancy rate of long internal lines is 24.2%.

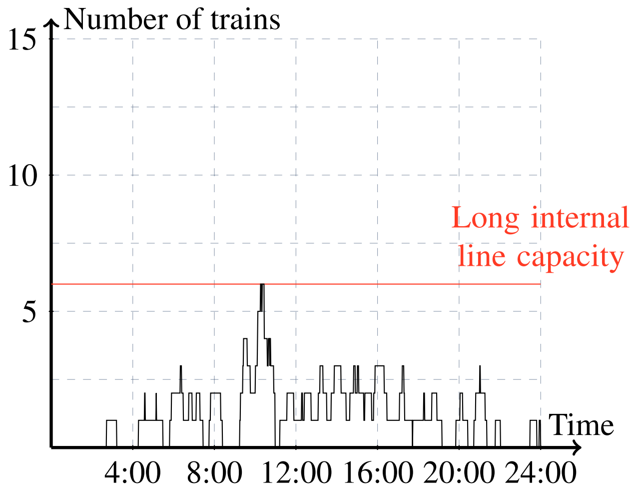


Fig. 5.1 the number of long trains with L=0

Occupancy of long and medium internal lines Similarly, long trains and medium trains are summed up in Figure 5.2. The capacity for long trains and medium trains is limited by 7 long or medium internal lines. Without considering the preference list of internal lines, we can find that the capacity of internal lines can meet the activities of long and medium length trains. There are at maximum 6 trains scheduled in parallel on the internal lines. The occupancy rate of long and medium length internal lines $\mu^{long+medium}$ is 28.2%.

Occupancy of all internal lines All trains activities on the internal lines are summed up in Figure 5.3. There are, in total, 15 internal lines in the railway station. Without considering the preference list of internal lines, we can find that the capacity of internal lines can meet the activities of all trains. The capacity limit is reached during the minute 491 (at 8:11). The occupancy rate of all internal lines $\mu^{long+medium+short}$ is up to 40.5%.

Occupancy of paths Clearly, the given tentative timetable mainly respects the internal lines capacity of the railway station but contains some difficult moments when the maximum internal line capacity is reached. To study the trains' activities in more detail, the movements scheduled in the southern part of the railway station are observed per minute in Figure 5.4. Based on the topology of the railway station in Figure 2.3, we can find 5 independent paths: [5,6,8], [4,7], [3,9], [2,10,16], [11,14]. So 5 trains at maximum are permitted to circulate in parallel. The capacity limit is reached within the time interval $[487,488)$, $[489,490)$, $[616,617)$, $[998,1002)$ and $[1067,1068)$. The trains' activities even exceed the path capacity

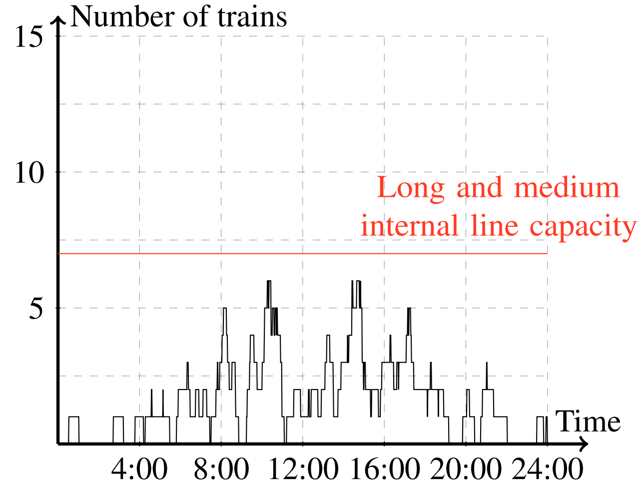


Fig. 5.2 the sum of long and medium trains with L=0

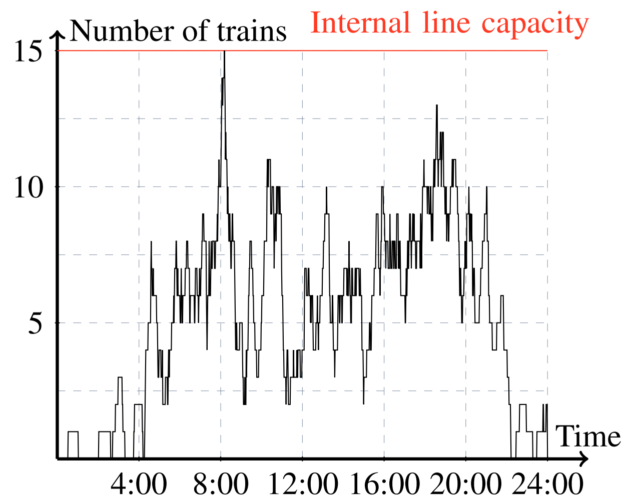


Fig. 5.3 the sum of all trains with L=0

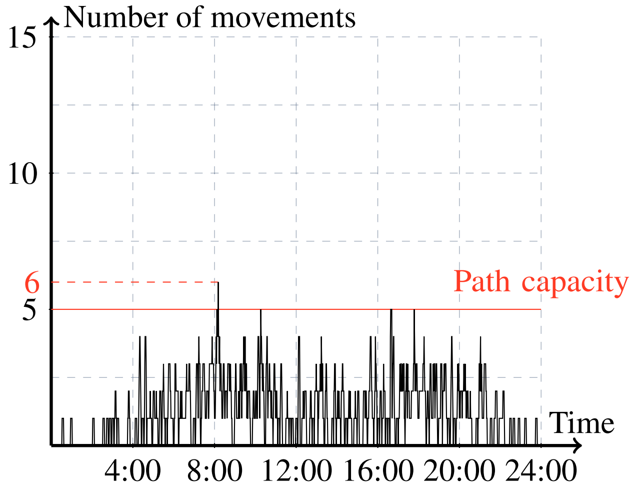


Fig. 5.4 the sum of all movements in the southern part with L=0

during the time interval $[491, 492)$. Six trains circulate in parallel at 8:11 through the railway station. The occupancy rate of path capacity is up to 62.3% calculated as below, with M cardinal number of $\mathbb{M}$.

$$\mu^{Path} = \frac{S * \mathsf{M}}{1440 * \text{Path Capacity}} \tag{5.2}$$

To eliminate the gap between trains' activities and path capacity, we take advantage of the time flexibility L which balances the paths allocation, but increases standstill durations on the internal line. But the increase of L extends the trains' occupation of internal lines. So the resolution of path conflict will make the internal line capacity more strained around 8:11 or even exceeded at 8:11. If the path and internal line conflict is unsolvable, a cancellation processing is necessary to eliminate the overflow. Another scheduling time will be proposed by railway station manager for the canceled train.

Otherwise, the resources capacity used above is only maximum theoretical capacity estimated at ideal conditions. It ignores the effects of variations in traffic and operations that occur in reality. Railway capacity is not static. It is extremely dependent on how it is used. If the theoretical capacity represents the upper theoretical bound, the practical capacity represents a more realistic measure. It is usually around 60-75% of the theoretical capacity, which has already been concluded by Kraft (1982). The complete definition and influencing factors of railway capacity are discussed in Abril et al. (2008).

Some resource allocations may decrease the capacity. The path capacity may drop to 2 trains when one train is allocated to the path $[1,2,3,7,8]$ which eliminates the majority of paths. The long internal line capacity may drop, when a short train stops at a long internal line. The trains' preference list of internal lines limits also the internal line capacity. In order to reduce the capacity loss, a precise resource allocation is highly required, for example, paths and lines allowing the maximum parallelism must be privileged.

However, the decrease of the capacity cannot be avoided. For example, the medium length train shown in Table 5.1 enters from the external line 18 and leaves by the external line 23. The preference list of internal lines is $[1,2,3,4,5,6,7]$. The distance between the two external lines means that the train must move nearly across the entire railway station. Clearly, trains will decrease the path capacity during their movements. For instance, in Table 5.2, the selectable internal lines are listed in the first column. The second and the third column represent separately the paths allocated to the entering movements and the residual path capacity. Similarly, the two last columns include the paths allocated to the leaving movements and the residual path capacity. The sum of the path capacity corresponding to entering and leaving movement is decreased to 7 or 8. Compared with the initial path capacity of 10, the path capacity loses 20%-30%. We can find that the path capacity decreases according to the path allocation. So the effective occupancy rate of path capacity is always higher than 62.3%.

	Entering Movement		Leaving Movement	
	Path	Path Capacity	Path	Path Capacity
Line 1	[1,2,10,13]	5	[1,2,3,7,8]	2
Line 2	[2,10,13]	5	[2,3,7,8]	2
Line 3	[3,10,13]	4	[3,7,8]	3
Line 4	[4,3,10,13]	3	[4,7,8]	4
Line 5	[5,4,3,10,13]	3	[5,6,8]	5
Line 6	[5,4,3,10,13]	3	[5,6,8]	5
Line 7	[5,4,3,10,13]	3	[5,6,8]	5

Table 5.2 Path capacity depending on the path allocation

5.2 Flexible time interval L

From the studies of real case above, we can see that the path occupancy rate (more than 62.3%) is much higher than the internal lines occupancy rate of 40.5%. To make the best use of the path resource, the flexible interval L is allowed for the technical movements to spread the movements on the time axis. On the other hand, the flexible interval L means also an additional standstill which intensifies the conflicts on internal lines. In this section, we define the "potential conflict" to discuss the relations between the value of L and the conflicts on the internal lines.

The number of trains which stay on internal lines during the potential time interval $[A_t{}^{Early}, B_t{}^{Late})$ is observed by minute in Figure 5.5. To study the effect of L on the potential conflicts, we draw three polylines of number of conflicting trains during the potential time interval with separately L=60, L=30 and L=0. The internal line capacity is 15 trains which is represented by the red bar.

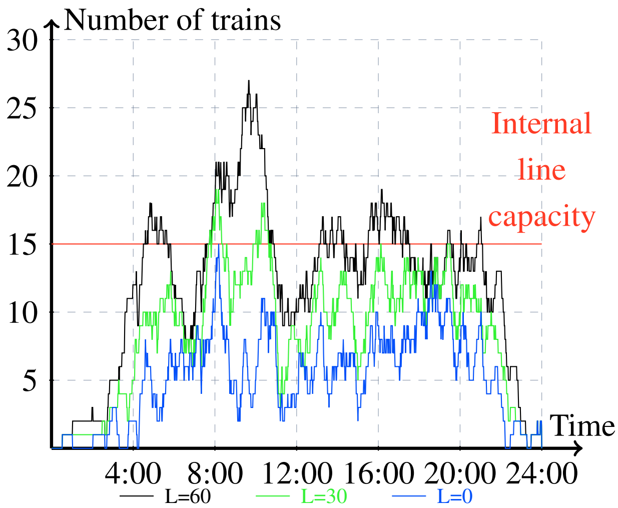


Fig. 5.5 Potential conflicts on the internal lines

The growth area of potential conflicts along with the increasing of L is the difference among the polylines. The growth area from 4:00 to 12:00 is obviously larger than that during the other time interval. So most of technical movements are executed from 4:00 to 12:00. The potential conflicting trains' numbers up to 27 trains with $L = 60$ are over the internal line capacity during several time intervals, specially from 7:00 to 10:00.

The occupancy rates of internal lines in one day 0:00-24:00 along with L is represented by the black lines in the Figure 5.6. The horizon axis is the value of L, and the vertical axis is the occupancy rate. The occupancy rate of long internal lines increases from 24.2% to 51.5%. The occupancy rate of long and medium internal lines increases from 28.2% to 64.3%. The occupancy rate of all internal lines is up to 101.6% which exceeds the capacity limit. On the whole, $L = 60$ is acceptable for the internal lines capacity.

On the other hand, the majority of potential conflicts exists from 4:00 to 22:00. The three internal lines occupancy rate from 4:00 to 22:00 are drawn in red. The occupancy rate of long internal lines increases from 34.7% to 71.7%. The occupancy rate of long and medium internal lines increase from 39.2% to 85.5%. The occupancy rate of all internal lines increase from 57.0% up to 136.8% which highly exceeds the capacity limit. In this case, we reduce L to $L^{opt} = 32$ to adapt the internal line capacity.

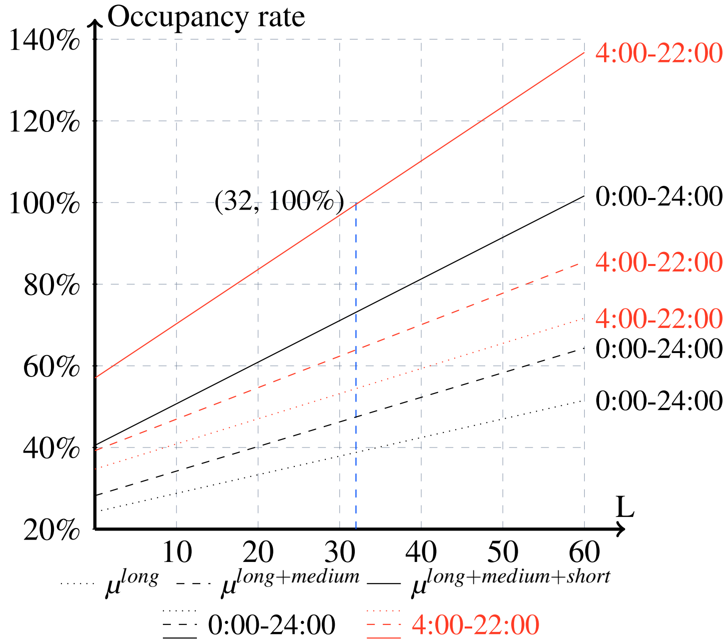


Fig. 5.6 Relation between L and occupancy rate

The reduced L will shorten the potential scheduling interval $[\alpha_m^{Early}, \beta_m^{Late})$ and $[A_t^{Early}, B_t^{Late})$, so the number of potential conflicts decreases. As a result, the number of constraints (3.31) and (3.32) is greatly reduced. But some feasible solutions may be eliminated by the reduction of L, a compensation measure is necessary to reschedule the cancelled trains with $L = 60$. The complete algorithm is described in the next section.

5.3 Complete algorithm

In this section we describe a complete algorithm for solving the problem of platforming trains in one-day timetable through a railway station, based on the formalization proposed in section 2.4.

The complete algorithm can be described as follows:

1. Initialization: generate the railway station layout $\mathbb{R} = (\mathbb{S}, \mathbb{L}, \mathbb{P})$ formalization and collect train data in the form of Table 5.1.
2. Preprocessing: try to reduce the problem instance in advance, thereby generating the intermediate parameters and sets with properly reduced flexible interval L^{opt} generated as in section 5.2.
3. Resolution: solve the one-day timetable with minimum trains cancellation by combination of mixed-integer linear programming (MILP) model described in section 3 and cumulative sliding window algorithm.
4. Reinsertion: reinsert cancelled trains and their movements within acceptable flexible time intervals.
5. Refinement: minimize the deviations of commercial movements.

The structure of the complete algorithm is shown in Figure 5.7.

In the first step, railway station layout is formalized as described in section 2.4.1. Train data are collected from railway station managers and expressed as in Table 5.1.

In the second step, the intermediate parameters and sets, listed as below, are generated according to the formalization obtained in the first step.

1. Preference list of internal lines for each train $\mathbb{L}_t^{Pref}$ as described in section 2.4.4.
2. Potential scheduling intervals $[\alpha_m^{Early}, \beta_m^{Late})$ and $[A_t^{Early}, B_t^{Late})$ with L^{opt} generated as in section 2.4.2.1.
3. Potential conflicts probes $C_{m,m'}^{refM}$ and $C_{t,t'}^{refT}$ defined in section 3.2.

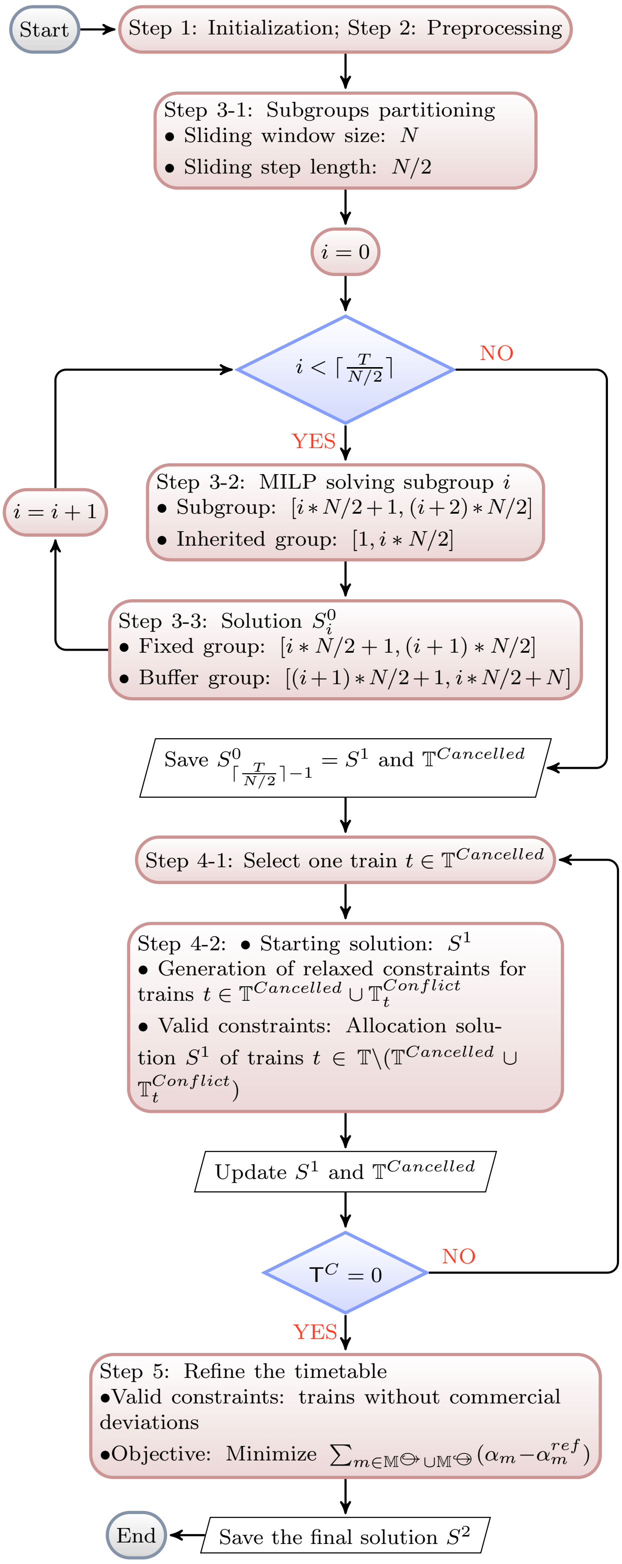


Fig. 5.7 Complete algorithm

4. Pairs of conflicting paths parameters $Y^P_{p,p'}$ defined in section 3.2.

Step 3 involves the generation of full-day timetable with cancellation processing. Due to the size of the whole problem, we propose to solve the problem step by step using a sliding window approach controlled by a parameter $N \in \mathbb{N}$ (sliding window width). At each step, N trains are considered in the mathematical model. The first $N/2$ trains solved will be stored, the reminder being relaxed to be solved again at next step. T trains are divided into $\lceil \frac{\mathsf{T}}{N/2} \rceil$ subgroups of N trains in chronological sequence until the end of the problem: $[1,N]$, $[N/2+1, 3N/2]$, $[N+1, 2N] \ldots [(\lceil \frac{\mathsf{T}}{N/2} \rceil - 1) \cdot N/2 + 1, \mathsf{T}]$. An illustrative execution is given in Figure 5.8, with $N = 60$ and $\mathsf{T} = 247$. All trains before the first train of one subgroup compose its *inherited group* as valid constraints. The first $N/2$ trains of one subgroup compose its *buffer group* which acts as a conflicts holding area between this subgroup and its previous subgroup. Solutions of this buffer group will be fixed at next step and will belong to next inherited group. The last $N/2$ trains of one subgroup compose its *new group*. To solve every subgroup of N trains, a mathematical model is formalized as described in section 3 and solved by CPLEX branch-and-cut algorithm.

A full-day conflict-free timetable with minimum train cancellations is obtained at the end of this step and is represented by S^0. $\mathbb{T}^{Cancelled}$ denotes the group of trains cancelled with cardinal number T^C.

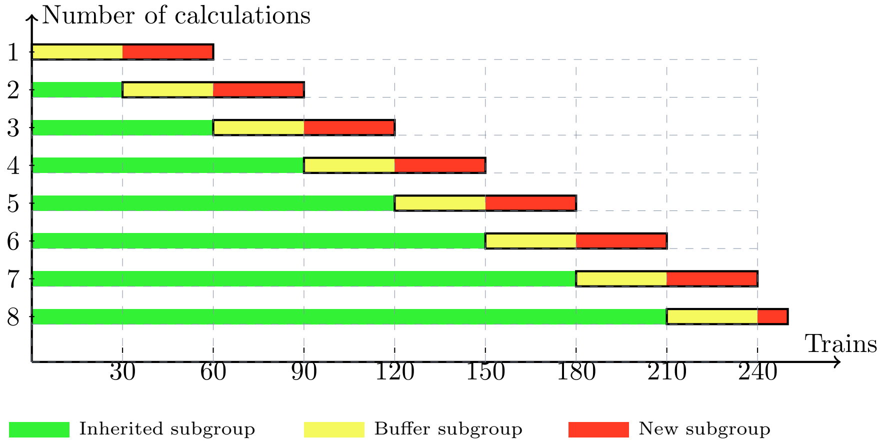


Fig. 5.8 Cumulative sliding window algorithm

As a compensation measure for step 3, reinsertion model described in section 4.1 is applied to reinsert cancelled trains, one by one, by relaxing constraints (2.4)-(2.7) in step 4. Every time a cancelled train $t \in \mathbb{T}^{Cancelled}$ is selected. The potential conflicting trains of the cancelled train $\mathbb{T}_t^{Conflict} = \{t' | C_{t,t'}^{refT} = 1\}$ and all cancelled trains $\mathbb{T}^{Cancelled}$ are given the full flexible interval L for technical movements and the acceptable deviation upper bound F for commercial movements. To meet the practical demand, we can set the value of F to provide different versions of timetables. With $F = 0$ and $L = 60$, we obtain the timetable with minimum train cancellations. In the next section, we set $F = 10$ and $L = 60$ to absorb train cancellations.

Trains without flexible time interval $\mathbb{T} \backslash (\mathbb{T}^{Cancelled} \cup \mathbb{T}_t^{Conflict})$ inherit the allocation solution S^0 and are considered as valid constraints. The objective of this step is to absorb train cancellations.

To avoid additional cancellations, an initial solution obtained in previous calculation is added to start a new reinsertion calculation. A full-day conflict-free timetable relaxed without train cancellation is obtained at the end of this step and is represented by S^1.

The last step aims to minimize the deviation of commercial movements by refinement model described in Section 4.2. The scheduling and routing solution of all trains without the time deviation of commercial movements obtained in step 4 is retained as valid constraints. The trains delayed are given the starting solution corresponding to S^1. A full-day conflict-free timetable with minimum deviation for commercial movements without train cancellation is obtained at the end of this step and is represented by S^2.

5.3.1 Preprocessing: conflicts on external lines

As the reference times of commercial movements are fixed in resolution step, the conflicts between commercial movements on external lines must be eliminated in preprocessing step. The compatibility of the given external lines is verified as shown in the equation (5.3). Obviously, the modification of commercial movements' reference time is required to be minimized.

$$\forall m, m' \in \mathbb{M}^{\ominus\!\!\rightarrow} \bigcup \mathbb{M}^{\leftarrow\!\!\ominus} \text{s.t.} l_m^e = l_{m'}^e$$

$$[\alpha_m^{ref}, \beta_m^{ref}) \cap [\alpha_{m'}^{ref}, \beta_{m'}^{ref}) = \varnothing \quad (5.3)$$

Parameters

- R is a sufficiently big constant.
- α_m^{ref} is the reference starting time of the movement m.
- β_m^{ref} is the reference ending time of the movement m.
- S is the time allocated to a movement. In our context, $S = 5$ minutes.

Variables

- α_m is the actual starting time of the movement m.
- β_m is the actual ending time of the movement m, $\alpha_m + S = \beta_m$.
- $X_{m,m'}^{OrderM}$ identifies the time order of two movements using the same external line. If m circulates before m', $X_{m,m'}^{OrderM} = 1$. Otherwise $X_{m,m'}^{OrderM} = 0$.
- Dev_m is the deviation of movements' reference time. $Dev_m = |\alpha_m - \alpha_m^{ref}|$.

Compatibility of external lines Constraint (5.4) indicates that two commercial movements cannot occupy the same external line at the same time. This rule is expressed as follows:

$$\forall m, m' \in \mathbb{M}^{\ominus\rightarrow} \bigcup \mathbb{M}^{\leftarrow\oplus}, m \neq m', [\alpha_m^{ref}, \beta_m^{ref}) \cap [\alpha_{m'}^{ref}, \beta_{m'}^{ref}) \neq \varnothing, \text{s.t. } l_m^e = l_{m'}^e$$
$$\beta_m \leq \alpha_{m'} + R \cdot (1 - X_{m,m'}^{OrderM}) \tag{5.4}$$
$$\forall m, m' \in \mathbb{M}^{\ominus\rightarrow} \bigcup \mathbb{M}^{\leftarrow\oplus}, m \neq m' [\alpha_m^{ref}, \beta_m^{ref}) \cap [\alpha_{m'}^{ref}, \beta_{m'}^{ref}) \neq \varnothing, \text{s.t. } l_m^e = l_{m'}^e$$
$$X_{m,m'}^{OrderM} + X_{m',m}^{OrderM} = 1 \tag{5.5}$$

The constraint (5.4) indicates that if two commercial movements m and m' are allocated to the same external line l and if the movement m circulates before m', then the term $1 - X_{m,m'}^{OrderM} = 0$. We have then $\beta_m \leq \alpha_{m'}$. Otherwise this term is larger than zero, and the constraint (5.4) is relaxed. The order of two movements is generated by equation (5.5).

Generation of time deviation The time deviations of commercial movements $Dev_m = |\alpha_m - \alpha_m^{ref}|$ are generated as follows:

$$
\begin{aligned}
&\forall m \in \mathbb{M}^{\ominus\!\rightarrow} \bigcup \mathbb{M}^{\hookrightarrow\!\ominus} \\
&Dev_m \geq \alpha_m - \alpha_m^{ref} \qquad (5.6) \\
&Dev_m \geq \alpha_m^{ref} - \alpha_m \qquad (5.7)
\end{aligned}
$$

Objective function The objective we focus on is to minimize the deviation of commercial movements' reference times:

$$
\min \sum_{m \in \mathbb{M}^{\ominus\!\rightarrow} \bigcup \mathbb{M}^{\hookrightarrow\!\ominus}} Dev_m \qquad (5.8)
$$

Applied on the tentative timetable, the commercial movements' conflicts on externa lines are eliminated by the minimal time deviations in the preprocessing step.

5.3.2 Subgroups partitioning strategies

Step 3-1 in Figure 5.7 is subgroups partitioning. In that way, full-day platforming problem is divided into several relatively small problems. While simplifying the full problem, an appropriate subgroups partitioning strategy avoids triggering extra insolvable conflicts. $\mathbb{T}_i^{sub}$ denotes the i^{th} trains subgroup. The principles of subgroups partitioning strategy can be summarized as follows:

1. No time intersection should exist between potential time interval of inherited group and new group in the same trains subgroup.

2. The size of trains subgroup should not be too big to handle. Computational time of each subgroup should be acceptable.

With L calculated in Section 5.2, potential scheduling time interval of trains $[A_t^{Early}, B_t^{Late})$ can be generated as shown in Section 2.4.2.1. So potential scheduling time interval of the i^{th}

trains subgroup, denoted by $[A_i^{sub}, B_i^{sub})$, is calculated as below:

$$[A_i^{sub}, B_i^{sub}) = \bigcup_{\forall t \in \mathbb{T}_i^{sub}} [A_t^{Early}, B_t^{Late}) \tag{5.9}$$

Similarly, potential scheduling time intervals of buffer and new group in trains subgroup $\mathbb{T}_i^{sub}$ are separately represented as:

$$[A_i^{buff}, B_i^{buff}) = \bigcup_{\forall t \in \mathbb{T}_i^{buff}} [A_t^{Early}, B_t^{Late}) \tag{5.10}$$

$$[A_i^{new}, B_i^{new}) = \bigcup_{\forall t \in \mathbb{T}_i^{new}} [A_t^{Early}, B_t^{Late}) \tag{5.11}$$

Trains in inherited group are already scheduled and generated by actual scheduling time:

$$[A_i^{inh}, B_i^{inh}) = \bigcup_{\forall t \in \mathbb{T}_i^{inh}} [A_t, B_t) \tag{5.12}$$

So the first partitioning principle can be expressed as follows:

$$[A_i^{inh}, B_i^{inh}) \cap [A_i^{new}, B_i^{new}) = \varnothing \tag{5.13}$$

In our case, trains are numbered in chronological sequence. If buffer groups do not contain enough trains to separate inherited and new groups, trains in new group may have unsolvable conflicts with trains in inherited group which are used as valid constraints. For example, in train subgroup i, $[A_i^{inh}, B_i^{inh}) = [20, 340)$ and $[A_i^{new}, B_i^{new}) = [200, 490)$, trains in inherited group during $[200, 340)$ are arranged without the agreement of trains in new group. As a result, some avoidable conflicts may be triggered. We call this kind of conflicts *structural conflicts*. Here we consider only direct conflicts. If indirect conflicts which are caused through "middle-train" are considered, time separation between inherited group and new group is favorable, the larger the better. In fact, the first principle tells us that big size of train subgroups is preferred to avoid *structural conflicts*.

Once a certain limit is reached, a change in the opposite direction is inevitable. If we have only one train subgroup including all trains in the day, the problem is not to be simplified. Huge computation effort is required. So the opposite limit appears as described in the second principle. Computation time depends on several elements, for example, computer operation speed, space of RAM and scale of problems.

5.4 From methods to implementation

After describing the complete algorithm, we present the data structure and programming environment. Our algorithm is mainly programmed in **C++**. The implementation is realized under Linux Operating System. C++ files are executed by the **shell** which is a command language interpreter. All input and output data are written in *JSON* data format.

5.4.1 Data structure

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language, Standard ECMA-262 3rd Edition - December 1999. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C# , Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.

JSON is built on two structures:

- A collection of name/value pairs. In various languages, this is realized as an object, record, struct, dictionary, hash table, keyed list, or associative array.
- An ordered list of values. In most languages, this is realized as an array, vector, list, or sequence.

These are universal data structures. Virtually all modern programming languages support them in one form or another. It makes sense that a data format that is interchangeable with programming languages is also based on these structures.

Data stream is expressed in *JSON* format through our whole algorithm architecture. In this way, original data can be delivered by Web to a powerful calculator in distance. The results obtained are returned to railway station by Web. So the performance of our algorithm will not be limited by the computer capacity. All computational experiments in our thesis are calculated on my own laptop. The efficiency of algorithm may be improved with a super-computer.

5.4.2 Programming architecture

The programming architecture can be described in Figure 5.9. The whole process is automated by the process control module which is updated according to the intermediate results obtained.

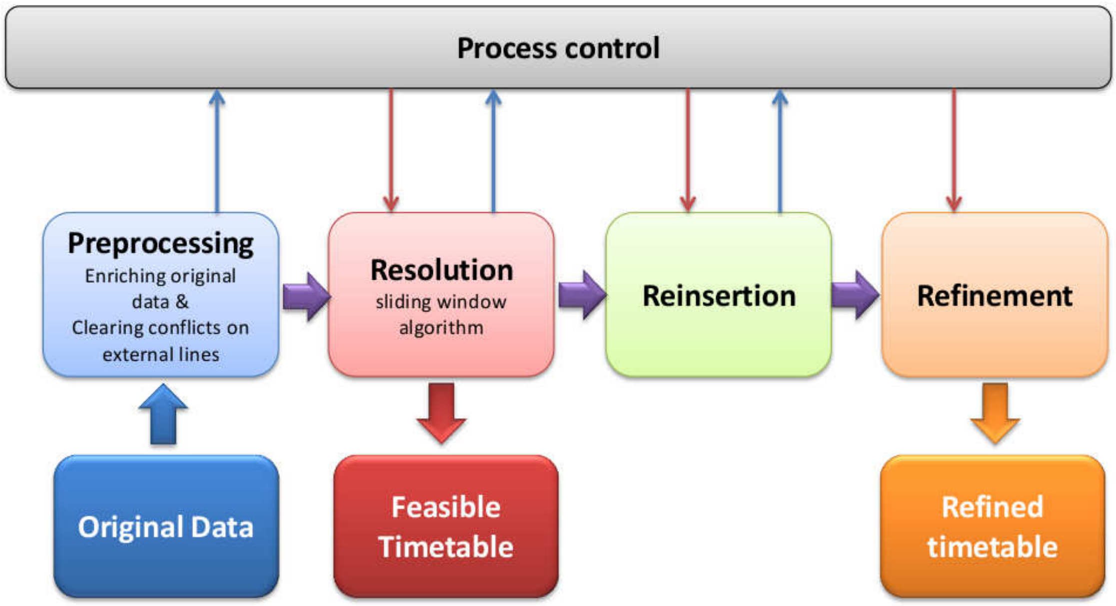


Fig. 5.9 Programming architecture

1. Initialization: Input the railway station layout $\mathbb{R} = (\mathbb{S}, \mathbb{L}, \mathbb{P})$ and trains' activities as original data in *JSON* data format.

2. Preprocessing:

   - Conflicts on external lines are cleared by MILP model found in Section 5.3.1 which is solved by CPLEX in C++.
   - C++ control file enriches original data with the intermediate parameters and sets based on the original data. JSON->C++->JSON.

3. Resolution: Cumulative sliding window algorithm process control is integrated in JSON data file including sub-groups partitioning information and stop criterion. MILP model described in section 3 of sub-problems is solved by CPLEX in C++. When the "Feasible timetable" is generated, the stop criterion automatically calls for the reinsertion calculation by adding Shell commands.

4. Reinsertion: The reinsertion MILP model is solved by CPLEX in C++. The train cancelled to be reinserted can be chosen by dispatcher. When the reinsertion is accomplished, the refinement is called by Shell commands.

5. Refinement: The refinement MILP model is solved by CPLEX in C++.

5.5 Computational results

The results of the complete algorithm with respect to the problem described in Section 5.1 are displayed in this section. The computational results are obtained by using CPLEX version 12.6 on a 64 bits computer under Linux with Intel i5-2520M CPU at 2.5GHz and 8GB memory RAM. For each group, the calculation time is limited to 500 seconds. Routing variables $X_{l,t}^{L^iT}$ are defined in the preference order of internal lines for the train t. The buffer group solved in the previous subgroup is generated as the starting solution for the next subgroup.

Solutions obtained in step 3 are presented in Table 5.3. The one-day timetable is divided into 8 subgroups of trains with $N = 60$ shown in the first column. The second and third column contains the numbers of technical and commercial movements for the relevant subgroup. The trains subgroup can be divided into three groups: inherited group, fixed group and buffer group which are described, in the 5th and 6th columns, by the number of trains and the time interval occupied in minutes. The inherited group is generated as valid contraints. With $N = 60$, there is no intersection between the time interval of buffer group and that of inherited group to avoid insolvable potentiel conflicts with inherited group. Once the variables and constraints shown in 7th and 8th columns are sent to the solver, CPLEX presolve eliminates redundant constraints and variables according to valid constraints. The reduced problem is described in 9th and 10th columns by the number of variables and constraints. The minimum number of trains cancelled solved within 500 seconds and the resolution information are shown in the last three columns in Table 5.3. At last, there are 9 trains cancelled in the one-day timetable. The group of trains cancelled $\mathbb{T}^{Cancelled}$ is $[35, 59, 71, 84, 91, 96, 97, 176, 191]$.

We relax potential conflicting trains and reinsert trains cancelled in step 4. The relaxation parameters F and L are assigned to values in order to meet different demands. If we need the timetable with minimum train cancellations, we set $F = 0$ and $L = 60$.

Trains Subgroup	Movements Tech.	Movements Comm.	Group Type	Trains	Time Interval(min)	Before presolve Var.	Before presolve Cons.	After presolve Var.	After presolve Cons.	TiLim=500s Obj	TiLim=500s GAP	TiLim=500s Solve time
0-59	45	78	Inherited	-	-	118480	10659	53808	1901	0	0.00%	50.23
			Buffer	0-29	30-406							
			New	30-59	322-530							
30-89	41	78	Inherited	0-29	30-406	228352	16786	71445	2296	2	100.00%	500.05
			Buffer	30-59	322-530							
			New	60-89	438-662							
60-119	48	73	Inherited	0-59	30-530	318805	22927	77158	2416	7	85.71%	500.07
			Buffer	60-89	438-662							
			New	90-119	575-795							
90-149	47	76	Inherited	0-89	30-662	364517	29051	45340	1957	7	33.33%	500.02
			Buffer	90-119	575-795							
			New	120-149	735-967							
120-179	40	84	Inherited	0-119	30-795	410493	35135	35476	2013	7	0.00%	165.09
			Buffer	120-149	735-967							
			New	150-179	892-1133							
150-209	31	92	Inherited	0-149	30-967	454655	41208	37333	2151	8	25.00%	500.04
			Buffer	150-179	892-1133							
			New	180-209	1012-1208							
180-239	32	89	Inherited	0-179	30-1133	505267	47109	37303	2152	9	12.50%	275.81
			Buffer	180-209	1012-1208							
			New	210-239	1143-1361							
210-246	22	53	Inherited	0-209	30-1208	517969	49001	22695	1269	9	9.37%	6.10
			Buffer	210-239	1143-1361							
			New	240-246	1215-1494							

Table 5.3 Cumulative sliding window algorithm solution

We need to absorb train cancellations within minimum time deviation of commercial movements. The deviation upper bound F for commercial movements is 10 minutes. The flexible time interval L grows up to 60 minutes. The calculation information of every reinsertion process is displayed in Table 5.4. At each line, we select a train t to reinsert. The conflicting trains $\mathbb{T}_t^{Conflict}$ are relaxed as shown in first column. The cancellation is absorbed step by step until no train is cancelled anymore. The group of trains cancelled after the reinsertion is shown in the last column.

In order to refine the solution obtained in step 4, the last step tries to minimize the time deviation of commercial movements. At last, 9 train cancellations are absorbed by 182 minutes deviation which involves 37 trains. Deviation of 3 trains reaches the deviation upper bound 10 minutes. 8 trains are postponed for more than 6 minutes. Others 29 trains have a delay of less than 5 minutes.

Steps 4 and 5 can also be used as a real-time platforming tool to insert additional trains.

Trains	Before presolve		After presolve		TiLim=500s			Trains
relaxed	Var.	Con.	Var.	Con.	Solution	GAP	Solve time	Cancelled
$\mathbb{T}_{35}^{Conflict}$	586236	49397	49125	1601	8	37.50%	500.01	53,72,84,91,96,97,176,191
$\mathbb{T}_{72}^{Conflict}$	606922	48919	115004	2582	6	16.67%	500.08	57,59,86,90,93,98,191
$\mathbb{T}_{59}^{Conflict}$	649567	48807	159307	3357	5	60.00%	500.12	65,91,96,176,191
$\mathbb{T}_{96}^{Conflict}$	689604	49845	101883	1940	4	25.00%	500.04	65,91,176,191
$\mathbb{T}_{176}^{Conflict}$	757380	50644	56821	2278	2	0.00%	126.95	65,91
$\mathbb{T}_{91}^{Conflict}$	802516	50973	90839	2350	1	0.00%	258.5	65
$\mathbb{T}_{65}^{Conflict}$	802381	50818	136958	2588	0	0.00%	394.69	-

Table 5.4 Reinsertion of cancelled trains.

5.6 Conclusion

In this chapter, we design a hybrid method based on sliding window algorithm to organize all functions together and to solve full-day timetable. At the beginning of this chapter, we study real cases by resources occupation indicators. Difficulty matter is evaluated and analysed by occupation indicators and flexible time interval L for technical movements. There are three difficulties in our problem:

- Complex railway station local networks.
- Flexible time interval for technical movements $L = 60$.

- Dense rail transport with infeasible cases.

These three elements give us an immense search space to find an optimum solution. To solve this large-scale problem, we apply sliding window algorithm to decompose the full-day timetable into tractable sub-problems. The complete algorithm can be described as follows:

1. Initialization: generate the railway station layout $\mathbb{R} = (\mathbb{S}, \mathbb{L}, \mathbb{P})$ formalization and collect train data in the form of Table 5.1.

2. Preprocessing: try to reduce the problem instance in advance, thereby generating the intermediate parameters and sets with properly reduced flexible interval L^{opt} generated as in section 5.2.

3. Resolution: solve the one-day timetable with minimum trains cancellation by combination of mixed-integer linear programming (MILP) model described in section 3 and cumulative sliding window algorithm.

4. Reinsertion: reinsert cancelled trains and their movements within acceptable flexible time intervals.

5. Refinement: minimize the deviation of commercial movements.

After the introduction of complete algorithm, we present preprocessing of conflict-free on external lines and subgroups partition strategies in details. At last, the hybrid method is tested on real case in railway station Bordeaux st Jean. The “Feasible timetable” of 247 trains (more than 500 movements) can be generated in around 48 minutes with 9 trains cancelled. From Table 5.4, we can see that there are only 2 sub-problems resulting with 0% GAP among 8 sub-problems. In the next Chapter, we aim to improve the result quality regarding computational time and GAP.

CHAPTER 6

Tri-level decomposition method: generation of initial solution

In this chapter, we try to generate an initial solution by heuristics methods for Decision model. The initial solution can be used as an upper bound in Branch-and-Bound calculation processing, so as to improve the quality of "Feasible timetable" and reduce computational time. In this case, we propose tri-level decomposition method which can not only generate a feasible solution but also provide us a comprehensible explanation of trains cancellation.

Train platforming problem can be formalised as three linear programming problems which consists of two integer linear programming models and one continuous linear programming model.

Train platforming problem consists of two sub-problems:

- Scheduling: modification of technical movements' departure or arrival time.
- Routing: allocation of internal lines and paths.

Diverse scheduling and routing decisions turn the identification of conflicts into a dynamic process. If we solve the two sub-problems at once by decision model, the combinatorial explosion constructs a huge search space. So a global optimal solution cannot be found within reasonable computation time. We need to find an equilibrium between quality of solution and computational time.

If we solve separately two sub-problems, scheduling decision variables are known as constant in routing sub-problem, and routing decision variables are known as constant in scheduling sub-problem. In this way, conflicts identification is a static process.

In our case, variables can be classified into three groups which are denoted separately by three vectors x, y and c:

- Routing variables $x = (X^{LiT}_{l,t}, X^{PM}_{p,m})$.
- Scheduling variables $y = (\alpha_m, \beta_m, A_t, B_t, X^{OrderT}_{t,t'}, X^{OrderM}_{m,m'})$.
- Cancellation variables $c = (X^{CancelM}_{m}, X^{CancelT}_{t})$.

The decision model described in Chapter 3 is represented by $min_{xyc}D(x,y,c)$ which involves all three variable vectors (x,y,c) to minimize the train cancellations, and the subscript vectors (x,y,c) are treated as variables. The decision model can be formalized as two sub-problems scheduling-routing. Upper level model described in section 6.2 is to solve scheduling sub-problem $min_{yc}S(x,y,c)$ where the routing variables x are obtained by solving the lower level routing sub-problem described in section 6.1. To avoid complex equations (3.31) and (3.32) in lower level programming model, routing subproblem is structured into two optimization programming steps. The lower level of routing subproblem described in section 6.1.1 decides the trains to be cancelled $min_c R_1(x,y,c)$ with weighted trains cancellation objective, and the trains cancelled are not allocated to any resources. With c solved by the lower level of routing sub-problem and y initialized by tentative timetable, the upper level of routing sub-problem R_2 described in section 6.1.2 aims at allocating internal lines and paths to the trains cancelled. Objective function of $max_x R_2(x,y,c)$ is to maximize the path tolerance and to maximize the headway time of all technical movements cancelled. The tri-level decomposition structure is shown in Figure 6.1.

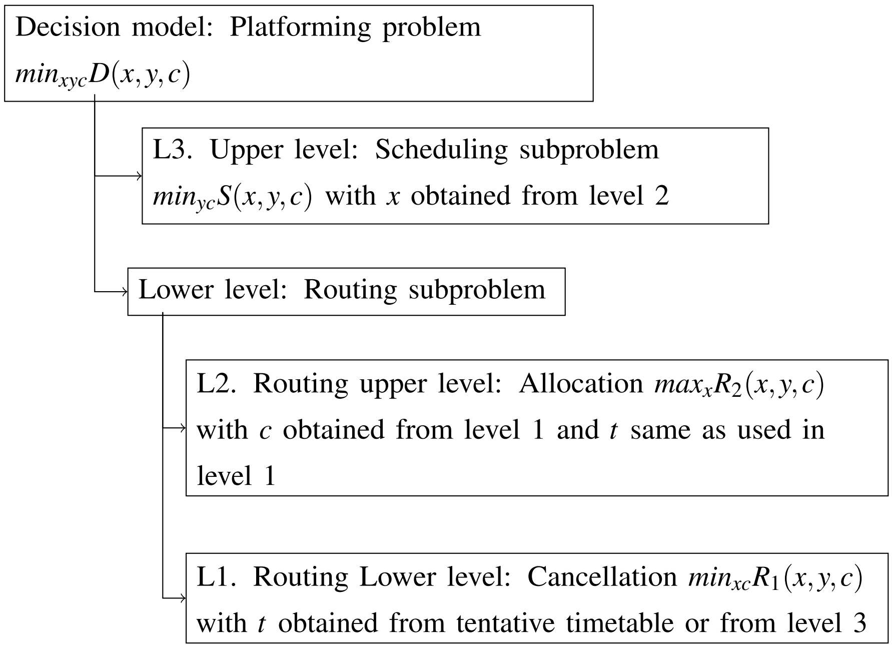


Fig. 6.1 Tri-level decomposition method

6.1 Lower level programming model: allocation of internal lines and paths

The lower level of train platforming problem aims to allocate internal lines to trains and allocate paths to movements with effective times. The effective times of trains and movements α_m, β_m, A_t and B_t are initialised from tentative timetable and improved by the upper level. With the given effective times, two parameters are proposed and used in lower level:

- $C^{effM}_{m,m'}$ identifies conflicts between two movements m and m'. If $[\alpha_m, \beta_m) \cap [\alpha_{m'}, \beta_{m'}) \neq \varnothing$, $C^{effM}_{m,m'} = 1$. Otherwise $C^{effM}_{m,m'} = 0$.
- $C^{effT}_{t,t'}$ identifies conflicts between two trains t and t'. If $[A_t, B_t) \cap [A_{t'}, B_{t'}) \neq \varnothing$, $C^{effT}_{t,t'} = 1$. Otherwise $C^{effT}_{t,t'} = 0$.

In order to cooperate with upper level, we not only need to provide a resources allocation strategy with minimal cancellations, but also assign reasonable resources to trains and movements cancelled which may help to reduce cancellations in upper level. As a result, the routing sub-problem is constructed as two sub-problems. Furthermore, this mechanism simplifies the complexity of routing problem through replacing resources compatibility constraints (3.31) and (3.32) by equations (6.4) and (6.5). In routing lower level, we concentrate to minimize the trains cancellation. Internal lines and paths are not to be allocated to trains and movements cancelled. In routing upper level, cancellation results are obtained from routing lower level, and re-allocation of resources aims to enforce resolvability of train cancellations in scheduling sub-problem while guaranteeing feasibility of the timetable.

6.1.1 Level 1 Routing lower level $min_{xc}R_1(x,y,c)$: minimization of weighted trains and technical movements cancellations

In routing lower level, the allocation principle is the same as used in the decision model, besides the trains and movements cancelled cannot be assigned with any internal line and path. In this case, we need to replace the routing constraints (3.11), (3.12) and (3.13) by following constraints (6.1), (6.2) and (6.3) with consideration of cancellation decisions.

Preference of internal lines. If the train t is cancelled $X_t^{CancelT} = 1$, the train t is not to be assigned with any internal line $\sum_{l_i \in \mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} = 0$.

$$\forall t \in \mathbb{T}, \sum_{l_i \in \mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} + X_t^{CancelT} = 1 \tag{6.1}$$

Allocation of paths. If the movement m is cancelled $X_m^{CancelM} = 1$, the movement m is not to be assigned with any path $\sum_{p \in \mathbb{P}^{l_e}} X_{p,m}^{PM} = 0$, as shown in equation (6.2). Despite the internal line allocated to the train, the movement cancellation $X_m^{CancelM} = 1$ will relax the equation (6.3).

$$\forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e}, \qquad \sum_{p \in \mathbb{P}^{l_e}} X_{p,m}^{PM} + X_m^{CancelM} = 1 \tag{6.2}$$

$$\forall t \in \mathbb{T}, \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \cap \mathbb{M}^t, \forall l_i \in \mathbb{L}_{l_e}^i$$
$$\sum_{p \in \mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} + X_m^{CancelM} \geq X_{l_i,t}^{L^iT} \tag{6.3}$$

Compatibility of lines. Two trains with time intersection $C_{t,t'}^{effT} = 1$ cannot be assigned with the same internal line. If the train t is cancelled, internal lines are not permitted to be allocated to the train $X_{l,t}^{L^iT} = 0$ expressed by (6.1). So compatibility of internal lines can be expressed as follows:

$$\forall t, t' \in \mathbb{T}, t \neq t', \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref}, \text{ s.t. } C_{t,t'}^{effT} = 1,$$
$$X_{l,t}^{L^iT} + X_{l,t'}^{L^iT} \leq 1 \tag{6.4}$$

Compatibility of switches. Two movements m and m' with time intersection $C_{m,m'}^{effM} = 1$ cannot pass two conflicting paths $Y_{p,p'}^{P} = 1$. If the movement m is cancelled, paths are not permitted to be allocated to the movement $X_{p,m}^{PM} = 0$ expressed by (6.2) and (6.3). So compatibility of paths can be expressed as follows:

$$\forall m, m' \in \mathbb{M}, m \neq m', \forall p, p' \in \mathbb{P}, p \neq p', \text{ s.t. } C_{m,m'}^{effM} = 1 \text{ and } Y_{p,p'}^{P} = 1,$$
$$X_{p,m}^{PM} + X_{p',m'}^{PM} \leq 1 \tag{6.5}$$

Objective function. Different from cancellation processing in decision model and upper level programming, trains and corresponding movements cancellations are no longer coincident as represented previously in equation (3.34), but distinguished to find the cause of train cancellations. $X_t^{CancelT} = 1$ represents that all internal lines preferred by train t are not available during $[A_t, B_t]$. $X_m^{CancelM} = 1$ represents that no conflict-free path preferred by movement m is available during $[\alpha_m, \beta_m]$.

To improve the timetable under the scheduling-routing mechanism, different cancellations are treated in different ways. In scheduling process, only the effective time of technical movements can be modified, and the standstill of trains on internal lines can only be prolonged. If a train is cancelled due to conflicts on internal line, the involved movements will be cancelled as well, as represented in equation (6.7). If a commercial movement is cancelled, the involved train and technical movements are also to be cancelled as shown in equation (6.8), because the effective times of commercial movements are fixed. If a technical movement is cancelled, we cannot make an involved cancellation, because the effective time of technical movements may be revised in scheduling sub-problem. Finally, a weighted objective function (6.6) is generated according to solvability of the three kinds of conflicts in the scheduling-routing mechanism. P denotes the cancellation penalty.

There are two versions of effective times for trains and movements: tentative timetable and timetable revised by upper level scheduling model. Standstills of trains on internal lines in tentative timetable are minimal. Standstills are prolonged in revised timetable to avoid

The routing upper level programming model is defined by:

$$R_1\text{: Minimize } \sum_{t\in\mathbb{T}} P * X_t^{CancelT} + \sum_{m\in\mathbb{M}^{\ominus\rightarrow}\bigcup\mathbb{M}^{\leftarrow\ominus}} P * X_m^{CancelM} + \sum_{m\in\mathbb{M}^{\ominus\rightarrow}\bigcup\mathbb{M}^{\leftarrow\ominus}} X_m^{CancelM} \tag{6.6}$$

$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t, \qquad X_m^{CancelM} \geq X_t^{CancelT} \tag{6.7}$$

$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t, \text{ s.t. } m \in \mathbb{M}^{\circleddash\rightarrow} \bigcup \mathbb{M}^{\hookleftarrow\oplus} \qquad X_m^{CancelM} = X_t^{CancelT} \tag{6.8}$$

$$\forall t \in \mathbb{T} \qquad \sum_{l_i\in\mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} + X_t^{CancelT} = 1 \tag{6.9}$$

$$\forall m \in \mathbb{M}^{l_e}, \quad \forall l_e \in \mathbb{L}^e \quad \text{s.t. } Y_{l_e,m}^{L^eM} = 1 \qquad \sum_{p\in\mathbb{P}^{l_e}} X_{p,m}^{PM} + X_m^{CancelM} = 1 \tag{6.10}$$

$$\forall t \in \mathbb{T}, \quad \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \cap \mathbb{M}^t, \forall l_i \in \mathbb{L}_{l_e}^i, \sum_{p\in\mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} + X_m^{CancelM} - X_{l_i,t}^{L^iT} \geq 0 \tag{6.11}$$

$$\forall t \neq t' \in \mathbb{T}, \quad \forall l \in \mathbb{L}^i \quad \text{s.t. } C_{t,t'}^{refT} = 1, \qquad X_{l,t}^{L^iT} + X_{l,t'}^{L^iT} \leq 1 \tag{6.12}$$

$$\forall m \neq m' \in \mathbb{M}, \forall p \neq p' \in \mathbb{P} \text{ s.t. } C_{m,m'}^{refM} = 1, Y_{p,p'}^{P} = 1, \qquad X_{p,m}^{PM} + X_{p',m'}^{PM} \leq 1 \tag{6.13}$$

$$\forall t \in \mathbb{T}, \quad \forall l \in \mathbb{L}, \qquad X_{l,t}^{L^iT} \in \{0,1\} \tag{6.14}$$

$$\forall m \in \mathbb{M}, \quad \forall p \in \mathbb{P}, \qquad X_{p,m}^{PM} \in \{0,1\} \tag{6.15}$$

$$\forall t \in \mathbb{T}, \qquad X_t^{CancelT} \in \{0,1\} \tag{6.16}$$

$$\forall m \in \mathbb{M}, \qquad X_m^{CancelM} \in \{0,1\} \tag{6.17}$$

Fig. 6.2 The routing lower level programming model

movement conflicts. As a result, if effective times come from tentative timetable (first loop of tri-level), the train cancellation is doubtless validated. If effective times come from revised timetable, train cancellation remains to be improved in tri-level mechanism and will be decided by the decision model.

6.1.2 Level 2 Routing upper level $max_x R_2(x, y, c)$: allocation of internal lines and paths for movements and trains cancelled

Routing lower level tries to identify insolvable conflicts of trains and commercial movements. Then we need to reinsert technical movements by the cooperation between routing and scheduling levels. The routing decision to enforce solvability of technical movements cancellation in scheduling process is prepared in routing upper level.

Cancellation decision solved in routing lower level is considered as given parameters in routing upper level. The allocation principle is same as used in decision model, so the routing constraints (3.11), (3.12) and (3.13) are reserved in the routing upper level. As the scheduling decisions and cancellation decisions are all given, the constraints (3.14) and (3.16) which guarantee compatibility of ressources are simplified as below:

Compatibility of lines. Two trains with time intersection $C^{effT}_{t,t'} = 1$ cannot occupy the same internal line. If one of the two trains is cancelled, the constraint is relaxed as $X^{L^iT}_{l,t} + X^{L^iT}_{l,t'} \leq 2$

$$\forall t,t' \in \mathbb{T}, t \neq t', \forall l \in \mathbb{L}^{Pref}_t \cap \mathbb{L}^{Pref}_{t'}, \text{ s.t. } C^{effT}_{t,t'} = 1,$$
$$X^{L^iT}_{l,t} + X^{L^iT}_{l,t'} \leq 1 + X^{CancelT}_t + X^{CancelT}_{t'} \tag{6.18}$$

Compatibility of switches. Two movements m and m' with time intersection $C^{effM}_{m,m'} = 1$ cannot pass the conflicting paths $Y^P_{p,p'} = 1$. If one of the two movements is cancelled, the constraint is relaxed as $X^{PM}_{p,m} + X^{PM}_{p',m'} \leq 2$.

$$\forall m,m' \in \mathbb{M}, m \neq m', \forall p,p' \in \mathbb{P}, p \neq p', \text{ s.t. } C^{effM}_{m,m'} = 1 \text{ and } Y^P_{p,p'} = 1,$$
$$X^{PM}_{p,m} + X^{PM}_{p',m'} \leq 1 + X^{CancelM}_m + X^{CancelM}_{m'} \tag{6.19}$$

Headway time H_t Originally, headway is the minimum time interval required between two trains using conflicting resources (paths or internal lines), for safety and signalling reasons. Here, we borrow the word "headway" to measure the spare time interval between the train t with a technical movement cancelled and another valid train t' on conflicting internal line, as shown in Figure 6.3.

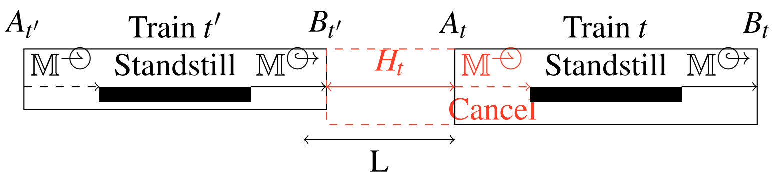


Fig. 6.3 Headway time for trains with technical entering movement cancelled

Within headway time, standstill can be extended without leading to conflicts on internal line. As a result, technical movements can be scheduled within headway time without considering of internal line conflicts, and the conflicts to be solved are on paths. Maximization of headway time enforces solvability of technical movements cancellation in scheduling process.

Headway time for technical movements cancelled is defined by equations (6.20)-(6.23). The technical movements to be reinserted involve trains not cancelled $X_t^{CancelT} = 0$ but containing technical movements cancelled. Headway time is generated in two cases depending on the technical movements direction (entering or leaving railway station).

- Firstly, if the first movement of the train t is cancelled $X_{m_1^t}^{CancelM} = 1$, we can say that the technical movement cancelled is entering railway station. In this case, headway time is evaluated by $A_t - B_{t'}$, as shown in Figure 6.3. In view of the flexible time interval for technical movement L, we consider only trains t' in accord with $A_t - L \leq B_{t'} < A_t$. If two trains are assigned with the same internal line, headway time needs to satisfy $H_t \leq A_t - B_{t'}$, as shown in equation (6.20).

$$\forall t,t' \in \mathbb{T}, t \neq t', \text{ s.t. } A_t - L \leq B_{t'} < A_t, X_t^{CancelT} = 0, X_{m_1^t}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref} \tag{6.20}$$

$$H_t \leq (A_t - B_{t'}) \cdot [1 + R \cdot (2 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT})]$$

If we have $B_{t'} = A_t$, headway is decribed in equation (6.21). If two trains are assigned with the same internal line, headway time need to satisfy $H_t \leq 0$.

$$\forall t,t' \in \mathbb{T}, t \neq t', \text{ s.t. } B_{t'} = A_t, X_t^{CancelT} = 0, X_{m_1^t}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref}$$

$$H_t \leq R \cdot (2 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT}) \tag{6.21}$$

- In the second case, if the last movement of the train t is cancelled $X_{m_{\mathbb{M}^t}^t}^{CancelM} = 1$, the technical movement cancelled is leaving railway station. Headway time is evaluated by $A_{t'} - B_t$, as shown in Figure 6.4.

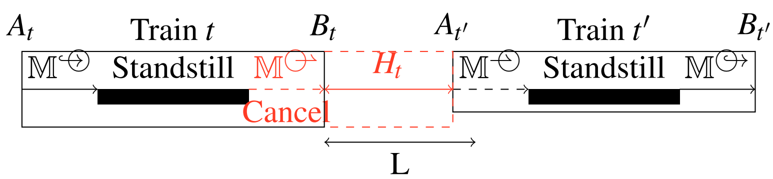


Fig. 6.4 Headway time for trains with technical leaving movement cancelled

In view of the flexible time interval for technical movement L, we consider only trains t' in accord with $B_t + L \geq A_{t'} > B_t$. If two trains are assigned with the same internal line, headway time need to satisfy $H_t \leq A_{t'} - B_t$.

$$\forall t, t' \in \mathbb{T}, t \neq t', \text{ s.t. } B_t + L \geq A_{t'} > B_t, X_t^{CancelT} = 0, X_{m^t_{\mathrm{M}^t}}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref}$$

$$H_t \leq (A_{t'} - B_t) \cdot [1 + R \cdot (2 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT})] \tag{6.22}$$

If we have $A_{t'} = B_t$, headway is decribed in equation (6.23). If two trains are assigned with the same internal line, headway time need to satisfy $H_t \leq 0$.

$$\forall t, t' \in \mathbb{T}, t \neq t', \text{ s.t. } A_{t'} = B_t, X_t^{CancelT} = 0, X_{m^t_{\mathrm{M}^t}}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref}$$

$$H_t \leq R \cdot (2 - X_{l,t}^{L^iT} - X_{l,t'}^{L^iT}) \tag{6.23}$$

Objective function. To enforce solvability of technical movement cancellations, we maximize the headway of all technical movements cancelled and the path tolerance of all paths allocated.

Path tolerance Tol_p is the number of trains which can pass in parallel through railway station network while one train is passing on the path p. Based on this idea, we evaluate every path by its tolerance index Tol_p. For example, tolerance of the path $[5,4,3,10,13,14]$ is 2 as shown in Figure 2.3, and tolerance of the path $[5,6,8]$ is 5. If a train is allocated with the path $[5,4,3,10,13,14]$, during the movement, only one another train is permitted to pass in parallel on path $[6,7,9]$ through the railway station.

$$R_2 : \text{Maximize} \sum_{t \in \mathbb{T}} H_t + \sum_{m \in \mathbb{M}, p \in \mathbb{P}} X_{p,m}^{PM} * Tol_p \tag{6.24}$$

The routing upper level programming model is defined by:

$$\text{Maximize} \sum_{t\in\mathbb{T}} H_t + \sum_{m\in\mathbb{M},p\in\mathbb{P}} X_{p,m}^{PM} \cdot Tol_p \tag{6.25}$$

$$\forall t \in \mathbb{T} \qquad \sum_{l_i\in\mathbb{L}_t^{Pref}} X_{l_i,t}^{L^iT} = 1 \tag{6.26}$$

$$\forall m \in \mathbb{M}^{l_e}, \quad \forall l_e \in \mathbb{L}^e \quad \text{s.t. } Y_{l_e,m}^{L^eM} = 1 \qquad \sum_{p\in\mathbb{P}^{l_e}} X_{p,m}^{PM} = 1 \tag{6.27}$$

$$\forall t \in \mathbb{T}, \quad \forall l_e \in \mathbb{L}^e, \forall m \in \mathbb{M}^{l_e} \cap \mathbb{M}^t, \quad \forall l_i \in \mathbb{L}_{l_e}^i, \quad \sum_{p\in\mathbb{P}^{(l_i,l_e)}} X_{p,m}^{PM} - X_{l_i,t}^{L^iT} \geq 0 \tag{6.28}$$

$$\forall t \neq t' \in \mathbb{T}, \quad \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref}, \quad \text{s.t. } C_{t,t'}^{effT} = 1, \tag{6.29}$$

$$X_{l,t}^{L^iT} + X_{l,t'}^{L^iT} \leq 1 + X_t^{CancelT} + X_{t'}^{CancelT}$$

$$\forall m \neq m' \in \mathbb{M}, \quad \forall p \neq p' \in \mathbb{P} \text{ s.t. } C_{m,m'}^{refM} = 1 \quad , Y_{p,p'}^{P} = 1, \tag{6.30}$$

$$X_{p,m}^{PM} + X_{p',m'}^{PM} \leq 1 + X_m^{CancelM} + X_{m'}^{CancelM}$$

$$\forall t,t' \in \mathbb{T}, t \neq t', \text{ s.t. } A_t - L \leq B_{t'} < A_t, X_t^{CancelT} = 0, X_{m_1^t}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref} \tag{6.31}$$

$$H_t + (A_t - B_{t'}) \cdot R \cdot (X_{l,t}^{L^iT} + X_{l,t'}^{L^iT}) \leq (A_t - B_{t'}) \cdot (1 + 2 \cdot R)$$

$$\forall t,t' \in \mathbb{T}, t \neq t', \quad \text{s.t. } B_{t'} = A_t, X_t^{CancelT} = 0, X_{m_1^t}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref}$$

$$H_t \cdot (X_{l,t}^{L^iT} + X_{l,t'}^{L^iT}) \leq 2 \cdot R \tag{6.32}$$

$$\forall t,t' \in \mathbb{T}, t \neq t', \text{ s.t. } B_t + L \geq A_{t'} > B_t, X_t^{CancelT} = 0, X_{m_{\mathsf{M}^t}^t}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref} \tag{6.33}$$

$$H_t + (A_{t'} - B_t) \cdot R \cdot (X_{l,t}^{L^iT} + X_{l,t'}^{L^iT}) \leq (A_{t'} - B_t) \cdot (1 + 2 \cdot R)$$

$$\forall t,t' \in \mathbb{T}, t \neq t', \quad \text{s.t. } A_{t'} = B_t, X_t^{CancelT} = 0, X_{m_{\mathsf{M}^t}^t}^{CancelM} = 1, \forall l \in \mathbb{L}_t^{Pref} \cap \mathbb{L}_{t'}^{Pref}$$

$$H_t + R \cdot (X_{l,t}^{L^iT} + X_{l,t'}^{L^iT}) \leq 2 \cdot R \tag{6.34}$$

$$\forall t \in \mathbb{T}, \quad \forall l \in \mathbb{L}, \quad X_{l,t}^{L^iT} \in \{0,1\} \tag{6.35}$$

$$\forall m \in \mathbb{M}, \quad \forall p \in \mathbb{P}, \quad X_{p,m}^{PM} \in \{0,1\} \tag{6.36}$$

Fig. 6.5 The routing upper level programming model

6.2 Level 3 Upper level programming model $min_{yc}S(x,y,c)$: rescheduling technical movements

In this thesis, the upper level of train platforming problem is to determine the effective times of technical movements, which extends also the train standstill on internal lines, and aims to minimize train cancellations. In train platforming process, trains scheduling sub-problem is influenced by the assignment of internal lines and paths. Routing variables $X_{l,t}^{L^iT}$ and $X_{p,m}^{PM}$ are solved by lower level and used as parameters in the upper level. Upper level programming model includes the time constraints (2.1)-(2.5), the objective function (3.34) is the same as the one used in the decision model and the compatibility constraints rewritten as below.

Compatibility of lines. If two trains in potential conflicts $C_{t,t'}^{refT} = 1$ are assigned to the same internal line $X_{l,t}^{L^iT} = X_{l,t'}^{L^iT} = 1$, a common time interval is not permitted for the two trains. If the term $1 - X_{t,t'}^{OrderT} + X_t^{CancelT} + X_{t'}^{CancelT} > 0$, the constraints are relaxed.

$$\forall t,t' \in \mathbb{T}, t \neq t', \forall l \in \mathbb{L}^i, \text{ s.t. } C_{t,t'}^{refT} = 1, X_{l,t}^{L^iT} = X_{l,t'}^{L^iT} = 1$$
$$B_t \leq A_t' + R \cdot (1 - X_{t,t'}^{OrderT} + X_t^{CancelT} + X_{t'}^{CancelT}) \tag{6.37}$$

Compatibility of switches. Similarly, if two movements in potential conflicts are assigned to two paths (p,p') which contain a common switch $Y_{p,p'}^P = 1$, a common time interval is not permitted for the two movements. If the term $1 - X_{m,m'}^{OrderM} + X_m^{CancelM} + X_{m'}^{CancelM} > 0$, the constraints are relaxed.

$$\forall m,m' \in \mathbb{M}, m \neq m', \forall p,p' \in \mathbb{P}, p \neq p', \text{ s.t. } C_{m,m'}^{refM} = 1 \text{ and } Y_{p,p'}^P = 1, X_{p,m}^{PM} = X_{p',m'}^{PM} = 1$$
$$\beta_m \leq \alpha_{m'} + R \cdot (1 - X_{m,m'}^{OrderM} + X_m^{CancelM} + X_{m'}^{CancelM}) \tag{6.38}$$

The upper level scheduling model is defined by:

$$\text{Minimize } \sum_{t\in\mathbb{T}} X_t^{CancelT} \tag{6.39}$$

$$\forall t \in \mathbb{T}, \forall m \in \mathbb{M}^t, \qquad X_m^{CancelM} = X_t^{CancelT} \tag{6.40}$$

$$
\begin{array}{llllrcll}
 & \forall m \in \mathbb{M} & & & \beta_m - \alpha_m & = & S & (6.41) \\
\forall t \in \mathbb{T} & & & & A_t - \alpha_{m_1^t} & = & 0 & (6.42) \\
\forall t \in \mathbb{T} & & & & B_t - \beta_{m_{\mathsf{M}^t}^t} & = & 0 & (6.43) \\
\forall t \in \mathbb{T} & & \forall i \in \{1, ..., \mathsf{M}^t - 1\} & & \beta_{m_i^t} - \alpha_{m_{i+1}^t} & <= & 0 & (6.44) \\
 & \forall m \in \mathbb{M}^{\ominus}, & & & \beta_m & \geq & \beta_m^{ref} - L & (6.45) \\
 & \forall m \in \mathbb{M}^{\ominus}, & & & \beta_m & \leq & \beta_m^{ref} & (6.46) \\
 & \forall m \in \mathbb{M}^{\ominus\rightarrow}, & & & \alpha_m & \geq & \alpha_m^{ref} & (6.47) \\
 & \forall m \in \mathbb{M}^{\ominus\rightarrow}, & & & \alpha_m & \leq & \alpha_m^{ref} + L & (6.48) \\
 & \forall m \in \mathbb{M}^{\rightarrow\ominus}, & & & \beta_m & = & \beta_m^{ref} & (6.49) \\
 & \forall m \in \mathbb{M}^{\ominus\rightarrow}, & & & \alpha_m & = & \alpha_m^{ref} & (6.50) \\
\forall t \neq t' \in \mathbb{T}, & \forall l \in \mathbb{L}^i & \text{s.t. } C_{t,t'}^{refT} = 1, & & X_{l,t}^{L^iT} = X_{l,t'}^{L^iT} = 1 & & & \\
 & & & & B_t - A_{t'} + R \cdot (X_{t,t'}^{OrderT} - X_t^{CancelT} - X_{t'}^{CancelT}) & \leq & R & (6.51) \\
\forall t \neq t' \in \mathbb{T}, & & & & X_{t,t'}^{OrderT} + X_{t',t}^{OrderT} & = & 1 & (6.52) \\
\forall m \neq m' \in \mathbb{M}, & \forall p \neq p' \in \mathbb{P} & \text{s.t. } C_{m,m'}^{refM} = 1, Y_{p,p'}^{P} = 1 & & , X_{p,m}^{PM} = X_{p',m'}^{PM} = 1 & & & \\
 & & & & \beta_m - \alpha_{m'} + R \cdot (X_{m,m'}^{OrderM} - X_m^{CancelM} - X_{m'}^{CancelM}) & \leq & R & (6.53) \\
 & \forall m \neq m' \in \mathbb{M}, & & & X_{m,m'}^{OrderM} + X_{m',m}^{OrderM} & = & 1 & (6.54) \\
\forall t \in \mathbb{T}, & & & & A_t & \in & [\![0, 1440]\!] & (6.55) \\
\forall t \in \mathbb{T}, & & & & B_t & \in & [\![0, 1440 + L]\!] & (6.56) \\
\forall t \in \mathbb{T}, & \forall m \in \mathbb{M}^t, & & & \alpha_m & \in & [\![0, 1440]\!] & (6.57) \\
\forall t \in \mathbb{T}, & \forall m \in \mathbb{M}^t, & & & \beta_m & \in & [\![0, 1440 + L]\!] & (6.58) \\
\forall t \neq t' \in \mathbb{T}, & & & & X_{t,t'}^{OrderT} & \in & \{0, 1\} & (6.59) \\
 & \forall m \neq m' \in \mathbb{M}, & & & X_{m,m'}^{OrderM} & \in & \{0, 1\} & (6.60) \\
\forall t \in \mathbb{T}, & & & & X_t^{CancelT} & \in & \{0, 1\} & (6.61) \\
 & \forall m \in \mathbb{M}, & & & X_m^{CancelM} & \in & \{0, 1\} & (6.62)
\end{array}
$$

Fig. 6.6 Scheduling programming model

6.3 Tri-level model operating mechanism

So far, the structure of tri-level model shown in Figure 6.1 and its three levels are presented above. Tri-level model is designed to provide an initial solution to the decision model. Decision model tries to solve the two sub-problems of platforming problem (scheduling and routing) at once. To reduce the calculation efforts, tri-level model plans to find a local

			Decision model	L1: Routing Lower level	L2: Routing Upper level	L3: Scheduling
Variables	Scheduling	Time (α_m, β_m, A_t, B_t)	Rational	Fixed	Fixed	Rational
		Order ($X_{t,t'}^{OrderT}$, $X_{m,m'}^{OrderM}$)	Bool	N/A	N/A	Bool
	Routing $X_{l,t}^{L^iT}$, $X_{p,m}^{PM}$		Bool	Rational	Bool	Fixed
	Cancellation $X_m^{CancelM}$, $X_t^{CancelT}$		Bool	Rational	Fixed	Bool
	Headway H_t		N/A	N/A	Rational	N/A
Constrains	Scheduling		Y	Y	Y	N
	Routing		Y	N	N	Y
Objective function			Minimize the number of trains cancelled	Minimize weighted trains' cancellation penalty	Maximize robustness	Minimize the number of trains cancelled

Table 6.1 Models comparison

optimum of platforming problem step by step. Variables and constraints of decision model and tri-level models (L1, L2, L3) are summarized in Table 6.1.

Routing variables include $X_{l,t}^{L^iT}$ and $X_{p,m}^{PM}$. Scheduling variables consist of α_m, β_m, A_t, B_t, $X_{t,t'}^{OrderT}$ and $X_{m,m'}^{OrderM}$. Cancellation variables are $X_m^{CancelM}$ and $X_t^{CancelT}$. Headway variable H_t is specially used in L2 to evaluate the robustness of routing solution.

As described in Section 3.4.1, scheduling sub-problem is proven as belonging to the network optimization problem family which can be solved by simplex methods in a rather short time. The scheduling solutions obtained are naturally integers. Inspired by this idea, we solve separately routing and scheduling sub-problems by tri-level decomposition method. We can see from Table 6.1 that all routing variables in L1 are defined as rational numbers, because the LP model in L1 is designed as a Network linear program. Specially, L1 only allocate internal lines and paths to trains invalided. If the train is cancelled, all its corresponding routing variables are assigned to 0.

As L3 only answers for scheduling sub-problem, L3 contains scheduling constraints and objective function. As L1 and L2 only answer for routing sub-problem, L1 and L2 contain routing constrains and objective function. Decision model includes objective function, scheduling and routing constraints.

The operating mechanism of tri-level model is shown in Figure 6.7. The local optimum is found after several loops among the three levels (L1, L2 and L3). In the first loop $i = 0$, departure and arrival time of trains come from tentative timetable with minimum standstill required by customer services. With minimum standstill and unchangeable commercial movements, L1 prioritizes allocation of paths to commercial movements and allocation of internal lines to trains' standstill. As a result, all trains cancelled in S_0^{L1} are due to the lack of internal lines accessible to paths suitable for their commercial movements. This kind of train cancellation is insolvable in next steps and need to be marked as *cancelled* until the end. Then, as the balance between scheduling and routing sub-problems, L2 takes the train cancellation result from S_i^{L1} and generates a robust solution. S_i^{L2} contains robust routing solution which makes train scheduling L1 flexible to absorb technical movements cancelled in L1. L3 minimizes train cancellations in the same way as Decision model by rescheduling technical movements and extending trains' standstill. The trains cancelled in S_i^{L3} are due to incompatible technical movements. At last, we verify whether S_i^{L3} is local optimum. If S_i^{L3} is local optimum, operating mechanism of tri-level ends and provides S_i^{L3} to Decision model as an initial solution. If not, the revised timetable with extended standstill is put out to L1, and $i = i + 1$.

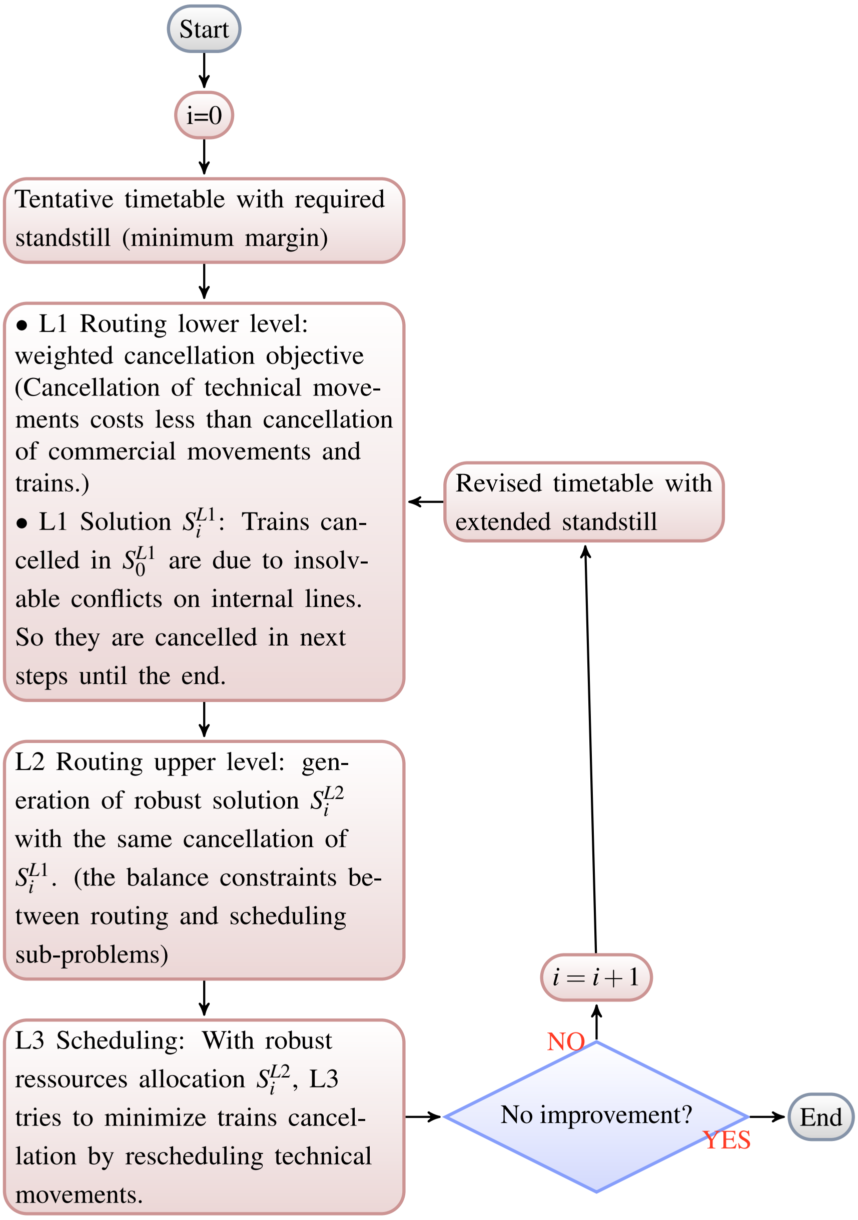


Fig. 6.7 Operating mechanism of tri-level model

6.4 Computational results

The complete algorithm is tested on real cases in St Jean Bordeaux railway station. The computational results are obtained by using CPLEX version 12.6 on a 64 bits computer under Linux with Intel i5-2520M CPU at 2.5GHz and 8GB memory RAM. For each group, the calculation time is limited to 500 seconds. Routing variables $X_{l,t}^{L^iT}$ are defined in the preference order of internal lines for the train t. The computational results are displayed in two ways:

- Decision model with initial solution given by tri-level model is tested on different sizes of problem: 5 trains, 10 trains, 15 trains, 20 trains, 25 trains and 30 trains. A benchmark is provided to evaluate the efficiency of this mechanism.
- The real full-day tentative timetable studied in Section 5.1 is solved by complete algorithm with tri-level model. The results are compared with the calculation process of complete algorithm without tri-level model.

6.4.1 Benchmark on different sizes of problems

Problems solved here are the same as ones used in Section 3.4.3. The flexible time interval permitted for technical movements is 10 minutes, $L = 10$. Limited by the real cases data we have, Decision model with initial solution given by tri-level model is tested on 117 problems of different sizes including 48 problems of 5 trains, 24 problems of 10 trains, 16 problems of 15 trains, 12 problems of 20 trains, 9 problems of 25 trains and 8 problems of 30 trains. 116 problems are solved to optimum trains cancellation with 0% gap. Only one problem of 30 trains is solved to proximity optimum with one train cancelled. CPLEX stops the calculation process of this instance before the solve time limit 500 seconds, and the solution status is given as “optimal”.

The whole numerical experiments consist of 6 different sizes problems. Then we analyse the results in the groups of the same size. In each group, the complexity of the problems is different. Each group is described by number of trains, number of technical movements, commercial movements and length of time interval covered by trains. Calculation performances of tri-level model and decision model are both displayed. We rank the problems in the order of total solve time which sums up the solve time of tri-level model and decision model. The problem solved in the minimum or the maximum solve time is presented in Table 6.2. The problem with the average solve time is to be constructed by the solve information of the whole group.

Tri-level model can generate an initial solution within short time. The maximal solve time is 3.4s, and the maximal average solve time is 0.34s. The solve time is stable and not sensitive to the growth of problem size. On the contrary, quality of initial cancellation solution is slightly decreased with the growth of size. The average of cancellation solution increase from 0.02 to 1.625 while the size grows from 5 trains to 30 trains. The maximal average cancellation 2.11 appears in groups of 25 trains.

With the initial solution provided by Tri-level model, Decision model performs well in improving solution and giving optimal solution. Gap between current solution and optimal solution is down to 0.00% in 116 problems out of 117 problems in total. Solution quality of the problem with gap 100% is affirmed as “optimal” by CPLEX without reach of the time limit of 500 seconds. So the optimality of solution can be guaranteed by Decision model. Solve time increases with the growth of problem size, but not in the exponential growth. It is related to the complexity and search space of problems.

	Trains	Movements		Time	Tri-level model		Decision model		
	per group	tech.	comm.	interval	Cancellation	Solve time	Cancellation	Gap	Solve time
Min	5	5	5	109	0	0,01	0	0.00%	0.01
Average	5	3	7	77,6	0.02	0,03	0	0.00%	0.04
Max	5	5	5	72	0	0.07	0	0.00%	0.89
Min	10	3	17	122	0	0,03	0	0.00%	0.03
Average	10	7	13	105,7	0.29	0,07	0	0.00%	0.11
Max	10	7	14	65	0	0.21	0	0.00%	0.42
Min	15	5	26	113	0	0.05	0	0.00%	0.07
Average	15	10	20	140,3	0.56	0.34	0	0.00%	0.77
Max	15	11	20	128	1	3.4	0	0.00%	7.19
Min	20	14	27	248	0	0,09	0	0.00%	0.17
Average	20	13	27	176.8	1.08	0.18	0	0.00%	4.41
Max	20	14	27	98	3	0.37	0	0.00%	45.95
Min	25	24	26	335	0	0.12	0	0.00%	0.33
Average	25	18	36	220,4	2.11	0.25	0,22	0.00%	3.92
Max	25	21	30	124	5	0.26	0	0.00%	21.23
Min	30	18	42	217	0	0.28	0	0.00%	0.29
Average	30	20	40	269,0	1.625	0.29	0.22	12.5%	24.79
Max	30	23	39	204	3	0.25	1	100.00%	171.56

Table 6.2 Decision model with initial solution given by tri-level model

The benchmark with 5 sizes of problem confirms the calculation efficiency of decision model with tri-level model in terms of solving small problems. However, our final objective is to generate a conflict-free full-day timetable in a complex and busy railway station. In the following section, we try to solve a real tentative timetable and evaluate the performance of complete algorithm with tri-level model.

6.4.2 Results on full-day timetable

In order to generate a full-day timetable, the calculation process follows the complete algorithm based on sliding window mechanism, described in Section 5. An accelerator, tri-level model, is added to decision model which aims to solve subgroups of trains.

To observe the performance of tri-level mechanism, solutions obtained with and without tri-level in step 3 are compared in Table 6.3. The one-day timetable is divided into 8 subgroups of trains with $N = 60$ shown in the first column. The second and third column contains the numbers of technical and commercial movements for the relevant subgroup. The trains subgroup can be divided into three groups: inherited group, fixed group and buffer group which are described, in the 5th and 6th columns, by the number of trains and the time interval occupied in minutes. The inherited group is generated as valid constraints. With $N = 60$, there is no intersection between the time interval of buffer group and the time interval of inherited group to avoid insolvable potential conflicts with inherited group. The minimum number of trains cancelled solved within 500 seconds and the resolution information are shown in the last two large columns in Table 6.3.

At last, there are 6 trains cancelled in the one-day timetable. The group of trains cancelled $\mathbb{T}^{Cancelled}$ is $[54, 57, 91, 92, 182, 188]$.

The detailed information of calculation process is summarized in Figure 6.8. Horizontal axis represents the cumulative solve time of the whole calculation process, and ordinate indicates the number of trains cancelled. We can see that the infeasible tentative one-day timetable with 247 trains is solved in 101 seconds with 6 trains cancelled. The feasibility of timetable solved is verified by simulation. The calculation process is divided into 8 segments according to the trains subgroup solved (including the inherited group which is also taken into account as valid constraints). Each segment begins with alternation of lower level (routing) and upper level (scheduling), and ends with solution of decision model. The final solution of every segment is marked in number under the polyline and displayed by the coloured bar. The colors of bar, same as used in Figure 5.8, helps to distinguish the trains cancelled in inherited group, in buffer group and in new group. All calculation steps are linked by black polyline in chronological sequence. Green polyline passes all upper level (scheduling) results. Red polyline passes all lower level (routing) results. The red and green polylines meet at the final result of each segment. Obviously, the lower level (red polyline) cancels less trains than the upper level (green polyline), because the train is not to be cancelled due to the technical movement cancellation in lower level. The first calculation of the lower level in each segment fixes the minimum trains cancellation due to the insolvable conflicts between trains or commercial movements. Then the cooperation between lower and upper level aims to absorb all technical movement cancellations. So the green line tends to go down. When the improvement of solution stops during two alternations, we say that the local optimum is found. Sometimes train cancellation is not reduced but increased, for example in (0-239t) and (0-246t). In this case, we take the best solution in the solution pool as the initial solution of Decision model. Based on the hybrid algorithm, we can also analyse the

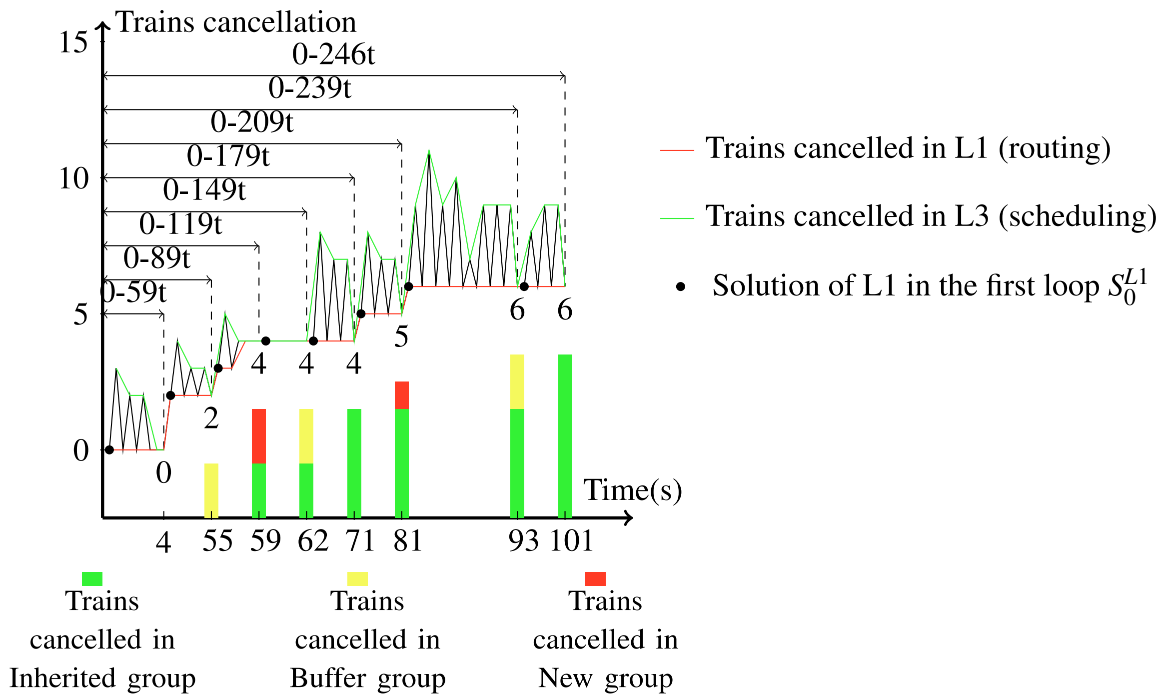


Fig. 6.8 Computational results

cause of train cancellations. In the first segment (0-59t), there is no trains cancellation. In the second segment (0-89t), two trains $[54,57]$ are cancelled due to the lack of internal lines, because the first routing solution is 2. In segment (0-119t), two trains $[54,57]$ cancelled are inherited from the previous segment, and one train $[91]$ is cancelled due to the lack of internal

Trains Subgroup	Movements		Group Type	Trains	Time Interval(min)	TiLim=500s with tri-level			TiLim=500s without tri-level		
	Tech.	Comm.				Obj	GAP	Solve time	Obj	GAP	Solve time
0-59	45	78	Inherited	-	-	0	0.00%	1.56	0	0.00%	50.23
			Buffer	0-29	00:30-06:46						
			New	30-59	05:22-08:50						
30-89	41	78	Inherited	0-29	00:30-06:46	2	0.00%	48.42	2	100.00%	500.05
			Buffer	30-59	05:22-08:50						
			New	60-89	07:18-11:02						
60-119	48	73	Inherited	0-59	00:30-08:50	4	0.00%	2.15	7	85.71%	500.07
			Buffer	60-89	07:18-11:02						
			New	90-119	09:35-13:15						
90-149	47	76	Inherited	0-89	00:30-11:02	4	0.00%	2.06	7	33.33%	500.02
			Buffer	90-119	09:35-13:15						
			New	120-149	12:15-16:07						
120-179	40	84	Inherited	0-119	00:30-13:15	4	0.00%	7.65	7	0.00%	165.09
			Buffer	120-149	12:15-16:07						
			New	150-179	14:52-18:53						
150-209	31	92	Inherited	0-149	00:30-16:07	5	0.00%	7.42	8	25.00%	500.04
			Buffer	150-179	14:52-18:53						
			New	180-209	16:52-20:08						
180-239	32	89	Inherited	0-179	00:30-18:53	6	0.00%	9.2	9	12.50%	275.81
			Buffer	180-209	16:52-20:08						
			New	210-239	19:03-22:41						
210-246	22	53	Inherited	0-209	00:30-20:08	6	0.00%	4.26	9	9.37%	6.10
			Buffer	210-239	19:03-22:41						
			New	240-246	20:15-24:54						

Table 6.3 Results comparison between cumulative sliding window algorithm with and without tri-level model

lines, and the other train [92] is cancelled due to the lack of paths for technical movements. In segments (0-149t), two trains cancelled $[54,57]$ are inherited from the previous segment, and two trains $[91,92]$ are cancelled due to the lack of internal lines. In (0-179t), there is no extra trains cancellation, and all four trains $[54,57,91,92]$ cancelled are inherited from the previous segment. In segment (0-209t), one train [182] is cancelled due to the lack of internal lines, and other four trains are inherited from the previous segment. In segment (0-239t), two trains $[182,188]$ are cancelled due to the lack of internal lines, and other four trains are inherited from the previous segment. In segment (0-246t), six trains cancelled are all inherited from the previous segment, and no extra train cancellation occurs. In a word, all six trains are cancelled due to the lack of internal lines. In step 3, the flexible interval used is $L = 32$ to reduce the calculation effort demanded by the decision model. Regarding the final solution, increase of L cannot help to reinsert the six trains cancelled due to the lack of internal lines, and only gives more feasibility to the trains cancelled due to the infeasible technical movement.

6.4.3 Representation of platforming results

In order to examine feasibility of platforming results, we express the final results by Gantt Chart. In addition, visualization of results is also a convenient way of implementing the resulted platform allocations. The platforming results are expressed in two levels: internal lines and switches.

In the internal lines Gantt Chart, the horizon axis is real-time in one day (24H), and the vertical axis is the list of internal lines in railway station. The train symbols are placed at the resulted actual time interval and the internal line allocated.

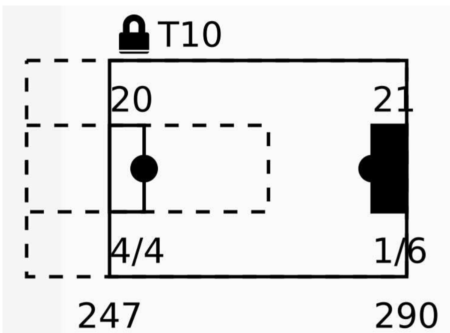


Fig. 6.9 Train symbol

Trains are represented by rectangles as shown in Figure 6.9. The train number $T10$ can be found on the top of the rectangle. Length of the horizon edge stands for the duration of standstill on internal lines and the last movement $[A_t, B_t + S[$ which are marked under the rectangle $[247, 290[$.

Expressed as small rectangles with a black point, the entering and leaving movements are stuck separately on two vertical edges. The movement numbers $m_1^t = 20$ and $m_2^t = 21$ are separately put on the top of small rectangles. The position of black points represents the direction of movements. If the point is stuck on the right vertical edge of small rectangle, the symbol represents a entering movement, as m_{20}. Otherwise the symbol represents a leaving movement, as m_{21}. The numbers under the small rectangles $4/4$ and $1/6$ indicate external lines and path chosen. So the movement m_{20} enters the railway station by the external line l_4^e and passes through the railway station on the path p_4. Filled small rectangles represent commercial movements. Hollow small rectangles are technical movements. The flexible time interval of technical movements is framed by dashed rectangles.

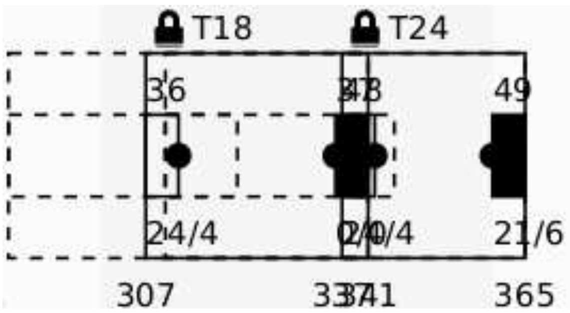


Fig. 6.10 Two trains on the same internal line

The feasibility of timetable is verified by the relative position of trains. **The lines Gantt Chart is designed in the same way of Lines Occupation graph (Graphe d'Occupation des Voies GOV), as described in Section 1.1, which is used to express planning decisions in railway stations.** As described in Section 2.4.5, if two trains allocated the same internal line in succession, the leaving movement of the early train and the entering movement of the late train are permitted to pass railway station in parallel. So the overlapping of two trains within 5 minutes is allowable. For example, two trains shown in Figure 6.10 are not conflicting with each other. Trains 18 and 24 are allocated on the same internal line. The leaving movement of T18 passes the railway station during $[336, 341[$, and the entering movement of T24 passes the railway station during $[337, 342[$. These two time intervals are overlapping during 4 minutes. However, only if the two movements are allocated to two conflict-free paths, this time intersections do not lead to conflicts on internal lines. When

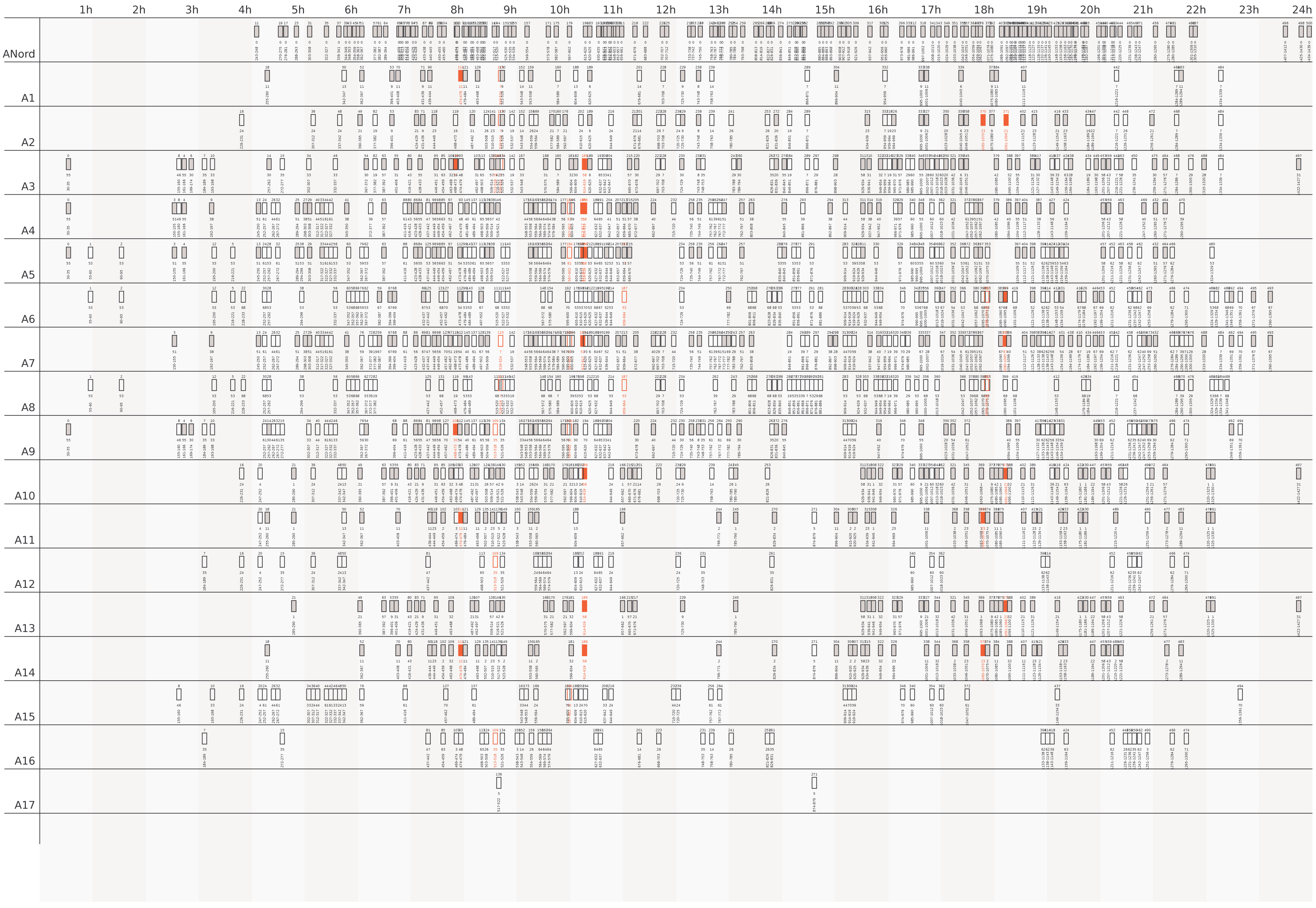
T24 arrives on the internal line, T18 has already left. In a word, only the intersection of two trains' standstills counts for conflicts on internal lines.

Fig. 6.11 Movement symbol

In the paths Gantt Chart, the horizon axis is real-time in one day (24H), and the vertical axis is the list of switches in railway station. **The movements planning is not taken into consideration by the practical timetabling process. The security within the local networks is ensured by dispatchers' operations. In our thesis, routing problem within the local networks is solved in details, and the allocation decision is expressed by the paths Gantt Chart.** The movement symbols are placed at the resulted actual time interval $[\alpha_m, \beta_m[$ and the switches belonging to the path allocated. For example, the movement m_{20}, which passes through the path $p_4 = [s_{11}, s_{10}, s_{12}, s_{15}]$, need to be placed on all the four switches' rows.

Movements are represented by rectangles as shown in Figure 6.11. The movement number 20 can be found on the top of the rectangle. The path allocated and the time interval scheduled to this movement are noted below the rectangle. The movement 20 passes through the path 4 within the time interval $[247, 252[$. Commercial movements are represented by gray rectangles, and technical movements are represented by white rectangles. Overlaps between movements are not permitted on switches except ANord.


1h
2h
3h
4h
5h
6h
7h
8h
9h
10h
11h
12h
13h
14h
15h
16h
17h
18h
19h
20h
21h
22h
23h
24h
V15
V1
V2
V3
V4
V5
V6
V7
V8
V9
V16
V11
V12
V17
V14


1h
2h
3h
4h
5h
6h
7h
8h
9h
10h
11h
12h
13h
14h
15h
16h
17h
18h
19h
20h
21h
22h
23h
24h
ANord
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
A15
A16
A17

6.5 Conclusion

In this chapter, we develop a tri-level decomposition method as an accelerator of decision model. As described in Figure 6.7, Level 1 takes the reference times as parameters and tries to solve the routing problem to minimal trains cancellation. Then, Level 2 takes the reference times and cancellation decision solved in Level 1 as parameters and tries to solve the routing problem to maximal robustness. Level 3 takes the routing decisions solved in Level 2 and tries to reschedule trains to minimal trains cancellation. Timetable corrected by Level 3 is reused by Level 1. So these three levels form a closed-loop. The cooperation among these three levels ends up without improvement of solution after several loops.

Operating mechanism of tri-level is summarized and tested on real case. Improvement of calculation efficiency is assessed by comparison of hybrid method with and without tri-level decomposition method. Computation time to generate “Feasible timetable” including 247 trains (more than 500 movements) reduces from 48 minutes to 101 seconds. Furthermore, all 8 sub-problems are solved to 0% GAP. Every train cancelled can be explained with a comprehensive reason, for example the lack of internal lines or infeasible technical movements.

Tri-level module is also implemented in Resolution module of the programming architecture described in Section 5.4. The process control automates the cooperation among these three levels and passes the initial solution to Decision model while the stop criterion is satisfied.

CHAPTER 7

Conclusion and Perspectives

Many countries have busy railway networks with highly complex patterns of train services. To meet the growing demand in rail traffic, railway networks are operated nearly at capacity margins. As bottleneck of networks, railway station demands eagerly a computer-based system to exploit its rail capacity. The management of rail traffic in stations requires careful scheduling to fit to the existing infrastructure, while avoiding conflicts between large numbers of trains, while satisfying safety or business policy and objectives.

This thesis combines operations research techniques with scientific and professional railway expertise in order to study the train platforming problem and to address the development of an advanced computerized dispatching support tool. The resulting timetable must ensure that no pair of trains is conflicting over paths and platforms, while allowing the coupling and uncoupling of trains at a platform and respecting their preferences of platforms and the accessibility of complete path of trains.

In the last years, this research area experienced an increasing interest due to the growth of train traffic and the limited possibilities of enhancing the infrastructure, which increase the demands for an efficient allocation of resources and the pressure on traffic dispatchers.

However, the train dispatching process is still dominated by human professional skills and experiences. Furthermore, the state-of-the-art in optimal train dispatching algorithms can handle only low traffic densities and a short time horizon within a reasonable amount of computation time.

We design and implement a decision support system for railway station dispatchers to generate a full-day conflict-free timetable which consists of two sets of circulations. The first set is made of commercial circulations given by several administrative levels (national, regional, freight) over a large time horizon (typically one year before the effective realization of the production). In France, commercial circulations are generated by RFF. Commercial circulations need to be strictly punctual. The other set corresponds to technical circulations added by the railway station managers to prepare or repair trains. The destination or origin of technical circulations is depot. A deviation interval L is permitted for the technical circulations depending on the direction and the reference time defined by railway station manager. This system is able to verify the feasibility of tentative timetable which is generated by RFF. Furthermore, commercial movements with unsolvable conflicts will return to their original activity managers with suggestions for the modification of the arrival and leaving times.

Considering accurately the switches and platforms occupation, cancellation of trains is operated by Decision model to guarantee the feasibility, and this version of timetable is called “Feasible timetable” including minimum trains cancellation. But the cancellation of trains is not the first choice for dispatchers. Better arrangement of all trains is achieved in “Revised timetable” by Reinsertion model and Refinement model. Trains cancelled in Decision model are reinserted into the “Feasible timetable”. To absorb this disturbance (reinsertion), an upper bound of time deviation F is given to commercial movements, and trains with conflict degree 1 with trains cancelled are relaxed to be rescheduled and rerouted. Reinsertion model guarantees the feasibility of timetable while absorbing trains cancellation with relaxed constraints. However, time deviation of commercial movements may cause delay propagation in railway networks. When trains reinsertion is completed, quality of “Revised timetable” is improved by Refinement model with minimum time deviations of all commercial movements. Both of “Feasible timetable” and “Revised timetable” are feedback to RFF. After several communications between RFF and railway stations, we could obtain finally an off-line timetable with high quality which ensure the smooth traffic within and between railway stations.

As the generation of full-day timetable in railway station is a large-scale problem, various methods are applied to reduce calculation efforts.

- The basic method is to establish an efficient formalization of the topology which requires less variables to describe the problem.
- Abundant constraints are detected and cut off during the design of model. In our thesis, resources compatibility constraints concentrate on potential conflicting trains and movements.
- Trains in full-day timetable are divided into several trains sub-groups in chronological sequence. Sliding window algorithm solves the large-scale problem step by step and enables the computation of effective dispatching solutions in an acceptable computation time.
- An initial solution provided by tri-level model accelerates greatly the calculation process led by branch-and-bound algorithm in Decision model.
- Tri-level optimization model is developed to provide global optimal approximate solutions in a rather short computation time. The whole problem is divided into three levels: 1) L1: scheduling sub-problem with minimal cancellation; 2) L2: routing sub-problem with minimal weighted cancellation; 3) L3: routing sub-problem with maximal robustness.

Extensive computational experiments are carried out on the complex dispatching areas of railway station Bordeaux St Jean. A benchmark including 117 problems of different sizes and complexities is provided to evaluate performance of Decision model with tri-level model as an accelerator. The complete algorithm is tested on a real full-day tentative timetable in railway station Bordeaux St Jean. Calculation performances show efficiency and effectiveness of our computerized dispatching support tool. Feasibility of resulting timetables is verified by two Gantt Chart separately corresponding to internal lines and switches. More numerical experiments may be carried on in different railway stations. In our algorithm, we do not apply any solution searching rule specially for station Bordeaux St Jean in calculation process. If topology of railway station and trains' activities are imported in the form designed, the algorithm can be easily applied on other railway stations. Professional railway expertise is integrated in internal lines preference lists and path lists of internal lines.

Through this thesis, we attempted to solve the off-line train platforming problem. Evidently, this thesis represents a step in this research avenue and works on the subject which can be pursued by solving on-line platforming problem. The algorithm proposed can be tested and improved in several aspects. The partitioning process of sub-groups in sliding window algorithm need to be studied with different N or dynamic subgroup sizes. The subgroups partitioning strategies are summarized in Section 5.3.2. In addition, the efficiency of the complete algorithm need to be tested on more real cases even in different railway stations. The effectiveness of our method can be evaluated by the comparison with the practical timetable. The robustness of the timetable, which is capable of neutralizing small deviations and stabilizing delay propagation, needs to be taken into account in future works. At last, in order to finish the development of this platforming tool, a user-friendly interface is required to communicate the results and data with dispatchers and railway station managers I look forward to discover these future researches development, which I hope not only to observe but in some way to participate in, too.

References

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

Adenso-Diaza, B., Gonzalez, M.O., and Gonzalez-Torre., P. (1999). On-line timetable re-scheduling in regional train services. *Transportation Research Part B*, 33, 383–397.

Arenas, D., Chevirer, R., Hanafi, S., and Rodriguez, J. (2014). Solving the periodic timetabling problem using a genetic algorithm. *CoRR*, abs/1411.6998.

Bai, L., Bourdeaud'huy, T., Castelain, E., and Babenasolo, B. (2013). Recherche de plans d'occupation de voies dans les réseaux ferroviaires. *14e conférence ROADEF de la société Française de Recherche Opérationnelle et Aide à la Décision*, February 13-15.

Bai, L., Bourdeaud'huy, T., Castelain, E., and Babenasolo, B. (2014). A mixed-integer linear program for routing and scheduling trains through a railway station. *3rd ICORES International Conference on Operations Research and Enterprise Systems, ICORES 2014, Angers, France, 6-8 March 2014*.

Bai, L., Bourdeaud'huy, T., Castelain, E., and Babenasolo, B. (2015). Incremental timetable generation in busy and complex railway stations: sliding-window algorithm with cancellation processing. *2015 IFAC Symposium on Information Control in Manufacturing, INCOM 2015, Ottawa, Canada, May 11-13, 2015*.

Balas, E. (1969). Machine sequencing via disjunctive graphs: an implicit enumeration approach. *Operations Research*, 17 (6), 941–957.

Barber, F., Tormos, P., Lova, A., Ingolotti, L., Salido, M., and Abril, M. (2006). A decision support system for railway timetabling (mom): the spanish case. *Computers in Railways*, X, 235–244.

Barber, F., Abril, M., Ángel Salido, M., Ingolotti, L.P., Tormos, P., and Lova, A. (2007). *Survey of automated Systems for Railway Management*. UNIVERSIDAD POLITÉCNICA DE VALENCIA.

Billionnet, A. (2003). Using integer programming to solve the train platforming problem. *Transportation Science*, 37(2), 213–222.

Caprara, A. (2010). Almost 20 years of combinatorial optimization for railway planning: from lagrangian relaxation to column generation. *In: Thomas Erlebach and Marco Lbbecke (Eds), Proceedings of the 10th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization and Systems(ATMOS). Schloss Dagstuhl. Germany, ATMOS 2010*, 1, 1–12.

Caprara, A. and Galli, L. (2011a). Solution of the train platforming problem. *Transportation Science*, 45(2), 246–257.

Cardillo, D.D.L. (1998). k l-list t colouring of graphs. *European Journal of Operational Research*, 106(1), 160–164.

Carey, M. (1994a). A model and strategy for train pathing with choice of lines, platforms and routes. *Transportation Research B*, 28 (5), 333–353.

Carey, M. (1994b). Extending a train pathing model from one-way to two-way track. *Transportation Research B*, 28 (5), 395–400.

Carey, M. and Carville, S. (2003). Scheduling and platforming trains at busy complex stations. *Transportation Research A*, 37 (3), 195–224.

Chang, S. and Chung, Y. (2005). From timetabling to train regulation – a new train operation model. *Information and Software Technology*, 47 (9), 575–585.

Corman, F., D'Ariano, A., Pacciarelli, D., and Pranzo, M. (2010). A tabu search algorithm for rerouting trains during rail operations. *Transportation Research Part B*, 44, 175–192.

D'Ariano, A. (2008). *Improving real-time train despatching: model, algorithms and applications*. PhD thesis, TU Delft.

Dündar, S. and Sahin, I. (2013). Train re-scheduling with genetic algorithms and artificial neural networks for single-track railways. *Transportation Research Part C journal*, 27, 1–15.

D'Ariano, A. and Albrecht, T. (2006a). Running time re-optimization during real-time timetable perturbations. *Allan J, Brebbia CA, Rumsey AF, Sciutto G, Sone S, Goodman CJ (eds) Computers in railways X.WIT, Southampton,*, 531–540.

D'Ariano, A., Corman, F., Pacciarelli, D., and Pranzo, M. (2006b). Real-time train conflict detection and resolution: global sequencing and local rerouting. *van Zuylen HJ (ed) Proceedings 9th TRAIL congress, selected papers. Delft University Press*, 77–92.

D'Ariano, A., Pacciarelli, D., and Pranzo, M. (2007a). A branch and bound algorithm for scheduling trains in a railway network. *Eur J Oper Res*, 183(2), 643–657.

D'Ariano, A. and Pranzo, M. (2009). An advanced real-time train dispatching system for minimizing the propagation of delays in a dispatching area under severe disturbances,. *Netw Spat Econ*, 9, 63–84.

D'Ariano, A., Pranzo, M., and Hansen, I. (2007b). Conflict resolution and train speed co-ordination for solving real-time timetable perturbations. *IEEE Trans Intell Transp Syst*, 8(2), 208–222.

D'Ariano, A., Pranzo, M., and Hansen, I. (2008). Reordering and local rerouting strategies to manage train traffic in real time. *Trainsportation Science*, 42(4), 405–419.

Egmond, R. (1999). Railway capacity assessment, an algebraic approach, trail studies in transportation science. *Delf University Press*, S99/2,.

Floudas, C. and Lin, X. (2004). Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. *Computers and Chemical Engineering*, 28, 2109–2129.

Forsgren, M. (2003). *Computation of Capacity on Railway Networks, Technical Report T2003-12*. Swedish Institute of Computer Science.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5), 533–549.

Goldberg, D. (1989). Genetic algorithms in search, optimization, and machine learning. *Addison Wesley, Longman, USA.*

Gupta, J. (1988). Two-stage, hybrid flow shop scheduling problem. *Journal of the Operational Research Society*, 39(4), 359–364.

Guéret, C., Prins, C., and Sevaux, M. (2000). *Programmation linéaire*. Edition Eyrolles ISBN 2-212-09202-4.

Hall, N. and Sriskandarajah, C. (1996). A survey on machine scheduling problems with blocking and no-wait in process. *Operations Research*, 44(3), 510–525.

Heller, I. and Tompkins, C. (1956). An extension of a theorem of dantzig's. In *Linear inequalities and related systems*, 247–254. Princeton Univ. Press, Princeton, NJ.

Higgins, A., Kozan, E., and Ferreira, L. (1996). Optimal scheduling of trains on a single line track. *Transportation Research Part B*, 30, 147–161.

Higgins, A., Kozan, E., and Ferreira, L. (1997). Heuristic techniques for single line train scheduling. *Journal of Heuristics*, 3 (1), 43–62.

Hoffman, A. and Kruskal, J. (1956). Integral boundary points of convex polyhedra. In *Linear inequalities and related systems*, 223–246. Princeton Univ. Press, Princeton, NJ.

Holland, J. (1975). Adaptation in natural and artificial systems. *University of Michigan Press, Ann Arbor, Michigan.*

Jacobs, J. (2004). Reducing delays by means of computer-aided on-the-spot rescheduling. *Allan J, Brebbia CA, Hill RJ, Sciutto G, Sone S (eds) Computers in railways IX. WIT, Southampton.*, 603–612.

Jain, A.S. and Meeran, S. (1991). A goal programming network for mixed integer linear programming: A case study for the job-shop scheduling problem. *International Journal of Neural Networks*, 2(3), 201–209.

Jain, A.S. and Meeran, S. (1998). A state-of-art review of job-shop scheduling techniques. *Department of Applied Physics, Electronic and Mechanical Engineering University of Dundee, Dundee, Scotland, UK, DD1 4HN.*

Kraft, E. (1982). Jam capacity of single track rail lines. *Proceedings of the Tansportation Research Forum*, 23 (1), 461–471.

Kroon, L., Romeijn, H., and Zwaneveld, P. (1997). Routing trains through railway stations: Complexity issues. *European Journal of Operational Research*, 98, 485–498.

Lucchini and Curchod (2001). Transalpine rail network: A capacity assessment model (capres). *First Swiss Transport Research Conference.*

Lüthi, M., Medeossi, G., and Nash, A. (2007). Evaluation of an integrated real-time rescheduling and train control system for heavily used areas.. *In: Hansen, I.A., Radtke, A., Pachl, J., Wendler, E. (Eds.), Proceedings of the 2nd International Seminar on Railway Operations Modelling and Analysis, Hannover, Germany.*

Manne, A. (1960). On the job-shop scheduling problem. *Operations Research*, 8, 219–223.

Mastrolilli, M. and Gambardella, L. (2000). Effective neighborhood functions for the flexible job shop problem. *Journal of Scheduling*, 3(1), 3–20.

Norio1, T. and Yoshiaki, T. (2005). Train rescheduling algorithm which minimizes passengers' dissatisfaction. *Transactions of Information Processing Society of Japan.*, 46(2), 26–38.

Pellegrini, P., Marlière, G., and Rodriguez, J. (2014). Optimal train routing and scheduling for managing traffic perturbations in complex junctions. *Transportation Research Part B*, 59, 58–80.

Petersen, E. (1974). Over-the-road transit time for a single track railway. *Transportation Science*, 8, 65–74.

Ping, L., Limin, J., and Xiong, K. (2001). Study on ga based train dispatching. *Proceedings of the 5th World Congress on Railway Research, Köln*, 949–953.

Rodriguez, J. (2007). A constraint programming model for real-time train scheduling at junctions. *Transportation Research Part B*, 41(2), 231–245.

Sahin, I. (1999). Railway traffic control and train scheduling based on inter-train conflict management. *Transportation Research Part B*, 33, 511–534.

Salim, V. and Cai, X. (1995). Scheduling cargo trains using genetic algorithms. *Proceedings of the 1995 IEEE International Conference on Evolutionary Computing*, 224–227.

Schrijver, A. and Steenbeek, A. (1994). Dienstregelingontwikkeling voor railned (timetable construction for railned). *Technical report, CWI, Center for Mathematics and Computer Science, Amsterdam. In Dutch.*

SNCF and Eurodecision (2004). A tool for the optimisation and the capacity assessment for railway infrastructure. *Demiurge*.

Tormos, P., Loval, A., Barber, F., Ingolotti, L., M.Abril, and Salido, M. (2008). A genetic algorithm for railway scheduling problems. *Xhafa, F., Abraham, A. (Eds.), Studies in Computational Intelligence (SCI), Springer-Verlag, Berlin*, 128, 255–276.

Tornquist, J. and Persson, J. (2005). rain traffic deviation handling using tabu search and simulated annealing. *Proceedings of the 38th Hawaii International Conference on System Sciences.*

Watson, R. (2001). The effects of railway privatisation on train planning. *Transport Reviews*, 21(2), 181–193.

Wegele, S. and Schneider, E. (2005). Dispatching of train operations using genetic algorithms. *Proceedings of the 1st International Seminar on Railway Operations Modelling and Analysis, Delft, 8–10 June 2005*.

Zwaneveld, P., Kroon, L., and Ambergen, H. (1996). A decision support system for routing trains through railway stations. In *Proceedings of Comp Rail*, volume 96, 217–226. Computational Mechanics Publications, Southampton, Berlin.

Zwaneveld, P., Kroon, L., and van Hoesel, S. (2001). Routing trains through a railway station based on a node packing model. *European Journal of Operational Research*, 128, 14–33.

List of symbols

Roman Symbols

$[{\alpha_m}^{Early}, {\beta_m}^{Late}[$ Potential scheduling time interval for movement m

$[\alpha_m, \beta_m[$ Occupation of path for movement m passing through railway station.

$[A_i^{buff}, B_i^{buff})$ Potential scheduling time interval of buffer group in the i^{th} trains subgroup.

$[A_i^{inh}, B_i^{inh})$ Potential scheduling time interval of inherited group in the i^{th} trains subgroup.

$[A_i^{new}, B_i^{new})$ Potential scheduling time interval of new group in the i^{th} trains subgroup.

$[A_i^{sub}, B_i^{sub})$ Potential scheduling time interval of the i^{th} trains subgroup.

$[{A_t}^{Early}, {B_t}^{Late}[$ Potential scheduling time interval for train t

$[A_t, B_t + S[$ All activities of train t occur within this time interval

$[A_t, B_t[$ Occupation of internal lines for train t.

${\alpha_m}^{Early}$ The earliest starting time of movement m.

α_m Actual starting time of the movement m.

$\alpha_m^{revised}$ Starting time of the movement m revised by reinsertion model.

α_m^{ref} Reference starting time of the movement m.

${\beta_m}^{Late}$ The latest arrival time of movement m.

β_m Actual ending time of the movement m, $\alpha_m + S = \beta_m$.

$\beta_m^{revised}$ Ending time of movement m revised by reinsertion model.

β_m^{ref}	Reference ending time of the movement m.
λ_t	Allocation of internal line to train t.
$\mathbb{L}$	A set of lines.
$\mathbb{L}^e$	The set of external lines.
$\mathbb{L}^i$	The set of internal lines.
$\mathbb{L}_{L_t}^{Comp}$	Compatible list of internal lines for train t with length L_t.
$\mathbb{L}_{Long}^{Comp}$	Compatible list of internal lines for long train.
$\mathbb{L}_{Medium}^{Comp}$	Compatible list of internal lines for medium train.
$\mathbb{L}_{Short}^{Comp}$	Compatible list of internal lines for short train.
$\mathbb{L}_t^{PrefD}$	Direction preference list of internal lines for train t.
$\mathbb{L}_t^{Pref}$	Preference list of internal lines for train t.
$\mathbb{L}_{D_m}^{Pref}$	Direction preference list of internal lines for movement m in the direction D_m.
$\mathbb{L}_{l_e}^i$	The subset of internal lines l_i reachable from an external line $l_e \in \mathbb{L}^e$.
$\mathbb{M}^{\rightharpoondown\!\bigcirc}$	The set of entering technical movements.
$\mathbb{M}^{\bigcirc\!\rightharpoondown}$	The set of leaving technical movements.
$\mathbb{M}^{\hookrightarrow\!\bigcirc}$	The set of entering commercial movements.
$\mathbb{M}^{\bigcirc\!\hookrightarrow}$	The set of leaving commercial movements.
$\mathbb{M}^t$	A ordered movements set of train t
$\mathbb{N}$	The set of natural numbers.
$\mathbb{P}$	A set of paths.
$\mathbb{P}^{(l_i,l_e)}$	The subset of paths that connect the internal line $l_i \in \mathbb{L}^i$ and the external line $l_e \in \mathbb{L}^e$.
$\mathbb{R}$	A railway station.
$\mathbb{S}$	A set of switches.

$\mathbb{S}^p$ A ordered switches set of path p.

$\mathbb{T}$ The set of all trains considered.

$\mathbb{T}_i^{buff}$ Buffer trains group in the i^{th} subgroup.

$\mathbb{T}^{Cancelled}$ Set of trains cancelled by Decision model.

$\mathbb{T}_t^{Conflict1}$ Group of trains in conflict degree 1 with the train t.

$\mathbb{T}_{t,t'}^{Conflict2}$ Group of trains in conflict degree 2 with the train t corresponding to train t'.

$\mathbb{T}^{Fixed}$ Group of trains without commercial time deviations $C_t^{dev} = 0$.

$\mathbb{T}_i^{inh}$ Inherited trains group in the i^{th} subgroup.

$\mathbb{T}^{Long}$ Set of long trains.

$\mathbb{T}_i^{new}$ New trains group in the i^{th} subgroup.

$\mathbb{T}^{Relaxed}$ Group of trains with commercial time deviations $C_t^{dev} = 1$.

$\mathbb{T}_i^{sub}$ The i^{th} trains subgroup.

L The cardinal of the set of lines $\mathbb{L}$.

M^{t} The cardinal number of set $\mathbb{M}^t$.

P The cardinal number of paths set $\mathbb{P}$

S The cardinal of the switches set $\mathbb{S}$.

S^p Number of switches included in the path p.

T The cardinal number of trains set $\mathbb{T}$.

$\mu^{long+medium+short}$ The occupancy rate of all internal lines.

$\mu^{long+medium}$ The occupancy rate of long and medium length internal lines.

μ^{Long} The occupancy rate of long internal lines.

μ^{Path} The occupancy rate of path capacity.

$\zeta(l)$ *"Entrance"* switch of line l

${A_t}^{Early}$ The earliest entering time of train t

A_t Starting time of occupation of the internal lines by the train t.

${B_t}^{Late}$ The latest leaving time of train t.

B_t Ending time of occupation of the internal lines by the train t.

c Vector of cancellation variables.

c Vector of cancellation variables.

C_t^{dev} Probes of trains containing commercial movements with time deviations. If $\exists m \in \mathbb{M}^t \cap \mathbb{M}^{Commercial}, \alpha_m^{ref} \neq \alpha_m^{revised}$, $C_t^{dev} = 1$. Otherwise 0.

$C_{m,m'}^{effM}$ Identification of conflicts between two movements m and m' with the given effective times. If $[\alpha_m, \beta_m) \cap [\alpha_{m'}, \beta_{m'}) \neq \varnothing$, $C_{m,m'}^{effM} = 1$. Otherwise $C_{m,m'}^{effM} = 0$

$C_{m,m'}^{refM}$ Probes of the potential conflicts between two movements m and m'.
If $[{\alpha_m}^{Early}, {\beta_m}^{Late}) \cap [{\alpha_{m'}}^{Early}, {\beta_{m'}}^{Late}) \neq \varnothing$,$C_{m,m'}^{refM} = 1$. Otherwise, $C_{m,m'}^{refM} = 0$.

$C_{t,t'}^{effT}$ Identification of conflicts between two trains t and t' with the given effective times. If $[A_t, B_t) \cap [A_{t'}, B_{t'}) \neq \varnothing$, $C_{t,t'}^{effT} = 1$. Otherwise $C_{t,t'}^{effT} = 0$.

$C_{t,t'}^{refT}$ Probes of the potential conflicts between two trains t and t'. If $[{A_t}^{Early}, {B_t}^{Late}) \cap [{A_{t'}}^{Early}, {B_{t'}}^{Late}) \neq \varnothing$, $C_{t,t'}^{refT} = 1$. Otherwise $C_{t,t'}^{refT} = 0$.

D_m Direction of movement m.

F Time deviation upper bound for commercial movements.

H_t the spare time interval between the train t with a technical movement cancelled and another valid train on conflicting internal line.

L The adjustable time interval for technical movements.

l A line in lines set $\mathbb{L}$.

L^i Number of internal lines.

L^{opt} Optimal flexible time interval for technical movements based on the internal line capacity.

l_e The e^{th} external line in set $\mathbb{L}^e$.

l_f The f^{th} line in set $\mathbb{L}$.

l_i The i^{th} internal line in set $\mathbb{L}^i$

l_m^e External line reserved by movement m.

L_{D_m} Cardinal number of set $\mathbb{L}_{D_m}^{Pref}$.

$l_{e'}$ Another external line different from l_e.

$max_x R_2(x,y,c)$ The routing upper level model involves all three variable vectors (x,y,c) to maximize robustness of timetable, and the subscript vector (x) is treated as variable.

$min_c R_1(x,y,c)$ The routing lower level model involves all three variable vectors (x,y,c) to minimize the weighted cancellation objective, and the subscript vector (c) is treated as variable.

$min_{xyc} D(x,y,c)$ The decision model involves all three variable vectors (x,y,c) to minimize train cancellations.

$min_{yc} S(x,y,c)$ The upper level (scheduling) model involves all three variable vectors (x,y,c) to minimize the train cancellations, and the subscript vectors (y,c) are treated as variables.

N Sliding window width: number of trains in each subgroup.

n_1 Number of inbound paths in railway station.

n_2 Number of outbound paths in railway station.

P The cancellation penalty used in weighted objective of routing upper level model.

p A path in set $\mathbb{P}$ consists of a set of ordered switches

p_c The c^{th} path in set $\mathbb{P}$

p_m Path allocated to movement m.

R A sufficiently big constant.

S Time duration used for a movement passing through railway station.

S^0 A full-day conflict-free timetable with minimum train cancellation obtained by Decision model.

S^1 The full-day conflict-free timetable relaxed without train cancellation solved by Reinsertion model.

S^2 The full-day conflict-free timetable with minimum deviation for commercial movements without train cancellation solved by Refinement model

S_i^{L1} Resulting timetable of L1 in the i^{th} loop.

S_i^{L2} Resulting timetable of L2 in the i^{th} loop.

S_i^{L3} Resulting timetable of L3 in the i^{th} loop.

s_1^p Internal switch of path p connecting with internal line.

$s_{\mathbb{S}^p}^p$ External switch of path p connecting with external line.

s_k The k^{th} switch in set $\mathbb{S}$.

s_k^p The k^{th} switch of path p

t A train in set $\mathbb{T}$.

t' Another train different from t.

t_i The i^{th} train in set $\mathbb{T}$.

Tol_p Tolerance index of path p:the number of trains which can pass in parallel through railway station network while one train passing on the path p.

VP_{m,l_i} Allocation priority of internal line l_i for movement m.

x Vector of routing variables.

$X_{l,t}^{L^iTRevised}$ Allocation of internal lines to train t by the reinsertion model.

$X_{l,t}^{L^iT}$ Allocation of internal lines to train t. If $\lambda_t = l$, $X_{l,t}^{L^iT} = 1$. Otherwise, $X_{l,t}^{L^iT} = 0$.

$X_{l_i,t}^{L^iTref}$ Internal line allocation decision for train t in Feasible timetable. If the train t is allocated to the internal line l_i, $X_{l_i,t}^{L^iTref} = 1$. Otherwise $X_{l_i,t}^{L^iTref} = 0$.

$X_{m,m'}^{OrderM}$ Time order of two movements using two conflicting paths. If m circulates before m', $X_{m,m'}^{OrderM} = 1$. Otherwise, $X_{m,m'}^{OrderM} = 0$.

$X_m^{CancelM}$ Cancellation of the movement m. If m is cancelled, $X_m^{CancelM} = 1$. Otherwise, $X_m^{CancelM} = 0$.

$X_{p,m}^{PMref}$ Path allocation decision for movement m in Feasible timetable.

$X_{p,m}^{PMRevised}$ Allocation of paths to movement m by the reinsertion model. If the path p is allocated to the movement m by the reinsertion model, $X_{p,m}^{PMRevised} = 1$.

$X_{p,m}^{PM}$ Allocation of paths to movement m. If $p = p_m$, $X_{p,m}^{PM} = 1$. Otherwise, $X_{p,m}^{PM} = 0$.

$X_{t,t'}^{OrderT}$ Time order of two trains using the same line. If t circulates before t', $X_{t,t'}^{OrderT} = 1$. Otherwise, $X_{t,t'}^{OrderT} = 0$.

$X_{t}^{CancelT}$ Cancellation of the train t. If t is cancelled, $X_{t}^{CancelT} = 1$. Otherwise, $X_{t}^{CancelT} = 0$.

y Vector of scheduling variables.

$Y_{p,p'}^{P}$ Identification of the conflicting paths. If $p \cap p' \neq \varnothing$, $Y_{p,p'}^{P} = 1$. Otherwise, $Y_{p,p'}^{P} = 0$.

L1 Lower level: Routing Lower level.

L2 Lower level: Routing Upper level.

L3 Upper level: Scheduling.

Titre: Ordonnancement des trains dans une gare complexe et à forte densité de circulation.

Cette thèse porte sur l'ordonnancement des trains dans les gares complexes et à forte densité de circulation. L'objectif final est de réaliser un outil pour aider les managers de la gare à générer un tableau des horaires sans conflits pour une journée complète. La gestion des circulations ferroviaires dans la gare nécessite un ordonnancement précis pour s'adapter aux ressources limitées en évitant les conflits entre les trains tout en satisfaisant l'objectif et les politiques à la fois économique et de sécurité.

Ce système consiste à vérifier la faisabilité des horaires donnés à la gare. Les trains à l'origine des conflits non résolus sont identifiés et les modifications d'horaires commerciaux proposées. Les détentes horaires des trains commerciaux sont minimisées pour diminuer la propagation des retards dans le réseau ferroviaire.

Mots-clés: Planification du transport ferroviaire, Allocation des ressources, Ordonnancement des trains, Réseau Ferroviaire, Méthodes de décomposition, décomposition trois-niveaux

Title: Train platforming problem in busy and complex railway stations.

This thesis focuses on the train platforming problem within busy and complex railway stations and aims to develop a computerized dispatching support tool for railway station dispatchers to generate a full-day conflict-free timetable. The management of rail traffic in stations requires careful scheduling to fit to the existing infrastructure, while avoiding conflicts between large numbers of trains and satisfying safety or business policy and objectives.

This system is able to verify the feasibility of tentative timetable given to railway station. Trains with unsolvable conflicts will return to their original activity managers with suggestions for the modification of arrival and departure times. Time deviations of commercial trains' activities are minimized to reduce the delay propagation within the whole railway networks.

Key words: Train Platforming, Train Scheduling, Train Routing, Railway networks, Decomposition method, Tri-level optimization