



**HAL**  
open science

# Metro Regenerative Braking Energy Optimization through Rescheduling: Mathematical Model and Greedy Heuristics Compared to MILP and CMA-ES

David Fournier

► **To cite this version:**

David Fournier. Metro Regenerative Braking Energy Optimization through Rescheduling: Mathematical Model and Greedy Heuristics Compared to MILP and CMA-ES. Computer Science [cs]. Paris-VIII, 2014. English. NNT: . tel-01102408

**HAL Id: tel-01102408**

**<https://theses.hal.science/tel-01102408>**

Submitted on 12 Jan 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - ShareAlike 4.0 International License

## THÈSE

Présentée pour obtenir

LE GRADE DE DOCTEUR EN SCIENCES  
DE L'UNIVERSITÉ PARIS DIDEROT

Spécialité : Informatique

par

David FOURNIER

# Metro Regenerative Braking Energy Optimization through Rescheduling: Mathematical Model and Greedy Heuristics Compared to MILP and CMA-ES

Présentée le 27 novembre 2014 devant le jury composé de :

- M. Thierry BENOIST
- M. Xavier DELORME
- M. Roberto DI COSMO (Président du jury)
- M. François FAGES (Directeur de thèse)
- M. Narendra JUSSIEN
- M. Denis MULARD (Co-directeur de thèse)

Rapporteurs :

- M. Thierry BENOIST
- M. Narendra JUSSIEN

Thèse préparée à  
**Inria Paris-Rocquencourt**  
EPI Lifeware  
Domaine de Voluceau, Rocquencourt, BP 105  
78 153 LE CHESNAY CEDEX

## Abstract

The use of regenerative braking is a key factor to reduce the energy consumption of a metro line. In the case where no device can store the energy produced during braking, only the metros that are accelerating at the same time can benefit from it. Maximizing the power transfers between accelerating and braking metros thus provides a simple strategy to benefit from regenerative energy without any other hardware device. In this thesis, we use a mathematical timetable model to classify various metro energy optimization rescheduling problems studied in the literature and prove their NP-hardness by polynomial reductions of SAT. We then focus on the problem of minimizing the global energy consumption of a metro timetable by modifying the dwell times in stations. We present a greedy heuristic algorithm which aims at locally synchronizing braking metros along the timetable with accelerating metros in their time neighbourhood, using a non-linear approximation of energy transfers. On a benchmark of six small size timetables, we show that our greedy heuristics performs better than CPLEX using a MILP formulation of the problem, even when it is able to prove the optimality of a linear approximation of the objective function. We also show that it runs ten times faster than a state-of-the-art evolutionary algorithm, called the covariance matrix adaptation evolution strategy (CMA-ES), using the same non-linear objective function on these small size instances. On real data leading to 10000 decision variables on which both MILP and CMA-ES do not provide solutions, the dedicated algorithm of our thesis computes solutions with a reduction of energy consumption ranging from 5% to 9%.

**Keywords** : Regenerative braking, MILP, Timetable optimization, Energy optimization, Mass rapid transit, Operations research, Heuristics.

OPTIMISATION DE L'ÉNERGIE DE RÉCUPÉRATION AU FREINAGE DES MÉTROS PAR  
MODIFICATION DE LA TABLE HORAIRE : MODÈLE MATHÉMATIQUE ET  
HEURISTIQUE GLOUTONNE COMPARÉE À LA PLNE ET À CMA-ES

**Résumé**

La réutilisation de l'énergie de freinage est un facteur clé pour réduire la consommation énergétique d'une ligne de métro. Si cette énergie ne peut pas être stockée, la seule manière de l'utiliser est d'en faire bénéficier les métros qui accélèrent au même moment. Maximiser les transferts de puissance entre les métros qui accélèrent et ceux qui freinent est donc une stratégie simple pour profiter de l'énergie de freinage. Dans cette thèse, nous utilisons un modèle mathématique de table horaire qui permet de classer des problèmes variés d'optimisation énergétique dans les métros, étudiés dans la littérature, et de prouver leur NP-difficulté par des réductions polynomiales de SAT. Nous nous concentrons particulièrement sur le problème de la minimisation de la consommation énergétique globale d'une table horaire de métro en ne modifiant que les temps d'arrêt en stations. Nous présentons un algorithme glouton qui vise à synchroniser localement, tout au long de la table horaire, les métros qui freinent avec les métros qui accélèrent dans leur voisinage temporel, en utilisant une approximation non linéaire des transferts d'énergie. Une évaluation sur six tables horaires de petite taille montre que notre heuristique gloutonne donne de meilleurs résultats qu'un modèle PLNE résolu par CPLEX. Ce même quand ce dernier est capable de prouver l'optimalité de solutions dont la fonction objectif est une approximation linéaire de la consommation énergétique. Notre heuristique donne aussi des résultats dix fois plus rapidement qu'un algorithme évolutionnaire de l'état de l'art nommé *covariance matrix adaptation evolution strategy* (CMA-ES), en utilisant la même fonction objectif non linéaire. Sur des données réelles contenant 10000 variables de décisions et sur lesquelles ni CPLEX ni CMA-ES ne sont pas capables de calculer une solution, l'algorithme dédié présenté dans notre thèse donne des solutions réduisant de 5% à 9% la consommation d'énergie.

**Mots-clés** : Energie de récupération au freinage, PLNE, Optimisation de table horaire, Optimisation énergétique, Transport collectif urbain, Recherche opérationnelle, Heuristique.



# Remerciements

Je tiens tout d'abord à remercier mon directeur de thèse, François FAGES, qui a su me guider avec constance et application tout au long de ma thèse. Je le remercie pour ses intuitions qui m'ont aidé à avancer et pour son goût de la perfection qui m'a permis d'écrire au mieux ce manuscrit.

Je remercie tout particulièrement Denis MULARD pour m'avoir fait confiance pendant ces trois années passées chez General Electric. Je le remercie de m'avoir soutenu et de m'avoir laissé les mains libres pour faire ma recherche dans les meilleures conditions possibles.

Je voudrais remercier aussi James GIARDINI pour le temps qu'il m'a consacré pendant la phase d'industrialisation de mes travaux. Sa patience et sa pédagogie m'ont été d'une grande aide.

Je remercie Steven GAY qui a été mon collègue de bureau pendant ma première année et avec qui j'ai partagé tant, voire trop de délires.

Merci à Sylvain SOLIMAN et Thierry MARTINEZ de m'avoir donné le goût du café et qui ont toujours répondu présent lors des moments difficiles, que ce soit pour relire mes manuscrits, me redonner le moral, ou pour faire des preuves de NP-complétude.

Je remercie bien évidemment tous mes collègues de Lifeware et de General Electric Transportation pour leurs interactions toujours fructueuses.

Je souhaite remercier particulièrement Thierry BENOIST et Narendra JUSSIEN pour avoir accepté d'être rapporteurs de ma thèse et pour les précieux conseils qu'ils m'ont transmis.

Je remercie aussi vivement Roberto DI COSMO pour avoir accepté d'être président du jury et Xavier DELORME pour avoir accepté d'en être membre.

Je remercie ma mère, mon père, ma sœur, Delphine, Vincent, mes deux grands-mères, Marie-Laure et toute ma famille pour le soutien et l'amour inconditionnels qu'ils m'ont porté.

Je remercie aussi mes amis, Vincent, Julien, Eric, Albert, Mathieu, Pierre, et les autres pour ne pas s'être trop moqué de moi et pour avoir fait semblant de s'intéresser à ce que je faisais.

Je remercie enfin mon épouse, Monireh SANAEI, pour avoir été présente dans les meilleurs comme dans les pires moments, de m'avoir soutenu et supporté pendant ces trois années de thèse, et pour tout l'amour qu'elle m'a donné.





# Contents

Introduction . . . . .	13
<b>1 Metro Energy Optimization Rescheduling Problems</b>	<b>21</b>
1.1 Metro Energy Optimization Rescheduling Model . . . . .	21
1.1.1 Variables . . . . .	22
1.1.2 Constraints . . . . .	22
1.1.3 Objective Function . . . . .	24
1.2 Problems Classification . . . . .	24
1.3 Complexity . . . . .	25
1.3.1 Membership to NP . . . . .	26
1.3.2 Global Energy Consumption Problems . . . . .	27
1.3.3 Maximum Power Peak Problems . . . . .	34
<b>2 Instant Power Demand</b>	<b>39</b>
2.1 Electrical Simulator . . . . .	40
2.1.1 Network Parameters . . . . .	40
2.1.2 Timetable Parameters . . . . .	41
2.1.3 Electric Variables . . . . .	41
2.1.4 Constraints and Instant Power Demand Value . . . . .	42
2.2 Instant Power Demand Approximations . . . . .	43
2.2.1 Power Flow and Non-Linear Approximation . . . . .	44
2.2.2 Power Flow Algorithm . . . . .	45
2.2.3 Linear Approximation for MILP . . . . .	47

<b>3</b>	<b>Metro Energy Optimization Timetabling</b>	<b>51</b>
3.1	Timetable Physical Feasibility . . . . .	52
3.1.1	Initial Timetable Parameters . . . . .	53
3.1.2	Metro Parameters . . . . .	53
3.1.3	Metro Constraints . . . . .	53
3.2	Metro Energy Optimization Timetabling Model . . . . .	54
3.2.1	Input Timetable Parameters . . . . .	55
3.2.2	Headway Pattern Parameters . . . . .	56
3.2.3	Headway Pattern Constraints . . . . .	56
3.2.4	Trip Allocation Algorithm . . . . .	58
3.3	Computational Results . . . . .	59
3.3.1	Headway Pattern Timetabling . . . . .	60
3.3.2	Energy Optimized Headway Pattern Timetabling . . . . .	61
<b>4</b>	<b>Greedy Heuristic Algorithm for <math>(G, dwe, nonlin)</math></b>	<b>65</b>
4.1	Acceleration Phase Shift Function . . . . .	66
4.1.1	Braking Phase Neighbourhood . . . . .	66
4.1.2	Shift and Propagation Algorithm . . . . .	66
4.2	Greedy Heuristics Optimization Algorithm . . . . .	67
4.3	Algorithm Optimizations . . . . .	71
4.3.1	Incremental Computation of the Objective Function . . . . .	71
4.3.2	Iterative Optimization . . . . .	71
<b>5</b>	<b>Performance Results</b>	<b>73</b>
5.1	Benchmark Instances . . . . .	73
5.2	Performance Comparison with MILP and CMA-ES . . . . .	74
5.2.1	Comparison with CMA-ES . . . . .	74
5.2.2	Comparison with MILP . . . . .	76
5.2.3	Robustness . . . . .	77

5.3 Performance Results on Real Data . . . . .	79
<b>6 Conclusion</b>	<b>85</b>



# Introduction

Reducing energy consumptions is a major issue for the future and has been the subject of increasing research activities over the last years. Transportation systems are the main energy consumers, estimated to represent 27% of the world energy production [1]. Among these systems, mass rapid transit, in particular metro systems, is a great consumer of electrical energy. As an example in 2006, the London Underground consumed 1173 GW.h [2], representing 2.8% of the Great London total electricity consumption [3].

Nowadays, almost all metros have *regenerative energy braking* systems. These systems are able to turn the electric motors into generators during braking phases, and thus to produce electricity. It has been shown that the raw energy discount provided by this technology is about 16.5% [4]. Some metros can directly use their own regenerative energy. Super capacitors allow much faster loads and unloads compared to classical batteries, and a metro equipped with super capacitors is able to collect the energy during braking, and give it back to the engine for its own accelerations [5]. The metros that cannot store their own regenerative energy can return it to the DC electrical network, but with important losses on long distances. The electrical substations (ESS) are the devices that convert AC to DC to feed the metro line. Some ESS are revertible and can convert the regenerative power of trains to AC power. In this case, the energy regenerated by metros can be used in other parts of the metro line without important loss, or be sold back to the electricity provider. Cornic showed that revertible ESS are able to convert over 99% of the regenerative braking energy, leading to a 18% saving of the annual energy consumption [6].

However, super capacitors, as well as revertible ESSs, are expensive equipment to buy and maintain, and may not be economically justified. González-Gil *et al.* [7] made an extensive review on energy efficient solutions for metros and classified them in terms of energy savings potential and in investment cost, as pictured in Figure 1. It is clear that the two methods with the best ratio *energy savings potential/investment cost* are the eco-driving techniques, in particular the optimization of the speed profiles, and the timetable optimization.

These two techniques do not require any extra equipment. The speed profiles opti-

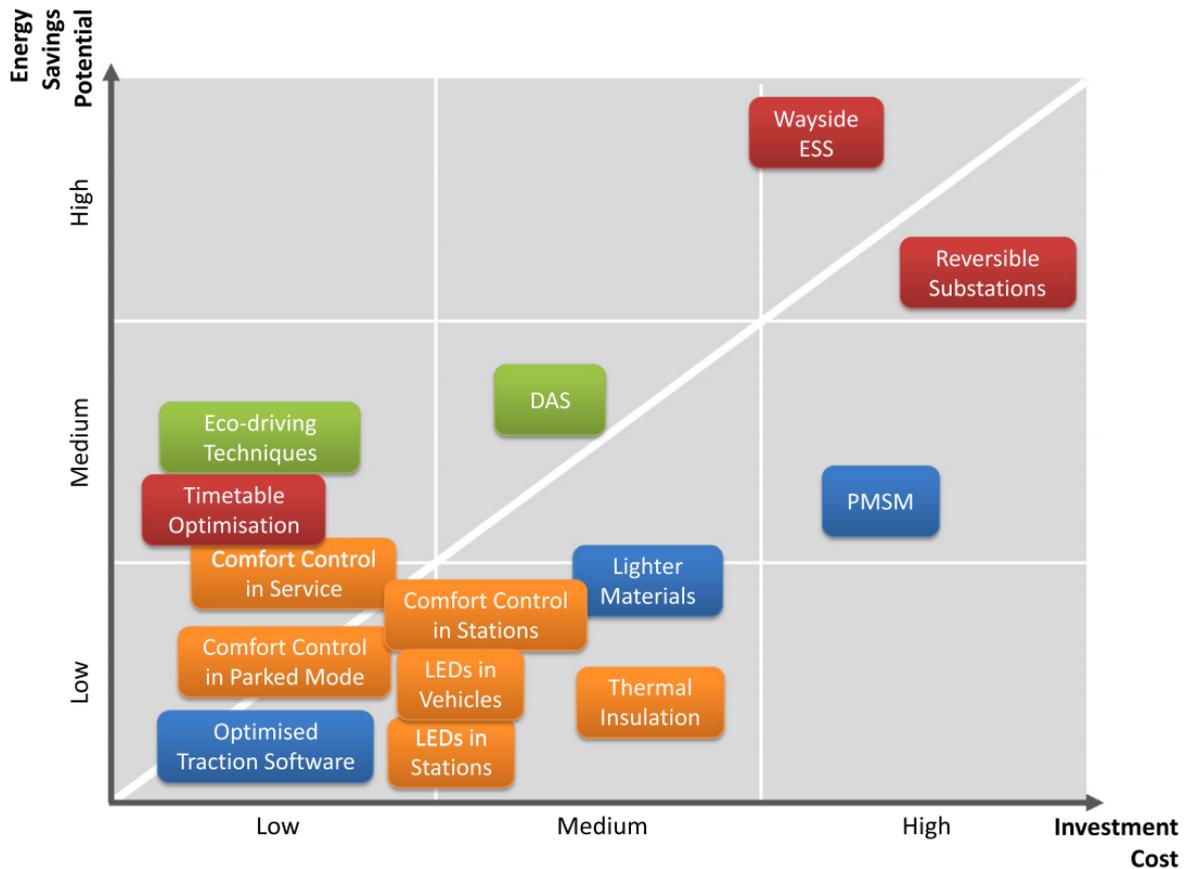


Figure 1: Comparison of energy efficient solutions in terms of investment costs and energy savings potential [7].

mization consist in finding the pattern of acceleration, coasting, cruise and braking that minimizes the energy consumption of a metro at a given interstation run. Chevrier *et al.* [8] or Bocharnikov *et al.* [9] used genetic algorithms to find energy efficient speed profiles, given the train characteristics, the gradient of the line, speed restrictions and some timetable constraints such as the minimum and maximum interstation time. Also, Su *et al.* showed that the speed profiles can not only be optimized to minimize the energy consumption the metro itself, but also to maximize the production of regenerative energy per run [10]. This regenerative energy surplus can then be used by other metros accelerating in the same time.

The metro energy optimization timetabling problem addresses the problem of synchronizing the braking of a metro with the acceleration of another metro in its close neighbourhood on the DC line, assuming that the speed profiles are fixed and have already been optimized beforehand. This synchronization can be done, for instance, by modifying the departure and arrival times of the metros in the stations, in order to shift the acceleration phases to the deceleration phases of some other metros in their neighbourhood.

---

## Thesis Contributions

This thesis first proposes a mathematical model to describe several metro energy optimization rescheduling problems under the same formalism. This mathematical model enables us to classify various problems of the literature, given their decision variables, their objective function and their energy consumption evaluation. We give a formal proof of NP-hardness of the energy optimization timetabling problem by polynomial reductions of SAT.

In our model, the energy consumption of the timetable is evaluated by a non-linear approximation of the real electrical behaviour of the metro line. This approximation is based on a power flow model – a particular lossy generalized flow problem – which models the energy transfers between braking and accelerating metros. The energy transfer losses between two points on the metro line are computed by the electrical simulator and are passed as parameters in the power flow approximation, giving a fast and accurate energy consumption evaluation.

This thesis then presents a greedy heuristic algorithm that addresses the problem of minimizing the global energy consumption of a timetable by solely rescheduling the dwell times. This heuristics not only increases the overlapping times between braking and accelerating metros but privileges those synchronizations that globally decrease the energy consumption. Extensive tests show that our approach gives robust solutions faster and of higher quality than MILP and a state-of-the-art evolutionary algorithm called the covariance matrix adaptation evolution strategy (CMA-ES) [11], and that it is suitable for real size timetables.

A set of benchmark instances, containing typical peak and off-peak hours timetables for a typical metro lines, is proposed to compare different optimization techniques on different problems. The set of these instances is available at <http://lifeware.inria.fr/wiki/COR14/Bench>.

Our greedy heuristic algorithms is fully implemented in the General Electric Transportation Tempo CBTC Solution <sup>1</sup> as part of the Automatic Train Supervision (ATS) system and has been subject to a patent granted by the United States [12] and by Europe [13].

---

<sup>1</sup>[http://media.gettransportation.com/sites/default/files/Brochure%20CBTC\\_040814.pdf](http://media.gettransportation.com/sites/default/files/Brochure%20CBTC_040814.pdf)

## Literature Review

Metro energy efficient techniques may address two main objectives: the minimization of power peaks, and the minimization of the global energy consumption over the day. A power peak occurs when too many metros are accelerating at the same time, which may cause in the worst case a momentary shut down of a part of the network as the metro network is sized with some maximum power capacity. Furthermore, when the energy consumed exceeds a certain limit during a given time period, typically around 15 minutes [14], the metro company will pay a fine to the electricity provider. On the other hand, the minimization of the global energy consumption simply allows the metro provider to use less electricity to run its metro line.

### Maximum Power Peak

Albrecht has shown in [15] that it is possible to reduce power peaks by utilizing the reserve time of metros, i.e. the remaining time that a metro has to finish its journey without disturbing the network. It is however tricky to use the reserve time for energy optimization reasons since it is primarily used for traffic regulation. Moreover, this optimization is done by modifying metro interstation times, which may be difficult to implement in a real-time application. Nevertheless, the implementation of this method using a genetic algorithm showed good results. Kim *et al.* have proposed in [16] to optimize the metro departure times in terminals instead of reserve times. They have partially solved a simplified model of this problem using Mixed Integer Linear Programming (MILP) [17, 18, 19, 20]. However, their approximation is not precise enough for real applications since regular timetables are typically second-accurate, whereas their model has a precision of 15 seconds. They improved their model in [21], refining the computation of the power peaks at each electric substation rather than for the whole metro line. Both Chen *et al.* [22] and Sansó and Girard [23] have described a precise electrical network simulator, which leads to an accurate evaluation of the metro power demands. They have managed to reduce the maximum power peak using a genetic algorithm.

### Global Energy Consumption

Miyatake and Ko proposed an ODE-based electrical simulator to model the energy transfers on the metro line [24]. They showed that a fine control of the speed profiles and the synchronization of departure times of two metros reduces the energy consumption of the line between 4% and 18%, depending on the relative positions of these two metros

on the line. Similarly, Bocharnikov *et al.* showed in [25] that an optimal control of the speed profiles not only includes eco-driving techniques, but also a better reuse of the regenerative braking. Xun *et al.* increased the synchronization between braking and acceleration phases of two following metros by modifying the speed profile coasting times [26]. Their approach is done such that the passenger waiting time is not worsened. Nasri *et al.* have shown in [27] that it is possible to decrease the energy consumption by modifying the dwell times, i.e. the stopping times in the stations. Their model uses an exact energy function and an accurate discretization of time at the scale of one second. This work provides an interesting proof of concept but has only been tested on a pilot system, consisting of 4 trains and 4 stations. Chang *et al.* used a notion of line receptivity in [28] to evaluate finely the potential transfers between braking and accelerating metros along the line. They were able to model an event-driven approach for an online control of railway operations, including an objective of energy consumption minimization, by a dynamic modification of timetable dwell times. On a more realistic size problem, Ramos *et al.* [29] and Peña *et al.* [30] have proposed a MILP model to optimize a night shift timetable for the metro of Madrid by maximizing the overlapping times between braking and acceleration phases of different metros. Peña *et al.* have added to the model described in [29] a notion of distribution matrix, weighing the energy transfers in function of the line receptivity between metros. To limit the number of binary variables, their model considers only one-to-one pairing between braking and accelerating metros. Their measurements on field have shown that the optimized timetable could reduce the global energy consumption by 3%. Recently, Yang *et al.* proposed a bi-objective timetable model minimizing both the passenger waiting time and the overlapping times. Complying with physical constraints (speed profiles) safety constraints (headways) and quality of service constraints (bounds on dwell times and trip times), they were able to generate a Beijing metro timetable reducing by 8% compared to the real timetable.

## Thesis Plan

**Chapter 1** The first chapter presents the mathematical model of the metro energy optimization rescheduling problems. This is a general timetable model, including domain bound constraints on all decision variables – the dwell times, interstation times and so on – and an energy-based objective function. This model enables us to classify various related problems of the literature, based on the variants of the problem. We prove the NP-hardness of some of the metro energy optimization rescheduling problems by polynomial reductions of SAT.

**Chapter 2** To evaluate the objective function of a metro energy optimization rescheduling problem, one must be able to evaluate the power demand of the metro line at

a given time. This chapter presents the model of the electrical simulator that has been used to compute this instant power demand. Then we propose a non-linear approximation of the instant power demand using a power flow model. We propose an algorithm to compute this power flow and use it as an objective function for CMA-ES [11] and in our heuristics. For MILP, a linear approximation of the objective function based on the overlapping times is also presented.

**Chapter 3** In this chapter we present a MILP model intended to unify both the creation of a timetable satisfying customer needs, safety and physical constraints, and the global energy consumption of this timetable. We first describe a model to check the feasibility of a fully parametrized timetable, given physical constraints to ensure that virtual trips can be effectively performed by metros. Then we describe a MILP model that unifies a *timetabling model* complying with a headway pattern and a global energy consumption optimization of this same timetable. To link created trips with physical metros, a trip allocation algorithm is also proposed. Finally, we show how an original handmade timetable can be greatly improved by the metro energy optimization timetabling model.

**Chapter 4** This chapter describes our greedy heuristic algorithm to solve the particular problem of minimizing the global energy consumption of a metro line by modifying solely the dwell times in stations ( $G, dwe, nonlin$ ). We formalize the notion of braking phase neighbourhoods and acceleration phase shifts that will be used in the algorithm. Our heuristics is given with two optimizations that increase the overall performance of it. The incremental computation of the objective function, made possible by the local modifications of the algorithm, allow a dynamic evaluation of the energy consumption over the process. Also, an iterative optimization is proposed to take benefit from the output solutions of an algorithm run to increase the efficiency of next runs.

**Chapter 5** The last chapter compares our heuristic algorithm with MILP and CMA-ES [11], on a benchmark of six small size timetables. We show that it gives better results than MILP and CMA-ES in both computation time and quality of the solutions. Furthermore, in order to prove that our heuristics can be utilized in an industrial context, we show that the output solutions are robust to small perturbations that naturally occur in a real time context. On real data, we show that the heuristics is able to reschedule two full timetables containing respectively 9585 and 7679 variables in 20 minutes. On these two examples, the two other methods fail at giving a solution within 30 minutes, CPLEX running out of memory and CMA-ES failing at computing its first iteration, while the heuristics computes solutions which decrease the total amount of energy consumed by respectively 5.15% and

7.54%. We also show that it is possible to save up to 8.91% energy by increasing the tolerance on trip times and headways.



# Chapter 1

## Metro Energy Optimization Rescheduling Problems

### Contents

---

<b>1.1</b>	<b>Metro Energy Optimization Rescheduling Model . . . . .</b>	<b>21</b>
1.1.1	Variables . . . . .	22
1.1.2	Constraints . . . . .	22
1.1.3	Objective Function . . . . .	24
<b>1.2</b>	<b>Problems Classification . . . . .</b>	<b>24</b>
<b>1.3</b>	<b>Complexity . . . . .</b>	<b>25</b>
1.3.1	Membership to NP . . . . .	26
1.3.2	Global Energy Consumption Problems . . . . .	27
1.3.3	Maximum Power Peak Problems . . . . .	34

---

## 1.1 Metro Energy Optimization Rescheduling Model

In this section, we define a generic mathematical model of metro energy optimization rescheduling which will be used to:

1. define and classify different metro energy optimization rescheduling problems from the literature (Section 1.2),
2. define the related MILP models,
3. define the instant power demand function (Chapter 2),

4. and define our greedy heuristic algorithm (Chapter 4).

The term *rescheduling* lies in the fact that we assume that physical trains and crew have been allocated to trips beforehand, so that the model of timetable described here does not consider metros depot movements, crew rostering, nor turnaround manoeuvres. A timetabling model, considering the metro allocations and the feasibility of the timetable regarding the metro line is presented in Chapter 3.

We assume that the metro line is composed of  $N$  stations,  $S = \{S_1, \dots, S_N\}$  and the metro timetable is represented as a sequence of  $M$  trips  $T = (T_1, \dots, T_{\frac{M}{2}}, T_{\frac{M}{2}+1}, \dots, T_M)$ , where  $M$  is even. For the sake of simplicity, we assume that the trips  $T_1, \dots, T_{\frac{M}{2}}$  cross all stations in the upstream sequence,  $(S_1, \dots, S_N)$ , and the trips  $T_{\frac{M}{2}+1}, \dots, T_M$  cross all stations in the downstream sequence,  $(S_N, \dots, S_1)$ . We note  $S_t(s)$  the  $s^{\text{th}}$  station crossed by the trip  $T_t$ , i.e. station  $S_s$  if  $1 \leq t \leq \frac{M}{2}$  or station  $S_{N-s+1}$  if  $\frac{M}{2} + 1 \leq t \leq M$ .

### 1.1.1 Variables

The variables are the dates of the departure and arrival times in stations for each trip, the dates of the starting of the braking phase and of the ending of the acceleration phase at each station for each trip. We consider that the time domain  $I = \{0, 1, \dots, I_{END}\}$  is discrete, with a precision of 1 second.

- $d_{t,s} \in I$  is the departure time of the trip  $T_t$ , with  $1 \leq t \leq M$ , at station  $S_t(s)$ , with  $1 \leq s \leq N - 1$ ,
- $a_{t,s} \in I$  is the arrival time of the trip  $T_t$ , with  $1 \leq t \leq M$ , at station  $S_t(s)$ , with  $2 \leq s \leq N$ ,
- $d_{t,s}^{acc} \in I$  is the ending time of the acceleration phase of the trip  $T_t$ , with  $1 \leq t \leq M$ , leaving station  $S_t(s)$ , with  $1 \leq s \leq N - 1$ ,
- $a_{t,s}^{brk} \in I$  is the beginning time of the braking phase of the trip  $T_t$ , with  $1 \leq t \leq M$ , arriving at station  $S_t(s)$ , with  $2 \leq s \leq N$ .

### 1.1.2 Constraints

Each trip  $t$  must leave its departure terminal (*dep*) within some bounds set by the metro company according to the scheduled timetable,

$$\underline{dep}_t \leq d_{t,1} \leq \overline{dep}_t \quad 1 \leq t \leq M. \quad (1.1)$$

Every trip  $t$  is given with an *interstation time* ( $int$ ), equal to the time required to travel two consecutive stations. The interstation times are bound according to the different speeds – e.g. economical, nominal or full throttle – a metro can take,

$$\underline{int}_{t,s} \leq a_{t,s+1} - d_{t,s} \leq \overline{int}_{t,s} \quad 1 \leq t \leq M, \quad 1 \leq s \leq N - 1, \quad (1.2)$$

The *dwell time* ( $dwe$ ) of a given trip at a station is the time it is required to stop in order to let passengers go in and out the metro. The dwell times are bound according to a minimum quality of service for the passengers,

$$\underline{dwe}_{t,s} \leq d_{t,s} - a_{t,s} \leq \overline{dwe}_{t,s} \quad 1 \leq t \leq M, \quad 2 \leq s \leq N - 1, \quad (1.3)$$

The *acceleration and braking phases* ( $acc$  and  $brk$ ). They represent the duration of the main acceleration and braking of each metro for every interstation. They are also bound

:

$$\underline{acc}_{t,s} \leq a_{t,s}^{acc} - d_{t,s} \leq \overline{acc}_{t,s} \quad 1 \leq t \leq M, \quad 1 \leq s \leq N - 1, \quad (1.4)$$

$$\underline{brk}_{t,s} \leq a_{t,s} - a_{t,s}^{brk} \leq \overline{brk}_{t,s} \quad 1 \leq t \leq M, \quad 2 \leq s \leq N, \quad (1.5)$$

It is worth noticing that every acceleration phase occurs right after a dwell time. Shifting the starting time of an acceleration phase  $d_{t,s}$  without modifying its length  $acc_{t,s}$  is thus equivalent to modify the length of the adjacent dwell time  $dwe_{t,s}$ .

The global *trip time* ( $trt$ ) is equal to the time a trip takes to run between its departure terminal and its arrival terminal. It is bound to ensure the feasibility of the timetable and the quality of service to the passengers,

$$\underline{trt}_t \leq a_{t,N} - d_{t,1} \leq \overline{trt}_t \quad 1 \leq t \leq M, \quad (1.6)$$

Finally the time intervals, called *headways* ( $hdw$ ), between two successive trips running in the same direction in a given station, are bound according to security requirements and quality of service,

$$\underline{hdw}_{t,s} \leq d_{t,s} - d_{t-1,s} \leq \overline{hdw}_{t,s} \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], \quad 1 \leq s \leq N. \quad (1.7)$$

The headways are thus checked here at each station. The tolerances on headways differ according to the hour of the day. During peak hours, the headways are small and the tolerance for stretching them is tight. During off peak hours, the tolerances are higher and more important modifications of the timetable are possible, leading to potentially greater energy savings.

### 1.1.3 Objective Function

All the constraints of the timetable model shown up to now are pretty trivial bound linear constraints. The difficulty of energy optimization problems lies in the objective function, and more precisely of the instant power demand  $P_i$  at each time.

Given an appropriate definition of  $P_i$ , the objective can then be to minimize either the global energy consumption  $G_{\mathcal{T}\mathcal{T}}$  of all trips,

$$G_{\mathcal{T}\mathcal{T}} = \sum_{i=0}^{I_{END}} P_i, \quad (1.8)$$

or the maximum power peak

$$PP_{\mathcal{T}\mathcal{T}} = \max_{i \in I} P_i. \quad (1.9)$$

It is worth noticing that the objective function  $PP_{\mathcal{T}\mathcal{T}}$  is widely used in the literature [15, 16, 22, 23, 21], but that a more realistic function would be the number of times that the power exceeds a certain threshold  $P_{MAX}$ ,

$$CP_{\mathcal{T}\mathcal{T}, P_{MAX}} = \text{card}(i \mid P_i > P_{MAX}). \quad (1.10)$$

Indeed, this function is more in accordance with the system of fines paid to the electricity provider when the quota is exceeded.

The most accurate evaluations of  $P_i$  are obtained by electrical simulators. In Chapter 2 we describe an instant power model, from which we derive two approximations: a non-linear approximation based on a power flow used in our algorithm and the linear approximation used in [29, 30]. Before that, the timetable model can already be used to classify several variants of the problem studied in the literature.

## 1.2 Problems Classification

As mentioned in the introduction, research is active for optimizing energy in the field of railways and some attempts to classify the studied problems have been made. Xun *et al.* [26] proposed a classification of the methods used to solve the problems when Li and Lo [31] listed without apparent classification some papers in the literature, detailing the decision variables or the algorithms used. We propose a classification based on the previous timetable model by a triple

$$(G/PP/CP, \text{dep}/\text{dwe}/\text{int}, \text{sim}/\text{nonlin}/\text{lin})$$

denoting the choice of the objective function ( $G$ ,  $PP$  or  $CP$ ), the decision variables (departure times, dwell times, interstation times or any combination of them) and the instant power demand evaluation (by an electrical simulator, a non-linear approximation or a linear approximation).

Problem	Equations	References
$(PP, dwe, sim)$	1.3, 1.4, 1.5, 1.6, 1.7, 1.9,	Chen <i>et al.</i> [22] Sansó and Girard [23]
$(PP, dwe - int, sim)$	1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.9	Albrecht <i>et al.</i> [15]
$(PP, dep, lin)$	1.1, 1.6, 1.7, 1.9	Kim <i>et al.</i> [16, 21]
$(G, dwe, nonlin)$	1.3, 1.6, 1.7, 1.8	Fournier <i>et al.</i> [12, 13, 32, 33] Chang <i>et al.</i> [28]
$(G, dwe, sim)$	1.3, 1.6, 1.7, 1.8	Nasri <i>et al.</i> [27]
$(G, dep - dwe, lin)$	1.1, 1.3, 1.6, 1.7, 1.8	Ramos <i>et al.</i> [29] Peña <i>et al.</i> [30] Yang <i>et al.</i> [34, 35]
$(G, int, sim)$	1.2, 1.4, 1.5, 1.6, 1.7, 1.8	Bocharnikov <i>et al.</i> [25]
$(G, int, nonlin)$	1.2, 1.6, 1.7, 1.8	Xun <i>et al.</i> [26]
$(G, dep - int, sim)$	1.1, 1.2, 1.4, 1.5, 1.6, 1.7, 1.8	Miyatake and Ko [24]

Table 1.1: Some metro timetabling energy optimization problems from the literature, classified by the problem triple they solve and the corresponding timetable equations.

Table 1.1 classifies different problems studied in the literature using these triples. In this paper, we shall focus on the problems  $(G, dwe, lin)$  and  $(G, dwe, nonlin)$ , that is, we focus on the problems of modifying solely the dwell times in order to minimize the global energy consumption of a metro line, evaluated using either a linear or a non-linear approximation. In this class of problems, the departure times, the interstation times, and the braking and acceleration phases are given. The modification of the dwell times modifies the arrivals  $a_{t,s}$  and departures  $d_{t,s}$  in stations, and hence the auxiliary variables.

### 1.3 Complexity

First of all, one can remark that without any objective function, the timetable feasibility problem is polynomial since all the equations of the timetable model are linear. Caprara *et al.* showed in [36] that minimizing the deviation of a solution timetable comparing to an initial one to satisfy capacity or overtaking constraints is NP-hard. Serafini *et al.* showed in [37] that the Periodic Event Scheduling Problem (PESP), the problem of periodically scheduling trains with precedence constraints, is NP-complete. Liebchen [38] finally showed that two natural variants of PESP are in MaxSNP, that is to say that they are optimization problems whose related decision problem is in *Strict NP*.

Based on an original idea by Thierry Martinez, we show the NP-hardness of the metro energy optimization rescheduling problems by polynomial reductions of SAT.

### 1.3.1 Membership to NP

Let us first define the decision problem associated with any problem of class  $(G, dep - dwe - int, lin)$ ,  $(G, dep - dwe - int, nonlin)$ ,  $(PP, dep - dwe - int, lin)$  and  $(PP, dep - dwe - int, nonlin)$  as the problem of checking if a solution timetable does not violate the timetable constraints and if its objective function is equal to a given value. In the following we adopt the convention that the value that must be checked for the objective function is given relatively to the value of the input timetable.

**Theorem 1.** *The decision problems associated with  $(G, dep - dwe - int, lin)$ ,  $(G, dep - dwe - int, nonlin)$ ,  $(PP, dep - dwe - int, lin)$  and  $(PP, dep - dwe - int, nonlin)$  belong to NP.*

*Proof.* Given an entirely instantiated solution of any of the  $(G, dep - dwe - int, lin)$ ,  $(G, dep - dwe - int, nonlin)$ ,  $(PP, dep - dwe - int, lin)$  or  $(PP, dep - dwe - int, nonlin)$  problems and an objective function value, the number of timetable inequalities to check is the sum of:

- $2M$  terminal departures inequalities (1.1),
- $2M.(N - 1)$  interstation times inequalities (1.2),
- $2M.(N - 2)$  dwell times inequalities (1.3),
- $4M.(N - 1)$  acceleration and braking phases inequalities (1.4, 1.5),
- $2M$  trip times inequalities (1.6) and
- $2N.(M - 2)$  headways inequalities (1.7),

that is to say,  $10M.N - 6M - 4N$  timetable inequalities, which is polynomial in the size of the input.

Furthermore, the evaluation of both the  $G$  or  $PP$  objective functions can be done in polynomial time, computing the  $I_{END}$  energy subproblems either using a linear or a quadratic approximation as shown in Chapter 2. For the objective function  $G$  (respectively  $PP$ ), the sum (respectively maximum) of the  $I_{END}$  subproblems values must be equal to the objective function value to check.

Since deciding if a solution timetable verifies the timetable constraints and the objective function value is done in polynomial time, the problems  $(G, dep - dwe - int, lin)$ ,  $(G, dep - dwe - int, nonlin)$ ,  $(PP, dep - dwe - int, lin)$  or  $(PP, dep - dwe - int, nonlin)$  belong to NP.

### 1.3.2 Global Energy Consumption Problems

**Theorem 2.** *The decision problem associated with  $(G, dep, lin)$  is NP-complete.*

The idea of the proof is to construct a polynomial reduction of SAT to the decision problem associated with the  $(G, dep, lin)$  problem. A particular timetable is constructed such that all the acceleration phases can be synchronized with only the periodic braking phases of a special metro  $T_0$ . It is possible to delay the departure time of each metro different from  $T_0$  by one time unit.

Each Boolean variable of the SAT problem is represented by one of the metros. The variable is true if the metro is delayed by one time unit and false if it is not. Each clause is encoded by a synchronization problem at each time  $i$  where  $T_0$  is braking. The clause is satisfied if at least one of the accelerating trains is synchronized with the braking of  $T_0$ . This ensures that one unit of energy is saved at this time. A Boolean formula in conjunctive normal form (CNF) is then satisfiable if and only if  $c$  units of energy can be saved in the timetable, where  $c$  is equal to the number of clauses in the SAT formula.

*Proof.* We show that there is a polynomial reduction of SAT to the decision problem of saving a certain amount of energy on a particular  $(G, dep, lin)$  problem. Let  $T = \{T_1, \dots, T_M\}$  be a set of  $M$  variables and  $\neg T = \{\neg T_1, \dots, \neg T_M\}$  be the set of their negations. Let  $\phi$  be a Boolean formula in conjunctive normal form:

$$\phi = \bigwedge_{i=1}^N c_i$$

where  $c_i$  are clauses of the form  $\bigvee_{j=1}^{M_i} l_{i,j}$  with  $l_{i,j} \in T \cup \neg T$ .

Let us consider the discrete time domain  $I = \{0, \dots, 6N - 1\}$ . Let  $\mathcal{TT}$  be the metro timetable composed of a sequence of  $M + 1$  trips  $T = (T_0, T_1, \dots, T_M)$  running in the same direction, such that each trip  $T_t \in T$  crosses a sequence of unique stations of length  $N + 1$ , and that the trips are:

- The trip  $T_0$  that cannot be shifted and that is crossing stations every 6 time units, departing from its first station at time  $i = 0$ :

- $d_{0,1} = 0$
- $\underline{int}_{0,s} = \overline{int}_{0,s} = int_{0,s} = 1 \quad 1 \leq s \leq N$
- $\underline{dwe}_{0,s} = \overline{dwe}_{0,s} = dwe_{0,s} = 5 \quad 2 \leq s \leq N$

Reminding the metro energy optimization rescheduling constraints:

- the interstation time for a given trip,

$$\underline{int}_{0,s} = \overline{int}_{0,s} = int_{t,s} = a_{t,s+1} - d_{t,s} \quad 0 \leq t \leq M, \quad 1 \leq s \leq N, \quad (1.11)$$

- the dwell time, or stopping time, in a station for a given trip,

$$\underline{dwe}_{0,s} = \overline{dwe}_{0,s} = dwe_{t,s} = d_{t,s} - a_{t,s} \quad 0 \leq t \leq M, \quad 2 \leq s \leq N, \quad (1.12)$$

let us prove by induction that  $d_{0,s} = 6(s-1)$ ,  $1 \leq s \leq N$ :

*Basis:* the statement holds for  $s = 1$ ,

$$d_{0,1} = 0 = 6(1-1)$$

*Inductive step:* if  $d_{0,s} = 6(s-1)$ , then  $d_{0,s+1} = 6s$ .

We can write

$$d_{0,s+1} = d_{0,s} + a_{0,s+1} - d_{0,s} + d_{0,s+1} - a_{0,s+1}$$

According to (1.11) and (1.12) we have:

$$\begin{aligned} d_{0,s+1} &= d_{0,s} + int_{t,s} + dwe_{t,s+1} \\ \Leftrightarrow d_{0,s+1} &= 6(s-1) + 1 + 5 \\ \Leftrightarrow d_{0,s+1} &= 6s \end{aligned}$$

We thus have:

$$d_{0,s} = 6(s-1), \quad 1 \leq s \leq N \quad (1.13)$$

$$a_{0,s} = d_{0,s-1} + int_{0,s-1} = 6(s-2) + 1, \quad 2 \leq s \leq N+1 \quad (1.14)$$

- The trips  $T_t$  with  $1 \leq t \leq M$  that are constructed according to the clauses  $c_i$  of  $\phi$ . The only possible shift applicable on trips  $T_t$  in this timetable is a delay of their departure time by 1 time unit, denoted  $\delta_t \in \{0, 1\}$ :

$$- d_{t,1} = \delta_t + \begin{cases} 0 & \text{if } T_t \in c_1 \\ 1 & \text{if } \neg T_t \in c_1 \\ 2 & \text{otherwise} \end{cases} \quad 1 \leq t \leq M$$

$$- \text{int}_{t,s} = \begin{cases} 4 & \text{if } T_t \in c_s \\ 3 & \text{if } \neg T_t \in c_s \\ 2 & \text{otherwise} \end{cases} \quad 1 \leq t \leq M, 1 \leq s \leq N$$

$$- \text{dwe}_{t,s} = \begin{cases} 2 & \text{if } T_t \in c_s \\ 3 & \text{if } \neg T_t \in c_s \\ 4 & \text{otherwise} \end{cases} \quad 1 \leq t \leq M, 2 \leq s \leq N$$

Like for  $T_0$ , we prove by induction that

$$d_{t,s} = 6(s-1) + \delta_t + \begin{cases} 0 & \text{if } T_t \in c_s \\ 1 & \text{if } \neg T_t \in c_s \\ 2 & \text{otherwise} \end{cases}, \quad 1 \leq t \leq M, 1 \leq s \leq N:$$

*Basis:* the statement holds for  $s = 1$ ,

$$d_{t,1} = \delta_t + \begin{cases} 0 & \text{if } T_t \in c_1 \\ 1 & \text{if } \neg T_t \in c_1 \\ 2 & \text{otherwise} \end{cases} = 6(1-1) + \delta_t + \begin{cases} 0 & \text{if } T_t \in c_1 \\ 1 & \text{if } \neg T_t \in c_1 \\ 2 & \text{otherwise} \end{cases}$$

$$\text{Inductive step: if } d_{t,s} = 6(s-1) + \delta_t + \begin{cases} 0 & \text{if } T_t \in c_s \\ 1 & \text{if } \neg T_t \in c_s \\ 2 & \text{otherwise} \end{cases},$$

$$\text{then } d_{t,s+1} = 6s + \delta_t + \begin{cases} 0 & \text{if } T_t \in c_{s+1} \\ 1 & \text{if } \neg T_t \in c_{s+1} \\ 2 & \text{otherwise} \end{cases}.$$

We can write

$$d_{t,s+1} = d_{t,s} + a_{t,s+1} - d_{t,s} + d_{t,s+1} - a_{t,s+1}$$

According to (1.11) and (1.12) we have:

$$\begin{aligned}
d_{t,s+1} &= d_{t,s} + int_{t,s} + dwe_{t,s+1} \\
\Leftrightarrow d_{t,s+1} &= 6(s-1) + \delta_t + \begin{cases} 0 + 4 & \text{if } T_t \in c_s \\ 1 + 3 & \text{if } \neg T_t \in c_s \\ 2 + 2 & \text{otherwise} \end{cases} + \begin{cases} 2 & \text{if } T_t \in c_{s+1} \\ 3 & \text{if } \neg T_t \in c_{s+1} \\ 4 & \text{otherwise} \end{cases} \\
\Leftrightarrow d_{t,s+1} &= 6(s-1) + \delta_t + 4 + 2 \begin{cases} 0 & \text{if } T_t \in c_{s+1} \\ 1 & \text{if } \neg T_t \in c_{s+1} \\ 2 & \text{otherwise} \end{cases} \\
\Leftrightarrow d_{t,s+1} &= 6s + \delta_t + \begin{cases} 0 & \text{if } T_t \in c_{s+1} \\ 1 & \text{if } \neg T_t \in c_{s+1} \\ 2 & \text{otherwise} \end{cases}
\end{aligned}$$

We thus have:

$$d_{t,s} = 6(s-1) + \delta_t + \begin{cases} 0 & \text{if } T_t \in c_s \\ 1 & \text{if } \neg T_t \in c_s \\ 2 & \text{otherwise} \end{cases} \quad 1 \leq t \leq M, \quad 1 \leq s \leq N \quad (1.15)$$

$$\begin{aligned}
a_{t,s} &= d_{t,s-1} + int_{t,s-1} = 6(s-2) + \delta_t + \begin{cases} 0 + 4 & \text{if } T_t \in c_{s-1} \\ 1 + 3 & \text{if } \neg T_t \in c_{s-1} \\ 2 + 2 & \text{otherwise} \end{cases} \\
\Leftrightarrow a_{t,s} &= 6(s-2) + \delta_t + 4, \quad 1 \leq t \leq M, \quad 2 \leq s \leq N+1 \quad (1.16)
\end{aligned}$$

Let the instant power demand function be

$$P_i = \max(0, \sum_{t=0}^M P_{t,i}) \quad i \in I, \quad (1.17)$$

where  $P_{t,i} \in \mathbb{R}$  is the power demand or production of the trip  $T_t$  at time  $i$ . The objective function is  $G_{\mathcal{T}\mathcal{T}}$ . Let the trip acceleration and braking phases last one time unit only,

$$\begin{aligned}
d_{t,s}^+ - d_{t,s} &= 1, \quad 0 \leq t \leq M, \quad 0 \leq s \leq N, \\
a_{t,s} - a_{t,s}^- &= 1, \quad 0 \leq t \leq M, \quad 1 \leq s \leq N+1.
\end{aligned}$$

Each trip will demand one unit of power when departing from a station, and will produce one unit of power when arriving to a station. The rest of the time, each trip will not

demand or produce any power. For all trips  $T_t$ , the instant power demand or production  $P_{t,i}$  at time  $i$  can be written as follows:

$$P_{t,i} = \begin{cases} 1 & \text{if there exists } 1 \leq s \leq N & \text{s.t. } d_{t,s} = i \\ -1 & \text{if there exists } 2 \leq s \leq N + 1 & \text{s.t. } a_{t,s} = i \\ 0 & \text{otherwise} \end{cases} \quad (1.18)$$

Now let us prove that the global energy consumption is not modified by the powers produced by trips  $\{T_1, \dots, T_m\}$  since they brake it when no trip is accelerating. According to equations (1.16) and (1.18) we have:

$$P_{t,i} = -1 \text{ if } i = 6(s - 2) + (4 \text{ or } 5), \quad 1 \leq t \leq M, \quad 2 \leq s \leq N + 1$$

Conversely and according to equations (1.13), (1.15) and (1.18), we have for all trips  $T_t \in T$ :

$$P_{t,i} = 1 \text{ if } i = 6(s - 1) + (0 \text{ or } 1 \text{ or } 2 \text{ or } 3), \quad 0 \leq t \leq M, \quad 1 \leq s \leq N \quad (1.19)$$

Thus, there is no time where the braking of any trip in  $\{T_1, \dots, T_M\}$  can be synchronized with the acceleration of any trip in  $T$ :

$$\nexists (i \in I, 1 \leq t \leq M, 0 \leq t' \leq M) \mid P_{t,i} = -1 \wedge P_{t',i} = 1$$

On the other hand, the braking phases of the trip  $x_0$  can be absorbed by the acceleration phases of the other trips, optimizing the objective function. According to equations (1.14) and (1.18) we have:

$$P_{0,i} = -1 \text{ if } i = a_{0,s} = 6(s - 2) + 1 \quad 2 \leq s \leq N + 1$$

Also according to equation (1.19),  $P_{t,s} = 1$  if  $i = d_{t,s}$ . To synchronize the acceleration of the trip  $T_t$  at station  $S_t(s)$  with one braking of the trip  $T_0$  we need:

$$\begin{aligned} a_{0,s+1} &= d_{t,s} \\ \Leftrightarrow 6(s + 1 - 2) + 1 &= 6(s - 1) + \delta_t + \begin{cases} 0 & \text{if } T_t \in c_s \\ 1 & \text{if } \neg T_t \in c_s \\ 2 & \text{otherwise} \end{cases} \\ \Leftrightarrow (\delta_t = 0 \wedge \neg T_t \in c_s) &\vee (\delta_t = 1 \wedge T_t \in c_s) \end{aligned}$$

In other terms, the timetable is constructed such that for each trip  $T_t$  and for each

station  $S_t(s)$  we have:

- If the variable  $T_t$  is in the clause  $c_s$ , then the acceleration phase of  $T_t$  at station  $S_t(s)$  is synchronized with the braking phase of  $T_0$  at station  $S_0(s+1)$  if and only if  $\delta_t = 1$ . Thus setting  $\delta_t = 1$  is equivalent to say that the variable  $T_t$  is true, satisfying all clauses  $c_s$  containing it.
- If the variable  $\neg T_t$  is in the clause  $c_s$ , then the acceleration phase of  $T_t$  at station  $S_t(s)$  is synchronized with the braking phase of  $T_0$  at station  $S_0(s+1)$  if and only if  $\delta_t = 0$ . Thus setting  $\delta_t = 0$  is equivalent to say that the variable  $\neg T_t$  is true, satisfying all clauses  $c_s$  containing it.
- If neither the variable  $T_t$  nor the variable  $\neg T_t$  are in the clause  $c_s$ , then the acceleration phase of  $T_t$  at station  $S_t(s)$  cannot be synchronized with any of the braking phases of  $T_0$ .

Every time one braking phase of  $T_0$  is synchronized with the acceleration phase of one trip  $T_t$ , one unit of power is saved. Therefore, there is a timetable and a set of  $\delta_t$  with  $1 \leq t \leq M$  that save  $N$  power units if and only if the set of  $\delta$  gives a valuation that satisfies  $\phi$ . Consequently, any SAT problem can be polynomially encoded into an instance of the decision problem associated with  $(G, dep, lin)$ .

Since the SAT problem is NP-complete, the decision problem associated with  $(G, dep, lin)$  is NP-hard, and since the latter belongs to NP (proof 1), the decision problem associated with  $(G, dep, lin)$  is NP-complete.

**Example 1.** *Let us consider the SAT formula  $(T_2 \vee T_3) \wedge (T_1 \vee \neg T_2 \vee T_3) \wedge (\neg T_1 \vee T_2)$ . The constructed timetable contains four trips  $T_0, T_1, T_2$  and  $T_3$ , and is divided in three periods during which metro  $T_0$  has the same behaviour. For each braking phase of  $T_0$  occurring at times 1, 7 and 13, either  $T_1, T_2$  or  $T_3$  can be synchronized to save one energy unit. At each time unit, a metro is either accelerating, consuming 1 energy unit, braking, producing 1 energy unit, or coasting or dwelling, producing or consuming nothing. The formula is satisfiable if and only if for each braking phase of  $T_0$ , one of the other metros can synchronize their acceleration phase. To synchronize their acceleration phases with the braking phases of  $T_0$ , each trip departure time can be delayed by one time unit. The following timetable  $\mathcal{TT}$ , represented by the energy consumed or produced by each metro at each time, encodes the formula  $(T_2 \vee T_3) \wedge (T_1 \vee \neg T_2 \vee T_3) \wedge (\neg T_1 \vee T_2)$ :*

*In this example, to save the 3 energy units produced by  $T_0$ , one solution can be to delay the trip  $T_3$  by one time unit. A feasible solution for this timetable would be then  $\delta_1 = 0, \delta_2 = 0, \delta_3 = 1$ .*

$\mathcal{TT}$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
$T_0$	1	-1	0	0	0	0	1	-1	0	0	0	0	1	-1	0	0	0	0
$T_1$	0	0	1	0	-1	0	1	0	0	0	-1	0	0	1	0	0	-1	0
$T_2$	1	0	0	0	-1	0	0	1	0	0	-1	0	1	0	0	0	-1	0
$T_3$	1	0	0	0	-1	0	1	0	0	0	-1	0	0	0	1	0	-1	0

Table 1.2: Decision problem associated with  $(G, dep, lin)$ : example timetable encoding a SAT formula of  $N$  clauses. Each cell represents the energy produced or consumed by the trip  $T_i$  at time  $i$ .

**Corollary 1.** *The decision problem associated with  $(G, dep, nonlin)$  is NP-complete.*

*Proof.* The objective function is of class *nonlin* if it involves at least one equation which is non-linear. Modifying the equation (1.17) of the previous timetable into the non-linear equation

$$\begin{cases} P_i = \max(0, \sum_{t=0}^M P_{t,i}) & \text{if } 6l \leq i \leq 6l + 3 \\ P_i = \max(0, \sum_{t=0}^M P_{t,i}^{(2+1)^k}) & \text{if } 6l + 4 \leq i \leq 6l + 5 \end{cases} \quad \forall 0 \leq l \leq N - 1$$

with  $k \in \mathbb{N}$ , does not change the result of the objective function. Indeed, the timetable is constructed in such a way that  $\sum_{t=0}^M P_{t,i} \leq 0$ , for all  $6l+4 \leq i \leq 6l+5$  and  $0 \leq l \leq N-1$ . Thus, like for the previous timetable,  $P_i = 0$ , for all  $6l+4 \leq i \leq 6l+5$  and  $0 \leq l \leq N-1$ . Finally the problem becomes  $(G, dep, nonlin)$  without changing its solutions, thus the problem associated with  $(G, dep, nonlin)$  is NP-complete.

**Corollary 2.** *The decision problem associated with  $(G, dep - dwe - int, lin)$  is NP-complete.*

*Proof.* A problem is of class *dwe* if at least one dwell time can be modified. Likewise, a problem is of class *int* at least one interstation time can be modified. To get a problem  $(G, dep - dwe - int, lin)$ , it suffices to add at the end of the timetable constructed in the proof of Theorem 2, one period of time where only  $T_0$  is running, and where its last dwell time and interstation time can be modified.

Let us thus consider the metro timetable  $\mathcal{TT}$  crossed by a sequence of  $M + 1$  trips  $T = (T_0, T_1, \dots, T_M)$  crossing  $N + 1$  stations each which encodes a Boolean formula in CNF containing  $n$  clauses. To correctly encode the formula, the timetable should have a length of  $6N - 1$  time units. Let us add at the end of the timetable 4 additional time units where only  $T_0$  is crossing a new station  $S_0(N + 2)$  such that:

$$\begin{aligned} d_{0,N+1} &= 6N \\ a_{0,N+2} &= 6N + 1 \end{aligned}$$

Now, the last dwell time of  $T_0$  can be extended by one time unit, encoded in  $\delta_{dwe} = \{0, 1\}$ ,

$$dwe_{0,N+1} = 5 + \delta_{dwe},$$

and the last interstation time of  $T_0$  can be extended by one time unit, encoded in  $\delta_{int} = \{0, 1\}$ ,

$$int_{0,N+1} = 1 + \delta_{int}$$

By adding these two tolerances, the particular decision problem associated with  $(G, dep, lin)$  encoding the SAT formula has become a problem  $(G, dep - dwe - int, lin)$ . As we have proved that the decision problem associated with  $(G, dep, lin)$  is NP-complete, then the  $(G, dep - dwe - int, lin)$  problem is also NP-complete.

**Example 2.** *The following timetable encodes a SAT formula with  $M$  variables and  $N$  clauses. The 4 last time units are dedicated to a last interstation trip for  $T_0$  whose dwell time and interstation can be extended by one time unit. The timetable still encodes the SAT formula but is now of class  $dep - dwe - int$  as at least one dwell time and one interstation time can be modified.*

$\mathcal{TT}$	$i_0$	—	$i_{6N-1}$	$i_{6N}$	$i_{6N+1}$	$i_{6N+2}$	$i_{6N+3}$
$T_0$	1		0	1	-1	0	0
.	.	SAT	0	0	0	0	0
.	.	formula	0	0	0	0	0
.	.	encoded	0	0	0	0	0
$T_M$	0		0	0	0	0	0

Table 1.3:  $(G, dep - dwe - int, lin)$  problem : example timetable encoding a SAT formula of  $N$  clauses between time 0 and  $6_{N-1}$ . Between time  $6_N$  and  $6_{N+3}$ , the dwell time of trip  $T_0$  can be lengthened by 1 time unit, as well as the last interstation duration.

### 1.3.3 Maximum Power Peak Problems

**Theorem 3.** *The decision problem associated with  $(PP, dep, lin)$  is NP-complete.*

The idea of the proof is similar to the proof 2, that is to construct a polynomial reduction of SAT to the decision problem associated with the  $(PP, dep, lin)$  problem. Here we construct a timetable such that the maximum power peak, whose value is 1, is the power demand of the special metro  $T_0$  during its acceleration phase. This special metro has  $N$  relevant acceleration phases that need to be synchronized with the braking phase of another metro. If all synchronizations are done, the maximum power peak is reduced by one power unit, from 1 to 0.

*Proof.* As for the Theorem 2 proof, we show that there is a polynomial reduction of SAT to the decision problem of reducing the power peak by one power unit on a particular  $(PP, dep, lin)$  problem. Let us consider the discrete time domain  $I = \{0, \dots, 6N + 3\}$ . Each trip  $T_t \in T$  crosses a sequence of unique stations of length  $N + 2$ . We construct a timetable and prove, using the same inductive proofs as  $(G, dep, lin)$ , that

$$\begin{aligned}
 d_{0,0} &= 0 \\
 d_{0,s} &= 6(s-1) + 3, \quad 1 \leq s \leq N \\
 a_{0,s} &= 6(s-1) + 2, \quad 1 \leq s \leq N+1 \\
 \\ 
 d_{t,s} &= 6s + \delta_t \quad 1 \leq t \leq M, \quad 0 \leq s \leq N \\
 a_{t,s} &= 6(s-1) + \delta_t + \begin{cases} 3 & \text{if } T_t \in c_s \\ 2 & \text{if } -T_t \in c_s \\ 4 & \text{otherwise} \end{cases} \quad 1 \leq t \leq M, \quad 1 \leq s \leq N \\
 a_{t,N+1} &= 6N + 2 \quad 1 \leq t \leq M
 \end{aligned}$$

Let the instant power demand function be

$$P_i = \sum_{t=0}^M P_{t,i} \quad \forall i \in I \quad (1.20)$$

and the objective function be  $PP_{\mathcal{T}\mathcal{T}}$ . Once again, the trip acceleration and braking phases last one time unit only,

$$\begin{aligned}
 d_{t,s}^+ - d_{t,s} &= 1, \quad 0 \leq t \leq M, \quad 0 \leq s \leq N, \\
 a_{t,s} - a_{t,s}^- &= 1, \quad 0 \leq t \leq M, \quad 1 \leq s \leq N+1.
 \end{aligned}$$

The trip power demands and productions are as follows

$$P_{0,i} = \begin{cases} \epsilon & \text{if } d_{0,0} = i \\ 1 & \text{if there exists } 0 \leq s \leq N \text{ s.t. } d_{0,s} = i \\ -1 & \text{if there exists } 1 \leq s \leq N+1 \text{ s.t. } a_{0,s} = i \\ 0 & \text{otherwise} \end{cases} \quad (1.21)$$

$$P_{t,i} = \begin{cases} \epsilon & \text{if there exists } 0 \leq s \leq N \text{ s.t. } d_{t,s} = i \\ -1 & \text{if there exists } 1 \leq s \leq N+1 \text{ s.t. } a_{t,s} = i \\ 0 & \text{otherwise} \end{cases} \quad 1 \leq t \leq M \quad (1.22)$$

where  $\epsilon$  is a negligible amount of power compared to 1 energy unit.

The maximum power peak  $PP_{\mathcal{T}\mathcal{T}}$  is not modified by the powers demanded by trips  $\{T_1, \dots, T_M\}$  as the power they demand is negligible compared to  $T_0$ .  $PP_{\mathcal{T}\mathcal{T}}$  thus becomes:

$$PP_{\mathcal{T}\mathcal{T}} = \max_{0 \leq i \leq 6N+3} P_{0,i} = \{0, 1\}$$

Every time one acceleration phase of  $T_0$  is synchronized with the braking phase of one trip  $T_t$ , the power peak at this instant is equal to 0. Given the period of time  $I$  and the structure of the timetable, the maximum power peak can be reduced by one power unit by synchronizing the  $N$  acceleration phases of  $T_0$  with the braking phase of another trip. Consequently, there is a timetable and a set of  $\delta_t$ ,  $1 \leq t \leq M$  that shave the  $N$  acceleration phases of  $T_0$  if and only if the set of  $\delta$  gives a valuation that satisfies  $\phi$ .

Since the SAT problem is NP-complete, the decision problem associated with  $(PP, dep, lin)$  is NP-hard, and since the latter belongs to NP (proof 1), the decision problem associated with  $(PP, dep, lin)$  is NP-complete.

**Example 3.** Let us consider the SAT formula  $(T_1 \vee T_2 \vee \neg T_3) \wedge (T_1 \vee \neg T_2 \vee T_3) \wedge (\neg T_1 \vee T_2)$ . The constructed timetable contains four trips  $T_0, T_1, T_2$  and  $T_3$ , and is divided in three main periods during which metro  $T_0$  has the same behaviour. For each acceleration phase of  $T_0$  occurring at times 3, 9 and 15, either  $T_1, T_2$  or  $T_3$  can be synchronized to save one energy unit. At each time unit, a metro is either accelerating, consuming 1 energy unit, braking, producing 1 energy unit, or coasting or dwelling, producing or consuming nothing. The formula is satisfiable if and only if for each acceleration phase of  $T_0$ , one of the other metros can synchronize their braking phase. To synchronize their braking phases with the acceleration phases of  $T_0$ , each trip departure time can be delayed by one time unit. The following timetable  $\mathcal{T}\mathcal{T}$ , represented by the energy consumed or produced by each metro at each time, encodes the formula  $(T_1 \vee T_2 \vee \neg T_3) \wedge (T_1 \vee \neg T_2 \vee T_3) \wedge (\neg T_1 \vee T_2)$ :

$\mathcal{T}\mathcal{T}$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
$T_0$	$\epsilon$	0	-1	1	0	0	0	0	-1	1	0	0	0	0	-1	1	0	0	0	0	-1	0
$T_1$	$\epsilon$	0	-1	0	0	0	$\epsilon$	0	-1	0	0	0	$\epsilon$	0	0	-1	0	0	$\epsilon$	0	-1	0
$T_2$	$\epsilon$	0	-1	0	0	0	$\epsilon$	0	0	-1	0	0	$\epsilon$	0	-1	0	0	0	$\epsilon$	0	-1	0
$T_3$	$\epsilon$	0	0	-1	0	0	$\epsilon$	0	-1	0	0	0	$\epsilon$	0	0	0	-1	0	$\epsilon$	0	-1	0

Table 1.4: Decision problem associated with  $(PP, dep, lin)$ : example timetable encoding a SAT formula of  $N$  clauses. Each cell represents the energy produced or consumed by the trip  $T_t$  at time  $i$ .

In this example, to shave the three energy peaks of  $T_0$ , one solution can be not to delay any trip. A feasible solution for this timetable would be then  $\delta_1 = 0$ ,  $\delta_2 = 0$ ,  $\delta_3 = 0$ .

**Corollary 3.** *The decision problem associated with  $(PP, dep, nonlin)$  is NP-complete.*

*Proof.* The objective function is of class *nonlin* if it exists at least one equation which is non-linear. Modifying the equation (1.20) of the previous timetable into the non-linear equation

$$P_i = \sum_{t=0}^M P_{t,i}^{(2+1)^k} \quad 0 \leq i \leq 6N + 3,$$

with  $k \in \mathbb{N}$ , does not change the result of the objective function. Indeed, for all  $i \neq 6(s-1) + 3$ ,  $1 \leq s \leq N$ ,  $\sum_{t=0}^M P_{t,i} \leq 0$ , involving that  $\sum_{t=0}^M P_{t,i}^{(2+1)^k} \leq 0$ . For all  $i = 6(s-1)+3$ ,  $1 \leq s \leq N$ ,  $\sum_{t=0}^M P_{t,i} = \{0, 1\}$  and thus  $\sum_{t=0}^M P_{t,i}^{(2+1)^k}$  does not modify the result. Finally the problem becomes  $(PP, dep, nonlin)$  without changing its resolution, thus the decision problem associated with  $(PP, dep, nonlin)$  is NP-complete.

**Corollary 4.** *The decision problem associated with  $(PP, dep - dwe - int, lin)$  is NP-complete.*

*Proof.* As in proof 2, it suffices to add an interstation trip to  $T_0$  where the dwell time and the interstation time are both modifiable to create a decision problem associated with  $(PP, dep - dwe - int, lin)$ , from  $(PP, dep, lin)$ , that encodes a SAT formula and prove that the decision problem associated with  $(PP, dep - dwe - int, lin)$  is NP-complete.

**Theorem 4.** *The problems  $(G, dep - dwe - int, lin)$ ,  $(G, dep - dwe - int, nonlin)$ ,  $(PP, dep - dwe - int, lin)$  and  $(PP, dep - dwe - int, nonlin)$  are NP-hard.*

*Proof.* Since the decision problems associated with  $(G, dep - dwe - int, lin)$ ,  $(G, dep - dwe - int, nonlin)$ ,  $(PP, dep - dwe - int, lin)$  and  $(PP, dep - dwe - int, nonlin)$  are NP-complete, thus the problems  $(G, dep - dwe - int, lin)$ ,  $(G, dep - dwe - int, nonlin)$ ,  $(PP, dep - dwe - int, lin)$  and  $(PP, dep - dwe - int, nonlin)$  are NP-hard.



# Chapter 2

## Instant Power Demand

### Contents

---

<b>2.1</b>	<b>Electrical Simulator</b>	<b>40</b>
2.1.1	Network Parameters	40
2.1.2	Timetable Parameters	41
2.1.3	Electric Variables	41
2.1.4	Constraints and Instant Power Demand Value	42
<b>2.2</b>	<b>Instant Power Demand Approximations</b>	<b>43</b>
2.2.1	Power Flow and Non-Linear Approximation	44
2.2.2	Power Flow Algorithm	45
2.2.3	Linear Approximation for MILP	47

---

### Introduction

The complexity of the metro line electrical network, the non-linearity of the electrical equations and the dynamic interactions between the network and the running metros make impossible to solve analytically the electrical behaviour of the line [39]. Usually, the energy consumption of a metro line is computed with an electrical simulator. To be realistic, the simulator must take into account both the energy consumed by accelerating metros and the energy produced by regenerative braking, and how they affect the energy supplied by electric substations [40]. We propose in this chapter a simplified electrical simulator to compute the instant power demands  $P_i$ . We then present an approximation based on a power flow model.

## 2.1 Electrical Simulator

The electrical circuit is composed of the  $N$  stations, the electrical substations (ESS) and the accelerating and braking trains at a given time.

The ESSs are electrical devices that convert the AC power provided by the electricity provider to DC power, directly usable by metros. We assume that they are not revertible, i.e. the regenerative braking can only be used by accelerating metros, and that they are directly connected to the stations. Part of the power provided by the ESSs is lost by Joule effects on the DC network.

Each station is connected to the next one by a resistive cable. The braking metros are connected to the station to which they arrive and are modelled as an ideal power source. The accelerating metros are connected to the station from which they depart and are modelled as an ideal power sink. Figure 2.1 depicts the circuit associated to a small network with five stations, three ESSs, one braking metro and one accelerating metro.

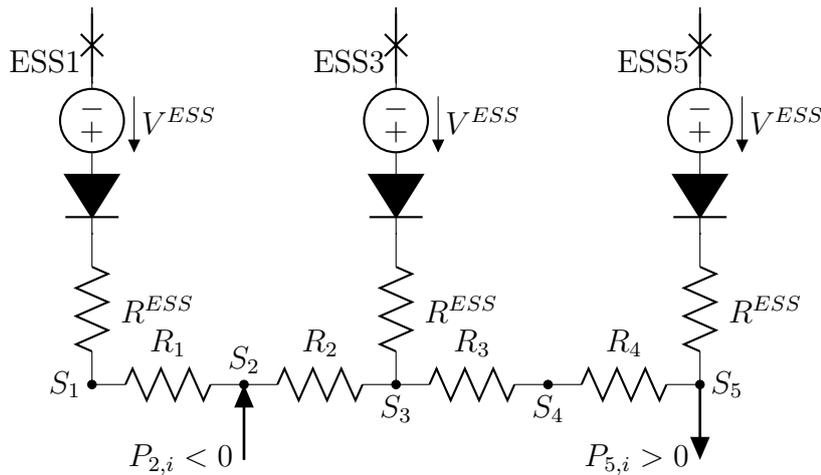


Figure 2.1: Electrical circuit associated to a metro line with five stations at time  $i$ . It is composed of three electric substations represented as ideal voltage sources in  $S_1$ ,  $S_3$  and  $S_5$ , a braking metro arriving in  $S_2$  and producing  $P_{2,i}$ , and an accelerating metro departing from  $S_5$  and consuming  $P_{5,i}$ . The points in the network are linked by resistive cables.

### 2.1.1 Network Parameters

The electrical properties of the metro line are fixed and given by the following set of parameters valid at any time:

- $V^{ESS} \in \mathbb{R}^+$  is the fixed voltage supplied by the ESSs to the line. Typical values are 750V [22] or 1500V [30].

- $R^{ESS} \in \mathbb{R}^+$  is the value of the internal resistance of the ESSs.
- $R_s \in \mathbb{R}^+$  is the resistance of the electric cable of the metro network between the stations  $S_s$  and  $S_{s+1}$ , for  $1 \leq s \leq N - 1$ . This is computed using the linear resistance equation  $R = \frac{\rho l}{a}$ ,  $\rho$  being the resistivity of the third rail,  $a$  its section and  $l$  its length.

## 2.1.2 Timetable Parameters

$P_{s,i} \in \mathbb{R}$  is the net power demand or production, generated by a metro, at time  $i$  and at station  $S_s$ . This value is different from 0 only if there is effectively a metro either braking or accelerating near the station at this particular time. These values are modified according to the acceleration and braking phases of the metros at each time point. We have

$$P_{s,i} \begin{cases} > 0 & \text{if metro accelerating near } S_s \\ < 0 & \text{if metro braking near } S_s \\ = 0 & \text{otherwise} \end{cases} \quad 1 \leq s \leq N, i \in I$$

The precise power values are supposed to be known beforehand by direct measurement on the metro motors. Figure 2.2 illustrates the net power demand and production of a metro during an interstation run. In a typical example depicted in Figure 2.2, the parameters  $P_{s,i}$  are given by the power curve for each time point  $i$ . The curve is then sampled and compiled in an energy profile table (Table 2.1).

## 2.1.3 Electric Variables

The energy transfers are modelled by the following variables. By convention, all voltages are positive and the currents can be negative.

- $v_{s,i} \in \mathbb{R}^+$  is the electric potential at station  $S_s$ , for  $1 \leq s \leq N$ , at time  $i \in I$ .
- $i_{s,i} \in \mathbb{R}$  is the current flowing through the cable between stations  $S_s$  and  $S_{s+1}$ , for  $1 \leq s \leq N - 1$ , at time  $i \in I$ .
- $i_{s,i}^{ESS} \in \mathbb{R}$  is the current flowing from the electrical substation connected to the station  $S_s$ , for  $1 \leq s \leq N$ , at time  $i \in I$ .
- $i_{s,i}^{MET} \in \mathbb{R}$  is the current flowing between the network and the metro located in  $S_s$ , for  $1 \leq s \leq N$ , at time  $i \in I$ .

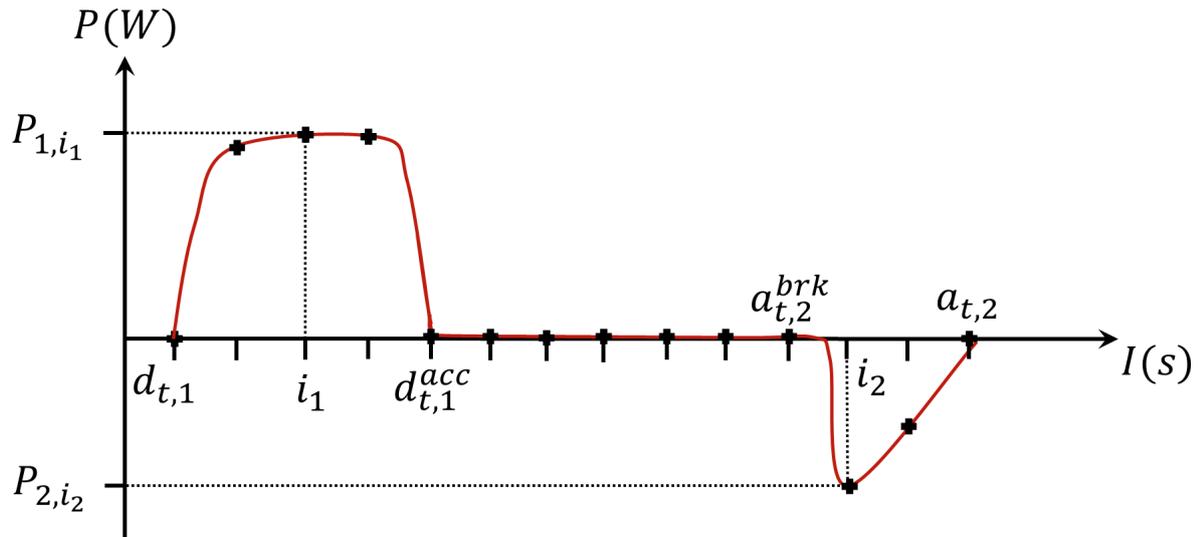


Figure 2.2: Net power demand and production curve as a function of real time for a metro accelerating and braking between two stations. The points on the curve represent the sampling over discrete time.

Timeslot	Net Power Demand	Comments
0	0	dwel
1	0.25	traction
2	0.75	
3	1	
4	1	
...	...	
15	1	
16	0	coasting
17	0	
...	...	
59	0	
60	-0.4	braking
61	-0.35	
62	-0.3	
...	...	
70	0	dwel

Table 2.1: Energy profile data for an interstation run. The net power demand is given in an arbitrary unit equal to 1 when the metro is at full throttle, for each timeslot of the interstation.

### 2.1.4 Constraints and Instant Power Demand Value

The following equations constrain the current and voltage at each metro station. Ohm's law gives

$$v_{s,i} - v_{s+1,i} = R_s \cdot i_{s,i} \quad 1 \leq s \leq N-1, i \in I \quad (2.1)$$

$$V_{ESS} - v_{s,i} = R_{ESS} \cdot i_{s,i}^{ESS} \quad 1 \leq s \leq N, i \in I \quad (2.2)$$

Kirchhoff's current law gives

$$i_{s,i} + i_{s+1,i} + i_{s,i}^{ESS} + i_{s,i}^{MET} = 0 \quad 1 \leq s \leq N, i \in I \quad (2.3)$$

The satisfaction of the instant power gives rise to a *non-linear equation*:

$$P_{s,i} = v_{s,i} \cdot i_{s,i}^{MET} \quad 1 \leq s \leq N, i \in I \quad (2.4)$$

It is worth remarking that the instant power demand of a metro line  $P_i$  is not equal to the sum of the net instant power demands of the metros  $\sum_{s=1}^N P_{s,i}$ , but to the power supplied by the electrical substations over the line to fulfil the metro power demands:

$$P_i = \sum_{s=1}^N V^{ESS} \cdot \max(0, i_{s,i}^{ESS}) \quad i \in I$$

This value represents the *net power consumption* of the metro line. The currents  $i_s^{ESS}$  flowing through each ESS can be negative, i.e. can flow backwards from the line to the grid, but this *negative power* is not counted in the instant power consumption. Indeed, ESSs possess a rectifier that works as a diode and forces the current to flow only in one direction. In reality, if an electrical substation receives energy, typically when too many metros are braking and none is accelerating, the energy is absorbed by resistors that are placed on the line or on the metros brakes.

## 2.2 Instant Power Demand Approximations

To simplify the evaluation of the power demand, some contributions in the literature have made the choice of directly computing the power transfers between braking and accelerating metros instead of calculating voltages and intensities, and deducing the power demand from it [32, 16, 30, 21, 12]. We introduce in this section the notion of *power flow network*, which is a particular case of a *generalized flow network*, to model these power transfers. The idea is that setting the flow along the paths of the power flow network is an approximation of the power transfers between braking and accelerating metros, the flow arriving in the sink of the graph representing the power saved by regenerative braking reuse.

Both the electrical simulator and the power flow, model the fact that one accelerating metro can benefit from the regenerative energy of several braking metros and that one braking metro can feed several accelerating metros. The power flow approximation does not intend to reproduce exactly the electrical behaviour of the metro line but rather to

give a fast evaluation of it for the optimization algorithm.

The *distribution matrix*  $\Delta$  is the matrix of *distribution ratios*  $\Delta_{s,s'} = (P - P_{s'}) / (P_s) \in [0, 1]$  between each stations  $S_s$  and  $S_{s'}$ , with  $1 \leq s' < s \leq N$ , where  $P_{s'}$  is the positive net power demand of a metro accelerating in station  $S_{s'}$ ,  $P_s$  the negative net power production of a metro braking in station  $S_s$ , and  $P$  the resulting power demand of the electrical network as computed by the electrical simulator. The distribution ratio represents the ratio of power a metro braking in station  $S_s$  effectively transfers to a metro accelerating in station  $S_{s'}$ . As  $P_{s'} \geq P$ , a value of 1 means that the energy is fully transferred and a value of 0 means that the energy is completely lost in resistors as a consequence of Joule effects.

### 2.2.1 Power Flow and Non-Linear Approximation

A generalized flow network is a finite directed graph  $G(V, E)$  given with capacities  $c(u, v)$  on edges in  $E$  and a flow  $f(u, v) \leq c(u, v)$ . The graph is given in addition with positive gains  $\gamma(u, v)$  such that, if a flow  $f(u, v)$  is entering at vertex  $v$ , then  $\gamma(u, v) \cdot f(u, v)$  is going out from  $v$ :

$$\sum_{v \in V | (u,v) \in E} \gamma(u, v) f(u, v) = \sum_{v \in V | (v,u) \in E} f(v, u), \quad (2.5)$$

Two vertices in  $V$  are distinguished, the source  $t$  which can produce flow and the sink  $t'$  which can absorb flow.

For each time  $i \in I$ , the *power flow network* is the generalized flow network defined by

$$\begin{aligned} V &= \{t, t'\} \cup V_i^{brk} \cup V_i^{acc}, \\ V_i^{brk} &= \{s, 1 \leq s \leq N | P_{s,i} < 0\}, \quad i \in I, \\ V_i^{acc} &= \{s', 1 \leq s' \leq N | P_{s',i} > 0\}, \quad i \in I, \\ E &= \{(t, s)\} \cup \{(s, s')\} \cup \{(s', t')\} \quad s \in V_i^{brk}, s' \in V_i^{acc} \end{aligned}$$

and

$$\begin{aligned} c(t, s)_i &= P_{s,i} & \gamma(t, s)_i &= 1 \\ c(s, s')_i &= +\infty & \gamma(s, s')_i &= \Delta_{s,s'} \\ c(s', t')_i &= P_{s',i} & \gamma(s', t')_i &= 1 \end{aligned}$$

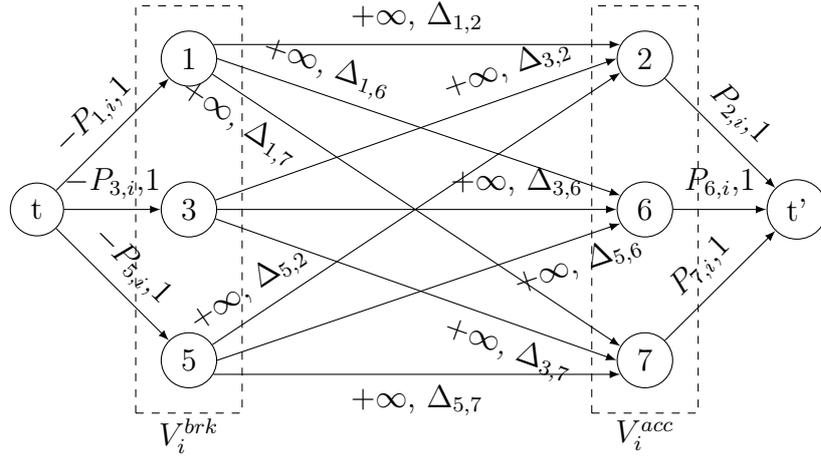


Figure 2.3: Example of a generalized flow network. Flows are created by the source  $t$  and absorbed by the sink  $t'$ . The left hand side of the graph contains the vertices corresponding to the braking metros, which are connected to the vertices corresponding to the accelerating metros on the right hand side. Each edge of the graph is characterized by a (capacity,gain) couple.

The instant power demand approximation  $P_i$  can then be defined as:

$$P_i = \sum_{s' \in V_i^{acc}} [c(s', t')_i - f(s', t')_i] \quad (2.6)$$

Figure 2.3 shows an example of a power flow network modelling power transfers between 3 metros braking in stations  $S_1$ ,  $S_3$  and  $S_5$ , and 3 metros accelerating in stations  $S_2$ ,  $S_6$  and  $S_7$ . The flows are attenuated between braking and accelerating metros according to their distribution ratio  $\Delta_{s,s'}$ . The flows are not bound and can be set freely between braking and accelerating metros. However, they are bound by the powers produced or demanded at the source or to the sink. The flows effectively arriving to the sink represent the regenerative power that has been saved, while the sum of the capacities' edges arriving to the sink represent the total power demanded by the accelerating metros. Subtracting these two values gives an approximation of the instant power demand.

## 2.2.2 Power Flow Algorithm

According to the equation (2.6), the power demand equation of the power flow is equal to the capacities minus the flows directed to the sink. We have:

$$c(s', t')_i = P_{s',i}, \quad s' \in V_i^{acc}$$

and, according to the flow conservation equation (2.5)

$$f(s', t')_i = \sum_{s \in V_i^{brk}} \gamma(s, s')_i \cdot f(s, s')_i \text{ with } \gamma(s, s')_i = \Delta_{s, s'}, \quad s \in V_i^{brk}, \quad s' \in V_i^{acc}$$

We can also reformulate the flow  $f(s, s')_i$  as the ratio of power transferred by the metro braking at station  $S_s$  multiplied by the power  $P_{s, i}$ :

$$f(s, s')_i = -x_{s, s', i} \cdot P_{s, i}, \quad s \in V_i^{brk}, \quad s' \in V_i^{acc}$$

The *power transfer ratio*  $x_{s, s', i} \in [0, 1]$  is the ratio of the power  $P_{s, i}$ , with  $s \in V_i^{brk}$ , transferred from the metro braking at station  $S_s$  to the metro accelerating at station  $S_{s'}$  such that  $x_{s, s', i} = -f(s, s')_i / P_{s, i}$ .

The following power flow algorithm computes the power transfer ratios by transferring the produced power of each braking metro in priority to the accelerating metro whose distribution ratio is maximum, until all the produced power is transferred. The algorithm returns the power transfer ratios  $x_{s, s', i}$ , either when all braking metros have transferred their power or when all accelerating metros have their demand fulfilled:

---

**Algorithm 1** Power transfer ratios computation at time  $i$

---

**Require:**  $V_i^{acc}$ ,  $V_i^{brk}$ ,  $\Delta_{s, s'}$ ,  $P_{s, i}$

- 1: Initialize vector  $x_{s, s', i} \leftarrow 0$
  - 2: **while**  $V_i^{brk} \neq \emptyset$  **do**
  - 3:     Choose randomly  $s \in V_i^{brk}$
  - 4:      $P^{INIT} \leftarrow P_{s, i}$
  - 5:     **while**  $P_{s, i} < 0$  **do**
  - 6:         **if**  $V_i^{acc} \neq \emptyset$  **then**
  - 7:             Choose  $s' \in V_i^{acc}$  s.t.  $s' = \arg \max_{s' \in V_i^{acc}} (\Delta_{s, s'})$
  - 8:             **if**  $-P_{s, i} \cdot \Delta_{s, s'} > P_{s', i}$  **then**
  - 9:                  $x_{s, s', i} \leftarrow (P_{s', i} / \Delta_{s, s'}) / P^{INIT}$
  - 10:                  $P_{s, i} \leftarrow P_{s, i} + P_{s', i} / \Delta_{s, s'}$
  - 11:                  $V_i^{acc} \leftarrow V_i^{acc} \setminus \{s'\}$
  - 12:             **else**
  - 13:                  $x_{s, s', i} \leftarrow P_{s, i} / P^{INIT}$
  - 14:                  $P_{s', i} \leftarrow P_{s', i} + P_{s, i} \cdot \Delta_{s, s'}$
  - 15:                  $P_{s, i} \leftarrow 0$
  - 16:             **end if**
  - 17:         **else**
  - 18:              $P_{s, i} \leftarrow 0$
  - 19:         **end if**
  - 20:     **end while**
  - 21:      $V_i^{brk} \leftarrow V_i^{brk} \setminus \{s\}$
  - 22: **end while**
  - 23: **return** vector  $x_{s, s', i}$
-

The instant power demand  $P_i$  can now be reformulated with the non-linear equation:

$$P_i = \sum_{s' \in V_i^{acc}} P_{s',i} + \sum_{s' \in V_i^{acc}} \sum_{s \in V_i^{brk}} (P_{s,i} \cdot x_{s,s',i} \cdot \Delta_{s,s'}), \quad (2.7)$$

### 2.2.3 Linear Approximation for MILP

Linear Programming (LP) is a branch of mathematical programming which addresses the problem of optimizing a linear objective function, subject to linear equality and linear inequality constraints. Linear programs can be applied to various fields of study and are easily solvable thanks to different resolution algorithms like the simplex algorithm developed by George Dantzig in 1947.

Mixed integer linear programming (MILP) involves linear programs in which some of the variables are restricted to be integers, while other variables are allowed to be non-integers. Unlike LP, MILP is usually NP-hard and one of the most efficient software to solve MILP problems is CPLEX, developed by ILOG [17]. For a complete study on LP and MILP, the reader can refer to [41] and [18], or [19] to put into perspective linear programming with other operations research techniques. For a short tutorial on MILP techniques, please refer to [20].

To avoid introducing too many variables in MILP, Ramos and Peña proposed in [29] and [30] a linear approximation of the power flow, which tends to maximize the overlapping times between acceleration and braking phases. The formulation of the power flow is simplified by authorizing only one single transfer between one braking metro and one accelerating metro. The objective function is then the sum, weighted by the distribution matrix, of the overlapping times of these transfers.

The MILP model introduces two new variables to describe the overlaps between metros :

- $\gamma_{t,s,t',s'} \in \{0, 1\}$  is a boolean variable equal to one if the trip  $T_t$ ,  $1 \leq t \leq M$  braking in station  $S_t(s)$ ,  $2 \leq s \leq N$ , transfers its power to the trip  $T_{t'}$ ,  $1 \leq t' \leq M$  accelerating in  $S_{t'}(s')$ ,  $1 \leq s' \leq N - 1$ . Otherwise it is equal to 0.
- $O_{t,s,t',s'} \in \mathbb{R}$  is the *overlapping time* between the braking phase of the trip  $T_t$ ,  $1 \leq t \leq M$  at station  $S_t(s)$ ,  $2 \leq s \leq N$ , and the acceleration phase of the trip  $T_{t'}$ ,  $1 \leq t' \leq M$  at station  $S_{t'}(s')$ ,  $1 \leq s' \leq N - 1$ .

Constraint

$$\sum_{t=1}^M \sum_{s=1}^N \gamma_{t,s,t',s'} \leq 1 \quad 1 \leq t' \leq M, 1 \leq s' \leq N \quad (2.8)$$

ensures that each accelerating metro receives the totality of the power of at most one braking metro, and constraint

$$\sum_{t'=1}^M \sum_{s'=1}^N \gamma_{t,s,t',s'} \leq 1 \quad 1 \leq t \leq M, 1 \leq s \leq N \quad (2.9)$$

ensures that each braking metro transfers the totality of its power to *at most one* accelerating metro, effectively modelling the fact that metros are only authorized to do one single pairing to transfer their power.

Constraint

$$O_{t,s,t',s'} \leq brk_{t,s} \cdot \gamma_{t,s,t',s'} \quad 1 \leq t < t' \leq M, 2 \leq s \leq N, 1 \leq s' \leq N - 1 \quad (2.10)$$

ensures that the overlapping time of a braking phase with an acceleration phase cannot be bigger than the braking phase time  $brk_{t,s}$ .

Constraints

$$O_{t,s,t',s'} \leq d_{t',s'}^{acc} - a_{t,s}^{brk} + m(1 - \gamma_{t,s,t',s'}) \quad (2.11)$$

$$O_{t,s,t',s'} \leq a_{t,s} - d_{t',s'} + m(1 - \gamma_{t,s,t',s'}) \quad (2.12)$$

$$1 \leq t < t' \leq M, 2 \leq s \leq N, 1 \leq s' \leq N - 1$$

ensure that the overlapping time is the minimum value of  $d_{v,u}^{acc} - a_{t,s}^{brk}$  and  $a_{t,s} - d_{v,u}$  when  $\gamma_{t,s,v,u} = 1$ . The member  $m(1 - \gamma_{t,s,v,u})$ , with  $m$  a big enough number, ensures that  $O$  is never negative.

The MILP objective function used in [29, 30]

$$\text{maximize} \sum_{t=1}^M \sum_{s=1}^N \sum_{t'=1}^M \sum_{s'=1}^N (O_{t,s,t',s'} \cdot \Delta_{t,s,t',s'}) \quad (2.13)$$

where  $\Delta_{t,s,t',s'}$  is equal to  $\Delta_{s'',s'''}$  such that  $S_{s''} = S_t(s)$  and  $S_{s'''} = S_{t'}(s')$ , is the sum of all overlapping times of the timetables, weighted by the distribution matrix. The weights tend to synchronize braking and accelerations that are close to each other.

Concerning the size of the generated MILP instances, the model contains:

- $M^2 \cdot N^2$  boolean variables  $\gamma$ ,  $M^2 \cdot N^2$  overlapping times variables  $O$  and  $M \cdot N$  dwell times variables  $dwe$ ,
- $3M^2 \cdot N^2 + 2M \cdot N$  MILP constraints (Equations 2.8, 2.9, 2.10, 2.11, 2.12) and  $5M \cdot N - 3M - 2N$  timetable constraints (Equations 1.1, 1.2, 1.3, 1.4, 1.5, 1.6,

1.7).

The MILP model thus introduces a quadratic number of variables and constraints in the number of trips and stations. A pre-processing have been proposed in [29, 30] to remove irrelevant constraints and variables:

The *no possible overlap* cut

$$\exists O_{t,s,t',s'}, \gamma_{t,s,t',s'} \Leftrightarrow (d_{t,1} < a_{t',N}) \wedge (d_{t',1} < a_{t,N})$$

prevents considering the variables and constraints that imply the synchronizations between trips that do not share a common time window in the timetable.

And the *no possible transfer* cut

$$\exists O_{t,s,t',s'}, \gamma_{t,s,t',s'} \Leftrightarrow \Delta_{t,s,t',s'} > 0$$

prevents considering the variables and constraints that imply a power transfer between two trips that are too remote from each other. As their distribution ratio is equal to 0, removing these variables and constraints does not modify the output solution. Anyway, even this pre-processing is not sufficient to handle real data timetables, as we will show in Section 5.3.



# Chapter 3

## Metro Energy Optimization Timetabling

### Contents

---

<b>3.1</b>	<b>Timetable Physical Feasibility . . . . .</b>	<b>52</b>
3.1.1	Initial Timetable Parameters . . . . .	53
3.1.2	Metro Parameters . . . . .	53
3.1.3	Metro Constraints . . . . .	53
<b>3.2</b>	<b>Metro Energy Optimization Timetabling Model . . . . .</b>	<b>54</b>
3.2.1	Input Timetable Parameters . . . . .	55
3.2.2	Headway Pattern Parameters . . . . .	56
3.2.3	Headway Pattern Constraints . . . . .	56
3.2.4	Trip Allocation Algorithm . . . . .	58
<b>3.3</b>	<b>Computational Results . . . . .</b>	<b>59</b>
3.3.1	Headway Pattern Timetabling . . . . .	60
3.3.2	Energy Optimized Headway Pattern Timetabling . . . . .	61

---

### Introduction

Since the first attempt to solve the train timetabling problem with optimization techniques in 1971 by Amit *et al.* [42], research has been very active on this subject and many timetable models and optimization techniques to solve them have been proposed. The introduction of the Periodic Event Scheduling Problem (PESP) by Serafini *et al.* [37] created a mathematical basis in 1989 of the metro timetabling problem, considering

it as a cyclic timetabling problem. Later on, further refinements by Liebchen [38] on the same problem allowed him to put in service in 2005 on the Berlin subway the first timetable resulting from a mathematical programming technique [43].

Many variants of the problem have been studied, including sophistications and different objective functions. Bampas *et al.* [44] proposed the Periodic Metro Scheduling problem (PMS), minimizing the maximum headway of the timetable. Nachtigall [45] or Cury *et al.* [46] proposed timetable models optimizing the passenger waiting times in stations and the rolling stock usage. Higgins *et al.* [47] proposed a single-track timetabling model minimizing the train waiting times by optimizing the overtaking manoeuvres along the line.

However, very few studies have included in their objective function an evaluation of the global energy consumption of the metro line. Bodhibrata *et al.* [48] added an energy model, aiming to minimize the metros' speed profiles, but did not take into account energy transfers between braking and accelerating metros. Recently, the works of Yang *et al.* [34, 35] or Li and Lo [31] represent a first attempt to tackle this problem, minimizing the global energy consumption and the passenger waiting time.

Based on the master's thesis work of Jing Yang [49], we propose in this chapter an extension of the metro energy optimization rescheduling model presented in Chapter 1, by creating from scratch a timetable, allocating trips to physical metros and creating trips in a way they comply with the metro line requirements. This timetabling model minimizes the global energy consumption, while following a specific headway pattern.

### 3.1 Timetable Physical Feasibility

In this section, we propose a purely linear model to check the *physical feasibility* of a given timetable, i.e. its capacity to satisfy the topological constraints of the physical metro line. In particular, the timetable should respect a minimal safety headway, trips should not overtake each other, and there should be enough time for the metros to perform their turn back manoeuvres in terminals.

### 3.1.1 Initial Timetable Parameters

The trips constituting the timetable are entirely defined in the initial timetable. Thus, all the general timetable bounds are shrunk to the single values:

$$\begin{aligned}
 \underline{dep}_t &= \overline{dep}_t & 1 \leq t \leq M \\
 \underline{int}_{t,s} &= \overline{int}_{t,s} & 1 \leq t \leq M, 1 \leq s \leq N-1 \\
 \underline{dwe}_{t,s} &= \overline{dwe}_{t,s} & 1 \leq t \leq M, 2 \leq s \leq N-1 \\
 \underline{acc}_{t,s} &= \overline{acc}_{t,s} & 1 \leq t \leq M, 1 \leq s \leq N-1 \\
 \underline{brk}_{t,s} &= \overline{brk}_{t,s} & 1 \leq t \leq M, 2 \leq s \leq N
 \end{aligned}$$

The trip times and headways are also directly computed with the above constraints.

### 3.1.2 Metro Parameters

As mentioned in Section 1.1, we assumed in our timetable model that a physical metro has been allocated to each trip beforehand. The initial timetable is thus given with a set of parameters that permits to link the virtual trips with physical metros:

$rst_t \in \mathbb{N}^*$	specific metro number, or <i>rolling stock</i> , performing the trip $T_t$ .
$next_t \in [1, M]$	trip $T_{t'}$ performed right after the trip $T_t$ by the same rolling stock, $next_t = t'$ s.t. $(rst_t = rst_{t'} \wedge d_{t,1} < d_{t',1})$ $\wedge (\nexists t''$ s.t. $rst_t = rst_{t''} \wedge d_{t,1} < d_{t'',1} < d_{t',1})$
$mhw \in I$	minimal headway to respect all over the line to ensure safety.
$tbm \in I$	time for a metro to make a U-turn in its arrival terminal to perform its next trip in the other direction, also called <i>turn back manoeuvre</i> duration.
$cap \in \mathbb{N}^*$	maximum number of metros that can wait for departure in each terminal at the same moment, or terminal capacity.

### 3.1.3 Metro Constraints

The four following constraints ensure that a given initial timetable is physically feasible given a metro line topology:

- No-crash constraint

$$a_{t,s} - d_{t-1,s} > 0, \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], \quad 1 \leq s \leq N \quad (3.1)$$

ensures that two consecutive trips are not dwelling at the same station at the same

moment.

- Minimal headway constraint

$$d_{t,s} - d_{t-1,s} \geq mhw, \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], \quad 1 \leq s \leq N \quad (3.2)$$

ensures that two consecutive trips are separated by a minimal safety headway all along the line.

- Turn back constraint

$$d_{nxt_t,1} - a_{t,N} \geq tbm, \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], \quad nxt_t \text{ defined} \quad (3.3)$$

ensures that a rolling stock has the necessary minimal time to turn back in terminal to perform its next trip.

- Terminal capacity constraint

$$a_{t,N} - d_{t-cap-1,1} \geq 0, \quad t \in [2 + cap, \frac{M}{2}] \cup [\frac{M}{2} + 2 + cap, M] \quad (3.4)$$

ensures that there is never more than  $cap$  metros waiting in each terminal.

Proving the satisfiability of the pure LP model consisting of the general timetable model and of the physical constraints described in this section will prove the physical feasibility of a given timetable. The timetable can be then displayed as a time/space graph. By convention, the vertical axis represents the time and the horizontal axis the metro line space. The leftmost and rightmost points represent the two terminals and each coloured line is a specific train travelling back and forth along the line.

Figure 3.1 highlights a specific train and shows its trip time and its turn back manoeuvres duration. Figure 3.2 highlights the rightmost terminal capacity, showing that no more than 2 trains can wait inside the terminal at the same moment. Also, due to the turn back constraints and because there was no other metro waiting in the terminal, the timetable shows that it had to add a new metro – the brown line on the graph – to comply with the schedule. In practice, the metro is taken from the depot and added to the metro line.

## 3.2 Metro Energy Optimization Timetabling Model

We propose in this section some additional constraints for the metro energy optimization rescheduling model to formalize timetabling model. A timetabling model is a model

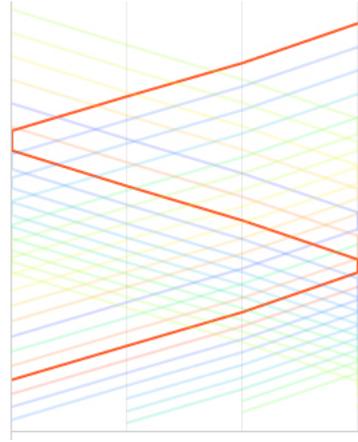


Figure 3.1: Highlighted specific metro running back and forth and stopping in terminals : metro line graph with the time on the vertical axis and the line space on the horizontal axis. The leftmost and rightmost points represent both terminals. Each coloured line represents a physical train running between both terminals.

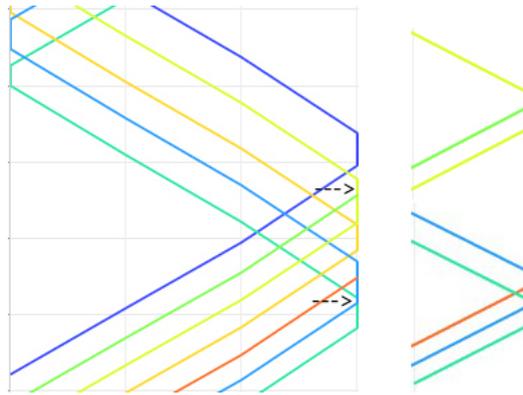


Figure 3.2: Terminal capacity focus : metro line graph with the time on the vertical axis and the line space on the horizontal axis. The leftmost and rightmost points represent both terminals. Each coloured line represents a physical train running between both terminals.

capable to generate an entire timetable from scratch, using input parameters given by the customer, namely the speed and dwell profiles, the first departure times of the day, and a headway pattern. Adding the instant power demand linear approximation to this timetabling model of Section 2.2.3 enables us to produce a unified MILP model for the creation of an energy optimized timetable from scratch.

### 3.2.1 Input Timetable Parameters

To compute the timetable from scratch, we assume that the metro running profiles have been set beforehand, so the way metro run, accelerate, brake or dwell is optimized and

is known in advance:

$$\begin{aligned} \underline{int}_{t,s} &= \overline{int}_{t,s} & 1 \leq t \leq M, 1 \leq s \leq N-1 \\ \underline{dwe}_{t,s} &= \overline{dwe}_{t,s} & 1 \leq t \leq M, 2 \leq s \leq N-1 \\ \underline{acc}_{t,s} &= \overline{acc}_{t,s} & 1 \leq t \leq M, 1 \leq s \leq N-1 \\ \underline{brk}_{t,s} &= \overline{brk}_{t,s} & 1 \leq t \leq M, 2 \leq s \leq N \end{aligned}$$

Finally, what will be computed by the timetabling model are the departure times of each metro from their terminal. Only the departure times of the first metros of the day are given:

- $dep_{UP}$  departure time of the first metro running upstream.
- $dep_{DOWN}$  departure time of the first metro running downstream.

### 3.2.2 Headway Pattern Parameters

As metros follow each other at a relatively high frequency – often more than every 6 minutes –, passengers are more aware of the headways, i.e. the frequency of metros at a station, than the schedule itself. Thus, the metro line provider is likely to give as an input parameter for its quality of service a *headway pattern*. The headway pattern is the headway that should follow the metros in function of the hour of the day. For example, a typical weekday will have two peak hours – one in the morning and one in the evening – when the headways are shorter than the rest of the timetable. The following additional parameters set the headway pattern, in function of the number of intervals of this pattern:

- $nit \in \mathbb{N}^*$  number of headway intervals.
- $tra_h \in I$  starting time of interval  $h$ , with  $1 \leq h \leq nit$ ,  
or ending time of interval  $N$  for  $h = nit + 1$ .
- $low_h \in I$  headway lower bound for the headway interval  $h$ , with  $1 \leq h \leq nit$ .
- $upp_h \in I$  headway upper bound for the headway interval  $h$ , with  $1 \leq h \leq nit$ ,  
verifying  $upp_h \geq low_h$ .

### 3.2.3 Headway Pattern Constraints

The two first trips are constrained by the given departure times

$$\begin{aligned} d_{1,1} &= dep_{UP} \\ d_{\frac{M}{2}+1,1} &= dep_{DOWN} \end{aligned}$$

The timetabling model must set for each trip in which interval of the headway pattern it belongs:  $b_{t,h} \in \{0, 1\}$  is a boolean variable equal to one if the trip  $T_t$ , with  $1 \leq t \leq M$ , starts after  $tra_h$ , with  $1 \leq h \leq nit + 1$ . Unlike pure LP models, MILP models support if-then-else conditions on variables. Adding a boolean variable for the condition, it is possible to re write it using inequalities:

$$b = (x > y)? \Leftrightarrow \begin{cases} x \leq y + m.b \\ y < x + m.(1 - b) \end{cases}$$

with  $b \in \{0, 1\}$  a boolean variable,  $x$  and  $y$  two free variables and  $m$  a big enough number, greater than the maximum values  $x$  and  $y$  can get.

- If  $x > y$ , then  $x \not\leq y + m \times 0$ , but  $x \leq y + m \times 1$ ; the first inequality forces  $b = 1$ .
- If  $y \geq x$ , then  $y \not< x + m \times (1 - 1)$ , but  $y < x + m \times (1 - 0)$ ; the second inequality forces  $b = 0$

Now, the if-then-else condition on  $b_{t,h}$  is linearized as

$$b_{t,h} = (d_{t,1} > tra_h)? \Leftrightarrow \begin{cases} d_{t,1} \leq tra_h + m_1.b_{t,h} \\ tra_h \leq d_{t,1} + m_1.(1 - b_{t,h}) \end{cases} \quad 1 \leq t \leq M, 1 \leq h \leq nit + 1 \quad (3.5)$$

where  $m_1 > I_{END}$  is a big enough number.

Given  $b_{t,h}$ , the lower and upper bounds on headways for each trip are constrained as follows:

- Lower bound constraint

$$\underline{hdw}_{t,s} \geq low_h.b_{t,h} - m_2.b_{t,h+1}, \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], 1 \leq s \leq N, 1 \leq h \leq nit \quad (3.6)$$

where  $m_2 > \max_{1 \leq h \leq nit} low_h$  is a big enough number, ensures that  $\underline{hdw}_{t,s} = low_h$ , where  $h$  is the interval the trip starts in. Indeed, the following table shows that the maximum value the expression  $low_h.b_{t,h} - m_2.b_{t,h+1}$  can get is the case where  $b_{t,h} = 1$  and  $b_{t,h+1} = 0$ . This case represent the interval when the trip has started after  $tra_h$  and before  $tra_{h+1}$ , meaning that the trip has started in the interval  $[tra_h, tra_{h+1}]$ . Constraint 3.6 enforces that all other possible cases give looser constraints, i.e. 0 or  $< 0$ .

$\underline{hdw}_{t,s}$	$b_{t,h} = 0$	$b_{t,h} = 1$
$b_{t,h+1} = 0$	0	$low_h$
$b_{t,h+1} = 1$	–	$< 0$

- Upper bound constraint

$$\overline{hdw}_{t,s} \leq upp_h \cdot (b_{t,h} - b_{t,h+1}) + m_3 \cdot (1 + b_{t,h+1} - b_{t,h}), \quad (3.7)$$

$$t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], \quad 1 \leq s \leq N, \quad 1 \leq h \leq nit$$

where  $m_3 > \max_{1 \leq h \leq nit} upp_h$  is a big enough number, ensures that  $\overline{hdw}_{t,s} = upp_h$ , where  $h$  is the interval the trip starts in. Like for the lower bound constraint, the following table shows that the minimum value the expression  $upp_h \cdot (b_{t,h} - b_{t,h+1}) + m_3 \cdot (1 + b_{t,h+1} - b_{t,h})$  can get is  $upp_h$  when  $b_{t,h} = 1$  and  $b_{t,h+1} = 0$ .

$\overline{hdw}_{t,s}$	$b_{t,h} = 0$	$b_{t,h} = 1$
$b_{t,h+1} = 0$	$m_3$	$upp_h$
$b_{t,h+1} = 1$	–	$m_3$

### 3.2.4 Trip Allocation Algorithm

The above MILP model compiles all trips' terminal departure times but does not link physical metros to these trips. Given the list of the departure times from the departure terminal  $d_{t,1}$  and the arrival times to the arrival terminal  $a_{t,N}$ , we can create a list of *tasks* as follows:

$$\text{task} = \left\{ \begin{array}{l} \text{trip} = t, \\ \text{type} = \begin{cases} \text{departure if } d_{t,1} \\ \text{arrival if } a_{t,N} \end{cases}, \\ \text{terminal} = \begin{cases} 1 \text{ if } (d_{t,1} \wedge 1 \leq t \leq \frac{M}{2}) \vee (a_{t,N} \wedge \frac{M}{2} + 1 \leq t \leq M) \\ N \text{ if } (d_{t,1} \wedge \frac{M}{2} + 1 \leq t \leq M) \vee (a_{t,N} \wedge 1 \leq t \leq \frac{M}{2}) \end{cases}, \\ \text{time} = d_{t,1} \vee a_{t,N}, \\ \text{metro} = \text{set by the trip allocation algorithm} \end{array} \right\}$$

The *trip allocation algorithm* sets for all tasks – and thus all trips – a specific metro to perform them. The algorithm sorts the list of tasks chronologically and assigns to each of them a *metro\_id*, starting from 1.

For each task, if the task represents a departure, the algorithm checks if a metro is waiting at the terminal. It is worth noticing that, to be available for an assignation to the task, a metro needs to be effectively in the terminal but also to have performed already its turn back manoeuvre. If it is the case, then this metro is assigned to the task. If not, a new metro is added by incrementing *metro\_id* and is assigned to the task. Incrementing *metro\_id* represents in reality adding a metro on the line from the yard.

If the task represents an arrival and that the queue in the terminal has not reached the terminal capacity  $cap$ , then the task is added to the queue. If the queue is full, the oldest task is removed from the queue to let enter the new one. Removing the oldest task represents in reality putting back a metro in the yard. The algorithm pseudo-code is summarized by Algorithm 2, returning the list of tasks with their metro assignation.

---

**Algorithm 2** Trip allocation algorithm
 

---

**Require:** list of *tasks*

```

1: Sort tasks by tasks.time in ascending order
2:  $Q_1, Q_N$  empty queues of task for each terminal
3: metro_id = 1
4: for  $i = 0 : i < \text{size}(\text{tasks})$  do
5:   if tasks[ $i$ ].terminal == 1 then
6:      $Q = Q_1$ 
7:   else
8:      $Q = Q_N$ 
9:   end if
10:  if tasks[ $i$ ].terminal == departure then
11:    if  $Q = 0 \parallel Q[0].\text{time} > \text{tasks}[i].\text{time} - \text{turn\_back}$  then
12:      tasks[ $i$ ].train = metro_id
13:      metro_id ++
14:    else
15:      tasks[ $i$ ].metro =  $Q[0].\text{metro}$ 
16:       $Q = Q - Q[0]$ 
17:    end if
18:  else
19:    if  $\text{size}(Q) \geq cap$  then
20:       $Q = Q - Q[0]$ 
21:    end if
22:     $Q = [Q, \text{tasks}[i]]$ 
23:  end if
24: end for
25: return tasks

```

---

### 3.3 Computational Results

This section demonstrates the capabilities of the metro energy optimization timetabling model on a real weekday handmade timetable. First, we check that the original timetable is feasible according to the physical feasibility model. Figure 3.3 shows the headways of metros along the timetable. The two weekday peak hours – with a headway of 300 seconds – are clearly visible.

The problem of the handmade timetable is how the transitions between different headway targets have been handled. Whereas the mid-day off peak hour headways are

clearly stable, the rest of the timetable presents many vibrations, due to the problem of adding or removing metros without affecting the headways.



Figure 3.3: Initial handmade timetable : graph representing the headways in seconds in function of the metro departure times. The red line represents the metros running upstream and the blue line the metros running downstream.

### 3.3.1 Headway Pattern Timetabling

Following the original handmade timetable headways – 300 seconds on peak hours and 600 seconds on off peak hours – we propose a headway pattern summarized in table 3.1.

$[tra_h, tra_{h+1}[$	$[low_h, upp_h]$	Period	Hour
$[0, 25200[$	$[600, 600]$	service starting	$< 07 : 00$
$[25200, 27000[$	$[300, 600]$	transition time	$07 : 00 - 07 : 30$
$[27000, 36000[$	$[300, 300]$	morning peak hour	$07 : 30 - 10 : 00$
$[36000, 37800[$	$[300, 450]$	transition time	$10 : 00 - 10 : 30$
$[37800, 59400[$	$[450, 450]$	mid-day off-peak hours	$10 : 30 - 16 : 30$
$[59400, 61200[$	$[300, 450]$	transition time	$16 : 30 - 17 : 00$
$[61200, 66600[$	$[300, 300]$	evening peak hour	$17 : 00 - 18 : 30$
$[66600, 68400[$	$[300, 600]$	transition time	$18 : 30 - 19 : 00$
$[68400, 90000[$	$[600, 600]$	service ending	$> 19 : 00$

Table 3.1: Lower and upper bounds on headways for every interval of the headway pattern, specifying the hours for each interval and the signification of the bounds.

Between a peak hour and an off peak hour, a transition time of 2000 seconds is let to adjust the number of metros on the line smoothly. Figure 3.4 presents the headways of

the compiled timetable, given only the headway pattern and the starting times  $dep_{DOWN}$  and  $dep_{UP}$ .



Figure 3.4: Compiled timetable following the given headway pattern : graph representing the headways in seconds in function of the metro departure times. The red line represents the metros running upstream and the blue line the metros running downstream.

Each trip has been allocated with a metro by the trip allocation algorithm and it is clear that the periodicity and the stability of the headways have been highly increased. This proves the utility of such a model, which given a relatively small amount of information as input, is able to produce a much more stable timetable than a handmade one.

### 3.3.2 Energy Optimized Headway Pattern Timetabling

Tolerances are added on the first departure times, allowing them to start in a time window of 20 seconds, and on each dwell time, allowing a shift of 1 second earlier or later:

$$\begin{aligned}
 dep_{UP} - 10 &\leq d_{1,1} \leq dep_{UP} + 10 \\
 dep_{DOWN} - 10 &\leq d_{\frac{M}{2}+1,1} \leq dep_{DOWN} + 10 \\
 \underline{dwe}_{t,s} + 2 &= \overline{dwe}_{t,s}, \quad 1 \leq t \leq M, \quad 2 \leq s \leq N - 1
 \end{aligned}$$

The computation of the energy optimized headway pattern timetable is intractable and runs out of memory on CPLEX. The following results are a proof of concept using a

relaxed objective function, where most transfers between braking and accelerating metros are not authorized. Figure 3.5 shows how the tolerances on dwell times modifies the headway pattern compiled timetable to improve the overlapping time between braking and acceleration phases.

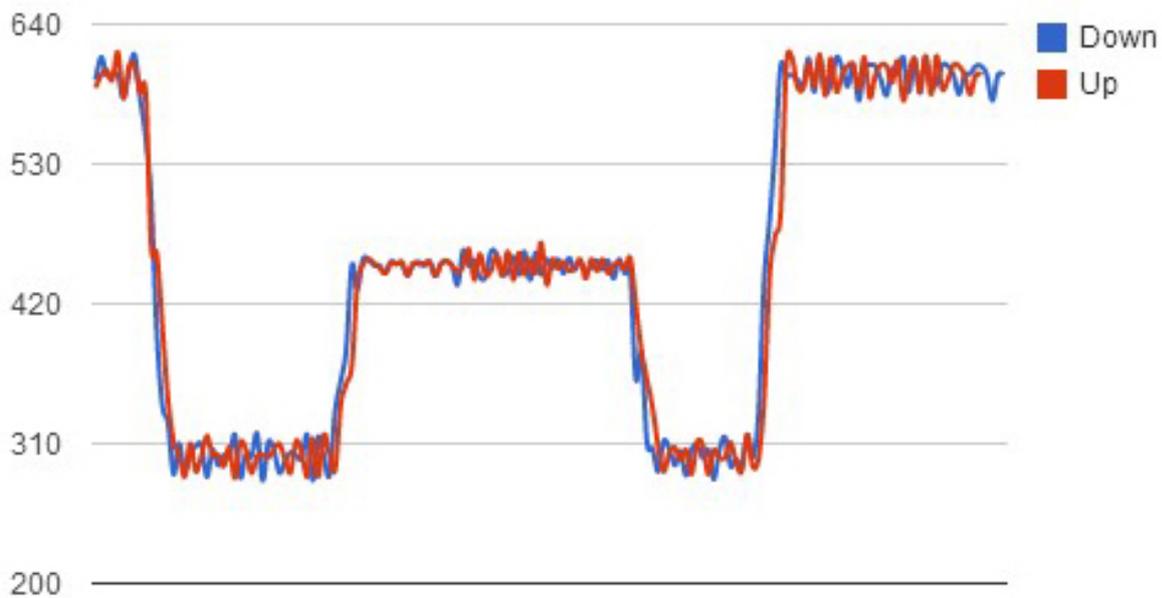


Figure 3.5: Compiled energy optimized timetable following the given headway pattern : graph representing the headways in seconds in function of the metro departure times. The red line represents the metros running upstream and the blue line the metros running downstream.

It is worth noticing that the energy optimized timetable is still more stable in terms of headways than the original handmade timetable. Figure 3.6 shows that, on the relaxed objective function, the cumulated overlapping times can be increased using the energy optimized headway timetabling model.

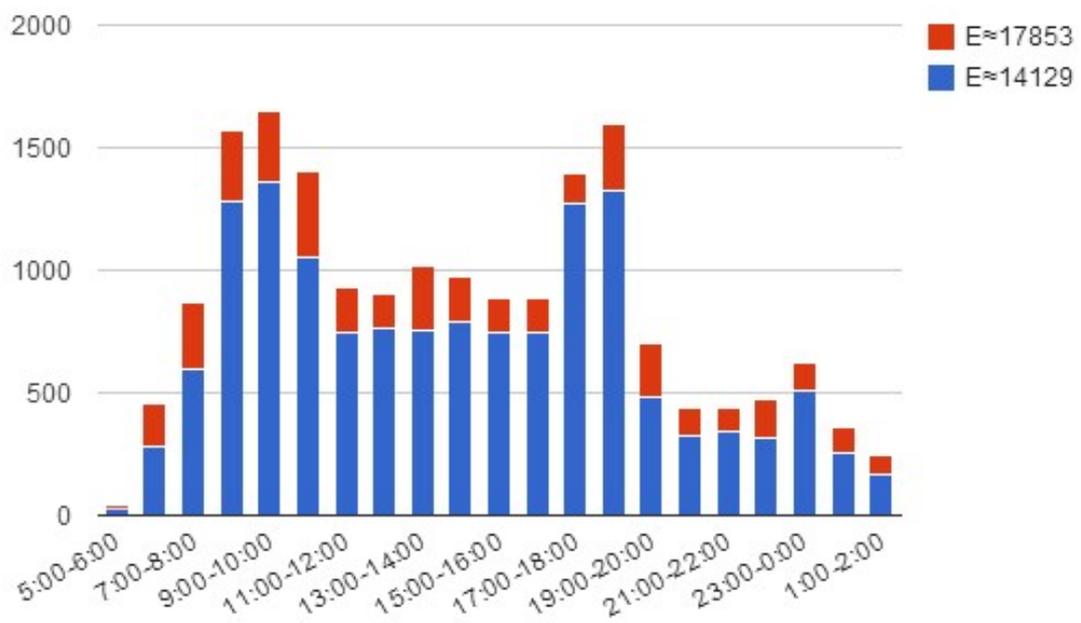


Figure 3.6: Overlapping time in seconds by intervals of 1 hour compared between the initial the timetable, in blue, and the energy optimized headway pattern timetable in red.



# Chapter 4

## Greedy Heuristic Algorithm for ( $G, dwe, nonlin$ )

### Contents

---

<b>4.1</b>	<b>Acceleration Phase Shift Function</b>	<b>66</b>
4.1.1	Braking Phase Neighbourhood	66
4.1.2	Shift and Propagation Algorithm	66
<b>4.2</b>	<b>Greedy Heuristics Optimization Algorithm</b>	<b>67</b>
<b>4.3</b>	<b>Algorithm Optimizations</b>	<b>71</b>
4.3.1	Incremental Computation of the Objective Function	71
4.3.2	Iterative Optimization	71

---

Since metro providers are unlikely to abandon the timetable in service for a brand new one, the literature has focused on metro rescheduling problems to optimize the energy consumption. This reluctance can be explained by the habits acquired by the metro drivers and passengers, as well as the validated robustness of the utilized timetable, that prevent a likeliness for change. Thus, for energy optimization, it is crucial to be able to reschedule the timetable in service. This rescheduling is done by modifying slightly the timetable inputs, so the output solution does not modify the crew rostering, the trip/metro allocations and the quality of service.

Chapter 3 was presenting a proof of concept of how should be a timetabling model taking into account the energy optimization. We present in this chapter the rescheduling solution now implemented in the General Electric Transportation Tempo ATS CBTC Solution, based on a heuristic algorithm modifying the dwell times.

Our heuristic algorithm allies the idea of maximizing the overlapping times between braking and acceleration phases and the idea of checking the global energy consumption

every time a modification is made. The algorithm thus shifts the acceleration phases to synchronize them with the braking phases of other metros, and recomputes the non-linear objective function to check for improvement. These local moves increase the power transfers between braking and accelerating metros, and possibly decrease the solution timetable global energy consumption.

The braking phase of  $a_{t,s}$  is the period of time delimited by  $a_{t,s}^{brk}$  and  $a_{t,s}$ , corresponding to the period where the trip  $T_t$  is braking and regenerating energy, arriving to the station  $S_t(s)$ . Likewise, the acceleration phase of  $d_{t,s}$  is the period of time delimited by  $d_{t,s}$  and  $d_{t,s}^{acc}$ , corresponding to the period where the trip  $T_t$  is accelerating and demanding energy, departing from the station  $S_t(s)$ .

## 4.1 Acceleration Phase Shift Function

The acceleration phase shift function  $Shift(d_{t,s}, a_{t',s'}^{brk})$  consists in modifying the departure time  $d_{t,s}$  of a trip  $T_t$  at a station  $S_t(s)$  to make it correspond to the beginning of the neighbour braking phase  $a_{t',s'}^{brk}$  of a trip  $T_{t'}$  at station  $S_{t'}(s')$ , and by fixing  $acc_{t,s}$ . Modifying  $d_{t,s}$  consists thus in modifying the duration of  $dwe_{t,s}$ , which will shift the future arrivals and departures times of the trip.

### 4.1.1 Braking Phase Neighbourhood

The time neighbourhood of a braking phase  $\mathcal{N}(a_{t,s})$  is defined as the set of acceleration phases that can overlap this braking phase within the given timetable tolerances. This means that every acceleration phase that may start before the end and finish after the beginning of a given braking phase, belongs to the neighbourhood of the latter:

$$\mathcal{N}(a_{t,s}) = \{d_{t',s'} | (t \neq t') \wedge (d_{t',s'}^{acc} + \overline{dwe_{t',s'}} > a_{t,s}^{brk}) \wedge (d_{t',s'} + \underline{dwe_{t',s'}} < a_{t,s})\}$$

### 4.1.2 Shift and Propagation Algorithm

The function considers the timetable constraints given by the bounds on dwell times, trip times and headways. If the function cannot shift the departure time of the acceleration to the beginning of the braking phase due to tolerance constraints, it will shift it to the closest time which respects these constraints.

The function  $Shift(d_{t,s}, a_{t',s'}^{brk})$  first checks whether  $d_{t,s}$  occurs before or after  $a_{t',s'}^{brk}$ . The algorithm then computes the possible shifts authorized by the three timetable bound constraints for dwell times, trip times and headways. If  $d_{t,s}$  occurs after (resp. before)

$a_{t',s'}^{brk}$ , these shifts are set according to the constraints' lower (resp. upper) bounds. Then, the maximum (resp. minimum) of these shifts and of the  $a_{t',s'}^{brk}$  date is added to  $d_{t,s}$ .

As only the adjacent dwell time of  $d_{t,s}$  is modified, the rest of the trip is shifted accordingly to the shift applied to  $d_{t,s}$ . Thus, the shift is also added to the departure times  $d_{t,s'}$  and the arrival times  $a_{t,s'}$ , such that  $s' > s$ . The shifting and propagation process is clearly visible on Figures 4.1 and 4.2 on trip 5 ; the shift applied on the first acceleration phase – the leftmost red rectangle – is propagated to the future braking and acceleration phases. Algorithm 3 summarizes the shift and propagation process.

---

**Algorithm 3** Acceleration phase shift and propagation
 

---

**Require:**  $\mathcal{T}\mathcal{T}$

```

1: Acceleration phase shift
2: if  $d_{t,s} > a_{t',s'}^{brk}$  then
3:    $shift_{dwe} = \overline{dwe_{t,s}} - (d_{t,s} - a_{t,s})$ 
4:    $shift_{trt} = \overline{trt_t} - (a_{t,N} - d_{t,1})$ 
5:    $shift_{hdw} = \overline{hdw_{t,s}} - (d_{t,s} - d_{t-1,s})$ 
6:    $shift = \max(shift_{dwe}, shift_{trt}, shift_{hdw}, a_{t',s'}^{brk} - d_{t,s})$ 
7:    $d_{t,s} \leftarrow d_{t,s} + shift$ 
8: else
9:    $shift_{dwe} = \overline{dwe_{t,s}} - (d_{t,s} - a_{t,s})$ 
10:   $shift_{trt} = \overline{trt_t} - (a_{t,N} - d_{t,1})$ 
11:   $shift_{hdw} = \overline{hdw_{t,s}} - (d_{t,s} - d_{t-1,s})$ 
12:   $shift = \min(shift_{dwe}, shift_{trt}, shift_{hdw}, a_{t',s'}^{brk} - d_{t,s})$ 
13:   $d_{t,s} \leftarrow d_{t,s} + shift$ 
14: end if
15: Shift propagation
16: for all  $d_{t,s'}$  s.t.  $d_{t,s'} > d_{t,s}$  do
17:    $d_{t,s'} \leftarrow d_{t,s'} + shift$ 
18: end for
19: for all  $a_{t,s'}$  s.t.  $d_{t,s'} > d_{t,s}$  do
20:    $a_{t,s'} \leftarrow a_{t,s'} + shift$ 
21: end for
22: return  $\mathcal{T}\mathcal{T}$ 

```

---

## 4.2 Greedy Heuristics Optimization Algorithm

The algorithm comprehensively searches the acceleration phase shift that minimizes the objective function in the time neighbourhood of each braking phase,. All the braking phases are first sorted in chronological order and the acceleration phases are shifted for the earliest braking phases.

For each braking phase, the algorithm computes its time neighbourhood  $\mathcal{N}(a_{t,s})$ , modifies each acceleration phase  $d_{t',s'}$  by applying the shift function  $\text{Shift}(d_{t',s'}, a_{t,s}^{brk})$

and checks whether this shift is decreasing the objective function. If it does, the best current objective function and the best shift are updated.

When all the acceleration phases have been shifted and evaluated, the algorithm shifts the acceleration phase that minimizes the most the objective function, or does nothing if none of them decreases the objective function. This monotonic behaviour ensures that the algorithm will converge after some iterations and does not worsen the initial timetable. Once an acceleration phase is shifted, it is removed from the pool of neighbour phases and cannot be shifted any more for another braking phase. Algorithm 4 summarizes the greedy heuristics optimization algorithm:

---

**Algorithm 4** Greedy heuristics optimization algorithm
 

---

**Require:**  $\mathcal{T}\mathcal{T}$

- 1: Sort  $a_{t,s}$ ,  $1 \leq t \leq M, 2 \leq s \leq N$  in chronological order
  - 2: **for all**  $a_{t,s}$  **do**
  - 3:    Compute initial objective function  $G_{\mathcal{T}\mathcal{T}}^{INIT}$
  - 4:    Initialize best objective function  $G_{\mathcal{T}\mathcal{T}}^{BEST} = G_{\mathcal{T}\mathcal{T}}^{INIT}$
  - 5:    Initialize best shift  $d^{BEST} = 0$
  - 6:    Compute  $\mathcal{N}(a_{t,s})$
  - 7:    **for all**  $d_{t',s'} \in \mathcal{N}(a_{t,s})$  **do**
  - 8:       $d^{INIT} \leftarrow d_{t',s'}$
  - 9:      Shift( $d_{t',s'}, a_{t,s}$ )
  - 10:     Compute  $G_{\mathcal{T}\mathcal{T}}$
  - 11:     **if**  $G_{\mathcal{T}\mathcal{T}} < G_{\mathcal{T}\mathcal{T}}^{BEST}$  **then**
  - 12:        $G_{\mathcal{T}\mathcal{T}}^{BEST} \leftarrow G_{\mathcal{T}\mathcal{T}}$
  - 13:        $d^{BEST} \leftarrow d_{t',s'}$
  - 14:     **end if**
  - 15:      $d_{t',s'} \leftarrow d^{INIT}$
  - 16:    **end for**
  - 17:    **if**  $d^{BEST} \neq 0$  **then**
  - 18:      Shift( $d^{BEST}, a_{t,s}$ )
  - 19:    **end if**
  - 20: **end for**
  - 21: **return**  $\mathcal{T}\mathcal{T}, G_{\mathcal{T}\mathcal{T}}^{BEST}$
- 

Figures 4.1 and 4.2 illustrate the behaviour of our heuristics on a sample timetable lasting 400 seconds. In this sample, 15 metros are running on the line and their braking and acceleration phases along the timetable are represented respectively with red and green rectangles on the figures. It is clear that the optimized timetable still looks like the initial one and that the modifications are not changing dramatically the shape of the trips. However, one can remark that these changes allow better overlapping times between braking and acceleration phases, as seen at time 36720.

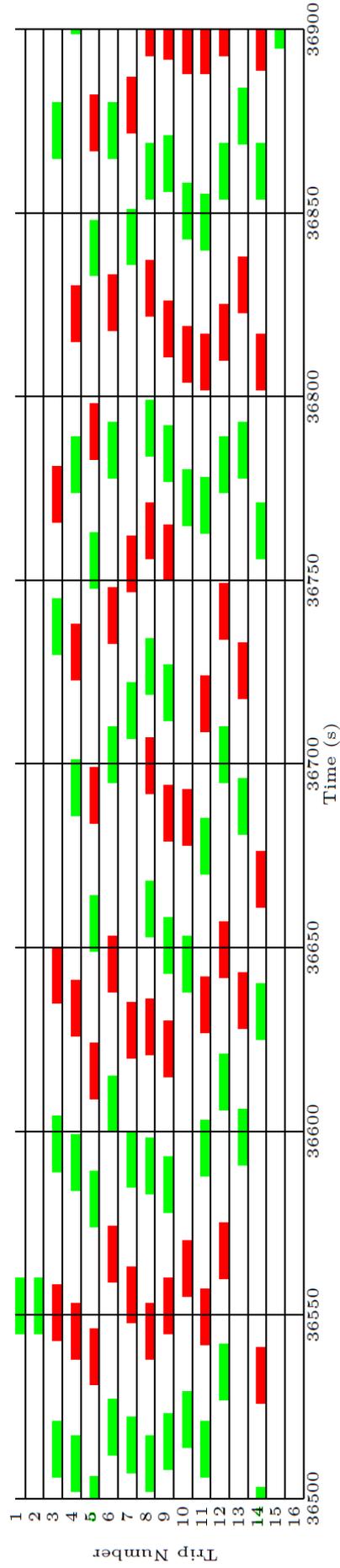


Figure 4.1: Initial timetable : braking and acceleration phases distribution over time on a part lasting 400 seconds. Each line represents a trip, green rectangles braking phases and red rectangles acceleration phases.

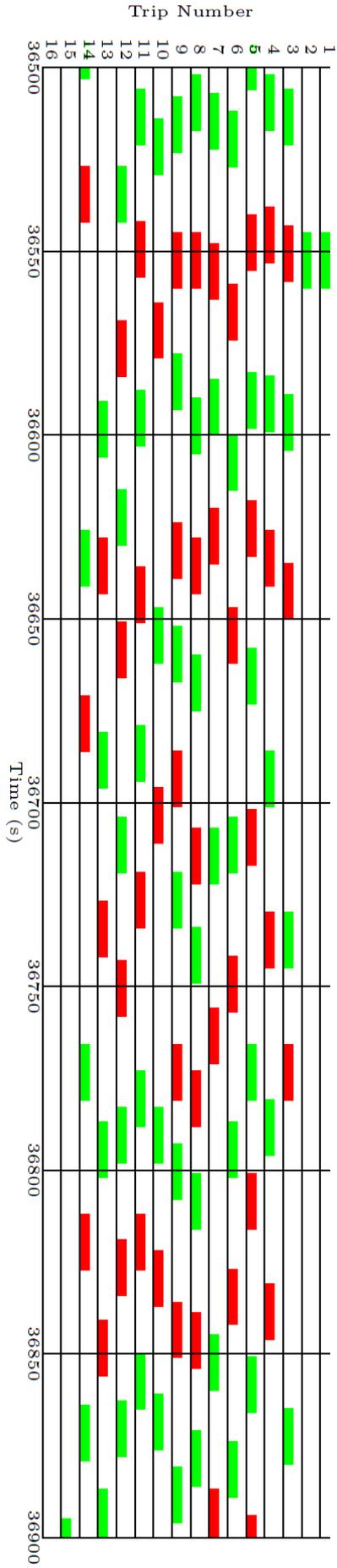


Figure 4.2: Optimized timetable : braking and acceleration phases distribution over time on a part lasting 400 seconds. Each line represents a trip, green rectangles braking phases and red rectangles acceleration phases.

## 4.3 Algorithm Optimizations

### 4.3.1 Incremental Computation of the Objective Function

Equation (2.7) shows that the instant power demand  $P_i$  is a function of the net power demands and productions  $P_{s,i}$ , the power transfer ratios  $x_{s,s',i}$  and the distribution matrix  $\Delta_{s,s'}$ . Since the distribution matrix is precomputed, only the power transfers are modified by the optimization process.

In our dedicated heuristics, the objective function is re-evaluated every time an acceleration phase is shifted. Since a shift consists in modifying the dwell time of a specific trip at a specific station, many time instants remain with the exact same net power demand and production along the timetable. Let  $P_i^{INIT}$  be the instant power demand at time  $i$  before a dwell time shift and  $P_{s,i}^{INIT}$  be the net power demands and productions at each station  $S_s$  at time  $i$ . After the dwell time shift, only the instant power demands where the power demands and productions have changed need recomputing, as follows:

$$P_i = \begin{cases} P_i^{INIT} & \text{if } P_{s,i} = P_{s,i}^{INIT} \quad \forall 1 \leq s \leq N, \\ \text{needs recomputation} & \text{otherwise} \end{cases}$$

The *incremental computation* avoids recomputing known values, which increases the computation time of the algorithm by an order of magnitude as shown in Table 4.1 on six benchmark instances detailed in the following section.

Instance	Length	# $d_{t,s}$	Computation Time (s)	
			Regular	Incremental
op1	15 min	127	34.9	7.31
op2	15 min	129	31.0	7.51
op3	60 min	449	530	52.5
p1	15 min	173	102	18.3
p2	15 min	186	134	25.7
p3	60 min	670	1495	168

Table 4.1: Computation time in seconds of one run of the greedy heuristics without and with incremental computation of the objective function. The two implementations are compared on the six benchmark instances described in Section 5.1 representing typical off-peak (op) or peak (p) hours timetables. The lengths of the timetable instances are either 15 or 60 minutes and the number of decision variables  $\#d_{t,s}$  are given.

### 4.3.2 Iterative Optimization

After one run of the algorithm, the optimized timetable can be utilized as input for a second run starting from the new solution. The optimization algorithm can thus

be executed either once or iteratively until the iterative algorithm stops improving the objective function.

Figure 4.3 shows the minimization of the objective function using iterative optimization. It shows that the first run largely improves the timetable and that the following iterations lead to further improvements.

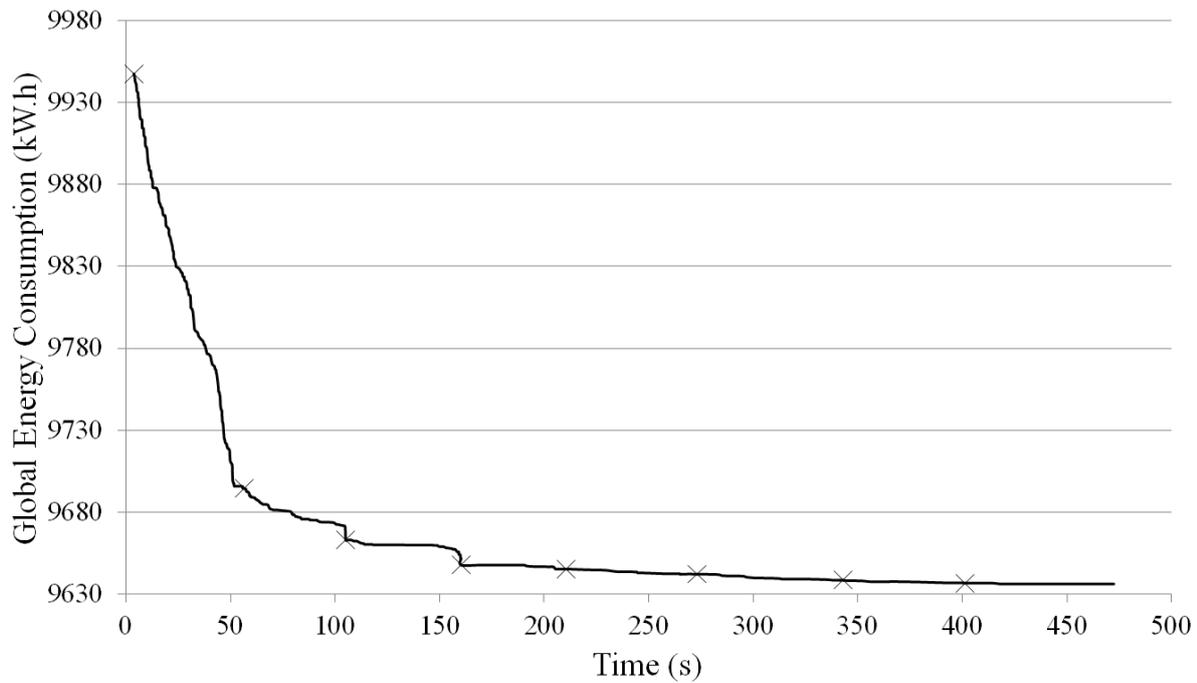


Figure 4.3: Evolution of the global energy consumption over computation time in seconds on a sample timetable during the iterative optimization process. The first cross represents the global energy consumption of the original timetable and the following ones the global energy consumption by iterating the greedy heuristics.

# Chapter 5

## Performance Results

### Contents

---

<b>5.1</b>	<b>Benchmark Instances</b>	<b>73</b>
<b>5.2</b>	<b>Performance Comparison with MILP and CMA-ES</b>	<b>74</b>
5.2.1	Comparison with CMA-ES	74
5.2.2	Comparison with MILP	76
5.2.3	Robustness	77
<b>5.3</b>	<b>Performance Results on Real Data</b>	<b>79</b>

---

In this section, the greedy heuristics algorithm is compared to MILP and CMA-ES on six small size benchmark instances on which the three methods give solutions. Both the greedy heuristics and CMA-ES [11] are implemented in C++, and the MILP model is solved using CPLEX 12. The machine used for the experiments is a PC with an Intel Core i5 with 3GB of RAM.

For all the following results, a timeout of 1500 seconds was set. If not specified otherwise, the objective function is computed using the electrical simulator described in section 2. The greedy heuristics used the incremental computation of the objective function and iterative optimization.

### 5.1 Benchmark Instances

The six timetables have been drawn from real data and represent relevant portions of the timetable, i.e. peak (p) and off-peak (op) parts of size of 15 minutes and one hour. These instances contain the initial parameters of the timetable ( $d_{t,1}^{INIT}$ ,  $dwe^{INIT}$ ,  $int^{INIT}$  and so on) as well as the tolerances, given by the customer, on which variables can be modified and by how much. The tolerances on dwell times, trip times and headways are equal

for all six instances. For the sake of simplicity, the tolerance values are given relatively to the initial timetable instance and  $\underline{dwe}_{t,s} = -3$  shall be read  $\underline{dwe}_{t,s} - dwe_{t,s}^{INIT} = -3$ . The departure times, the interstation times, the braking and acceleration phases lengths are fixed :

$$\begin{aligned} \underline{dwe}_{t,s} &= -3 & 1 \leq t \leq M, 1 \leq s \leq N \\ \overline{dwe}_{t,s} &= 9 & 1 \leq t \leq M, 1 \leq s \leq N \\ \underline{trt}_t &= -30 & 1 \leq t \leq M \\ \overline{trt}_t &= 30 & 1 \leq t \leq M \\ \underline{hdw}_{t,s} &= -30 & t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], 1 \leq s \leq N \\ \overline{hdw}_{t,s} &= 30 & t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], 1 \leq s \leq N \end{aligned}$$

## 5.2 Performance Comparison with MILP and CMA-ES

### 5.2.1 Comparison with CMA-ES

Evolution strategies are stochastic search algorithms that try to minimize an arbitrary objective function called fitness function. The covariance matrix adaptation evolution strategy (CMA-ES) [11] applies to vectors of real-valued variables and arbitrary real-valued fitness functions. This algorithm is a multi-point method which at each iteration, samples the search space according to multivariate normal distributions, estimates its covariance matrix, determines a move to make in the most promising direction and updates the multivariate normal distributions for the variables. One important characteristic of CMA-ES compared to other meta-heuristics, is the limited number of parameters that need to be set, namely the initial standard deviation and the termination criteria. The other parameters are automatically adapted during the execution.

In our experiments, we use the default value for the population size  $4 + 3 \log(\#d_{t,s})$ . The optimization is stopped after 10 iterations without improvement of the objective function. The initial distribution has a default variance of  $(\overline{dwe}_{t,s} - \underline{dwe}_{t,s})/7$  for each trip at each station.

The CMA-ES algorithm does not handle internally bound domains on variables. To

avoid infeasible solutions, a penalty function  $p(\mathcal{T}\mathcal{T})$  is added to the objective function,

$$G_{\mathcal{T}\mathcal{T}} = \sum_{i=0}^{I_{END}} P_i + p(\mathcal{T}\mathcal{T}). \quad (5.1)$$

This function adds the difference, to the power of 4, of each variable out of its domain minus the bound it violates. This enforces the algorithm to search solutions within the given tolerances. Algorithm 5 summarizes the penalty function computation.

---

**Algorithm 5** CMA-ES penalty function
 

---

**Require:**  $\mathcal{T}\mathcal{T}$

```

1:  $p(\mathcal{T}\mathcal{T}) = 0$ 
2: for all  $dwe_{t,s}$  do
3:   if  $dwe_{t,s} < \underline{dwe_{t,s}}$  then  $p(\mathcal{T}\mathcal{T}) \leftarrow p(\mathcal{T}\mathcal{T}) + (dwe_{t,s} - \underline{dwe_{t,s}})^4$ 
4:   else
5:     if  $dwe_{t,s} > \overline{dwe_{t,s}}$  then  $p(\mathcal{T}\mathcal{T}) \leftarrow p(\mathcal{T}\mathcal{T}) + (dwe_{t,s} - \overline{dwe_{t,s}})^4$ 
6:     end if
7:   end if
8: end for
9: for all  $trt_t$  do
10:  if  $trt_t < \underline{trt_t}$  then  $p(\mathcal{T}\mathcal{T}) \leftarrow p(\mathcal{T}\mathcal{T}) + (trt_t - \underline{trt_t})^4$ 
11:  else
12:    if  $trt_t > \overline{trt_t}$  then  $p(\mathcal{T}\mathcal{T}) \leftarrow p(\mathcal{T}\mathcal{T}) + (trt_t - \overline{trt_t})^4$ 
13:    end if
14:  end if
15: end for
16: for all  $hdw_{t,s}$  do
17:  if  $hdw_{t,s} < \underline{hdw_{t,s}}$  then  $p(\mathcal{T}\mathcal{T}) \leftarrow p(\mathcal{T}\mathcal{T}) + (hdw_{t,s} - \underline{hdw_{t,s}})^4$ 
18:  else
19:    if  $hdw_{t,s} > \overline{hdw_{t,s}}$  then  $p(\mathcal{T}\mathcal{T}) \leftarrow p(\mathcal{T}\mathcal{T}) + (hdw_{t,s} - \overline{hdw_{t,s}})^4$ 
20:    end if
21:  end if
22: end for
23: return  $p(\mathcal{T}\mathcal{T})$ 

```

---

Table 5.1 shows the results of CMA-ES against our heuristics. Due to its stochastic behaviour, CMA-ES has been run 100 times for each instance. The table compiles the average computation time, and both the average and best value found for the objective function over the 100 runs. The results show that the greedy heuristics performs better than the best run of CMA-ES on four of the six benchmark instances. On *op1*, our heuristics is better than the average result of CMA-ES but not than its best result. Finally, CMA-ES is slightly better than the greedy heuristics in average only for *p3*. The better performance of the heuristic algorithm is partly due to the incremental computation of the objective function, which cannot be implemented in CMA-ES since all solutions are sampled randomly.

Inst.	Length	# $d_{t,s}$	Initial Value	CMA-ES		Greedy Heuristics		
				Value		Time	Value	Time
				Average	Best			
op1	15 min	127	2514	2401	2381	256	<b>2394</b>	<b>45.6</b>
op2	15 min	129	2516	2402	2388	223	<b>2381</b>	<b>38.0</b>
op3	60 min	449	9956	9724	9716	761	<b>9579</b>	<b>648</b>
p1	15 min	173	3433	3300	3285	503	<b>3261</b>	<b>178</b>
p2	15 min	186	3651	3516	3483	669	<b>3450</b>	<b>291</b>
p3	60 min	670	13067	<b>12696</b>	<b>12675</b>	<b>1030</b>	12717	1500

Table 5.1: Compared performance in computation time (Time in seconds) and energy consumption (Value in kW.h) between the average and best values found over 100 runs of CMA-ES and the greedy heuristics on six benchmark instances. The instances *op* represent an off-peak hour timetable and the instances *p* represent a peak hour timetable, both of either 15 minutes or 60 minutes long.

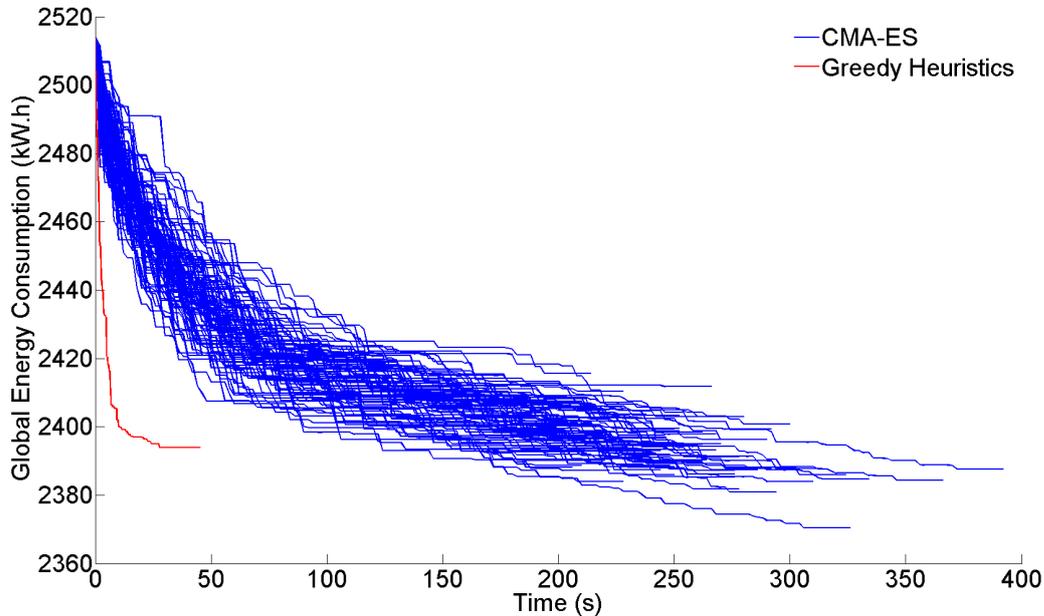


Figure 5.1: Global energy consumption (in kW.h) evolution over time (in seconds) of 100 CMA-ES runs (blue lines) against a greedy heuristics iterative optimization run (red line) on benchmark instance *op1*

## 5.2.2 Comparison with MILP

As shown in Table 5.2, CPLEX is able to prove the optimality on the four smallest instances and outperforms the greedy heuristics on the linear objective function (Equation 2.13). However, when comparing both optimization methods on the objective function computed with the electrical simulator (Table 5.3), our greedy heuristics performs better on five of the six instances.

This is due to the fact that the linear objective function is less accurate than the

Instance	Length	# $d_{t,s}$	Initial Value	Greedy Heuristics Value	MILP	
					Value	Integrality gap
op1	15 min	127	12.48 s	101.8 s	<b>318.1 s</b>	optimal
op2	15 min	129	11.48 s	159.1 s	<b>351.6 s</b>	optimal
op3	60 min	449	45.97 s	817.9 s	<b>1637 s</b>	10.62%
p1	15 min	173	250.5 s	414.8 s	<b>772.5 s</b>	optimal
p2	15 min	186	279.1 s	533.8 s	<b>835.8 s</b>	optimal
p3	60 min	670	1019 s	1576 s	<b>3003 s</b>	20.44%

Table 5.2: Compared performances over the MILP objective function (Value in s) between CPLEX and the greedy heuristics on six benchmark instances. The MILP solutions are given with their integrality gap, *optimal* standing for 0%. The instances *op* represent an off-peak hour timetable and the instances *p* represent a peak hour timetable, both of either 15 minutes or 60 minutes long.

Instance	Length	# $d_{t,s}$	Initial Value	MILP		Greedy Heuristics	
				Value	Time	Value	Time
op1	15 min	127	2514	2427	<b>0.72</b>	<b>2394</b>	45.6
op2	15 min	129	2516	2419	<b>1.00</b>	<b>2381</b>	38.0
op3	60 min	449	9956	<b>9579</b>	1500	<b>9579</b>	<b>648</b>
p1	15 min	173	3433	3281	460	<b>3261</b>	<b>178</b>
p2	15 min	186	3651	3494	<b>82.5</b>	<b>3450</b>	291
p3	60 min	670	13067	<b>12711</b>	<b>1500</b>	12717	<b>1500</b>

Table 5.3: Compared performances in computation time (Time in seconds) and energy consumption (Value in kW.h) between MILP and the greedy heuristics on six benchmark instances. The instances *op* represent an off-peak hour timetable and the instances *p* represent a peak hour timetable, both of either 15 minutes or 60 minutes long.

one used in the greedy heuristics (Equation 2.7). The main differences between these two objective functions are that the MILP model is only able to pair one braking with one acceleration (Equations 2.8, 2.9), when the power flow objective function is able to dispatch dynamically to different braking and accelerations over time as described in Section 2.2.2. Thus eventhough our algorithm does not prove optimality, it better approximates the real behaviour of the electricity flows and leads to better solutions.

### 5.2.3 Robustness

The output solutions of the greedy heuristics must be robust to be effectively used in an industrial context [50]. Indeed, even if the timetable optimizer is more likely to be used in a fully automated metro context, small perturbations can still occur [51]. Thus, the output solutions should still save a significant amount of energy compared to the initial timetable, even when they are lightly disrupted. To test the robustness of solutions, a noise  $\theta = \{1, 2, 3\}$  seconds is randomly applied to each decision variable  $d_{t,s}$  that has

been adjusted by the greedy heuristics, such that

$$d_{t,s} \leftarrow d_{t,s} + \text{rand}\{-\theta, \dots, \theta\}$$

For each benchmark instance, 100 random solutions have been generated for three different noise amplitude. Tables 5.4, 5.5 and 5.6 summarize the statistical analysis of the robustness for respectively 1, 2 and 3 seconds of noise amplitude. The 100 randomly generated solutions are compared to the initial timetable and the optimized timetable in terms of average value and statistical dispersion of the global energy consumption.

The tables show that the average value of the global energy consumption is degraded proportionally to the noise amplitude but that even the worst randomly generated solution is still saving more energy than the initial timetable on every instance. Also, the degradation of the objective function is predictable in function of the noise as the standard deviation is small and the distribution of the solutions is condensed around the average value. This means that our greedy heuristics leads to solutions that are inside attraction basins of local optima, which are robust to small perturbations. However, it seems better to re-optimize the current solution as soon as the noise becomes greater than 3 seconds, Table 5.6 showing that a 3 seconds noise on *p3* greatly decreases the optimized solution.

Inst.	Init.	Opt.	Robustness Analysis – 1 second noise					
			Average	Std.Dev.	Min.	Max.	1 <sup>st</sup> Quart.	3 <sup>rd</sup> Quart.
op1	2514	2394	2404	3.20	2396	2416	2402	2406
op2	2516	2381	2389	2.94	2383	2400	2387	2390
op3	9956	9579	9649	12.90	9623	9679	9640	9658
p1	3433	3261	3273	3.30	3266	3285	3271	3273
p2	3651	3450	3468	4.13	3459	3478	3466	3471
p3	13067	12717	12792	13.35	12758	12820	12782	12801

Table 5.4: One second noise : statistical analysis of the greedy heuristics solutions robustness on six benchmark instances. Given the optimized solutions, 100 random solutions with noise on decision variables values are generated and evaluated regarding their global energy consumption in kW.h.

Figure 5.2 is a box plot of the statistical dispersion of the randomly generated solutions for the instance *op1*, both on the initial timetable and on the optimized timetable. The box plots of the initial timetable show that the more the noise, the bigger the amplitude of the statistical distribution, but always centred around the initial global energy consumption. On the other hand, the box plots of the optimized solution show that the increasing noise constantly decreases the average global energy consumption, which tends to come back to its initial value.

This means that the initial solution has not been set in order to optimize the global

Inst.	Init.	Opt.	Robustness Analysis – 2 seconds noise					
			Average	Std.Dev.	Min.	Max.	1 <sup>st</sup> Quart.	3 <sup>rd</sup> Quart.
op1	2514	2394	2414	5.42	2400	2431	2411	2417
op2	2516	2381	2400	6.32	2385	2421	2396	2403
op3	9956	9579	9716	20.05	9664	9778	9703	9730
p1	3433	3261	3289	6.99	3273	3313	3283	3294
p2	3651	3450	3489	7.32	3470	3505	3483	3494
p3	13067	12717	12876	18.68	12838	12914	12862	12890

Table 5.5: Two seconds noise : statistical analysis of the greedy heuristics solutions robustness on six benchmark instances. Given the optimized solutions, 100 random solutions with noise on decision variables values are generated and evaluated regarding their global energy consumption in kW.h.

Inst.	Init.	Opt.	Robustness Analysis – 3 seconds noise					
			Average	Std.Dev.	Min.	Max.	1 <sup>st</sup> Quart.	3 <sup>rd</sup> Quart.
op1	2514	2394	2424	5.83	2410	2438	2420	2428
op2	2516	2381	2410	6.60	2393	2430	2407	2414
op3	9956	9579	9762	19.48	9709	9807	9748	9778
p1	3433	3261	3305	8.48	3290	3334	3300	3312
p2	3651	3450	3511	10.07	3486	3534	3504	3517
p3	13067	12717	12931	23.73	12864	12994	12914	12944

Table 5.6: Three seconds noise : statistical analysis of the greedy heuristics solutions robustness on six benchmark instances. Given the optimized solutions, 100 random solutions with noise on decision variables values are generated and evaluated regarding their global energy consumption in kW.h.

energy consumption and that the initial energy value represents a fair approximation of the consumption of a not energy-aware metro line. Finally, adding noise to the optimized solutions adds randomness that makes it eventually resemble any not energy-aware timetable.

### 5.3 Performance Results on Real Data

Our greedy heuristics has also been applied on a major city metro line comprising 16 stations for optimizing one full day timetable in two typical situations:

- a *weekday timetable* comprising 694 trips and 9585 dwell times,
- a *Sunday timetable* comprising 556 trips and 7679 dwell times.

Both the C++ implementation of CMA-ES and the MILP resolution by CPLEX have failed to tame problems of this size. The MILP model contains, after its pre-processing, 230908 constraints and 165760 variables, whose 52872 are binary, and runs out of memory on CPLEX on a PC with an Intel Core i5 with 3GB of RAM.. The size of this

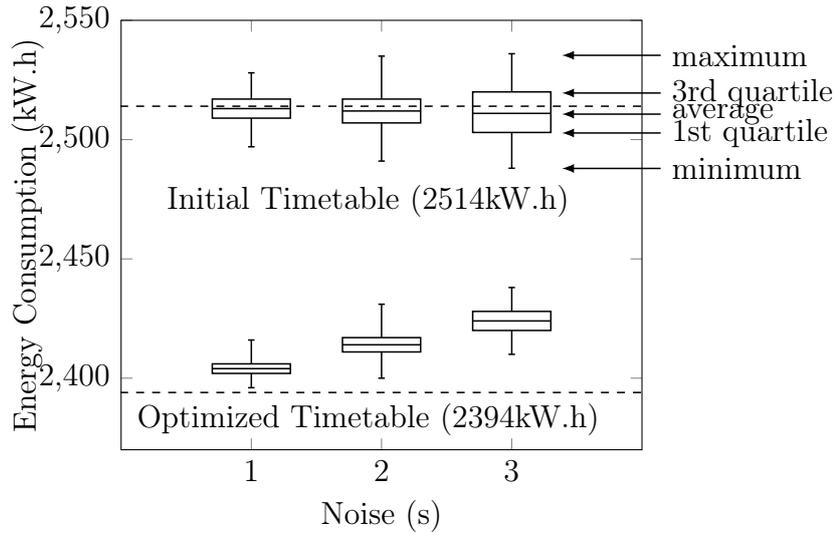


Figure 5.2: Box plot of the statistical distribution of 100 randomized solutions generated with a noise (in seconds) on decision variables. Compared global energy consumption (in kW.h) between the initial timetable and the timetable optimized by the greedy heuristics for the benchmark instance *op1*.

instance is to relate with the size of the problem handled in [30] which was containing only 17850 constraints and 13860 variables, whose 4780 were binary, or in [29] which was containing 7700 constraints and 4200 variables, whose 600 were binary. For CMA-ES, it fails at computing the global energy consumption of the initial population within 30 minutes. On the other hand, our greedy heuristics is able to compute a solution in 20 minutes. The tolerances on the dwell times, trip times and headways have been first set such that there is no visible change in the quality of service for the passengers. Their relatively small values are as follows:

$$\begin{aligned}
 \underline{dwe}_{t,s} &= -3 & 1 \leq t \leq M, 1 \leq s \leq N \\
 \overline{dwe}_{t,s} &= 3 & 1 \leq t \leq M, 1 \leq s \leq N \\
 \underline{trt}_t &= -15 & 1 \leq t \leq M \\
 \overline{trt}_t &= 15 & 1 \leq t \leq M \\
 \underline{hdw}_{t,s} &= -15 & t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], 1 \leq s \leq N \\
 \overline{hdw}_{t,s} &= 15 & t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], 1 \leq s \leq N
 \end{aligned}$$

For the *Sunday timetable*, the trip times and headways tolerances have been enlarged to 20 seconds as follows:

$$\begin{aligned} \underline{trt}_t &= -20 & 1 \leq t \leq M \\ \overline{trt}_t &= 20 & 1 \leq t \leq M \\ \underline{hdw}_{t,s} &= -20 & t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], 1 \leq s \leq N \\ \overline{hdw}_{t,s} &= 20 & t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M], 1 \leq s \leq N \end{aligned}$$

While the optimized timetable with regular tolerances is saving energy by 7.54%, the solution with increased tolerances can save up to 8.91%, increasing possibilities to synchronize phases better. Table 5.7 summarizes these results.

Instance	Length	$\#d_{t,s}$	Initial	CMA-ES	MILP	Greedy heuristics
weekday	1 day	9585	218294	-	-	207052 (-5.15%)
sunday15	1 day	7679	189953	-	-	175638 (-7.54%)
sunday20	1 day	7679	189953	-	-	173036 (-8.91%)

Table 5.7: Compared performances in terms of energy consumption, given in kW.h, of CMA-ES, CPLEX and the greedy heuristics on three full size timetables. CMA-ES and CPLEX did not manage to output a solution. The ratios represent the energy savings compared to the initial timetable energy consumption.

Figures 5.3, 5.4 and 5.5 compare the initial and optimized timetable energy consumptions on three real data instances. For the *weekday timetable*, the two peak hour periods are clearly visible, from 8am to 11am and from 5pm to 9pm. It appears that more energy is saved during these hours. This is due to the fact that, according to the computed distribution matrix  $\Delta_{s,s'}$ , the energy transfers can be done only between metros that are very close from each other. Since the density of metros on the line is higher during peak hours, it is thus possible to save more energy during these times. The extrapolation of these savings shows that the metro company could save 3.65 GW.h of electrical energy per year.

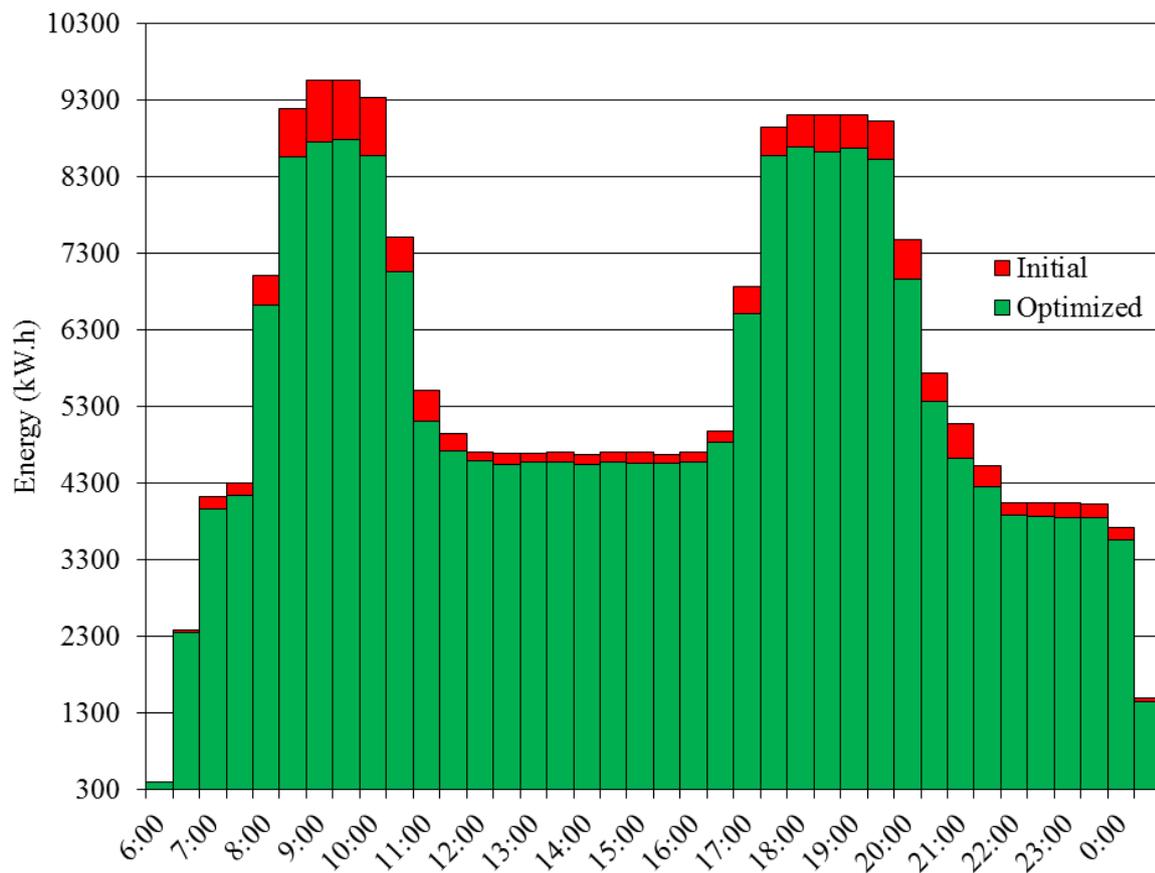


Figure 5.3: *Weekday timetable* between 6am and 1am: energy consumption by intervals of 30 minutes compared between the initial the timetable, in red, and timetable computed by the greedy heuristics in green.

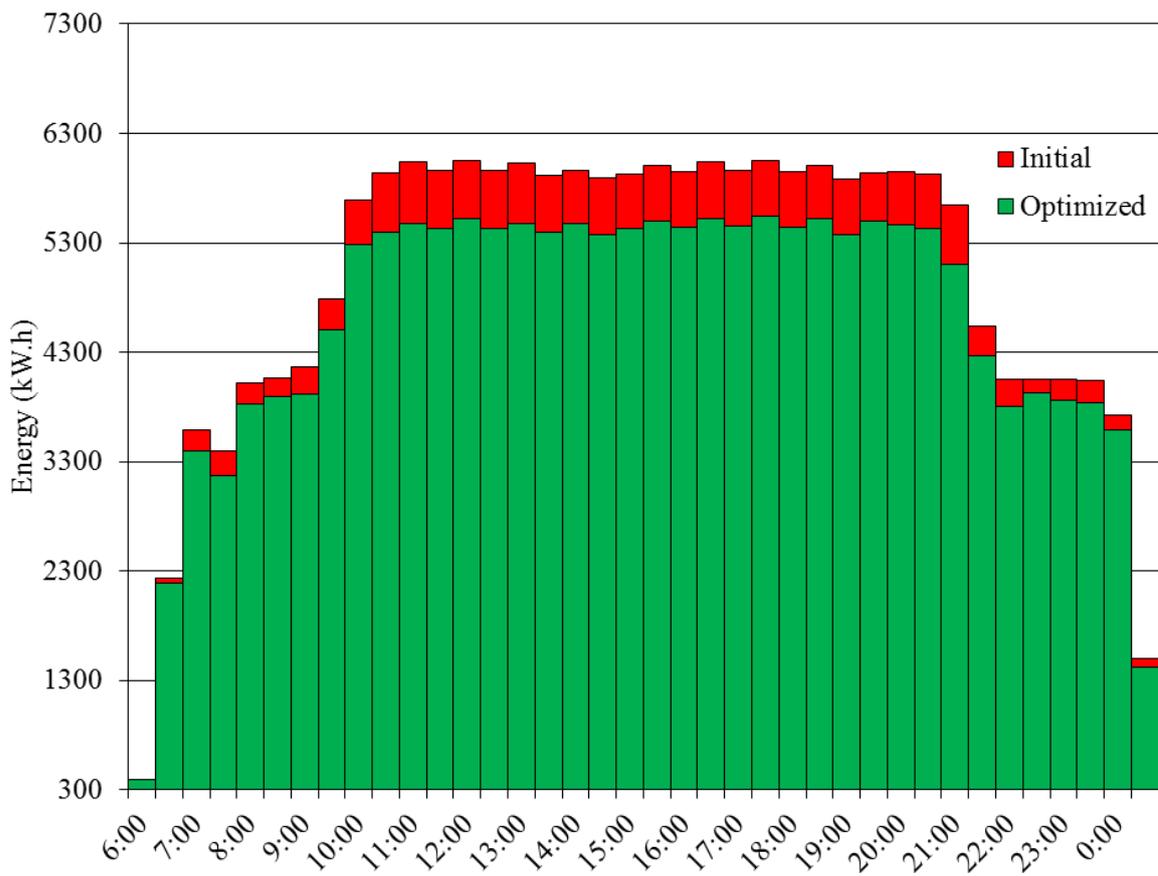


Figure 5.4: 15 seconds tolerance on trips and headways *Sunday timetable* between 6am and 1am : energy consumption by intervals of 30 minutes compared between the initial timetable, in red, and the timetable computed by the greedy heuristics in green.

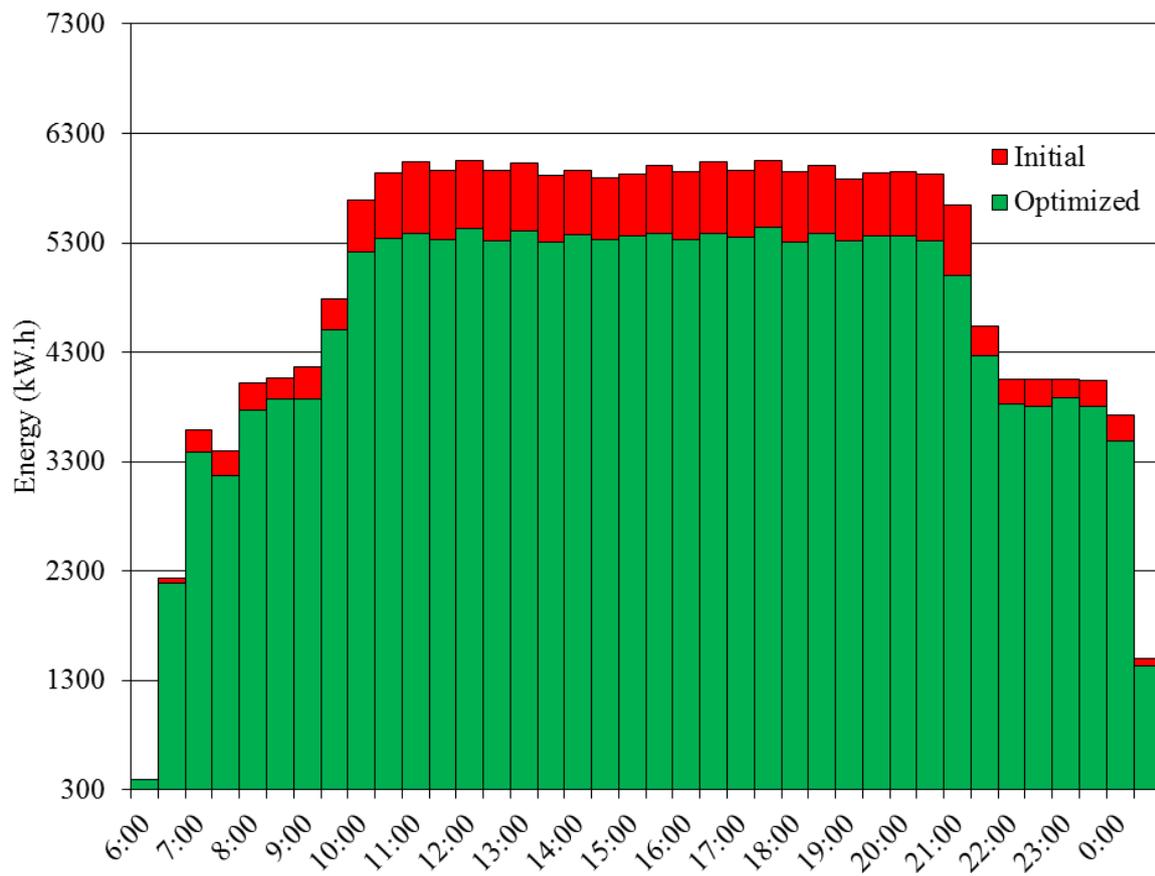


Figure 5.5: 20 seconds tolerance on trips and headways *Sunday timetable* between 6am and 1am : energy consumption by intervals of 30 minutes compared between the initial timetable, in red, and the timetable computed by the greedy heuristics in green.

# Chapter 6

## Conclusion

In this thesis, We have proposed mathematical rescheduling model capable to represent the diversity of the metro energy optimization problems. This model has been used to classify by triples (*objective function, decision variables, instant power demand evaluation*) several problems in the literature. By polynomial reductions of SAT, we have shown that both the problems minimizing the global energy consumption or the maximum power peak are NP-hard, no matter the decision variables they use or whether they approximate the instant power demand by linear or non-linear equations.

We have presented a greedy heuristic algorithm that addresses the particular problem of minimizing the global energy consumption by solely modifying the dwell times ( $G, dwe, nonlin$ ). It locally synchronizes the braking phases with acceleration phases while globally checking that the timetable energy consumption is reduced. Shifting acceleration phases by modifying the dwell times lengths thus increases the energy transfers and the regenerative braking usage.

We have shown that our heuristic algorithm performs better than the classical optimization methods used in the literature. On six small size benchmark instances on which MILP and CMA-ES could be run, we have shown that our heuristic algorithm computes solutions faster and of higher quality. In particular for the MILP formulation, the results computed by CPLEX are of lesser quality due to the inaccuracy of the linear approximation of the objective function. It also performs better than the state-of-the-art metaheuristic CMA-ES on these instances mainly for implementation reasons, in particular the possibility of incrementally computing the objective function over iterations. By adding a random noise on the output solutions, we have shown that our heuristics is computing robust timetables, justifying its use in an industrial context where small perturbations occur frequently.

Moreover, our dedicated heuristics is the only method able to solve a full day

timetable of 7679 variables for the *Sunday* configuration and 9585 variables for the *weekday* configuration, decreasing the global energy consumption respectively by 5.15% and 7.54%, and up to 8.91% by increasing the tolerances on dwell times, trip times and headways. These results show the applicability of this algorithm in an industrial context.

In this thesis, we have pointed out that a key factor to efficiently solve metro energy optimization rescheduling problems is how the instant power demand of the metro line is defined or approximated. While an analytical computation of the energy consumption is out of reach because of the complexity of the electricity equations, there is a trade-off on instant power demand approximations between accuracy and rapidity to deal with.

On one hand, electrical simulators are very accurate but are only suited to compute the global energy consumption of a static timetable. During optimization processes, the computation time make them costly to use. On the other hand and in order to be computed by MILP solvers, linear approximations based on the computation of the braking and acceleration phase overlapping times are not accurate enough ; the use of an auxiliary objective function instead of an energy consumption evaluation preventing some high quality solutions to be considered during the optimization.

We have proposed a non-linear approximation of the instant power demand, which provides an evaluation of the global energy consumption, fast enough to be used iteratively in an optimization algorithm. This approximation is based on a power flow – a particular generalized flow with lossy edges – that simulates the power transfers between braking and accelerating metros. To better model the line receptivity and the Joule effect losses, these transfers are attenuated by a distribution matrix which has been computed beforehand by an electrical simulator. Eventhough this model is not suitable for MILP due to non-linear equations, the algorithm computing this power flow gives a fast approximation of the instant power demand. The power flow model does not claim to be an accurate instant power demand evaluation, but rather a fast approximation leading the optimization process towards high quality solutions.

We have proposed to the community a set of benchmark timetables, available at <http://lifeware.inria.fr/wiki/COR14/Bench>, to get a standard framework that can be used to compare different methods and approaches to solve any metro energy optimization rescheduling problem. Our greedy heuristic algorithm is now implemented as part of the ATS of the General Electric Transportation signalling system, Figure 6.1 illustrating the look of the optimization module. It needs now to be validated with its future deployments on real cases. Our model unifying the timetable creation and its energy optimization presented in Chapter 3 is now under study to improve the regenerative braking usage.



Figure 6.1: Screenshot of the ATS optimization module of the General Electric Transportation signalling system.



# Bibliography

- [1] I. E. Agency, “Energy balance for world.”  
[http://www.iea.org/stats/balancetable.asp?COUNTRY\\_CODE=29](http://www.iea.org/stats/balancetable.asp?COUNTRY_CODE=29), 2009.
- [2] T. O’Toole, “Environment report,” tech. rep., London Underground, 2006.
- [3] W. Rose and T. Rouse, “Sub-national electricity consumption statistics and household energy distribution analysis for 2010,” tech. rep., Department of Energy and Climate Change, Mar. 2012.
- [4] J. Greatbanks, “Review of the discount for using regenerative braking,” *AEA Technology*, 2005.
- [5] M.-Y. Ayad, S. Pierfederici, S. Raël, and B. Davat, “Voltage regulated hybrid DC power source using supercapacitors as energy storage device,” *Energy Conversion and Management* 48, 2007.
- [6] D. Cornic, “Efficient recovery of braking energy through a reversible DC substation,” in *Electrical Systems for Aircraft, Railway and Ship Propulsion*, pp. 1–9, Oct. 2010.
- [7] A. González-Gil, R. Palacin, P. Batty, and J. Powell, “A systems approach to reduce urban rail energy consumption,” *Energy Conversion and Management*, vol. 80, pp. 509–524, 2014.
- [8] R. Chevrier, G. Marlière, B. Vulturescu, and J. Rodriguez, “Multi-objective evolutionary algorithm for speed tuning optimization with energy saving in railway: Application and case study,” in *RailRome 2011*, (Rome, Italy), 2011.
- [9] Y. Bocharnikov, A. Tobias, C. Roberts, S. Hillmansen, and C. Goodman, “Optimal driving strategy for traction energy saving on dc suburban railways,” *IET Electric Power Applications*, 2007.
- [10] S. Su, X. Li, T. Tang, and Z. Gao, “A subway train timetable optimization approach based on energy-efficient operation strategy,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, pp. 883–893, 2013.

- [11] N. Hansen and A. Ostermeier, “Completely derandomized self-adaptation in evolution strategies,” *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.
- [12] D. Fournier and D. Mulard, “Method and system for timetable optimization utilizing energy consumption factors,” Mar. 11 2014. US Patent App. 13/676,279.
- [13] D. Fournier and D. Mulard, “A method and system for timetable optimization utilizing energy consumption factors,” Jan. 2014. EP Patent App. EP20,130,174,982.
- [14] I. A. of Public Transport, “Reducing energy consumption in underground systems – an important contribution to protecting the environment.” 52nd International Congress of UITP, 1997.
- [15] T. Albrecht, “Reducing power peaks and energy consumption in rail transit systems by simultaneous metro running time control,” *Computers in Railways IX*, 2004.
- [16] K. M. Kim, S.-M. Oh, and M. Han, “A mathematical approach for reducing the maximum traction energy: The case of Korean MRT trains,” *IMECS 2010*, Mar. 2010.
- [17] ILOG, “V12.1: User’s manual for CPLEX,” *International Business Machines Corporation*, vol. 46, no. 53, p. 157, 2009.
- [18] G. Sierksma, *Linear and Integer Programming; Theory and Practice*. Marcel Dekker Inc., 1996.
- [19] F. Hillier and G. Lieberman, *Introduction to Operations Research*. McGraw-Hill Series in Industrial Engineering and Management Sciences, 9 ed., 2010.
- [20] J. Smith and Z. Taskin, “Tutorial guide to mixed-integer programming models and solution techniques,” in *Optimization in medicine and biology* (G. Lim and E. Lee, eds.), pp. 522–546, Auerbach Publications, 2008.
- [21] K. Kim, K. Kim, and M. Han, “A model and approaches for synchronized energy saving in timetabling,” *WCRR 2011*, May 2011.
- [22] J.-F. Chen, R.-L. Lin, and Y.-C. Liu, “Optimization of an MRT train schedule: Reducing maximum traction power by using genetic algorithms,” *IEEE Transactions on power systems*, vol. 20, pp. 1366–1372, Aug. 2005.
- [23] B. Sansó and P. Girard, “Trains scheduling desynchronization and power peak optimization in a subway system,” *IEEE*, 1995.
- [24] M. Miyatake and H. Ko, “Numerical analyses of minimum energy operation of multiple trains under DC power feeding circuit,” in *European Conference on Power Electronics and Applications*, pp. 1–10, Sept. 2007.

- [25] Y. Bocharnikov, A. Tobias, and C. Roberts, "Reduction of train and net energy consumption using genetic algorithms for trajectory optimisation," in *IET Conference on Railway Traction Systems*, pp. 1–5, Apr. 2010.
- [26] J. Xun, X. Yang, B. Ning, T. Tang, and W. Wang, "Coordinated train control in a fully automatic operation system for reducing energy consumption," *Computers in Railways XIII*, pp. 3–13, 2012.
- [27] A. Nasri, M. F. Moghadam, and H. Mokhtari, "Timetable optimization for maximum usage of regenerative energy of braking in electrical railway systems," *SPEEDAM 2010*, 2010.
- [28] C. Chang, Y. Phoa, W. Wang, and B. Thia, "Economy/regularity fuzzy-logic control of DC railway systems using event-driven approach," *IEE Proceedings of Electric Power Applications*, vol. 143, Jan. 1996.
- [29] A. Ramos, M. Peña, A. Fernández, and A. Cucala, "Mathematical programming approach to underground timetabling problem for maximizing time synchronization," in *International Conference on Industrial Engineering and Industrial Management*, pp. 88–98, Sept. 2007.
- [30] M. Peña, A. Fernández, A. P. Cucala, A. Ramos, and R. Pecharromán, "Optimal underground timetable design based on power flow for maximizing the use of regenerative-braking energy," *Journal of Rail and Rapid Transit*, vol. 226, pp. 397–408, July 2012.
- [31] X. Li and H. K. Lo, "An energy-efficient scheduling and speed control approach for metro rail operations," *Transportation Research Part B: Methodological*, vol. 64, no. 0, pp. 73–89, 2014.
- [32] D. Fournier, F. Fages, and D. Mulard, "A greedy heuristic for optimizing metro regenerative energy usage," in *Proceedings of the second international conference on railway technology: research, development and maintenance* (J. Pombo, ed.), (Ajaccio), Civil-Comp Press, 2014.
- [33] D. Fournier, T. Martinez, F. Fages, and D. Mulard, "Metro energy optimization through rescheduling: Mathematical model and heuristic algorithm compared to milp and cma-es," *Submitted to COR*, 2014.
- [34] X. Yang, X. Li, Z. Gao, H. Wang, and T. Tang, "A cooperative scheduling model for timetable optimization in subway systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, pp. 438–447, Mar. 2013.

- [35] X. Yang, B. Ning, X. Li, and T. Tang, "A two-objective timetable optimization model in subway systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, pp. 1–9, Feb. 2014.
- [36] A. Caprara, M. Fischetti, and P. Toth, "Modeling and solving the train timetabling problem," *Operations Research*, vol. 50, pp. 851–861, Sept./Oct. 2002.
- [37] P. Serafini and W. Ukovich, "A mathematical model for periodic scheduling problems," *Society for Industrial and Applied Mathematics*, vol. 2, pp. 550–581, Nov. 1989.
- [38] C. Liebchen, "Periodic timetable optimization in public transport," in *Operations Research Proceedings 2006* (K.-H. Waldmann and U. Stocker, eds.), vol. 2006 of *Operations Research Proceedings*, pp. 29–36, Springer Berlin Heidelberg, 2007.
- [39] P. Firpo and S. Savio, "Optimized train running curve for electrical energy saving in autotransformer supplied ac railway systems," *Electric Railways in United Europe*, Mar. 1995.
- [40] J.-C. Jong and E.-F. Chang, "Models for estimating energy consumption of electric trains," *Journal of the Eastern Asia Society for Transportation Studies*, vol. 6, pp. 278–291, 2005.
- [41] L. Wolsey, *Integer Programming*. Wiley-Interscience, 1 ed., Sept. 1998.
- [42] I. Amit and D. Goldfarb, "The timetable problem for railways," *Developments in Operations Research*, vol. 2, pp. 379–387, 1971.
- [43] C. Liebchen, "The first optimized railway timetable in practice," *Transportation Science*, vol. 42, no. 4, pp. 420–435, 2008.
- [44] E. Bampas, G. Kaouri, M. Lampis, and A. Pagourtzis, "Periodic metro scheduling," *ATMOS 2006*, 2006.
- [45] K. Nachtigall, "A branch and cut approach for periodic network programming," tech. rep., Hildesheim Univ. (Germany). Inst. fuer Mathematik, 1994.
- [46] J. Cury, F. Gomide, and M. Mendes, "A methodology for generation of optimal schedules for an underground railway system," in *18th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*, vol. 2, pp. 897–902, Dec. 1979.
- [47] A. Higgins, E. Kozan, and L. Ferreira, "Optimal scheduling of trains on a single line track," *Transportation Research Part B: Methodological*, vol. 30, no. 2, pp. 147–161, 1996.

- 
- [48] N. Bodhibrata and N. Manabendra, “Optimal design of timetables to maximize schedule reliability and minimize energy consumption, rolling stock and crew deployment,” in *Proceedings of 2nd UIC(International Congress of Railways) Energy Efficiency Conference*, Feb. 2004.
- [49] J. Yang, “Metro timetabling energy optimization,” Master’s thesis, École Polytechnique, Sept. 2014.
- [50] V. Cacchiani and P. Toth, “Nominal and robust train timetabling problems,” *European Journal of Operational Research*, vol. 219, pp. 727–737, 2012.
- [51] M. Khan and X. Zhou, “Stochastic optimization model and solution algorithm for robust double-track train-timetabling problem,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, pp. 81–89, Mar. 2010.



# List of Figures

1	Comparison of energy efficient solutions in terms of investment costs and energy savings potential [7]. . . . .	14
2.1	Electrical circuit associated to a metro line with five stations at time $i$ . It is composed of three electric substations represented as ideal voltage sources in $S_1$ , $S_3$ and $S_5$ , a braking metro arriving in $S_2$ and producing $P_{2,i}$ , and an accelerating metro departing from $S_5$ and consuming $P_{5,i}$ . The points in the network are linked by resistive cables. . . . .	40
2.2	Net power demand and production curve as a function of real time for a metro accelerating and braking between two stations. The points on the curve represent the sampling over discrete time. . . . .	42
2.3	Example of a generalized flow network. Flows are created by the source $t$ and absorbed by the sink $t'$ . The left hand side of the graph contains the vertices corresponding to the braking metros, which are connected to the vertices corresponding to the accelerating metros on the right hand side. Each edge of the graph is characterized by a (capacity,gain) couple. . . . .	45
3.1	Highlighted specific metro running back and forth and stopping in terminals : metro line graph with the time on the vertical axis and the line space on the horizontal axis. The leftmost and rightmost points represent both terminals. Each coloured line represents a physical train running between both terminals. . . . .	55
3.2	Terminal capacity focus : metro line graph with the time on the vertical axis and the line space on the horizontal axis. The leftmost and rightmost points represent both terminals. Each coloured line represents a physical train running between both terminals. . . . .	55
3.3	Initial handmade timetable : graph representing the headways in seconds in function of the metro departure times. The red line represents the metros running upstream and the blue line the metros running downstream. . . . .	60

3.4	Compiled timetable following the given headway pattern : graph representing the headways in seconds in function of the metro departure times. The red line represents the metros running upstream and the blue line the metros running downstream. . . . .	61
3.5	Compiled energy optimized timetable following the given headway pattern : graph representing the headways in seconds in function of the metro departure times. The red line represents the metros running upstream and the blue line the metros running downstream. . . . .	62
3.6	Overlapping time in seconds by intervals of 1 hour compared between the initial the timetable, in blue, and the energy optimized headway pattern timetable in red. . . . .	63
4.1	Initial timetable : braking and acceleration phases distribution over time on a part lasting 400 seconds. Each line represents a trip, green rectangles braking phases and red rectangles acceleration phases. . . . .	69
4.2	Optimized timetable : braking and acceleration phases distribution over time on a part lasting 400 seconds. Each line represents a trip, green rectangles braking phases and red rectangles acceleration phases. . . . .	70
4.3	Evolution of the global energy consumption over computation time in seconds on a sample timetable during the iterative optimization process. The first cross represents the global energy consumption of the original timetable and the following ones the global energy consumption by iterating the greedy heuristics. . . . .	72
5.1	Global energy consumption (in kW.h) evolution over time (in seconds) of 100 CMA-ES runs (blue lines) against a greedy heuristics iterative optimization run (red line) on benchmark instance <i>op1</i> . . . . .	76
5.2	Box plot of the statistical distribution of 100 randomized solutions generated with a noise (in seconds) on decision variables. Compared global energy consumption (in kW.h) between the initial timetable and the timetable optimized by the greedy heuristics for the benchmark instance <i>op1</i> . . . . .	80
5.3	<i>Weekday timetable</i> between 6am and 1am: energy consumption by intervals of 30 minutes compared between the initial the timetable, in red, and timetable computed by the greedy heuristics in green. . . . .	82
5.4	15 seconds tolerance on trips and headways <i>Sunday timetable</i> between 6am and 1am : energy consumption by intervals of 30 minutes compared between the initial timetable, in red, and the timetable computed by the greedy heuristics in green. . . . .	83

- 
- 5.5 20 seconds tolerance on trips and headways *Sunday timetable* between 6am and 1am : energy consumption by intervals of 30 minutes compared between the initial timetable, in red, and the timetable computed by the greedy heuristics in green. . . . . 84
- 6.1 Screenshot of the ATS optimization module of the General Electric Transportation signalling system. . . . . 87



# List of Tables

1.1	Some metro timetabling energy optimization problems from the literature, classified by the problem triple they solve and the corresponding timetable equations. . . . .	25
1.2	Decision problem associated with $(G, dep, lin)$ : example timetable encoding a SAT formula of $N$ clauses. Each cell represents the energy produced or consumed by the trip $T_t$ at time $i$ . . . . .	33
1.3	$(G, dep - dwe - int, lin)$ problem : example timetable encoding a SAT formula of $N$ clauses between time 0 and $6_{N-1}$ . Between time $6_N$ and $6_{N+3}$ , the dwell time of trip $T_0$ can be lengthened by 1 time unit, as well as the last interstation duration. . . . .	34
1.4	$(PP, dep, lin)$ problem : example timetable encoding a SAT formula of $N$ clauses. Each cell represents the energy produced or consumed by the trip $T_t$ at time $i$ . . . . .	36
2.1	Energy profile data for an interstation run. The net power demand is given in an arbitrary unit equal to 1 when the metro is at full throttle, for each timeslot of the interstation. . . . .	42
3.1	Lower and upper bounds on headways for every interval of the headway pattern, specifying the hours for each interval and the signification of the bounds. . . . .	60
4.1	Computation time in seconds of one run of the greedy heuristics without and with incremental computation of the objective function. The two implementations are compared on the six benchmark instances described in Section 5.1 representing typical off-peak (op) or peak (p) hours timetables. The lengths of the timetable instances are either 15 or 60 minutes and the number of decision variables $\#d_{t,s}$ are given. . . . .	71

5.1	Compared performance in computation time (Time in seconds) and energy consumption (Value in kW.h) between the average and best values found over 100 runs of CMA-ES and the greedy heuristics on six benchmark instances. The instances <i>op</i> represent an off-peak hour timetable and the instances <i>p</i> represent a peak hour timetable, both of either 15 minutes or 60 minutes long. . . . .	76
5.2	Compared performances over the MILP objective function (Value in s) between CPLEX and the greedy heuristics on six benchmark instances. The MILP solutions are given with their integrality gap, <i>optimal</i> standing for 0%. The instances <i>op</i> represent an off-peak hour timetable and the instances <i>p</i> represent a peak hour timetable, both of either 15 minutes or 60 minutes long. . . . .	77
5.3	Compared performances in computation time (Time in seconds) and energy consumption (Value in kW.h) between MILP and the greedy heuristics on six benchmark instances. The instances <i>op</i> represent an off-peak hour timetable and the instances <i>p</i> represent a peak hour timetable, both of either 15 minutes or 60 minutes long. . . . .	77
5.4	One second noise : statistical analysis of the greedy heuristics solutions robustness on six benchmark instances. Given the optimized solutions, 100 random solutions with noise on decision variables values are generated and evaluated regarding their global energy consumption in kW.h. . . . .	78
5.5	Two seconds noise : statistical analysis of the greedy heuristics solutions robustness on six benchmark instances. Given the optimized solutions, 100 random solutions with noise on decision variables values are generated and evaluated regarding their global energy consumption in kW.h. . . . .	79
5.6	Three seconds noise : statistical analysis of the greedy heuristics solutions robustness on six benchmark instances. Given the optimized solutions, 100 random solutions with noise on decision variables values are generated and evaluated regarding their global energy consumption in kW.h. . . . .	79
5.7	Compared performances in terms of energy consumption, given in kW.h, of CMA-ES, CPLEX and the greedy heuristics on three full size timetables. CMA-ES and CPLEX did not manage to output a solution. The ratios represent the energy savings compared to the initial timetable energy consumption. . . . .	81

# List of Algorithms

1	Power transfer ratios computation at time $i$ . . . . .	46
2	Trip allocation algorithm . . . . .	59
3	Acceleration phase shift and propagation . . . . .	67
4	Greedy heuristics optimization algorithm . . . . .	68
5	CMA-ES penalty function . . . . .	75



# List of Symbols

$\delta_t$	Departure time shift of trip $T_t$
$\Delta_{s,s'}$	Distribution ratio between stations $S_s$ and $S_{s'}$
$\epsilon$	Negligible net power demand generated by a metro
$\gamma(u, v)$	Edge $(u, v)$ gain
$\gamma_t, s, t', s'$	Boolean variable equal to one a power transfer occurs between the braking phase of trip $T_t$ at station $S_t(s)$ and the acceleration phase of trip $T_{t'}$ at station $S_{t'}(s')$
$\mathcal{N}(a_{t,s})$	Neighbourhood of the braking phase of $a_{t,s}$
$\phi$	Boolean formula in conjunctive normal form
$\rho$	Third rail resistivity
$\theta$	Noise amplitude
$\underline{acc}_{t,s}, \overline{acc}_{t,s}$	Lower and upper bounds of the acceleration phase of trip $T_t$ departing from station $S_t(s)$
$\underline{brk}_{t,s}, \overline{brk}_{t,s}$	Lower and upper bounds of the braking phase of trip $T_t$ arriving at station $S_t(s)$
$\underline{dep}_t, \overline{dep}_t$	Lower and upper bounds of the departure time of trip $T_t$ at its departure terminal
$\underline{dwe}_{t,s}, \overline{dwe}_{t,s}$	Lower and upper bounds of the dwell time of trip $T_t$ at station $S_t(s)$
$\underline{hdw}_{t,s}, \overline{hdw}_{t,s}$	Lower and upper bounds of the headway between trip $T_t$ and trip $T_{t-1}$ at station $S_t(s)$
$\underline{int}_{t,s}, \overline{int}_{t,s}$	Lower and upper bounds of the interstation time of trip $T_t$ between stations $S_t(s)$ and $S_t(s+1)$
$\underline{trt}_t, \overline{trt}_t$	Lower and upper bounds of the trip time of trip $T_t$
$a$	Third rail section
$a_{t,s}^{brk}$	Beginning of braking phase of trip $T_t$ arriving at station $S_t(s)$
$a_{t,s}$	Arrival time of trip $T_t$ at station $S_t(s)$

---

$b_{t,h}$	Boolean variable equal to 1 if trip $T_t$ starts after $tra_h$
$c(u, v)$	Edge $(u, v)$ capacity
$c_i$	$i^{th}$ clause of $\Phi$
$cap$	Terminal capacity
$CP_{\mathcal{T}\mathcal{T}, P_{MAX}}$	Number of power peaks higher than $P_{MAX}$ of timetable $\mathcal{T}\mathcal{T}$
$d_{t,s}^{acc}$	End of acceleration phase of trip $T_t$ departing from station $S_t(s)$
$d_{t,s}$	Departure time of trip $T_t$ at station $S_t(s)$
$f(u, v)$	Edge $(u, v)$ flow
$G_{\mathcal{T}\mathcal{T}}$	Global energy consumption of timetable $\mathcal{T}\mathcal{T}$
$i_{s,i}^{ESS}$	Current flowing between station $S_s$ and the electric substation
$i_{s,i}^{MET}$	Current flowing between a metro located in station $S_s$ and the network
$i_{s,i}$	Current flowing between stations $S_s$ and $S_{s+1}$ at time $i$
$l$	Third rail length
$low_h$	Lower bound for the headway interval $h$
$M$	Number of trips
$mhwh$	Minimum headway authorized on the metro line
$N$	Number of stations
$nit$	Number of headway intervals
$next_t$	Trip $T_{t'}$ performed right after the trip $T_t$ by the same rolling stock
$Ot, s, t', s'$	Overlapping time between the braking phase of trip $T_t$ at station $S_t(s)$ and the acceleration phase of trip $T_{t'}$ at station $S_{t'}(s')$
$p(\mathcal{T}\mathcal{T})$	Penalty function of the timetable $\mathcal{T}\mathcal{T}$
$P_i$	Instant power demand at time $i$
$P_{MAX}$	Power peak threshold
$P_{s,i}$	Net power demand or production generated by a metro at time $i$ at station $S_s$
$P_{t,i}$	Net power demand or production generated by trip $T_t$ at time $i$
$PP_{\mathcal{T}\mathcal{T}}$	Maximum power peak of timetable $\mathcal{T}\mathcal{T}$
$R^{ESS}$	Internal resistance of the electric substation
$R_s$	Third rail resistance between stations $S_s$ and $S_{s+1}$

---

$rst_t$	Rolling stock performing the trip $T_t$
$S_s$	Station $s$
$S_t(s)$	$s^{th}$ station crossed by trip $T_t$
$T_t$	Trip number $t$
$tbm$	Turn back manoeuvre duration
$tra_h$	Starting time of interval $h$
$upp_h$	Upper bound for the headway interval $h$
$V^{ESS}$	Voltage supplied by the electric substation
$v_{s,i}$	Electric potential at station $S_s$ at time $i$
$x_{s,s',i}$	Power transfer ratio between stations $S_s$ and $S_{s'}$ at time $i$

