Metro Energy Optimization through Rescheduling: Mathematical Models and Heuristic Algorithm Compared to MILP and CMA-ES

David Fournier, Thierry Martinez, François Fages, Denis Mulard

Metro Energy Optimization through Rescheduling: Mathematical Models and Heuristic Algorithm Compared to MILP and CMA-ES☆

David Fournier[a,b], Thierry Martinez[a], François Fages[a], Denis Mulard[b]

[a] *Inria Saclay - EPI Lifeware - 1 rue Honor d'Estienne dOrves - Campus de l'cole Polytechnique 91120 Palaiseau (France)*
[b] *General Electric Transportation ITS Delta - Tour Europlaza 28D5, 20 avenue André Prothin 92063 Paris La Défense Cedex (France)*

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 paper, we use a mathematical timetable model to classify various metro energy optimization 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 trains along the timetable with accelerating trains in their time neighbourhood, using a non-linear approximation of energy transfers. On a benchmark of the litterature composed of six small size timetables, we show that our greedy heuristics performs better than CPLEX using a MILP formulation of the problem with 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, our dedicated algorithm computes solutions with a reduction of energy consumption ranging from 5% to 9%.

Keywords: Energy Optimization, Timetable Optimization, Regenerative

☆This article is an extended version of an article presented at the Railways 2014 conference [1]

Email addresses: `david.l.l.fournier@gmail.com` (David Fournier), `thierry.martinez@inria.fr` (Thierry Martinez), `francois.fages@inria.fr` (François Fages), `denis.mulard@ge.com` (Denis Mulard)

Braking, Mass Rapid Transit, Greedy Heuristics

1. Introduction

Energy consumption is a major issue for the future and has been the subject of increasing research activities over the last years. The total energy consumed in transportation is estimated to represent 27% of the world energy production [2]. In 2006, the London Underground consumed 1173 GW.h [3], representing 2.8% of the Great London total electricity consumption [4]. The transportation field is thus of particular importance and industrial companies try to optimize energy by different means, in particular in mass rapid transit such as metros.

Nowadays, almost all metros have *energy regenerative braking* systems. These devices turn the electric motors into generators during braking, and produce electric energy. It has been shown that the raw energy discount provided by this technology is about 16.5% [5]. 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 [6]. 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. However, super capacitors, as well as revertible ESSs, are expensive equipment to buy and maintain, and may not be economically justified.

Another way to use the regenerative energy returned to the DC line, without requiring any extra equipment, is to synchronize the braking of a metro with the acceleration of another metro in its close neighbourhood on the DC line. 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. The objective function may then combine two criteria: the minimization of power peaks, and the minimization of the global energy consumption during the day. A power peak occurs when too many metros are accelerating at the same time, which may cause two problems. First, a metro network is sized with some maximum power capacity. When the demand exceeds that threshold, a control system prevents the destruction of the electric equipment by a momentary shut down of a part of the network, dropping the quality of service. Second, more than 80% of metro companies pay their energy provider for a certain amount of energy per given time period, typically around 15 minutes [7]. When the energy consumed exceeds a certain limit, the metro company pays a fine to the electricity provider.

Albrecht has shown in [8] 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 in [8]. Kim et al. have proposed in [9] to optimize the metro departure times in terminals instead of reserve times. They have partially solved a simplified model of this problem using MILP. However, their approximation is not precise enough for a real application since regular timetables are typically second-accurate, whereas their model has a precision of 15 seconds. In [10], Chen et al. 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 that chooses between only either short or long stopping times for metros in each station.

Concerning the minimization of the total energy consumption on the line, Nasri et al. have shown in [11] 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 be tested on a pilot system, consisting of 4 trains and 4 stations. On a more realistic size problem, Peña et al. [12] have proposed a MILP model to optimize a night shift timetable for the metro of Madrid by maximizing the overlap of braking and acceleration phases of different metros. Their measurements on field have shown that the optimized timetable could reduce the global energy consumption by 3%.

In this paper, we focus on the problem of minimizing the global energy consumption of a metro timetable by modifying the dwell times in stations. We present a heuristic algorithm which gives better results than general purpose optimization methods. In particular, we compare our algorithm with MILP and with a state-of-the-art evolutionary algorithm called the covariance matrix adaptation evolution strategy (CMA-ES) [13]. The MILP formulation of Peña et al. [12] maximizes the overlapping time between accelerating and braking phases and not directly the global energy consumption. Also, to limit the number of binary variables, their model considers only one-to-one pairing between braking and accelerating metros, whereas our objective function takes into account the redistribution of the regenerative energy to several accelerating metros. We show that the MILP approximations of the objective function lead to solutions of lesser quality, eventhough CPLEX [14] is able to prove the optimality of the computed solutions. Furthermore, this approach does not scale-up to real data which lead to instances of 10000 decision variables by running out of memory, while handled with our heuristics in 20 minutes. CMA-ES using the same non-linear approximation as us does not scale up on real data either, failing at evaluating the initial population objective functions. On small size instances, our heuristic algorithm also performs better than CMA-ES mainly for the implementation reason that our algorithm computes the objective function incrementally .

The rest of the paper is organised as follows. Section 2 presents a mathematical model of metro timetables taking into account both time constraints and energy optimization objectives. Section 3 uses this model to classify various related problems of the literature, and prove their NP-hardness by polynomial reductions of SAT. Then in Section 4 we model the instant power demand as an electrical network which simulates the energy consumption of the metro line, and describe in Section 5 a power flow approximation based on a distribution matrix. We propose an algorithm to compute this power flow and use it as an objective function for CMA-ES and in our heuristics. This formulation is also used to derive the linear approximation of the objective function for MILP. Section 6 describes our heuristics for minimizing the global energy consumption of a metro line by modifying solely the dwell times in stations. Section 7 compares the results of this heuristics on a benchmark of six small size instances, and shows that it gives better results than MILP and CMA-ES in both computation time and quality of the solutions. The set of these instances is available at `http://lifeware.inria.fr/wiki/COR14/Bench`. Then in Section 8 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.

2. Timetable Model

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

1. define and classify different metro energy optimization problems from the literature (Section 3),
2. define the related MILP models,
3. define the instant power demand function (Section 4 and 5),
4. and define our greedy heuristic algorithm (Section 6).

For the sake of simplicity, we assume that the metro line is composed of N stations, $S = \{S_1, ..., S_N\}$ and the metro timetable is represented as a sequence of M trips, with M being pair, $T = (T_1, ..., T_{\frac{M}{2}}, T_{\frac{M}{2}+1}, ..., T_M)$. We assume that the trips $T_1, ..., T_{\frac{M}{2}}$ cross all stations in the upstream sequence, $(S_1, ..., S_N)$, and the trips $T_{\frac{M}{2}+1}, ..., T_M$ cross all stations in the downstream sequence, $(S_N, ..., S_1)$. We note $S_t(s)$ the s^{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 $\frac{M}{2} + 1 \leq t \leq M$.

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.

2.1. 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 $I = \{0, 1, ..., I_{END}\}$ 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 $2 \leq s \leq N$,
- $d^{acc}_{t,s} \in I$ 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 $1 \leq s \leq N-1$,
- $a^{brk}_{t,s} \in I$ 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 $2 \leq s \leq N$.

2.2. Auxiliary Variables

To simplify the equations, it is useful to introduce the following auxiliary variables as functions of the variables. These variables have the dimension of a time in I:

- the interstation time for a given trip,

$$int_{t,s} = a_{t,s+1} - d_{t,s} \quad 1 \leq t \leq M,\ 1 \leq s \leq N-1,$$

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

$$dwe_{t,s} = d_{t,s} - a_{t,s} \quad 1 \leq t \leq M,\ 2 \leq s \leq N-1,$$

- the total trip time,

$$trt_t = int_{t,1} + \sum_{k=2}^{N-1} (dwe_{t,k} + int_{t,k}) = a_{t,N} - d_{t,1} \quad 1 \leq t \leq M,$$

- the time interval, called headway, between two successive trips running in the same direction in a given station,

$$hdw_{t,s} = d_{t,s} - d_{t-1,s} \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M],\ 1 \leq s \leq N.$$

- The duration of the acceleration phase for a given trip to leave a given station,
$$acc_{t,s} = d^{acc}_{t,s} - d_{t,s} \quad 1 \leq t \leq M,\ 1 \leq s \leq N-1,$$
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 duration of the braking phase for a given trip before arriving at a given stations,
$$brk_{t,s} = a_{t,s} - a^{brk}_{t,s} \quad 1 \leq t \leq M,\ 2 \leq s \leq N.$$

2.3. Constraints

Each trip t must leave its departure terminal within some bounds set by the metro company according to the quality of service provided to passengers,

$$\underline{d_{t,1}} \leq d_{t,1} \leq \overline{d_{t,1}} \quad 1 \leq t \leq M. \tag{1}$$

The interstation times are bound according to the different speeds – e.g. economical, nominal or full throttle – a metro can take,

$$\underline{int_{t,s}} \leq int_{t,s} \leq \overline{int_{t,s}} \quad 1 \leq t \leq M,\ 1 \leq s \leq N-1, \tag{2}$$

The dwell times are bound according to a minimum quality of service for the passengers,

$$\underline{dwe_{t,s}} \leq dwe_{t,s} \leq \overline{dwe_{t,s}} \quad 1 \leq t \leq M,\ 2 \leq s \leq N-1, \tag{3}$$

The acceleration and braking phases are also bound according to the possible interstation times for a given station :

$$\underline{acc_{t,s}} \leq acc_{t,s} \leq \overline{acc_{t,s}} \quad 1 \leq t \leq M,\ 1 \leq s \leq N-1, \tag{4}$$

$$\underline{brk_{t,s}} \leq brk_{t,s} \leq \overline{brk_{t,s}} \quad 1 \leq t \leq M,\ 2 \leq s \leq N, \tag{5}$$

The global trip time is also bound to ensure the feasibility of the timetable and the quality of service to the passengers,

$$\underline{trt_t} \leq trt_t \leq \overline{trt_t} \quad 1 \leq t \leq M, \tag{6}$$

Finally, the headways are bound according to security requirements and quality of service,

$$\underline{hdw_{t,s}} \leq hdw_{t,s} \leq \overline{hdw_{t,s}} \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2}+2, M],\ 1 \leq s \leq N. \tag{7}$$

It is worth noticing that the headways must be checked 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.

2.4. 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 definition of 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 $G_{\mathcal{TT}}$ of all trips,

$$G_{\mathcal{TT}} = \sum_{i=0}^{I_{END}} P_i, \tag{8}$$

or the maximum power peak

$$PP_{\mathcal{TT}} = \max_{i \in I} P_i. \tag{9}$$

It is worth noticing that the objective function $PP_{\mathcal{TT}}$ is widely used in the literature [8, 9, 10, 15, 16], but that a more realistic function would be the number of times that the power exceeds a certain threshold P_{MAX},

$$CP_{\mathcal{TT}} = \mathrm{card}(i \mid P_i > P_{MAX}). \tag{10}$$

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 Section 4 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 [12]. Before that, the timetable model can already be used to classify several variants of the problem studied in the literature.

3. Classification and Complexity of Metro Timetabling Energy Optimization Problems

3.1. Problem 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. [17] proposed a classification of the methods used to solve the problems and Li et al. [18] listed without apparent classification some papers in the literature, detailing the decision variables or the algorithms used. To date, there is no formal classification of metro energy optimization problems. We propose a classification based on the previous timetable by a triple

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

denoting the choice of the objective function (G, PP or CP), the decision variables (departure times, dwell times, interstation times or any combination

Problem	Equations	References
(PP, dwe, sim)	3, 6, 7, 4, 5, 9,	Chen et al. [10]
$(PP, dep-dwe, sim)$	1, 3, 6, 7, 4, 5, 9	Sanso et al. [15]
$(PP, dwe-int, sim)$	2, 3, 6, 7, 4, 5, 9	Albrecht et al. [8]
(PP, dep, lin)	1, 6, 7, 9	Kim et al. [9, 16]
$(G, dwe, nonlin)$	3, 6, 7, 8	Fournier et al. [1, 19]
(G, dwe, sim)	3, 6, 7, 8	Nasri et al. [11]
$(G, dep-dwe, lin)$	1, 3, 6, 7, 8	Peña et al. [12]
(G, int, sim)	2, 6, 7, 4, 5, 8	Bocharnikov et al. [20]

Table 1: Some metro timetabling energy optimization problems from the literature, classified by the problem triple they solve and the corresponding timetable equations.

of them) and the instant power demand evaluation (by an electrical simulator, a non-linear approximation or a linear approximation).
Table 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 dep_t, the interstation times $int_{t,s}$, and the braking and acceleration phases $brk_{t,s}$ and $acc_{t,s}$ 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.

3.2. Complexity

First of all, one can remark that in absence of objective function, the timetable satisfiability problem is polynomial since all the equations of the timetable model are linear. Caprara in *et al.* showed in [21] 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 [22] that the problem of periodically scheduling trains with precedence constraints is NP-complete. Here we show the NP-hardness of several energy optimization timetable problems by polynomial reductions of SAT.

Theorem 1. *The problem (G, dep, lin) is NP-hard.*

Corollary 1. *The problem $(G, dep, nonlin)$ is NP-hard.*

Corollary 2. *The problem $(G, dep-dwe-int, lin)$ is NP-hard.*

The proofs are given in appendix. The idea of the proof of the first result is to construct a polynomial reduction of SAT to a particular (G, dep, lin) problem. A particular timetable is constructed such that all the acceleration phases can be synchronized with the periodic braking phases of only one special metro x_0.

In this construction, it is possible to delay the departure time of each metro different from x_0 by one time unit only. This ensures that one unit of energy is saved at this time. A Boolean formula in conjunctive normal form (CNF) with c clauses is then satisfiable if and only if c units of energy can be saved in the timetable.

Example 1. Let us consider the SAT formula $(x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee y)$. The constructed timetable contains four trips x_0, x_1, x_2 and x_3, and is divided in three periods during which metro x_0 has the same behaviour. For each braking phase of x_0 occurring at times i_1, i_7 and i_{13}, either x_1, x_2 or x_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, coasting or dwelling, producing or consuming nothing. Each trip departure time can be delayed by one time unit. The formula is satisfiable if and only if for each braking phase of x_0, one other metro can synchronize its acceleration phase with it. The timetable with the energy consumed or produced by each metro at each time is shown in Table 2.

$\mathcal{TT}$	i_0	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}	i_{12}	i_{13}	i_{14}	i_{15}	i_{16}	i_{17}
x_0	1	-1	0	0	0	0	1	-1	0	0	0	0	1	-1	0	0	0	0
x_1	0	0	1	0	-1	0	1	0	0	0	-1	0	0	1	0	0	-1	0
x_2	1	0	0	0	-1	0	0	1	0	0	-1	0	1	0	0	0	-1	0
x_3	1	0	0	0	-1	0	1	0	0	0	-1	0	0	0	1	0	-1	0

Table 2: Timetable energy problem encoding the satisfiability of $(x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_2)$. Each cell represents the power produced (-1) or consumed (1) by the trips x_i at each time. In this example, to save the 3 energy units produced by x_0, and satisfy the SAT formula in this encoding, it suffices to delay x_3 by one time unit since x_1 and x_2 already save the 2 other energy units.

4. Instant Power Demand Model

The energy consumption of a metro line can be computed with an 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 1 depicts the circuit associated to a small network with five stations, three ESSs, one braking metro and one accelerating metro.

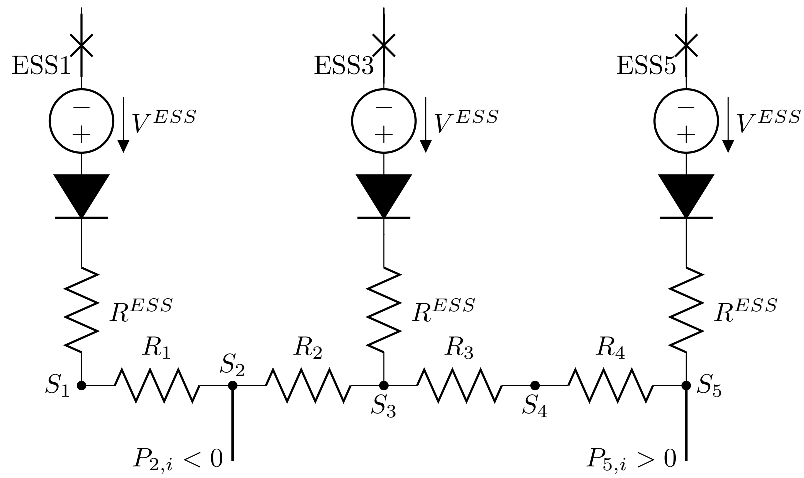

Figure 1: Electrical circuit associated to a metro line with five stations at time i. It is composed of three electric sub-stations in S_1, S_3 and S_5, a braking metro arriving in S_2 and producing $P_{2,i}$, and an accelerating metro departing from S_5 and consuming $P_{5,i}$. The points in the network are linked by resistive cables.

4.1. 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 [10] or 1500V [12].
- $R^{ESS} \in \mathbb{R}^+$ 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 $R = \frac{\rho.l}{a}$, ρ being the resistivity of the third rail, a its section and l its length.

4.2. Timetable Parameters

$P_{s,i} \in \mathbb{R}$ is the net power demand or production, involved by a metro, at time i for every 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

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

The precise power values are supposed to be known beforehand by direct measurement on the metro motors. Figure 2 illustrates the net power demand and production of a metro during an interstation run. In a typical example depicted in Figure 2, the parameters $P_{s,i}$ are given by the power curve for each time point i.

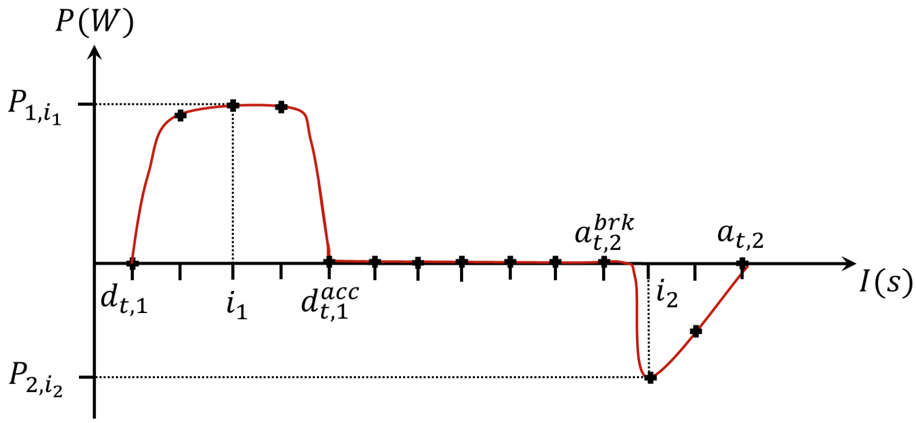

Figure 2: Net power demand and production curve as a function of real time for a metro accelerating and braking between two stations. The points on the curve represent the sampling over discrete time.

4.3. 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 $i \in I$.

4.4. Constraints and Instant Power Demand Value

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

$$v_{s,i} - v_{s+1,i} = R_s.i_{s,i} \quad 1 \leq s \leq N-1, i \in I \tag{11}$$

$$V_{ESS} - v_{s,i} = R_{ESS}.i_{s,i}^{ESS} \quad 1 \leq s \leq N, i \in I \tag{12}$$

Kirchhoff's current law gives

$$i_{s,i} + i_{s+1,i} + i^{ESS}_{s,i} + i^{MET}_{s,i} = 0 \quad 1 \leq s \leq N, i \in I \tag{13}$$

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

$$P_{s,i} = v_{s,i}.i^{MET}_{s,i} \quad 1 \leq s \leq N, i \in I \tag{14}$$

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:

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

This value represents the *net power consumption* of the metro line. The currents i^{ESS}_s 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.

5. 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 [1, 9, 12, 16, 19]. 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* Δ 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.

5.1. Power Flow 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 a flow $f(u, v)$ is entering at vertex v, then $\gamma(u, v).f(u, v)$ is going out from v:

$$\sum_{v \in V | (u,v) \in E} \gamma(u, v) f(u, v) = \sum_{v \in V | (v,u) \in E} f(v, u), \tag{15}$$

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

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

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

$$P_i = \sum_{s' \in V_i^{acc}} [c(s', t')_i - f(s', t')_i] \tag{16}$$

Figure 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.

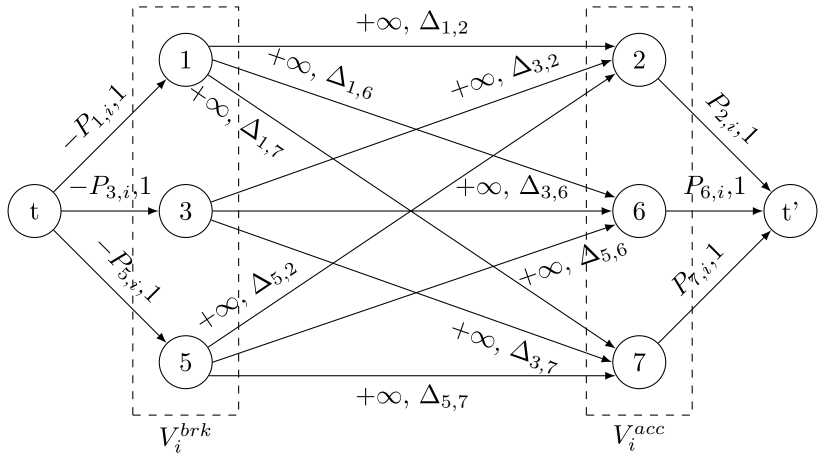

Figure 3: Example of a generalized flow network. Flows are created by the source t and absorbed by the sink t'. The left hand side of the graph contains the vertices corresponding to the braking metros, which are connected to the vertices corresponding to the accelerating metros on the right hand side. Each edge of the graph is characterized by a (capacity,gain) couple.

5.2. Power Flow Algorithm

According to the equation (16), 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 (15)

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

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 $P_{s,i}$:

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

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}, $\Delta_{s,s'}$, $P_{s,i}$

Initialize vector $x_{s,s',i} \leftarrow 0$
while $V_i^{brk} \neq \emptyset$ **do**
    Choose randomly $s \in V_i^{brk}$
    $P_{INIT} \leftarrow P_{s,i}$
    while $P_{s,i} < 0$ **do**
        if $V_i^{acc} \neq \emptyset$ **then**
            Choose $s' \in V_i^{acc}$ s.t. $s' = \arg\max_{s' \in V_i^{acc}}(\Delta_{s,s'})$
            if $-P_{s,i}.\Delta_{s,s'} > P_{s',i}$ **then**
                $x_{s,s',i} \leftarrow (P_{s',i}/\Delta_{s,s'})/P_{INIT}$
                $P_{s,i} \leftarrow P_{s,i} + P_{s',i}/\Delta_{s,s'}$
                $V_i^{acc} \leftarrow V_i^{acc} \backslash \{s'\}$
            else
                $x_{s,s',i} \leftarrow P_{s,i}/P_{INIT}$
                $P_{s',i} \leftarrow P_{s',i} + P_{s,i}.\Delta_{s,s'}$
                $P_{s,i} \leftarrow 0$
            end if
        else
            $P_{s,i} \leftarrow 0$
        end if
    end while
    $V_i^{brk} \leftarrow V_i^{brk} \backslash \{s\}$
end while
return vector $x_{s,s',i}$

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

$$P_i = \sum_{s' \in V_i^{acc}} P_{s',i} + \sum_{s' \in V_i^{acc}} \sum_{s \in V_i^{brk}} (P_{s,i}.x_{s,s',i}.\Delta_{s,s'}), \tag{17}$$

5.3. Linear Approximation for MILP

To avoid introducing too many variables in MILP, Peña proposed in [12] 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 $T_t,\ 1 \le t \le M$ at station $S_t(s),\ 2 \le s \le N$, and the acceleration phase of the trip $T_{t'},\ 1 \le t' \le M$ at station $S_{t'}(s'),\ 1 \le s' \le N-1$.

Constraint

$$\sum_{t=1}^{M}\sum_{s=1}^{N} \gamma_{t,s,t',s'} \le 1 \quad 1 \le t' \le M,\ 1 \le s' \le N \tag{18}$$

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

$$\sum_{t'=1}^{M}\sum_{s'=1}^{N} \gamma_{t,s,t',s'} \le 1 \quad 1 \le t \le M,\ 1 \le s \le N \tag{19}$$

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

$$O_{t,s,t',s'} \le brk_{t,s}.\gamma_{t,s,t',s'} \quad 1 \le t < t' \le M,\ 2 \le s \le N,\ 1 \le s' \le N-1 \tag{20}$$

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

$$O_{t,s,t',s'} \le d^{acc}_{t',s'} - a^{brk}_{t,s} + m(1 - \gamma_{t,s,t',s'}) \tag{21}$$

$$O_{t,s,t',s'} \le a_{t,s} - d_{t',s'} + m(1 - \gamma_{t,s,t',s'}) \tag{22}$$

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

ensure that the overlapping time is the minimum value of $d^{acc}_{v,u} - a^{brk}_{t,s}$ 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 [12]

$$\text{maximize} \sum_{t=1}^{M}\sum_{s=1}^{N}\sum_{t'=1}^{M}\sum_{s'=1}^{N} (O_{t,s,t',s'}.\Delta_{s,s'}) \tag{23}$$

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:

- $M^2.N^2$ boolean variables γ, $M^2.N^2$ overlapping times variables O and $M.N$ dwell times variables dwe,
- $3M^2.N^2 + 2M.N$ MILP constraints (Equations 18, 19, 20, 21, 22) and $5M.N + 2M$ timetable constraints (Equations 1, 2, 3, 4, 5, 6, 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 [12] to remove irrelevant constraints and variables. In particular, the number of overlapping variables $O_{t,s,t',s'}$ to consider can be reduced by noticing that two trips T_t and $T_{t'}$ such that $d_{t,1} > d_{t',1} + trt_{t'}$ cannot overlap in any case. However, this pre-processing is not sufficient to handle real data timetables as we will show in Section 8.

6. Greedy Heuristics Algorithm for $(G, dwe, nonlin)$

The idea of our heuristic algorithm is to shift the acceleration phases to synchronize them with braking phases and to recompute 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.

6.1. 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:

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

6.2. Acceleration Phase Shift Function

The acceleration phase shift function 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 $a^{brk}_{t',s'}$ of a trip $T_{t'}$ at station $S_{t'}(s')$, and by fixing $acc_{t,s}$. 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: $\text{Shift}(d_{t,s}, a^{brk}_{t',s'}) =$

$$\begin{cases} d_{t,s} := \max(a^{brk}_{t',s'}, d_{t,s} - \max(\underline{dwe_{t,s}}, \underline{trt_t}, \underline{hdw_{t,s}})) & \text{if } d_{t,s} > a^{brk}_{t',s'} \\ d_{t,s} := \min(a^{brk}_{t',s'}, d_{t,s} - \min(\overline{dwe_{t,s}}, \overline{trt_t}, \overline{hdw_{t,s}})) & \text{if } d_{t,s} < a^{brk}_{t',s'} \end{cases}$$

Modifying $d_{t,s}$ consists thus in modifying the duration of $dwe_{t,s}$, which will shift the future arrivals and departures $a_{t,s'}$ and $d_{t,s'}$ such that $s' > s$.

6.3. 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 $\text{Shift}(d_{t',s'}, a^{brk}_{t,s})$ 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. The following pseudo-code summarizes the greedy heuristics optimization algorithm:

Algorithm 2 Greedy heuristics optimization algorithm

Require: $\mathcal{TT}$

Sort $a_{t,s},\ 1 \le t \le M, 2 \le s \le N$ in chronological order
for all $a_{t,s}$ **do**
    Compute initial objective function $G^{INIT}_{\mathcal{TT}}$
    Initialize best objective function $G^{BEST}_{\mathcal{TT}} = G^{INIT}_{\mathcal{TT}}$
    Initialize best shift $d^{BEST} = 0$
    Compute $\mathcal{N}(a_{t,s})$
    for all $d_{t',s'} \in \mathcal{N}(a_{t,s})$ **do**
        $d^{INIT} \leftarrow d_{t',s'}$
        $\text{Shift}(d_{t',s'}, a_{t,s})$
        Compute $G_{\mathcal{TT}}$
        if $G_{\mathcal{TT}} < G^{BEST}_{\mathcal{TT}}$ **then**
            $G^{BEST}_{\mathcal{TT}} \leftarrow G_{\mathcal{TT}}$
            $d^{BEST} \leftarrow d_{t',s'}$
        end if
        $d_{t',s'} \leftarrow d^{INIT}$
    end for
    if $d^{BEST} \neq 0$ **then**
        $\text{Shift}(d^{BEST}, a_{t,s})$
    end if
end for
return $\mathcal{TT},\ G^{BEST}_{\mathcal{TT}}$

6.4. Incremental Computation of the Objective Function

Equation (17) 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:

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

The *incremental computation* avoids recomputing known values, which increases the computation time of the algorithm by an order of magnitude as shown in Table 3 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 3: 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 7 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.

6.5. 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 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.

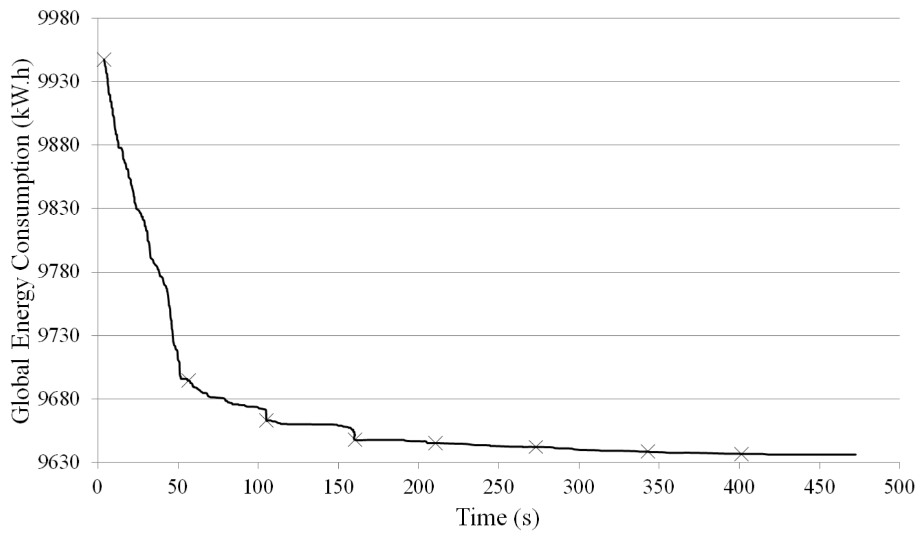

Figure 4: Evolution of the global energy consumption over computation time in seconds on a sample timetable during the iterative optimization process. The first cross represents the global energy consumption of the original timetable and the following ones the global energy consumption by iterating the greedy heuristics.

7. Performance Results Compared to MILP and CMA-ES

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 [13] 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 4. The greedy heuristics used the incremental computation of the objective function and iterative optimization.

7.1. 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 :

$$
\begin{aligned}
\underline{dwe_{t,s}} &= -3 \quad 1 \leq t \leq M,\ 1 \leq s \leq N \\
\overline{dwe_{t,s}} &= 9 \quad 1 \leq t \leq M,\ 1 \leq s \leq N \\
\underline{trt_t} &= -30 \quad 1 \leq t \leq M \\
\overline{trt_t} &= 30 \quad 1 \leq t \leq M \\
\underline{hdw_{t,s}} &= -30 \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M],\ 1 \leq s \leq N \\
\overline{hdw_{t,s}} &= 30 \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M],\ 1 \leq s \leq N
\end{aligned}
$$

7.2. 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) [13] 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. The CMA-ES

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. A quadratic penalty function is added to the objective function for each variable out of its domain, to enforce the algorithm to search solutions within the given tolerances.

Table 4 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.

7.3. Comparison with MILP

As shown in Table 5, CPLEX is able to prove the optimality on the four smallest instances and outperforms the greedy heuristics on the linear objective

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	**2394**	**45.6**
op2	15 min	129	2516	2402	2388	223	**2376**	**38.0**
op3	60 min	449	9956	9724	9716	761	**9556**	**648**
p1	15 min	173	3433	3300	3285	503	**3262**	**178**
p2	15 min	186	3651	3516	3483	669	**3442**	**291**
p3	60 min	670	13067	**12696**	**12675**	**1030**	12713	1500

Table 4: 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.

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

Table 5: 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.

function (Equation 23). However, when comparing both optimization methods on the objective function computed with the electrical simulator (Table 6), our greedy heuristics performs better on five of the six instances.

Instance	Length	$\#d_{t,s}$	Initial Value	MILP		Greedy Heuristics	
				Value	Time	Value	Time
op1	15 min	127	2514	2427	**0.72**	**2394**	45.6
op2	15 min	129	2516	2419	**1.00**	**2376**	38.0
op3	60 min	449	9956	9579	1500	**9556**	**648**
p1	15 min	173	3433	3281	460	**3262**	**178**
p2	15 min	186	3651	3494	**82.5**	**3442**	291
p3	60 min	670	13067	**12711**	**1500**	12713	**1500**

Table 6: 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.

This is due to the fact that the linear objective function is less accurate that the one used in the greedy heuristics (Equation 17). The main differences between these two objective functions are that the MILP model is only able to pair one braking with one acceleration (Equations 18, 19), when the power flow objective function is able to dispatch dynamically to different braking and accelerations over time as described in Section 5.2. Thus eventhough our algorithm does not prove optimality, it better approximates the real behaviour of the electricity flows and leads to better solutions.

8. 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 instance is to relate with the size of the problem handled in [12] which was containing only 17850 constraints and 13860 variables, whose 4780 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:

$$\begin{aligned}
\underline{dwe_{t,s}} &= -3 \quad 1 \leq t \leq M,\ 1 \leq s \leq N \\
\overline{dwe_{t,s}} &= 3 \quad 1 \leq t \leq M,\ 1 \leq s \leq N \\
\underline{trt_t} &= -15 \quad 1 \leq t \leq M \\
\overline{trt_t} &= 15 \quad 1 \leq t \leq M \\
\underline{hdw_{t,s}} &= -15 \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M],\ 1 \leq s \leq N \\
\overline{hdw_{t,s}} &= 15 \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M],\ 1 \leq s \leq N
\end{aligned}$$

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

$$\begin{aligned}
\underline{trt_t} &= -20 \quad 1 \leq t \leq M \\
\overline{trt_t} &= 20 \quad 1 \leq t \leq M \\
\underline{hdw_{t,s}} &= -20 \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M],\ 1 \leq s \leq N \\
\overline{hdw_{t,s}} &= 20 \quad t \in [2, \frac{M}{2}] \cup [\frac{M}{2} + 2, M],\ 1 \leq s \leq N
\end{aligned}$$

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 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 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.

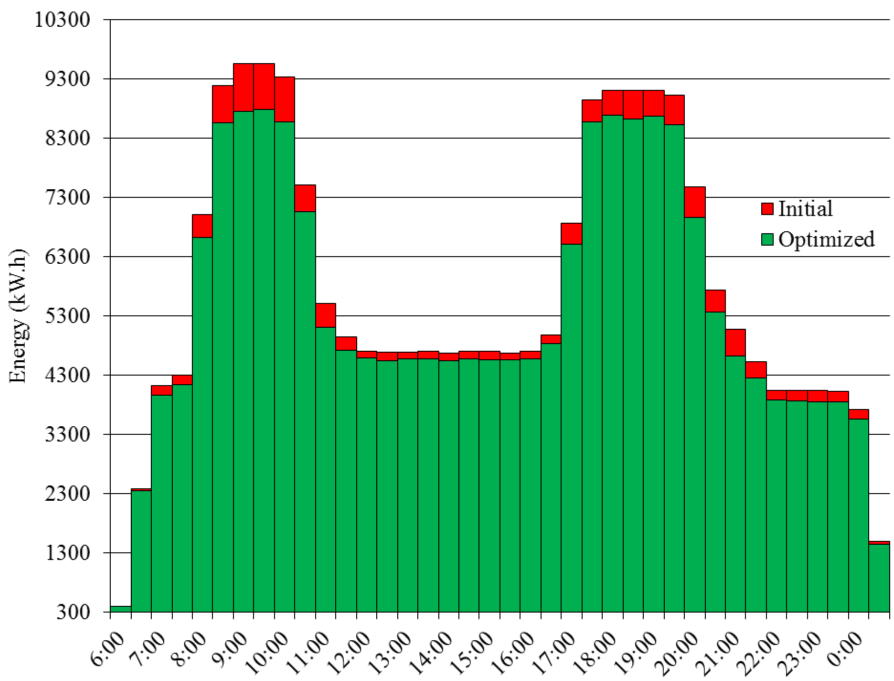

Figure 5: *Weekday timetable* between 6am and 1am: energy consumption by intervals of 30 minutes compared between the initial the timetable, in red, and timetable computed by the greedy heuristics in green.

Figures 5, 6 and 7 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

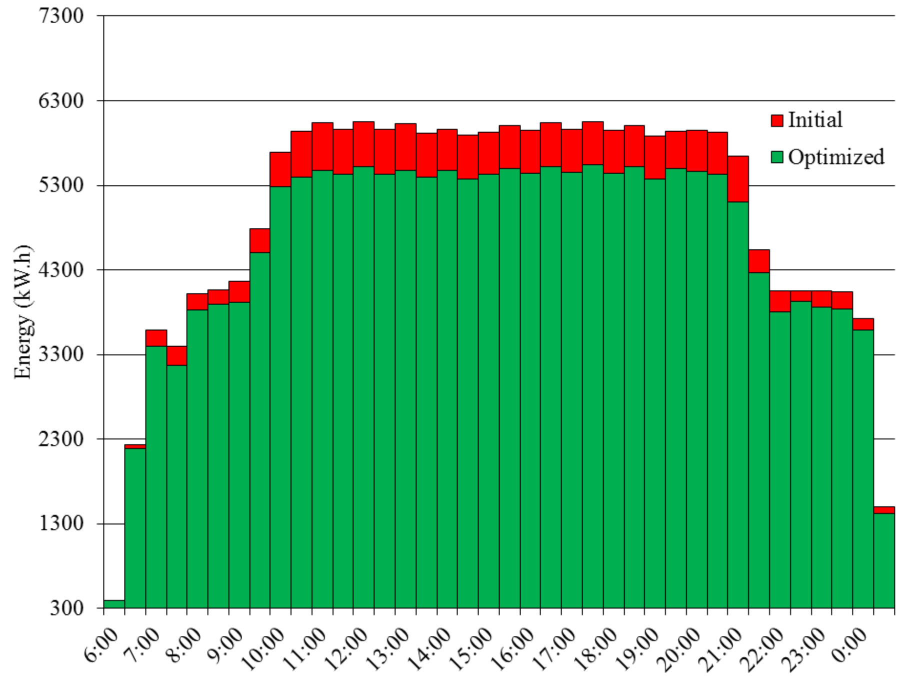

Figure 6: *Sunday timetable* between 6am and 1am with a tolerance of 15 seconds on trips and headways: energy consumption by intervals of 30 minutes compared between the initial timetable, in red, and the timetable computed by the greedy heuristics in green.

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.

9. Conclusion

In this paper, we have proposed a generic mathematical model for metro energy optimization problems and a dedicated heuristics for solving the global energy consumption optimization problem by the sole modification of dwell times. This model has led us to a simple classification by triples of several similar problems of the literature, and to prove their NP-hardness.

We have shown that our heuristic algorithm for the problem $(G, dwe, nonlin)$ 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 better solutions. In particular for the MILP formulation, the results computed by CPLEX were of lesser quality due to the linear approximation of the objective function whereas our heuristics uses a non-linear power flow approximation. It was also shown to perform better than the state-of-the-art metaheuristic CMA-ES on these

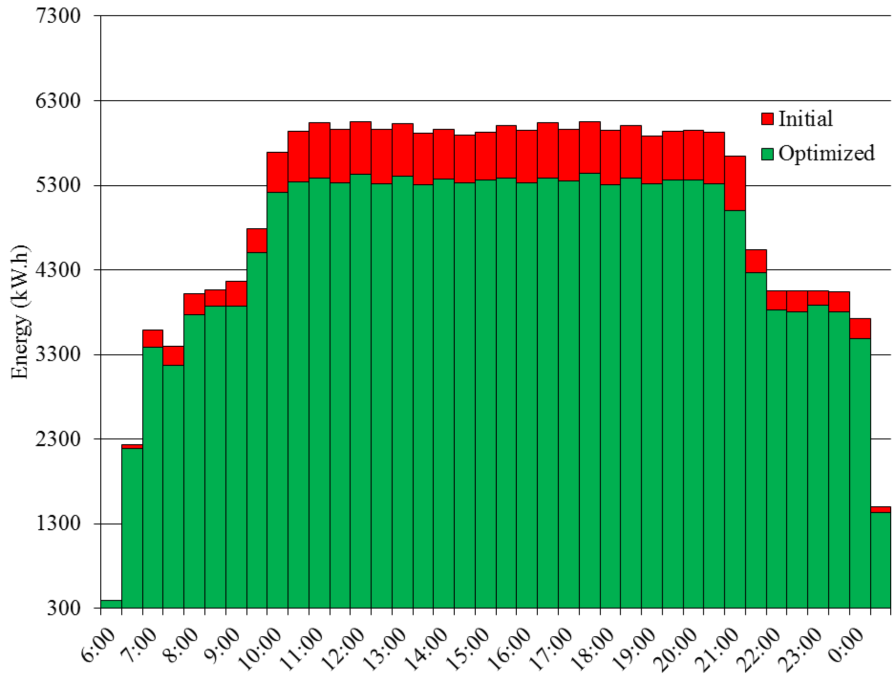

Figure 7: *Sunday timetable* between 6am and 1am with a tolerance of 20 seconds on trips and headways: energy consumption by intervals of 30 minutes compared between the initial timetable, in red, and the timetable computed by the greedy heuristics in green.

instances thanks to the possibility of incrementally computing the objective function over iterations.

Furthermore, our dedicated heuristics was 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 by 5.15% and 7.54%, and up to 8.91% by increasing the tolerances on the variables. These results show the applicability of this method in an industrial context.

References

[1] 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.

[2] I. E. Agency, "Energy balance for world." `http://www.iea.org/stats/balancetable.asp?COUNTRY_CODE=29`, 2009.

[3] T. O'Toole, "Environment report," tech. rep., London Underground, 2006.

[4] 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.

[5] J. Greatbanks, "Review of the discount for using regenerative braking," *AEA Technology*, 2005.

[6] 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.

[7] I. A. of Public Transport, "Reducing energy consumption in underground systems – an important contribution to protecting the environment." 52nd International Congress of UITP, 1997.

[8] T. Albrecht, "Reducing power peaks and energy consumption in rail transit systems by simultaneous metro running time control," *Computers in Railways IX*, 2004.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.

[14] ILOG, "V12.1: User's manual for CPLEX," *International Business Machines Corporation*, vol. 46, no. 53, p. 157, 2009.

[15] B. Sansó and P. Girard, "Trains scheduling desynchronization and power peak optimization in a subway system," *IEEE*, 1995.

[16] K. Kim, K. Kim, and M. Han, "A model and approaches for synchronized energy saving in timetabling," *WCRR 2011*, May 2011.

[17] 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.

[18] 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.

[19] D. Fournier and D. Mulard, "Method and system for timetable optimization utilizing energy consumption factors," Mar. 11 2014. US Patent App. 13/676,279.

[20] 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.

[21] A. Caprara, M. Fischetti, and P. Toth, "Modeling and solving the train timetabling problem," *Operations Research*, vol. 50, pp. 851–861, Sept./Oct. 2002.

[22] 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.

Appendix A. NP-completeness proofs

Theorem 1. *The problem (G, dep, lin) is NP-hard.*

Proof 1. We show that there is a polynomial reduction of SAT to (G, dep, lin). Let $X = \{x_1, ..., x_m\}$ be a set of m variables and $\neg X = \{\neg x_1, ..., \neg x_m\}$ 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 $l_{i,j} \in X \cup \neg X$.

Let us consider the discrete time domain $I = \{0, ..., 6n - 1\}$. Let $\mathcal{TT}$ be the metro timetable composed of a sequence of $m + 1$ trips $T = (x_0, x_1, ..., x_m)$ running in the same direction, such that each trip $x_t \in T$ crosses a sequence of unique stations of length $n + 1$, and that the trips are :

- The trip x_0 cannot be shifted and is crossing stations every 6 time units, departing from its first station at time $i = 0$:
  - $d_{0,0} = 0$
  - $int_{0,s} = 1 \quad 1 \leq s \leq n$
  - $dwe_{0,s} = 5 \quad 2 \leq s \leq n$

  Reminding the auxiliary variables equations:

  - the interstation time for a given trip,

  $$int_{t,s} = a_{t,s+1} - d_{t,s} \quad 0 \leq t \leq m,\ 1 \leq s \leq n, \tag{A.1}$$

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

  $$dwe_{t,s} = d_{t,s} - a_{t,s} \quad 0 \leq t \leq m,\ 2 \leq s \leq n, \tag{A.2}$$

  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 (A.1) and (A.2) 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:

$$d_{0,s} = 6(s-1), \quad 1 \leq s \leq n \tag{A.3}$$
$$a_{0,s} = d_{0,s-1} + int_{0,s-1} = 6(s-2) + 1, \quad 2 \leq s \leq n+1 \tag{A.4}$$

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

  – $d_{t,1} = \delta_t + \begin{cases} 0 \text{ if } x_t \in c_1 \\ 1 \text{ if } \neg x_t \in c_1 \\ 2 \text{ otherwise} \end{cases} \quad 1 \leq t \leq m$

  – $int_{t,s} = \begin{cases} 4 \text{ if } x_t \in c_s \\ 3 \text{ if } \neg x_t \in c_s \\ 2 \text{ otherwise} \end{cases} \quad 1 \leq t \leq m, \ 1 \leq s \leq n$

  – $dwe_{t,s} = \begin{cases} 2 \text{ if } x_t \in c_s \\ 3 \text{ if } \neg x_t \in c_s \\ 4 \text{ otherwise} \end{cases} \quad 1 \leq t \leq m, \ 2 \leq s \leq n$

  Like for x_0, we prove by induction that

  $d_{t,s} = 6(s-1) + \delta_t + \begin{cases} 0 \text{ if } x_t \in c_s \\ 1 \text{ if } \neg x_t \in c_s \\ 2 \text{ otherwise} \end{cases}, \ 1 \leq t \leq m, \ 1 \leq s \leq n$:

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

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

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

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

  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 (A.1) and (A.2) we have:

$$d_{t,s+1} = d_{t,s} + int_{t,s} + dwe_{t,s+1}$$

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

$$\Leftrightarrow d_{t,s+1} = 6(s-1) + \delta_t + 4 + 2 \begin{cases} 0 \text{ if } x_t \in c_{s+1} \\ 1 \text{ if } \neg x_t \in c_{s+1} \\ 2 \text{ otherwise} \end{cases}$$

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

We thus have:

$$d_{t,s} = \ 6(s-1) + \delta_t + \begin{cases} 0 \text{ if } x_t \in c_s \\ 1 \text{ if } \neg x_t \in c_s \\ 2 \text{ otherwise} \end{cases} \qquad 1 \leq t \leq m,\ 1 \leq s \leq n \quad \text{(A.5)}$$

$$a_{t,s} = \ d_{t,s-1} + int_{t,s-1} = 6(s-2) + \delta_t + \begin{cases} 0+4 \text{ if } x_t \in c_{s-1} \\ 1+3 \text{ if } \neg x_t \in c_{s-1} \\ 2+2 \text{ otherwise} \end{cases}$$

$$\Leftrightarrow a_{t,s} = \ 6(s-2) + \delta_t + 4, \quad 1 \leq t \leq m,\ 2 \leq s \leq n+1 \qquad \text{(A.6)}$$

Let the instant power demand function be

$$P_i = \max(0, \sum_{t=0}^{m} P_{t,i}) \quad i \in I, \qquad \text{(A.7)}$$

where $P_{t,i} \in \mathbb{R}$ is the power demand or production of the trip x_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. 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 x_t, the instant power demand or production $P_{t,i}$ at time i can be written as follows:

$$P_{t,i} = \begin{cases} 1 \text{ if } \exists\ 1 \leq s \leq n & \text{s.t. } d_{t,s} = i \\ -1 \text{ if } \exists\ 2 \leq s \leq n+1 & \text{s.t. } a_{t,s} = i \\ 0 \text{ otherwise} \end{cases} \qquad \text{(A.8)}$$

Now let us prove that the global energy consumption is not modified by the powers produced by trips $\{x_1, ..., x_m\}$ since they brake it when no trip is accelerating. According to equations (A.6) and (A.8) we have:

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

Conversely and according to equations (A.3), (A.5) and (A.8), we have for all trips $x_t \in T$:

$$P_{t,i} = 1 \text{ if } i = 6(s-1) + (0 \text{ or } 1 \text{ or } 2 \text{ or } 3), \quad 0 \leq t \leq m, \ 1 \leq s \leq n \tag{A.9}$$

Thus, there is no time where the braking of any trip in $\{x_1, ..., x_m\}$ 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 (A.4) and (A.8) 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 (A.9), $P_{t,s} = 1$ if $i = d_{t,s}$. To synchronize the acceleration of the trip x_t at station s with one braking of the trip x_0 we need:

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

In other terms, the timetable is constructed such that for each trip x_t and for each station s we have:

- If the variable x_t is in the clause c_s, then the acceleration phase of x_t at station $S_t(s)$ is synchronized with the braking phase of x_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 x_t is true, satisfying all clauses c_s containing it.

- If the variable $\neg x_t$ is in the clause c_s, then the acceleration phase of x_t at station $S_t(s)$ is synchronized with the braking phase of x_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 $\neg x_t$ is true, satisfying all clauses c_s containing it.

- If neither the variable x_t nor the variable $\neg x_t$ are in the clause c_s, then the acceleration phase of x_t at station $S_t(s)$ cannot be synchronized with any of the braking phases of x_0.

Every time one braking phase of x_0 is synchronized with the acceleration phase of one trip x_t, one unit of power is saved. Given the period of time I and the structure of the timetable, there is n power units that an optimal synchronization could save. Consequently, considering the decision problem of saving n power units, there is a timetable and a set of δ_t with $1 \leq t \leq m$ that save n units of power if and only if the set of δ corresponds to a valuation that satisfies ϕ ; the n synchronizations corresponding to the n satisfied clauses.

Corollary 1. *The problem* $(G, dep, nonlin)$ *is NP-hard.*

Proof 2. A problem is of class *nonlin* if it exists at least one equation which is non-linear. Modifying the equation (A.7) of the previous timetable into the non-linear equation

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

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 resolution, thus the problem $(G, dep, nonlin)$ is NP-hard.

Corollary 2. *The problem* $(G, dep - dwe - int, lin)$ *is NP-hard.*

Proof 3. 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 1, one period of time where only x_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 $T = (x_0, x_1, ..., x_m)$ 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 x_0 is crossing a new station $S_0(n+2)$ such that:

$$d_{0,n+1} = 6n$$
$$a_{0,n+2} = 6n+1$$

Now, the last dwell time of x_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 x_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 problem (G, dep, lin), which encodes the SAT formula and which is NP-hard, has become a problem $(G, dep - dwe - int, lin)$. Thus the problem $(G, dep - dwe - int, lin)$ is NP-hard.