Solution algorithms for the generalized train unit shunting problem

Franck Kamenga, Paola Pellegrini, Joaquin Rodriguez, Boubekeur Merabet

EURO Journal on Transportation and Logistics 10 (2021) 100042

Contents lists available at ScienceDirect

EURO Journal on Transportation and Logistics

journal homepage: www.journals.elsevier.com/euro-journal-on-transportation-and-logistics

Solution algorithms for the generalized train unit shunting problem

Franck Kamenga [a,c,*], Paola Pellegrini [b], Joaquin Rodriguez [c], Boubekeur Merabet [a]

[a] SNCF Réseau, DGEX Solutions, F-75013, France
[b] COSYS-LEOST, Univ Gustave Eiffel, IFSTTAR, UnivLille, F-59650, Villeneuve d'Ascq, France
[c] COSYS-ESTAS, Univ Gustave Eiffel, IFSTTAR, UnivLille, F-59650, Villeneuve d'Ascq, France

ARTICLE INFO

Keywords:
Railway traffic
Shunting problem
Rolling stock management
Maintenance scheduling
Routing
Matching

ABSTRACT

This paper proposes different algorithms to tackle the Generalized Train Unit Shunting Problem (G-TUSP). This is the pre-operational problem of managing rolling stock in a station, between arrivals and departures. It includes four sub-problems: the Train Matching Problem, the Track Assignment Problem, the Shunting Routing Problem, and the Shunting Maintenance Problem. In our algorithms, we consider different combinations for the integrated or sequential solutions of these sub-problems, typically considered independently in the literature. We assess the performance of the algorithms proposed in real-life and fictive instances representing traffic in Metz-Ville station, which includes four shunting yards. It is a main junction between two dense traffic lines in the east of France. In a thorough experimental analysis, we study the contribution of each sub-problem to the difficulty of the G-TUSP, and we identify the best algorithms. The outcomes of our algorithms are superior to solutions manually designed by experienced railway practitioners.

1. Introduction

Rolling stock planning must manage *train units* so that a timetable can be operated. A specific part of this planning is *shunting*, i.e., the management of train units between an arriving and a departure trip in a station. Inside stations, train units are prepared for departure and possibly stored if they are not needed immediately. More precisely, they are cleaned and undergo maintenance checks. Moreover, train units can be coupled or uncoupled to match train configurations required for departure. This preparation is done on siding tracks located around platform tracks. Parallel siding tracks form *shunting yards*. Some of these tracks have specific amenities such as train washing machines for external cleaning or pits for maintenance checks. To be stored in yards, train units need first of all to be moved from their arrival platform. Then, they may need to be moved from one shunting track to another. Finally they need to be moved to their departure platform. Movements arriving or departing from a yard are called *shunting movements* and must respect traffic safety rules imposed by signaling system and ground-agents instructions. Indeed, *routes* used for shunting movements must not create conflicts with the rest of train traffic in the station.

Shunting operation planning includes several decisions. First, arriving train units must be assigned to departures: these are matching decisions. This matching must take into account rolling stock features required for departures. Other decisions concern train units location: they must be parked on one or several shunting tracks depending on amenities required by maintenance operations. Similarly, movements are set to reach parking locations. For these movements, route planning decisions are to be made: routes are assigned to train units and movements are scheduled based on running times and potential conflicts. Finally, respecting maintenance crews and amenities availability, maintenance operations must be scheduled. Although these four types of decisions are often made separately, they are usually strongly interdependent. For instance, some matching plans make train units parking or maintenance scheduling impossible.

The Generalized Train Unit Shunting problem (G-TUSP) is the problem of shunting operations planning. It considers a station and a timetable with arriving and departing trains that need to be shunted. In the classic timeline of the railway planning process (Marinov et al., 2013), it is solved in the so-called pre-operational planning phase. In this process, *strategic* planning concerns long term development. *Tactical* planning deals with medium term decisions, as timetables and schedules. The *operational* planning is for short term: timetables and schedules are implemented on a "day-to-day" basis in order for the system to provide the service. The latter can be split in *pre-operational* and *real-time* planning

* Corresponding author. SNCF Réseau, DGEX Solutions, F-75013, France.
E-mail addresses: franck.kamenga@reseau.sncf.fr (F. Kamenga), paola.pellegrini@univ-eiffel.fr (P. Pellegrini), joaquin.rodriguez@univ-eiffel.fr (J. Rodriguez), boubekeur.merabet@reseau.sncf.fr (B. Merabet).

https://doi.org/10.1016/j.ejtl.2021.100042
Received 19 July 2020; Received in revised form 11 April 2021; Accepted 7 May 2021

phases. The definition of the precise timing of these phases may slightly change from country to country. In France, for example, the former goes from 6 days to 4 h before operations. The latter then starts and continues until operations are actually performed.

Being a pre-operational problem, the G-TUSP aims to minimize departure delays and cancellations if perturbations are expected at least some hours in advance, as well as maintenance cancellation and other costs related to shunting. Examples of perturbations are the closure of a track in a station due to maintenance, or the presence of a temporary speed limitation on a line, which will make trains arrive later than planned in the timetable. All details on infrastructure and maintenance crews are considered, so that the solution of the problem is a set of precise decisions that can be implemented to operate the yard and the station, unless unpredictable perturbations occur. Moreover, a predefined matching of arriving and departing train units is as far as possible respected. Such matching is typically defined in the solution of the *tactical G-TUSP*, which is solved during tactical planning, when the timetable is defined. It is a macroscopic problem neglecting short-term conditions of yard operations. Specifically, the tactical G-TUSP defines arrival and departure train unit matching in coherence with the rolling stock rostering for the whole network. Here, heavy maintenance activities are scheduled, and expected demand is matched with capacity. Standard values are considered for track, crew and amenity capacities, as well as for movement and light maintenance activity duration.

This paper proposes solution approaches for the G-TUSP based on the integration of several sub-problems. To the best of our knowledge, this is the first work dealing with the overall G-TUSP, and proposing algorithms that could actually be used to manage a shunting yard. Kamenga et al. (2019) present a mixed integer linear formulation which integrates the four sub-problems of G-TUSP. This formulation gives satisfying solutions compared to decisions made by dispatchers. Nevertheless, computation times are quite high. To obtain good solutions to the G-TUSP in shorter time, processing sequentially the different sub-problems is a natural strategy that we investigate in this paper. However, as above mentioned, intuitively at least some of these sub-problems would rather be solved in an integrated way. In this study, we assess the importance of the interdependence between sub-problems considering their sequential or integrated solution. Then, we select the best algorithm to solve the G-TUSP considering the most appropriate sub-problems integration. We model the G-TUSP on an infrastructure microscopically represented, so as to exploit the whole station and shunting yard capacity available in reality. All the algorithms proposed are tested on several instances which cover different types of perturbation. Some of these instances replicate actually occurred situations, others are artificially generated starting from them.

The rest of the paper is organized as follows. Section 2 describes the G-TUSP. Then, Section 3 gives a literature review of models and solution approaches for shunting problems. Section 4 presents modeling aspects for G-TUSP sub-problems, while the comprehensive description of the model is reported in the supplementary material. Then, solution algorithms are presented in Section 5. Section 6 reports experiments and Section 7 concludes the paper.

2. Problem description

The G-TUSP integrates four sub-problems:

- The Train Matching Problem (TMP), the problem of matching arriving and departing train units;
- The Track Assignment Problem (TAP), the problem of choosing train units location;
- The Shunting Routing Problem (SRP), the problem of determining train units routing during shunting movement;
- The Shunting Maintenance Problem (SMP), the problem of defining train units maintenance scheduling.

The definition of three of these four problems is quite uniform across the whole literature: The TMP is the problem of matching arriving a departing train units, respecting constraints linked to type of rolling stock and schedule. The SRP is the problem of routing train units to or from the shunting yard, or within it. The routes train units go through are chosen and movements are scheduled. In some cases routes are already fixed and we have a Shunting Routing Problem with fixed routes, that we denote SRP^1. The SMP consists in defining train units maintenance scheduling between their arrival and departure.

Instead, several different variants of the TAP exist. In most of them several train units can be parked on the same shunting track. Their total length can not exceed the track's length. This is the *length constraint*. Also, when a train leaves a shunting track it must not be blocked by another train parked in front of it. This is the *crossing constraint*. Shunting yards can contain dead-end tracks, those tracks are also called *last-in-first-out (LIFO)* tracks because they are like stacks. There can be tracks in which trains arrive at one side and leave at the opposite side. These tracks are like queues and are called *first-in-first-out (FIFO)* tracks. Finally, shunting yards may contain *regular tracks* in which trains can enter or leave at both sides. Different variants of the problem deal with one or more of these types of tracks. Others consider shunting tracks as single-capacity resources. Then, two train units can not be parked on the same track at the same time. We have a single-capacity track assignment problem, denoted *TAP-cap1*. Thus, the TAP-cap1 does not deal with crossing and length constraints. In most variants of the problem, each train unit needs to be parked on only one shunting track. This is the *standard TAP*, that we denote TAP^1. In this case, the time at which train units leave or enter a track in the shunting yard is considered to be known. We can also consider a version of the TAP in which each train unit is allowed to be parked on up to k tracks during a planning period. Then we have a k-Track Assignment Problem, denoted TAP^k. The time at which train units leave or enter tracks becomes variable. In the most general case, the number of tracks where a train unit has to be parked is not bounded. We have a *multiple track assignment problem*, denoted TAP*.

3. Literature review

Several contributions focus on G-TUSP sub-problems, and some propose solution approaches to deal with large size instances.

Some papers deal with combinations of sub-problems considering the TAP for maintenance. Here, each train unit must be parked successively on different tracks to use various equipment necessary for its maintenance. Following the taxonomy defined above, this is a TAP*. Tomii and Zhou (2000) tackle the SMP, the TAP*-cap1 and the SRP. Trains have to be shunted to appropriate tracks to undergo specific operations. A track can be used by only one train at a time. The authors consider the choice of shunting routes between sidings. Scheduling is performed thanks to a PERT network and resource assignments are selected with a genetic algorithm. Qi et al. (2017) propose a discrete time model which integrates the SRP, the TAP*-cap1 and the SMP. As Tomii and Zhou (2000), they consider that a track can be occupied by only one train at a time. The model is solved with a Lagrangian relaxation-based algorithm on Beijing South Railway Station instances. Jacobsen and Pisinger (2011) consider the TAP* and the SMP with a discrete time model. Shunting tracks are supposed to be LIFO. The paper uses three metaheuristics: guided fast local search, guided local search and simulated annealing. The authors carry out experiments on virtual instances containing up to 10 trains. Guided local search provides results close to the ones obtained by solving an Integer Linear Programming (ILP) model on test instances. However, a result is obtained after a few seconds by guided local search while it takes several hours for the ILP. The authors consider the use of simulated annealing to improve solutions obtained by local search. Some contributions consider the TAP* with a fixed maintenance schedule. Li et al. (2017) assume that arrival and departure times on shunting tracks are known. In this case, the TAP* can be reduced to the TAP^1. In the paper, all train units are of the same length. Tracks can be LIFO or regular and are set to contain at most two train units. An ILP which models crossing

conflicts and maximizes the number of parked trains is proposed. The authors solve real Chinese railway instances using CPLEX.

Other papers focus on the standard TAP. Di Stefano and Koči (2004) consider the TAP[1] without length constraints. The authors first assume that all departures occur after all arrivals, this is the *midnight condition*. The authors try to minimize the number of tracks used considering four shunting yard configurations. The first configuration is the one in which n trains enter and leave a track on only one side. It gathers LIFO and FIFO cases. Then, under this configuration the TAP is equivalent to permutation graph coloring. For this case, the paper provides optimal solution with an algorithm which requires $O(n \ln n)$ time. In the three other configurations trains can enter or leave through both sides of tracks. In these cases the problem is NP-complete. The paper also proves that solving the TAP[1] with LIFO/FIFO tracks configuration and without midnight condition is NP-complete. Demange et al. (2012) tackle the online TAP[1]. The departure time of each train is known as soon as it arrives at the shunting yard. The authors study regular and LIFO/FIFO tracks with length constraints. They establish a conflict graph on which a coloring is performed when a train arrives. Gilg et al. (2018) deal with both LIFO, FIFO and regular tracks. They propose two ILP formulations. In a first formulation, arrival times are included in crossing constraints, while a second one simply considers conflicts. They note that the second formulation provides a better linear relaxation. A robust extension and a stochastic version are proposed to take into account possible delays. Experiments are based on European stations and are performed with Gurobi solver.

Part of the literature deals with the TAP[1] and the TMP, in railway or other modes of transport. Winter and Zimmermann (2000) study several algorithms to manage these problems in tram depots. Trams have the same length and their timetable respect the midnight condition. Trams do not have to be coupled or uncoupled. The authors provide ILP formulations to minimize the number of crossing conflicts or the number of departures performed by substitute types of trains. They also propose a greedy search, a Tabu search and a dynamic programming algorithm. Cardonha and Borndörfer (2009) extend the previous paper for schedules that do not respect midnight condition. They propose a column generation approach with a dynamic programming based pricing. Gallo and Miele (2001) provide an extension for buses in which vehicles can have different lengths and the midnight condition is not respected. Buses are parked in lanes that are similar to FIFO tracks. A Lagrangian relaxation based heuristic is proposed and is compared with an exact solution. For many of the tackled instances, CPLEX is not able to find an integer solution within 3 h or terminates with an out of memory message, while the heuristic provides solutions in a few minutes. Freling et al. (2005) adapt this problem to passenger train shunting yards, also including train units coupling and uncoupling. This is the Train Unit Shunting Problem (TUSP). The authors solve the TMP in a first phase and the TAP in a second phase, once a train matching is set. An ILP based on a graph associated to each train models the TMP. The TAP[1] is tackled with column generation which is based on dynamic programming like in Cardonha and Borndörfer (2009). Haijema et al. (2006) solve the TUSP with a heuristic that tackles the problem in sub-planning periods. Still for the TAP[1] and the TMP, Kroon et al. (2008) propose an ILP formulation. The new issue considered is the fact that the position of train units must be taken into account so that there is no crossing after uncoupling. The authors also include valid inequalities based on the conflict graph cliques. Haahr et al. (2017) provide a column generation method and allow trains to be parked at platforms. The pricing problem is based on a shortest path search. This method is compared with constraint programming formulations, a greedy algorithm, the one-step ILP proposed by Kroon et al. (2008) and the two-step approach by Freling et al. (2005).

Other contributions deal with problems strictly related to the TMP. Hoogervorst et al. (2020) tackle the Passenger Delay Reduction Problem (PDRP). This is the problem of minimizing passenger delays in a network by rescheduling rolling stock after a disruption occurs. In a sense, this is a network version of the TMP (*netTMP*), as train matching is decided for the whole network at once, rather than at a single station. The authors propose two models for this problem, which they test on instances of Netherlands Railways. The results show that modifying rolling stock rotation can significantly reduce delay propagation. A strictly connected problem to the netTMP is the rolling stock rostering one. This is a tactical problem: a set of services needs to be covered with the available fleet of train units, guaranteeing the fulfillment of maintenance needs. The objective is minimize the assignment cost, which can be defined in various ways. A noticeable amount of literature exists on this problem (Maróti, 2006). A remarkable example is the work by Borndörfer et al. (2011), who introduce a hypergraph model for the train unit assignment. Infrastructure capacity at stations is also considered, with macroscopic constraints stating that the total length of trains that occupy a station simultaneously must not exceed the total length available on tracks. This model is the basis of a tool currently used by Deutsche Bahn for carrying out actual rolling stock rostering (Borndörfer et al., 2020).

Another part of the literature focuses on the SRP. Riezebos and Van Wezel (2009) study combinatorial aspects of shunting movements. They provide a two-step algorithm which searches for the shortest paths across a set of shunting tracks. Lentink et al. (2006) propose an additional step to the TUSP solution approach by Freling et al. (2005), in which the SRP is solved thanks to an A* algorithm. Van Den Broek and Kroon (2007) consider the SRP with a solution of the TAP[1]. Given shunting routes, shunting movement times are determined within a given time range. The authors also focus on a variable route variant. Mixed integer linear programming (MILP) formulations model the fixed-route and variable-route variants. Abbink (2006) tackle both the TAP[1] and the SRP with a discrete time model. Tracks are divided in portions that can contain a train unit. A constraint programming algorithm is proposed to achieve conflict-free planning.

In summary, all existing contributions, with the only exception of Kamenga et al. (2019), focus on subsets of the four sub-problems discussed. The specific combinations are reported in Table 1. For each sub-problem, we indicate if each paper deals with it or not and, if so, which variant is considered when many exist. In the last column we specify if the solution of several sub-problems is done in an integrated way. Remark that we study different possibilities for integrating problems and solving them sequentially. This is indicated in the table by the mention of the presence of some integration. In this paper, we consider the G-TUSP proposed by Kamenga et al. (2019) which tackles four shunting problems. Kamenga et al. (2019) present a MILP formulation for the problem and, as Freling et al. (2005), Jacobsen and Pisinger (2011) and Lentink et al. (2006), remark that solving the TAP to optimality requires a lot of computation compared to other shunting problems.

Table 1
Summary of contributions in shunting for passengers trains.

Contribution	TMP	TAP	SMP	SRP	Integrated
Tomii and Zhou (2000)	No	TAP*-cap1	Cont.	Cont.	Yes
Freling et al. (2005)	TMP	TAP[1]	No	No	No
Abbink (2006)	No	TAP[1]	No	Disc.	Yes
Haijema et al. (2006)	TMP	TAP[1]	No	No	No
Kroon et al. (2008)	TMP	TAP[1]	No	No	Yes
Lentink et al. (2006)	TMP	TAP[1]	No	Cont.	No
Jacobsen and Pisinger (2011)	No	TAP*	Disc.	No	Yes
Haahr et al. (2017)	TMP	TAP[1]	No	No	Yes
Li et al. (2017)	No	TAP[1]	No	No	–
Qi et al. (2017)	No	TAP*-cap1	Disc.	Disc.	Yes
Gilg et al. (2018)	No	TAP[1]	No	No	–
Hoogervorst et al. (2020)	netTMP	No	No	No	–
Kamenga et al. (2019)	TMP	TAP*	Cont.	Cont.	Yes
This paper	TMP	TAP*	Cont.	Cont.	Some

4. Problem modeling

Being the G-TUSP a very detailed representation of a complex real problem, its complete model includes a large number of choices and definitions. For readability, we report the complete model in the supplementary material of this paper, together with its MILP formulation. In this section, we focus on some specific aspects of the problem. In particular, we first focus on train management, then on maintenance scheduling and finally on movement and parking modeling. We focus on these aspects for different reasons. Concerning train management (Section 4.1), we wish to highlight the introduction of a set of trains named *intermediate trains*. To the best of our knowledge, the concept they stand for has never been defined in the literature while they allow a quite convenient representation of the problem. As for maintenance scheduling (Section 4.2), we aim to give the flavor of the very high level of detail we consider in the model. Similarly, for movements and parking (Section 4.3), we report our most main modeling choices as they are typically neglected in the literature on G-TUSP sub-problems.

4.1. Train management

In the G-TUSP, train units can be coupled or uncoupled to form trains. If a train unit must be both uncoupled from others it arrived with and coupled to further ones it will depart with, we assume that the uncoupling is carried out first. Three sets of trains are introduced to model coupling and uncoupling operations: *arriving*, *intermediate* and *departing trains*. Arriving trains are moved from a platform track to the shunting yard. Once there, the corresponding train units are uncoupled if needed, and they become intermediate trains, which are moved in the shunting yard and possibly submitted to maintenance. Finally, the train units constituting intermediate trains are coupled if necessary and become departing trains to be moved to the suitable platform track.

The notation related to train management and introduced in this section is summarized in Table 2. Formally, we denote T_T the set of arriving trains. The arrival time of $t \in T_T$ is denoted a_t. Every train is composed of one or several train units. Train units are divided into types that are denoted *TU*: Train units of the same type are considered interchangeable in the matching problem. For a train t and a train unit type $tu \in TU$, $m_{t,tu}$ is the number of train units of type tu in t. Every arriving train disappears when it enters in the shunting yard. Then, one or more intermediate trains appear. If more than one intermediate train appears, an uncoupling operation takes place. Its cost is denoted Q_H. For an arriving train t, $T_I(t)$ is the set of intermediate trains that can be created with its train units. The set of departing trains is denoted T_S. For a departing train t, we denote $T_I(t)$ the set of intermediate trains which are compatible with t and d_t its expected departure time. $T_I(t)$ contains intermediate trains which can be used, maybe after coupling, to obtain t. The cost of a coupling operation is denoted Q_C. Intermediate trains in $T_I(t)$ must arrive before t's expected departure time d_t. Nevertheless, we remark that, as operations can be performed on intermediate trains, they may not be ready to depart as soon as they arrive at the shunting yard. Then for a departing train t, $T_I(t)$ may contain some trains that can not be ready at t's departure time, unless an operation is canceled or the departure delayed. Let $T_I^N(t) \subseteq T_I(t)$, be the subset of intermediate trains that cannot be ready on time for t's departure if all planned operations are executed. $T_I^N(t)$ can be computed by considering the total duration of the operations planned to be carried out on each intermediate train and then deducing its earliest exit time from the shunting yard. When intermediate trains are assigned to a departing train, they disappear from the shunting yard. If no train is assigned to a departing train, the latter is canceled. The cost of a departure cancellation is denoted B_t and the cost of one time unit of delay Q_t.

Not all intermediate trains disappear to become departing trains: Some intermediate trains may remain in the shunting yard at the end of the planning period τ_M. For intermediate trains that are present in the shunting yard before the planning period, dummy arriving trains are introduced. These arriving trains enter the station at the beginning of the planning period. The shunting track where the present intermediate trains have been stored before the planning period is known. They are considered parked on this shunting track at the beginning of the planning period.

As introduced in Section 2, the TMP is to allocate intermediate trains to arriving trains and departing trains to intermediate trains such that matching constraints, stated afterwards, are respected. In the problem, a maximum of departing trains must be covered on time while assignment cost and number of coupling or uncoupling operations are minimized. The first matching constraint states that the number of train units is preserved. The second one states that an intermediate train can be assigned to at most one arriving train. In the last constraint, at most one departure train can be assigned to an intermediate train.

In Fig. 1, we report an example to picture the model using intermediate trains. Here, three types of train units are considered: hashed ones, full colored ones and white ones. For each arriving train, the set of its intermediate trains is represented by a thick-line dashed box. For each departing train, the set of compatible intermediate trains is represented with a thin-line dashed box. Arrows represent matching, that is a possible combination of coupling and uncoupling using the train units available to compose the two departing trains. Here, the arriving train t_1 is uncoupled in order to obtain train t_A and two intermediate trains are coupled to obtain train t_B.

The initial matching decided in the tactical G-TUSP defined in the introduction is an optional input datum. Let $\omega_{t,t'}$ for $t \in T_S$, $t \in T_I(t)$ be the cost of the assignment of intermediate train t' to departing train t. $\omega_{t,t'}$ is positive if the assignment does not belong to the initial matching or if $t' \in T_I^N(t)$ and is equal to 0 otherwise. This cost represents the fact that it is preferable to keep the initial assignment if possible: if a precise assignment has been made in the tactical G-TUSP, then we may avoid the violation of macroscopic constraints by keeping it, as mentioned in the introduction. Indeed, if by doing so major infeasibilities occur in the station under consideration, then changes are allowed, which motivates the relevance of the TMP.

By definition, the sets $\{T_I(t)\}_{t \in T_T}$ are disjoint. For readability, we introduce $T_I = \cup_{t \in T_T} T_I(t)$ that is the set of all intermediate trains. We can remark that a departing train t and an arriving train t' use the same set of train units if and only if $T_I(t) = T_I(t')$.

Finally, trains that do not stop at the station under study or stop without being shunted are named *passing trains*. The set of passing trains is denoted T_P. An instance contains then a set of trains $T = T_T \cup T_I \cup T_S \cup T_P$.

Remark that T_I is typically not explicitly included in the input data, but it can be generated starting from set T_T.

With a slight abuse of notation, for readability in the following we will refer to a train $t \in T$ considering t also as its index in the set of trains. Therefore, for two trains t and t', $t < t'$ means that the index of t is lower than the index of t'.

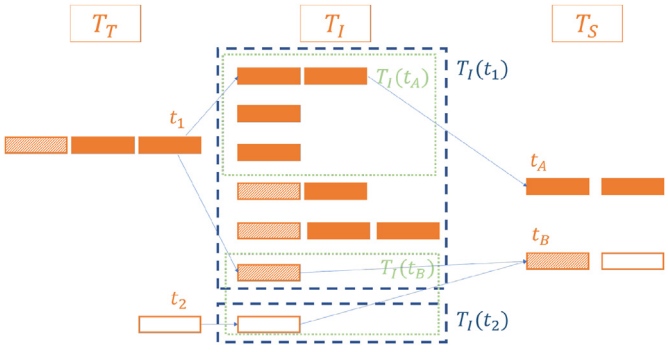

Fig. 1. Train matching. Arriving trains T_T on the left are used for the departing trains T_S on the right through the set of intermediate trains T_I. A possible matching is represented with arrows.

4.2. Maintenance scheduling

Cleaning or maintenance operations may be included in the rolling stock plan. They are considered to be made on intermediate trains. Table 3 resumes notation introduced in this section. In particular, the operations carried out on an intermediate train $t \in T_I$ form the set O_t. An operation $o \in O_t$ can only be performed on shunting tracks with specific facilities. P_t is the set of shunting tracks compatible with t. This set contains tracks which are accessible and long enough for t. If t is an electric train, any track of P_t has to be electrified. Set $P^o \subseteq P_t$, includes all shunting tracks where operation o can be carried out. In addition, an operation requires the use of specific human resources. We consider that an operation o requires a crew among the set HR^o of crews which can perform o. Each crew hr is available from its shift start time sR_{hr} to its shift end time eR_{hr}.

The duration of operation o is denoted pR^o and the cost of its cancellation ω^o. The sequence of operations to be carried on an intermediate train t is given. We denote E_t the set of pairs of successive operations: $(o, o') \in E_t$ if and only if operation o' has to be performed right after operation o. If two successive operations require the train movement from a shunting track to another, we denote mr the maximum duration of the shunting movement. We assume that at most one operation can be carried out on a track at any time, independently on the number of trains simultaneously present on the track itself.

We also specify bt, i.e., the minimum time that must separate the arrival of a train on a track and the departure of another train which used the same track before.

The SMP is modeled considering a TMP solution as given or as constructed simultaneously. In the SMP, each operation has to be assigned to a crew and a track so that it does not get canceled. The resulting total delay of departing trains and the number of operations canceled must be minimized. First, operations carried out on a train must be scheduled after its arrival. If all the operations cannot finish by the planned departure time of the assigned departing train, the latter is delayed. Then, the operations schedule must follow a defined order. If a train has to change track between two operations it needs time to be moved. Maintenance or cleaning operations must be performed when the assigned crew is available. Otherwise, two operations can not utilize the same crew or the same track at the same time.

Also, when an operation is in progress, the shunting track where it is carried out must be protected to ensure staff safety. Thus, during this period, no other train can enter this shunting track or leave it. This constraint only affects the TAP but is generated with an SMP solution.

Table 2
Notation for train management.

Notation	Description
T_T, T_I, T_S, T_P	set of arriving trains, intermediate trains, departing trains, passing trains
$T = T_T \cup T_I \cup T_S \cup T_P$	set of trains
$T^* = T_T \cup T_I \cup T_S$	set of shunted trains
a_t, d_t	arrival time of the train $t \in T_T$, departure time of the train $t \in T_S$
TU, $m_{t,tu}$	set of train unit types, number of train units of type $tu \in TU$ in the train $t \in T^*$
$T_I(t)$	set of intermediate trains compatible with the arriving or departing train $t \in T_T \cup T_S$
$T_I^N(t)$	set of intermediate trains that may be assigned to the departing train $t \in T_S$ but must either skip maintenance operations or make t late
Q_C, Q_H, mc	cost of coupling, uncoupling, minimum time required for coupling or uncoupling
$\omega_{t,t'}$	cost of assigning the departing train t to the intermediate train $t' \in T_I(t)$
B_t, Q_t	departure cancellation cost, cost associated to the delay of the train $t \in T_S$
τ_M	end of planning period
$t < t'$	index of train $t \in T$ is lower than index of train $t' \in T$

Table 3
Notations for maintenance operations scheduling.

Notation	Description
O_t, P_t	set of operations to carry out on $t \in T_I$, set of shunting tracks compatible with t
HR^o, $P^o \subseteq P_t$	set of crews and shunting tracks which can be assigned to operation $o \in \bigcup_{t \in T_I} O_t$
pR^o, ω^o	duration and cancellation cost of operation $o \in \bigcup_{t \in T_I} O_t$
E_t	set of successive operations on $t \in T_I$. $(o, o') \in E_t$ if and only if operation o' has to performed right after operation o
sR_{hr}, eR_{hr}	shift start time and shift end time of crew hr
bt	buffer time for utilization of a track
mr	maximum duration of a shunting movement

4.3. Movement and parking modeling

Trains move on an infrastructure modeled microscopically through a *track-circuit* scale representation. A track-circuit is a portion of track on which the presence of a train is automatically detected. Thanks to this infrastructure model, detailed characteristics of *interlocking system* are taken into account and train safety is ensured through suitable separation. Specifically, the structure of block sections is known, where a block section is a sequence of track-circuits delimited by light signals and in which at most one train can circulate at any time, unless particular procedures are put in place.

Fig. 2 represents a simple example in which an orange, a green and a blue routes are shown with their respective track-circuits named z followed by a number. Both orange and blue routes use track-circuit z_{15}, therefore they cannot utilize it at the same time. The train with the orange route is an intermediate train whose route starts on shunting track 21. This train results from the arriving train using the green route and has to be cleaned. It is parked on shunting track 29 for cleaning. The train with the blue route is a departing train which uses platform A.

The notation introduced in this section is summarized in Table 4. The length of each intermediate train $t \in T_I$ is denoted l_t. As introduced in Section 4.2, the set of shunting tracks compatible with t is denoted P_t. We introduce $P_T = \bigcup_{t \in T_I} P_t$ the set of shunting tracks. The set of sides in a shunting track $p \in P_T$ is denoted $Ex(p)$, those are the sides where trains can leave or enter the shunting track. This set contains at most two elements which indicate a geographical direction. We use the direction left and right, respectively denoted L and R. In particular if $|Ex(p)| = 1$ then track p is a dead-end track. The length of p is denoted L_p.

The TAP* deals with intermediate trains parking and uses both the TMP and the SMP solution. In the TAP*, we need to assign a sequence of shunting tracks to each intermediate train, and choose sides and times at which the train enters and leaves them. The assignment is made with the minimum number of tracks. The TAP* must also respect four main sets of constraints. First, every intermediate train must occupy a track at all times at which it exists. The total length of intermediate trains that occupy a track at the same time can not exceed the track length. Also crossing constraint must be considered. They can be conveniently visualized with a graphical representation inspired to the one proposed by Kroon et al. (2008) and Di Stefano and Koči (2004). Fig. 3 shows an example of this representation for a specific track. A circle represents the track for a fixed time horizon. The circle is divided in two halves, indicating a side each. Time grows from top to bottom. Arrival and departure time of intermediate trains using the track are points on this circle. Trains are represented by segments which link their arrival and the departing points. If two segments cross, then the crossing constrains are violated. In Fig. 3, intermediate train t_1 enters on the right side before t_2 enters on the left side. Then, t_1 leaves on the left side before t_2 leaves on the same side. The segments representing t_1 and t_2 cross, then the crossing constraints are not respected for these trains. Indeed, the sequence of events is infeasible. Furthermore, intermediate trains cannot enter or leave a track at the same time on the same side. These events should be separated by the buffer time bt, except if the intermediate trains leave (or enter) the

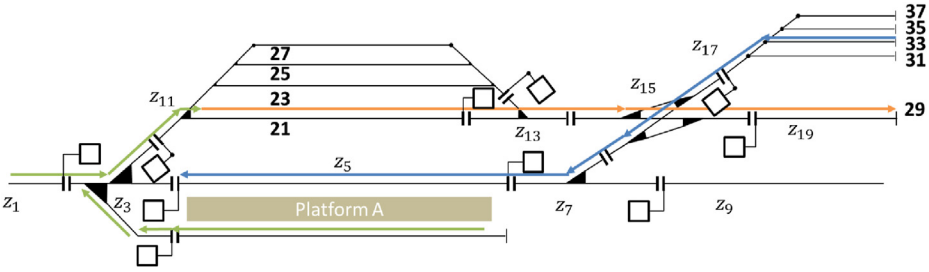

Fig. 2. Example of station layout. Signals are represented by squares. The green arriving train becomes the orange train on shunting track 21. The blue departing train leaves the shunting track and is moved to platform A. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

Table 4
Notations for movement and parking modeling.

Notation	Description
P_t	set of shunting tracks compatible with $t \in T_I$
$P_T = \cup_{t \in T_I} P_t$	set of shunting tracks
$Ex(p)$	set of sides of the shunting track $p \in P_T$
L, R	left side, right side
L_p	length of the shunting track $p \in P_T$
bt	buffer time for utilization of a track
l_t	length of the intermediate train $t \in T_I$

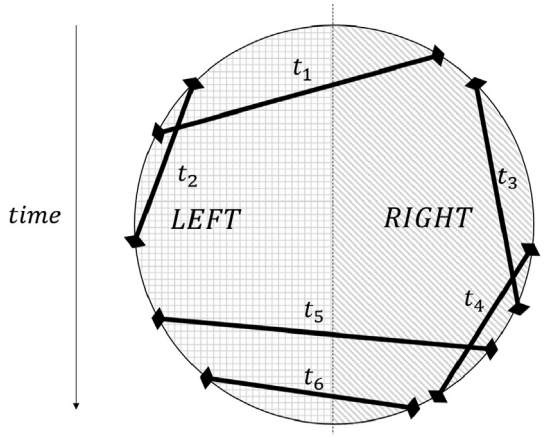

Fig. 3. Graphical representation of crossing constraints on a regular track. Intermediate trains are shown as segments connecting the planned entry and exit side of a track. If two segments cross, the plan is infeasible.

track together because they are coupled. The TAP* must also be consistent with the maintenance schedule: the sequence of tracks assigned to a train and specific to maintenance operations should follow the sequence of operations of the train itself. Moreover, intermediate trains need to stay long enough on these specific tracks so that operations can be performed. The TAP* must also take into account specific aspects due to coupling and uncoupling. Coupling operations take place on the last shunting track used by an intermediate train, and uncoupling operations on the first one. Then, the intermediate trains to be coupled (or uncoupled) must be parked on the same track at the end (or the beginning) of their parking sequence.

The SRP consists in scheduling train movements in the shunting yard and in the station so that parking decisions are executed while minimizing number and duration of intermediate train movements as well as delays which may be caused by conflicts. Also the movements of passing trains must be scheduled. Several sets of constraints must be respected, mostly following the classic representation of train movements on microscopic infrastructures available in the literature (Pellegrini et al., 2015). However, a different modeling must be considered due to the frequent utilization of turnaround routes in shunting movements. In these movements, trains are moved to a track-circuit to then reverse and leave in the opposite direction. As an example, in Fig. 2, the green route goes to track-circuit z_1 to then reverse and travel up in the infrastructure. By doing so, the green route uses twice the same track-circuit z_3. In layouts slightly more complex than the one shown in the figure, a train may go through the green route, arrive in z_1 by passing on z_3 a first time, and then have to wait for another train to pass by z_3 before using it again. Hence, the first train would both precede and follow the second on z_3. This cannot be captured in the classic modeling. Hence, in addition to actual track-circuits, we consider formal track-circuits. Several *formal track-circuits* may correspond to the same actual one, but none of them may be used twice by the same train along a route. Hence, running time consistency constraints along routes are linked to formal track-circuits for each train, while disjunctive constraints to ensure non overlapping utilization by different trains are imposed on actual ones. Another difference with respect to the classic modeling is that, here, intermediate trains must use several routes to move between parking tracks. In the literature, instead, each train must use exactly one route. Hence, for each intermediate train, we remove the single route constraints and we add constraints imposing that for each route used there must be another one that precedes it and one that follows it.

5. Solution algorithms for the G-TUSP

In this section, we introduce different algorithms based on various combinations of integrated and sequential solution of the G-TUSP sub-problems.

Kamenga et al. (2019) propose a comprehensive MILP formulation which suffers from computation time issues when its exact solution is attempted. In this section, we investigate different possibilities to reduce the computation time necessary to achieve a good quality solution. Remark that, on one hand, the literature shows that many sub-problems are already difficult to solve when they are tackled independently from one another. On the other hand, the interaction between the sub-problems significantly complicates the solution task. The possibilities we propose concretize in sequential algorithms, in which different phases integrate different sub-problems.

In the sequential algorithms, we independently tackle a sub-problem or a group of sub-problems and we use the so obtained solutions as input to the following ones. In sequential order, the TMP is the first problem to be solved in the G-TUSP. Indeed, it is thanks to a TMP solution that the time at which a train must be ready for departure is known. Then, the time spent by every train in the shunting yard can be determined. Moreover, the TMP can be instantiated without the need for a solution to other sub-problems. After the TMP, the SMP can be solved. In particular, compatible crews and shunting tracks are allocated to maintenance operations. In the next step, the TAP* assigns shunting tracks for trains to be parked on when they are not undergoing maintenance operations. Finally, the SRP is solved.

Table 5 reports the structure of different sequential algorithms. We

consider four algorithms in which we progressively find exact or heuristic solutions to different sub-problems. All algorithms are in three steps. In the first step, a first part of the G-TUSP sub-problems are solved. Therefore a partial solution is found, since a subset of sub-problems is solved. This step is named Partial solution. In this step, depending on the algorithm, the TMP and the SMP are solved with the MILP formulations detailed in Sections 5.1 and 5.2.

In addition to seeking for the coherence with the algorithm definitions, we report the detailed equations only for these sub-problems aiming to a balance between readability and description of original modeling choices.

The TAP* is solved heuristically, using the algorithm described in Section 5.3. In the second step, the missing sub-problems are integrated, but only one route is considered in the SRP to link each origin-destination pair. This step is hence named FIX-Solution. In this step, we use the formulation of Kamenga et al. (2019) in which we fix the value of the decision variables corresponding to some of the problems solved in the first step. Finally, in the third step, all sub-problems are solved considering all available alternative routes. This step is named VAR-Solution. Also this step uses the formulation of Kamenga et al. (2019) in which we fix the same variables as in the second step. The search is initialized with the FIX-solution.

The first algorithm reported in Table 5 is named S-FM: shunting with fixed matching. Here, first, the TMP is exactly solved. Then, the three remaining sub-problems are exactly solved altogether considering the matching as given and non-modifiable: the TMP is not solved again in the second and third step. Instead, all other sub-problems are solved in both these steps: the solution found in the former is used as initial solution in the search of the latter.

The second algorithm is S-FMM: shunting with fixed matching and maintenance. Here, the integrated problem composed of the TMP and the SMP is exactly solved, and its solution is used as input for the integrated exact solution of the TAP* and the SRP.

In the third algorithm, S-FMMP for shunting with fixed matching, maintenance and parking, the first step starts as in S-FMM. However, here the solution is used as an input of both a heuristic algorithm for the TAP* and the exact solution of the SRP.

Finally, the fourth algorithm is the S-FP: shunting with fixed parking. Here, as in S-FMMP, we have the exact integrated solution of the TMP and the SMP, and the heuristic one of the TAP*. However, while the TAP* cannot be modified in the last two steps, the TMP and the SMP ones can.

The reason for using sub-problem solutions as non-modifiable inputs for subsequent steps of the algorithms is the attempt of limiting the size of their search space, so that they can be explored more efficiently. Indeed, this efficiency comes at the cost of possibly excluding the overall optimal solution of the G-TUSP from the explored space. Instead, when solutions are only used to initialize the search but can be modified, it is a different approach to try to increase efficiency, this time without excluding the optimum *a priori*. As the number of alternative route appears to be an element strongly increasing the difficulty of the instances, we limit it thanks to the application of a pre-processing technique presented in Section 5.4.

Table 5
Schematic representation of the algorithms proposed. The solution found in the first step is passed to step 2, where other sub-problems are solved, including the SRP where only one route is considered available for each movement. In step 3, alternative routes are also considered. When a problem is mentioned twice, the former-step solution is used to initialize the search in the latter.

Name	Step 1		Step 2		Step 3
	Partial solution		**FIX-Solution**		**VAR-Solution**
S-FM	$MILP_{TMP}$	→	$MILP_{SMP,TAP^*,SRP^1}$	→	$MILP_{SMP,TAP^*,SRP}$
S-FMM	$MILP_{TMP,SMP}$	→	$MILP_{TAP^*,SRP^1}$	→	$MILP_{TAP^*,SRP}$
S-FMMP	$MILP_{TMP,SMP}$ ↓ $Heuristic_{TAP^*}$	→	$MILP_{SRP^1}$	→	$MILP_{SRP}$
S-FP	$MILP_{TMP,SMP}$ ↓ $Heuristic TAP^*$	→	$MILP_{TMP,SMP,SRP^1}$	→	$MILP_{TMP,SMP,SRP}$

5.1. MILP formulation for the TMP

We consider the following MILP formulation for the TMP. It is based on data introduced in Table 2. We first consider the following binary variables:

- xT_t, with $t \in T_I$, is equal to 1 if t is created and 0 otherwise;
- $xS_{t,t'}$, with $t \in T_S$, $t' \in T_I(t)$, is equal to 1 if t' is assigned to t and 0 otherwise;
- qS_t, with $t \in T_S$, is equal to 1 if t is canceled and 0 otherwise.

We also introduce the following integer variables:

- u_t, with $t \in T_T$ gives the number of uncoupling operations on t;
- v_t, with $t \in T_S$ gives the number of coupling operations on t.

The objective function to minimize integrates several penalties (1). First, it takes into account the cost of departure cancellations. The function includes uncoupling and coupling operations cost. Then, penalties for intermediate trains assignment to departing trains are added, to push towards the respect of the initial matching.

$$\min \sum_{t\in T_S} B_t qS_t + \sum_{t\in T_T} Q_C u_t + \sum_{t\in T_S} Q_H v_t + \sum_{t\in T_S}\sum_{t'\in T_I(t)} \omega_{t,t'} xS_{t,t'} \tag{1}$$

First, we need to check train compositions. We introduce constraints for the number of train units of a specific type in trains. For each type, each arriving train must have the same number of train units as the sum of all intermediate trains created after uncoupling (2). Also, each departing train must have the same number of train units as the sum of all intermediate trains assigned to it, unless it is canceled (3). If an intermediate train is not created, it can not be assigned to a departing train (4). A departure train is canceled if no intermediate train is assigned to it (5). Then, the number of uncoupling operations on an arriving train or coupling operations on a departing train is equal to the number of intermediates trains assigned minus one (6) (7).

$$m_{t,tu} = \sum_{t'\in T_I(t)} m_{t',tu} xT_{t'} \qquad \forall t \in T_T, tu \in TU \tag{2}$$

$$m_{t,tu}(1 - qS_t) = \sum_{t'\in T_I(t)} m_{t',tu} xS_{t,t'} \qquad \forall t \in T_S, tu \in TU \tag{3}$$

$$\sum_{t'\in T_S : t\in T_I(t')} xS_{t',t} \le xT_t \qquad \forall t \in T_I \tag{4}$$

$$1 - qS_t \ge xS_{t,t'} \qquad \forall t \in T_S, t' \in T_I(t) \tag{5}$$

$$u_t \ge \sum_{t'\in T_I(t)} xT_{t'} - 1 \qquad \forall t \in T_T \tag{6}$$

$$v_t \ge \sum_{t'\in T_I(t)} xS_{t,t'} - 1 \qquad \forall t \in T_S \tag{7}$$

We remark that formulation (1)–(7) allows the relaxation of integrality constraints of variables u_t and v_t.

5.2. MILP formulation for the SMP and the TMP

In this section, we present a MILP formulation which integrates the SMP and the TMP. It uses notation defined in Tables 2 and 3 as well as variables used in Section 5.1. We also define M a large constant.

We introduce non-negative continuous variables:

- $sO_{o,p,hr}$, with $o \in O_t\,(t \in T_I)$, $p \in P^o$, $hr \in HR^o$, time at which operation o starts on shunting track p with the crew hr;
- dO_t, with $t \in T_I$, time at which intermediate train t ends all its operations;
- D_t, with $t \in T_S$, delay suffered by departing train t when exiting the control area.

Moreover, we introduce binary variables:

- $xO_{o,p,hr}$, with $o \in O_t\,(t \in T_I)$, $p \in P^o$, $hr \in HR^o$, is equal to 1 if o is carried out on shunting track p by crew hr and 0 otherwise;
- $y_{o,o',hr}$ with $o \in O_t$, $o' \in O_{t'}\,(t, t' \in T_I, t < t')$, $hr \in HR^o \cap HR^{o'}$, is equal to 1 if crew hr performs operation o before operation o' and 0 otherwise;
- $yP_{o,o',p}$, with $o \in O_t$, $o' \in O_{t'}\,(t, t' \in T_I, t < t')$, $p \in P^o \cap P^{o'}$, is equal to 1 if operation o is carried out before o', and they are both carried out on shunting track p, 0 otherwise.

The objective function to minimize includes function (1) and adds penalties for departures delay and operations cancellation (8). We note that we can have an operation cancellation penalty only if the intermediate train concerned by the operation is actually created.

$$\min \sum_{t \in T_S} Q_t D_t + \sum_{t \in T_I, o \in O_t} \omega_o \left(xT_t - \sum_{p \in P^o, hr \in HR^o} xO_{o,p,hr} \right) + \sum_{t \in T_S} B_t q S_t + \sum_{t \in T_T} Q_C u_t + \sum_{t \in T_S} Q_H v_t + \sum_{t \in T_S} \sum_{t' \in T_I(t)} \omega_{t,t'} xS_{t,t'} \tag{8}$$

First of all, all constraints in the TMP formulation (2)–(7) must be respected. Moreover, any operation carried out on t must use exactly one crew and one shunting track (9). The starting time of an operation is set to 0 if it is not assigned to a shunting track (10). Operations must start after the concerned train arrives on a shunting track (11), if they are performed. Remark that, when solving this problem, we have no information on the precise time that will see the train entering a shunting track. Hence, to be conservative, we consider this time equal to the train's arrival time plus (a_t) the maximum duration of a shunting movement (mr). If uncoupling operations take place on the train, for each of them we also add the minimum uncoupling time (mc) and the time needed for another shunting movement. An operation performed by crew hr must start after the shift start time of hr (12) and end before the shift end time (13). Note that (12) imposes that the starting time of an operation is 0 if it is not assigned to a crew. If operation o' follows operation o, then o' starts after the end of o. We consider the case in which the successive operations are performed on the same track (14) and the one in which they are performed on different tracks and a shunting movement is necessary (16). Constraints (17) specify when intermediate trains end all their performed operations. If these operations end after the departure time of the associated departing train, then the latter is delayed (18).

$$\sum_{hr \in HR^o, p \in P^o} xO_{o,p,hr} \leq xT_t \quad \forall t \in T_I, o \in O_t \tag{9}$$

$$\sum_{hr \in HR^o} sO_{o,p,hr} \leq M \sum_{hr \in HR^o} xO_{o,p,hr} \quad \forall t \in T_I, o \in O_t, p \in P^o \tag{10}$$

$$sO_{o,p,hr} \geq a_t + mr + (mc + mr)u_t - M(1 - xO_{o,p,hr}) \quad \forall t \in T_I, t' \in T_I(t), \tag{11}$$

$$o \in O_{t'}, p \in P^o, hr \in HR^o$$

$$sO_{o,p,hr} \geq sR_{hr} xO_{o,p,hr} \quad \forall t \in T_I, o \in O_t, p \in P^o, \tag{12}$$

$$hr \in HR^o$$

$$sO_{o,p,hr} \leq (eR_{hr} - pR^o) xO_{o,p,hr} \quad \forall t \in T_I, o \in O_t, p \in P^o, \tag{13}$$

$$hr \in HR^o$$

$$sO_{o',p,hr'} \geq sO_{o,p,hr} + pR^o xO_{o,p,hr} - M(1 - xO_{o',p,hr'}) \quad \forall t \in T_I, (o, o') \in E_t, \tag{14}$$

$$p \in P^o \cap P^{o'},$$

$$hr \in HR^o, hr' \in HR^{o'} \tag{15}$$

$$sO_{o',p',hr'} \geq sO_{o,p,hr} + (pR^o + mr) xO_{o,p,hr} - M(1 - xO_{o',p',hr'}) \quad \forall t \in T_I, (o, o') \in E_t, \tag{16}$$

$$p \in P^o, p' \in P^{o'}, p \neq p',$$

$$hr \in HR^o, \quad hr' \in HR^{o'}$$

$$dO_t \geq sO_{o,p,hr} + pR^o xO_{o,p,hr} \quad \forall t \in T_I, o \in O_t, p \in P^o, \tag{17}$$

$$hr \in HR^o$$

$$D_t \geq dO_{t'} - d_t - M(1 - xS_{t,t'}) \quad \forall t \in T_S, t' \in T_I(t) \tag{18}$$

As two operations can not use a crew at the same time, we set disjunctive constraints (5.2), (5.2).

$$sO_{o',p',hr} \geq sO_{o,p,hr} + pR^o xO_{o,p,hr} - M(1 - y_{o,o',hr}) \tag{19}$$

$$\forall t, t' \in T_I, o \in O_t, o' \in O_{t'}, hr \in HR^o \cap HR^{o'}, p \in P^o, p' \in P^{o'}, t < t'$$

$$sO_{o,p,hr} \geq sO_{o',p',hr} + pR^{o'} xO_{o',p',hr} - My_{o,o',hr} \tag{20}$$

$$\forall t, t' \in T_I, o \in O_t, o' \in O_{t'}, hr \in HR^o \cap HR^{o'}, p \in P^o, p' \in P^{o'}, t < t'$$

Also, as two operations cannot be performed on the same shunting track concurrently, we set disjunctive constraints (5.2), (5.2).

$$sO_{o',p,hr'} \geq sO_{o,p,hr} + (pR^o + bt) \cdot xO_{o,p,hr} - M(1 - yP_{o,o',p}) \tag{21}$$

$$\forall t, t' \in T_I, o \in O_t, o' \in O_{t'}, p \in P^o \cap P^{o'}, hr \in HR^o, hr \in HR^{o'}, t < t'$$

$$sO_{o,p,hr} \geq sO_{o',p,hr'} + (pR^{o'} + bt) \cdot xO_{o',p,hr} - MyP_{o,o',p} \tag{22}$$

$$\forall t, t' \in T_I, o \in O_t, o' \in O_{t'}, p \in P^o \cap P^{o'}, hr \in HR^o, hr \in HR^{o'}, t < t'$$

5.3. Heuristic for the TAP*

In two of the algorithms we propose, the TAP* is solved heuristically, taking in input a solution of the integration of TMP and SMP. This solution can be obtained with the MILP formulation in Section 5.2, for example.

The heuristic for the TAP* we propose greedily assigns parking time slots to trains in the shunting yards. Such greedy assignment takes into account the constraints on parking time slots imposed by the input solution: the maintenance schedule and the coupling and uncoupling operations planned impose the assignment of shunting tracks to specific trains at some time instant. Once all these constraints are set, the greedy

approach assigns the remaining parking time slots, which we refer to as *unassigned slots*.

Algorithm 1. Greedy algorithm for the TAP*

The heuristic is described in Algorithm 1. In an initialization phase, we create the slots associated to maintenance operations: we associate to each train a slot on the track necessary to carry out each operation, starting and ending when imposed in the SMP solution. Then, we sort trains according to their arrival time in the yard. If many intermediate trains arrive at the same time, the one whose first maintenance operation

```
Create maintenance slots
Sort intermediate trains in increasing order of arrival time
for t – intermediate train do
    for o – operations of t do
        s̄ ≡ aimed starting time;   start slot on track of o as early as possible;
        if slot cannot start at s̄ then
            if o is not the first operation and track of o′ predecessor of o is free then
                extend slot of t on track of o′;
            else
                create an unassociated slot for t;
    if t needs uncoupling then
        start uncoupling operation management (Algorithm 2);
    if t needs coupling then
        start coupling operation management;
Sort unassociated slots in increasing order of starting time
for unassociated slot do
    determine set of feasible tracks;
    select a track with the shortest remaining length in the set;
```

Algorithm 2. Greedy algorithm for the TAP*: uncoupling operation management

```
1  ô ≡ first operation of t; t̂r ≡ track of ô; ŝ ≡ start time of t's slot on t̂r; ĝ ≡ uncoupling
     group of t; â ≡ arrival time;
2  if t is the first treated in ĝ then
3      if t̂r is free between â and ŝ then
4          extend slot of t on t̂r;   set uncoupling track of ĝ to t̂r;
5      else
6          create an unassociated slot for t;   set uncoupling track of ĝ to dummy track;
7  else
8      if uncoupling track of ĝ is dummy then
9          if t̂r is free between â and ŝ then
10             extend slot of t on t̂r;   set uncoupling track of ĝ to t̂r;
11             for t̃ – already considered intermediate train of ĝ do
12                 remove initial unassociated slot of t̃;
13                 associate to t̃ a slot on t̂r, ending as late as possible;
14                 if t̂r is not free until the beginning of t̃'s first operation then
15                     advance the starting time of slot corresponding to t̃'s first operation s̃;
16                     if latest ending time of slot on t̂r < earliest starting time of s̃ then
17                         create unassociated slot for t̃
18         else
19             create an unassociated slot for t;
20     else
21         ẽ ≡ latest ending time of a slot of a train of ĝ on uncoupling track;
22         if uncoupling track of ĝ is free until ŝ then
23             create slot for t on uncoupling track;
24         else
25             create an early slot on uncoupling track;
26             if t̂r is free from ẽ then
27                 extend slot of t on t̂r;
28             else
29                 create an unassociated slot for t between ẽ and ŝ;
```

starts the earliest is sorted first. For sake of readability, with a slight abuse of notation, we refer to an intermediate train arrival and departure time to indicate the arrival and departure time of the corresponding arriving and departing train. Similarly, we will say that an intermediate train needs coupling or uncoupling.

Following the defined order, we deal with one train at a time. First, we increase the duration of maintenance slots whenever possible. To define the aimed starting time of the slot corresponding to the first operation, we check if the train needs uncoupling. If so, we set the aimed starting time to the arrival time plus the decoupling operation time. Otherwise, we set it to the arrival time. For the following operations, we set the aimed starting time as the end of the previous operation following the input solution. After this definition, we create a slot on the necessary track as early as possible. If the starting time of the slot cannot be equal to the aimed starting time, due to previously assigned slots, we try to leave the train on the track it occupied for the preceding operation, if any; if this track was already assigned for a different operation and we cannot leave the train there, or if we are considering the first operation, we create an unassociated slot. Second, we deal with necessary uncoupling and coupling operations, in this order. For brevity, we only detail the uncoupling operation management (formalized in Algorithm 2), as the coupling one is equivalent, taking place at the end of the train's presence in the shunting yard rather than at the beginning. If the intermediate train needs to be uncoupled, we include it in the group of all intermediate trains that are associated to its arriving one: we need to assign to all these trains a slot on the same track at their arrival at the shunting yard. If this is the first train dealt with in the group, we try to assign it a slot on the track where its first operation takes place, starting at the arrival time: this will be the uncoupling track of the group. If it is not possible, we create an unassociated slot for the train and we record the absence of assigned track to the group, by defining a dummy track as the uncoupling one. If the train is not the first one considered in the group, we check if the uncoupling track has already been assigned. If it has not, we try to assign the one where t's first operation is carried out. If it is not possible, we create an unassociated slot for t. If it is possible, we advance the starting time of t's slot as suitable, and we associate slots on this same track to all the already considered intermediate trains of the group. These slots will have the same starting time of the one of t and they will end as late as possible in coherence with the already scheduled slots. If necessary, unassociated slots are created. If instead an uncoupling track had already be assigned to the group, we create a slot for t on this track ending as late as possible. If the slot corresponding to the first operation of t cannot start when the one on the uncoupling track must end, then we create an unassociated slot.

Finally, we consider all unassociated slots in order of starting time. For each of these slots, we identify the set of feasible tracks, i.e., the ones that satisfy four criteria:

1. They must be compatible with the intermediate train, e.g., they must be electrified if the train is electric;
2. They must respect the length constraint, i.e., have a sufficiently long available portion for the whole slot duration;
3. They must respect the crossing constraint, i.e., allow the association of the slot without the occurrence of crossing issues;
4. They must be coherent with previously assigned slots in case of necessary coupling or uncoupling.

Then, a slot is assigned to the track with the shortest remaining length. The yard layout and the latter criterion may impose an entrance side to the shunting track. When the entrance side is not constrained, we set an entrance side defined in input for each shunting track. In principle, this algorithm may return an infeasibility if the shunting yard is saturated. Such infeasibility may be solved by defining a backtracking procedure. However, as we never experienced an infeasibility, we do not introduce such procedure in this paper.

We illustrate the functioning of the algorithm in Fig. 4. Maintenance slots appear with full-colored rectangles, and they are assigned following the SMP input solution. Different colors correspond to different shunting tracks. The first train considered is t_1. Its blue slot is extended to start as soon as the train arrives. Its yellow slot cannot start right after the end of the blue operation, due to t_5 yellow maintenance slot, and the blue one cannot end right before the start of the yellow operation due to t_4 blue maintenance slot. Hence, an unassigned slot is created. The train needs uncoupling. As the blue slot can start at the arrival time, then this is set as the uncoupling track of the group. Then, we consider t_2. We can advance the beginning of its brown slot at the arrival time plus the uncoupling duration. Then, right after the first operation, it can move to the green track where its second operation is carried out. As the green track is free afterwards, the slot can be extended until departure time. t_2 requires both uncoupling and coupling. As for the former, it is the second train considered in the group, and the uncoupling track has already been set: we associated a short blue slot at the train arrival. As for the latter, t_2 is the first train of the group, and we can set the green one as the coupling track. The train that needs to be coupled with it is t_3, which first has a dark blue slot in which two operations are carried out, and then a red one. As the red track is occupied by t_5 before t_3 while the dark blue one is free, it is the latter that is assigned to the train between the two operations. As t_3 needs coupling and a coupling track for the group has been set, it is assigned a green slot before departure. The assignment of the remaining slots are straightforward. What shall be noted is that there are cases in which uncoupling or coupling tracks cannot be defined. This is the case of t_5 and t_6, and t_6 and t_7. Indeed, the track of the last operation of t_5 cannot be used as t_3 has a slot there, and t_6 and t_7 only have unassigned slots as they have no operations scheduled. When dealing with unassigned slots, we have to ensure the coherence of the final slots of t_5 and t_6 and of the initial ones of t_6 and t_7.

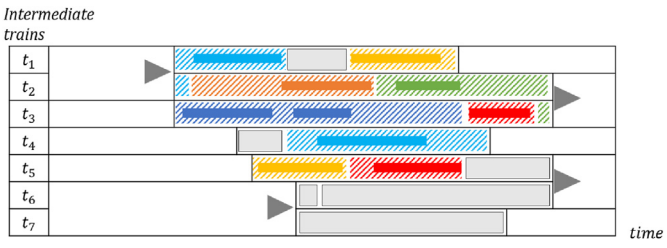

Fig. 4. Example of parking time slots assigned in the greedy heuristic for the TAP*.

5.4. Pre-processing technique for limiting alternative routes

The size of the formulation proposed by Kamenga et al. (2019) and detailed in the supplementary material for integrating, in particular, the SRP and the TAP is strongly influenced by the number of routes available to a train. The same holds for the part of the reference formulation used in this paper. Indeed, the number of variables generated by the model can vary up to power 4 of the number of routes per train. In particular, for example, parking variables concern two trains to control their coherent occupation of a track. They are indexed on four route, i.e., the entrance and exit routes of both trains. For keeping the size manageable while preserving routing flexibility, we implement a selection of alternative routes based on a simplification of the approach of Riezebos and Van Wezel (2009). Specifically, for each origin-destination pair, we consider only the k shortest ones in terms of nominal running time, with k being a fixed integer number. Differently from Riezebos and Van Wezel (2009), we do not need to consider the sequence of shunting tracks traversed by a route, since for us shunting tracks can only be either the origin or the destination of the route itself. In addition, some routes that are trivially useless are removed. More precisely, the routes in which more than two turnarounds are performed are not taken into account as well as the

routes that link two shunting tracks passing through platforms.

6. Experimental analysis

In this section, we present the results of the assessment of the four proposed algorithms (Table 5). The algorithms are coded in Java, and MILP models are solved with the commercial solver CPLEX 12.8. The experiments are run with an Intel®Xeon™CPU X5650 2.67 GHz, 12 cores, 24 GB RAM. We study traffic in Metz-Ville station. It is a major hub for Eastern France railway traffic. We tackle real scenarios which include perturbations such as arrival delay or track closure. We also consider scenarios in which we insert fictive perturbations in order to enrich the experimental analysis.

We set running time limits for each algorithm's phase that are mentioned in Table 5:

- In the first step, that gives a partial solution (partial solution), the running time is limited to 30 s. In S-FM and S-FMM, this is the running time for solving a MILP formulation, while in S-FMMP and S-FP the running time includes both a MILP solution and a heuristic run.
- In the second step, where a complete solution without alternative routes (FIX-solution) is sought, the running time is limited to 600 s.
- In the last step, where a complete solution considering alternative routes (VAR-solution) is sought, the running time is limited to 900 s.

We set these time limits to fit the practical needs expressed by operation experts of the Metz-Ville station. To understand whether slightly different values may bring performance improvements, we perform further tests where the time limit of the last step is 1200 s. The results are qualitatively equivalent with the two set-ups. Hence, in the following we only report and discuss the ones achieved in 900 s.

As discussed in the introduction, the G-TUSP objective function includes penalties due to delays, departures and maintenance operations canceling, coupling and uncoupling operations, modifications of the initial matching, as well as the number and the duration of shunting movements. Coefficients in the objective function reported in Table 6 have been set by operation experts of Metz-Ville station. Remark that these values are not monetary estimations: they capture the relative importance of an increase of the specific objective of one unit, whatever the unit represents. For example, according to the experts, canceling a maintenance operation is 13.5 times preferable to canceling a departure, if the least valued operation and departure are considered. These coefficients make canceling departures extremely unlikely: the penalty associated with canceling a departure is extremely high with respect to the other objective function components. Indeed, this is realistic, as departures are actually canceled only when no viable alternatives exist or when their cost is disproportionate.

6.1. Case study

We consider traffic in Metz-Ville station and its passengers shunting yards represented in Fig. 5. It is a major junction where the Nancy-Luxembourg and Metz-Strasbourg lines intersect. The station mainly hosts regional trains. Many of these trains start or end their service in Metz-Ville. The area is 3.8 km long and has 10 platforms including a dead-end one. Yards F1 and F2 are controlled from a signal box, while switches are directly handled by a ground-agent in yards F3 and F4. The infrastructure is composed of 138 track-circuits, 68 signals, 421 block sections and 405 routes. Remark that the number of block sections is much larger than the one of track-circuits. This is typical in shunting yards, where, on the one hand, most tracks are bi-directional: track-circuits belong in general to at least two block sections, one per direction. On the other hand, a very large portion of track-circuits include switches, to allow maximal flexibility to shunting movements, hence often belonging to more than a block section in each direction. 18 shunting tracks are available for passenger train units. Yard F3 contains a track with a washing machine. Two tracks in yard F3 and one track in yard F4 have equipment for technical inspection. A predefined coupling exists between the other shunting track and rolling stock types. Hence, each train can be parked only on a subset of these tracks.

Table 6
Coefficient of penalties in the G-TUSP objective function.

Type of cost	unit	range
departure cancellation	per departing train	135 K - 0.5 M
maintenance operation cancellation	per operation	10K–13.5K
delay	per second of delay	900–1800
number of shunting movements	per movement	15
duration of shunting movements	per second of movement	1
modification to the initial matching	per modification	7
coupling or uncoupling operation	per operation	900

As for the real-life scenarios, we consider two regular and two perturbed week days from 2018 traffic data. One perturbed day includes several trains suffering arrival delays coming form Luxembourg between 4:30pm and 7:40pm. These delays were due to urgent infrastructure maintenance works during the day. These delays were known 4 h before operations. As trains arrive late in the evening peak hour, their maintenance can not start on time. In this scenario, in reality as cleaning crews shift ended too early, some cleaning operations were actually postponed to the morning or canceled. During the other perturbed day, one of the two north side shunting necks is closed because of a major track failure. A shunting neck is a track used for train turnarounds during shunting movements. The closed shunted neck is circled in red in Fig. 5 (north side, up). Another neck remains available on the north side of the station, and it is circled in green in the figure. The track closure scenario reduces the set of possible routes and increase the likelihood of the occurrence of conflicts. For example, if a train has to be moved from yard F2 to yard F4, in this scenario it has to use the green shunting neck and traverse the main station tracks on which passing trains travel too. For each of the four days, we consider a day and a night time scenario. The former includes traffic between the morning (7am) and the evening peak hour (7pm). In this interval, between 14 and 18 train units have to be shunted and there are between 241 and 243 passing trains. The latter includes trains between evening (6:30pm) and the next morning peak hour (7:30am). In this interval, between 19 and 26 train units have to be shunted and there are between 158 and 165 passing trains.

In each scenario, there are 7 types of train units, on which 4 different maintenance or cleaning operations may have to be performed: arrival check, internal cleaning, WC cleaning and external cleaning. The number of alternative routes mentioned in Section 5.4 does not exceed 5.

Additional scenarios are created by adding fictive perturbation in the real-life ones. First, we generate scenarios in which two trains suffer a 2 h arrival delay. These trains are randomly chosen with uniform distribution among all trains requiring shunting. We set the 2-h delay in agreement with the operation experts of Metz-Ville station, based on the consideration that the G-TUSP is a pre-operational problem, as mentioned in the introduction. Hence, train delays are the ones expected due to some planned perturbation. As these perturbations are typically quite important, as a few-hour track possession for infrastructure maintenance along a line, expected delays are often quite large. Second, we increase the number of delayed trains to four. Third, we consider a new perturbation, in which track 74 is closed. This is one of the two south side shunting necks, it is circled in blue in Fig. 5 (south side, up). Therefore, trains have to use shunting neck 29 circled in brown in Fig. 5 to perform a turnaround on the south side. When this happens, a high number of shunting movements encounter station traffic.

In summary, 32 instances including 8 real-life scenarios are tackled and the Appendix reports their details:

- 4 instances do not contain perturbations (all real-life scenarios),
- 14 instances only contain delays (2 real-life and 12 fictive scenarios),

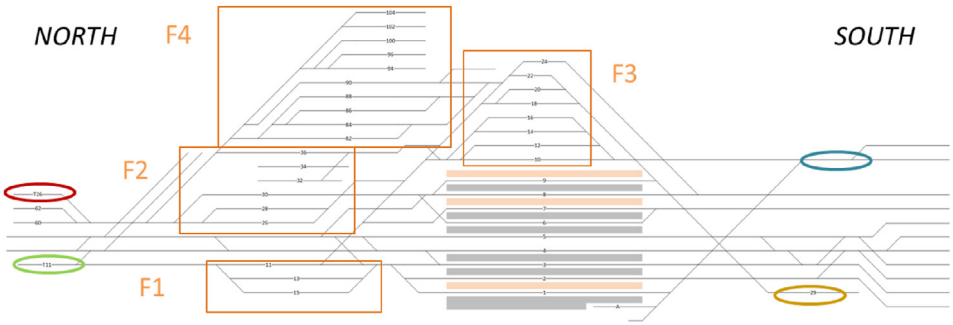

Fig. 5. Layout of Metz-Ville station: Filled rectangles represent platforms. Yards are orange squared and shunting necks are circled. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

- 8 instances contain only track closures (2 real-life and 6 fictive scenarios),
- 6 instances contain both delays and track closures (all fictive scenarios).

Before running the algorithms on each of these instances, we execute the feasibility test described by Haahr et al. (2017). It is based on an aggregated capacity assessment. Specifically, we check that the total length of trains that occupy the shunting yard at any time does not exceed the total shunting tracks length. For this check, we consider the actual train arrival times and the planned departure ones. This aggregated check requires $O(|T_T| + |T_S|)$ time.

6.2. Experimental results

In this section, we focus first on computation times to compare our algorithms, then we tackle objective function values.

The first analysis shows the impact of the integration of different sub-problems on the difficulty of the G-TUSP. Computation times are summarized in Table 7 for each step in the algorithms. In the table, we report the minimum, mean and maximum computation time used by each algorithm in the three steps. Let us remark that the maximum computation time (30, 600 and 900 s for the three steps, respectively) plays a role only if CPLEX does not prove the optimality of a solution earlier. If this happens, the corresponding algorithm step is stopped and the computation time is equal to the time limit. Otherwise, the algorithm step stops with the optimality proof. All instances are considered in the mean computation time, disregarding the reason why the execution stops.

The main observation concerns the solution of the TAP*. This problem has a great impact on the difficulty of the G-TUSP. Computation times increase significantly if the TAP* is integrated with other G-TUSP sub-problems and an exact solution for this problem is searched. Recall that S-FM and S-FMM neglect the TAP* in Step 1 and solve it exactly in Steps 2 and 3, integrating it with the SRP and both the SMP and the SRP, respectively. Differently, in S-FMMP and S-FP, Step 1 is the same and it solves the TAP* with a heuristic. The TAP* solution is considered fixed from there on. If we focus on Step 1, the computation time of the last two algorithms does not exceed 6.6 s with an average only slightly higher than the one of the first two algorithms (4.1 s more than S-FM and 1.2 s more than S-FMM). In Step 2, when the TAP* is considered the computation time reaches 362 (S-FM) and 373 (S-FMM) seconds, while it remains lower than 1 min otherwise. In Step 3, only S-FMMP always achieves optimality: the computation time is at most 715 s. S-FMM, which in this step only differs for the integration of TAP*, only solves 12.5% of the instances to optimality within 900 s. The latter algorithm exits the search with an average optimality gap of 4.5% and a maximum of 7.3%. Although reaching in some cases the time limit, S-FP solves 93.9% of the instances to optimality in step 3 with an average gap of 0.3% and a maximum gap of 0.4%. Finally, S-FM never manages to close the gap, exiting with a value of 4.5% in average, and getting to a maximum of 9.5%. Indeed, this algorithm re-assess the highest number of sub-problems in Step 3, including the TAP*. A quantitative measure of the impact of the integration of this problem is also given by the number of binary variables included in the MILP formulations of Step 3 of the different algorithms: they are 1838K and 1831K for S-FM and S-FMM, respectively, while only 49K and 162K for S-FMMP and S-FP.

Focusing on the other sub-problems, we observe that the computation time to solve Step 1 in S-FM, and hence to solve the TMP, does not exceed 0.8 s. Indeed, As mentioned by Freling et al. (2005), the TMP alone is rather easy to handle. The same holds when this problem is integrated with the SMP: solving the MILP that constitutes Step 1 of S-FMM takes between 2.2 and 5.1 s. Apparently, integrating the SMP has a minor impact on the difficulty of the problem. Indeed, in Step 2, when solving to optimality the TAP and the SRP (S-FMM) or the TAP, the SRP and the SMP (S-FM) the average computation times differ by less than 10%. In Step 3, where alternative routes are allowed, S-FM has an average computation only slightly higher than S-FMM.

After this analysis on difficulty, we focus on the impact of integrating G-TUSP sub-problems on solution quality. We discuss the quality of solutions according to objective function components, first, and scenario types, second.

Table 8 reports the mean values of the G-TUSP objective function and some of its components returned by the four algorithms as their final solution, i.e., after Step 3. Departure cancellations are not mentioned in Table 8 since no departure is canceled in the solutions. Indeed, cancellations are extremely rare in Metz-Ville station: it is a major hub where

Table 7
Computation times of the algorithms proposed.

	Step 1			Step 2			Step 3		
	partial solution			FIX - solution			VAR - solution		
	(sec)			(sec)			(sec)		
algorithm	min	mean	max	min	mean	max	min	mean	max
S-FM	0.2	0.6	0.8	110	192.4	362	900	900	900
S-FMM	2.2	3.5	5.1	115	177.1	373	572	870.6	900
S-FMMP	3.1	4.7	6.6	12	23.9	51	86	257.3	715
S-FP	3.1	4.7	6.6	12	25.6	56	153	428.8	900

extra train units are often available and the re-utilization times are typically long enough to allow the necessary flexibility. Recall that the objective function represents an overall penalty associated to the specific decisions made, and as such it has no specific unit. S-FP gives solutions with the best average objective function, while S-FMMP gives the worst. This is not surprising since, in Steps 2 and 3, the latter considers fixed the solutions of all sub-problems but the SRP, while S-FP only does so for the TAP*. Indeed, here the specific solution of the TAP* does not really make a difference, as S-FM and S-FMM solutions are in average worse than S-FP and better than S-FMMP.

If we look at the number of modifications to the planned train matching (third column of Table 8), we see that S-FM is the algorithm that modifies the planned train matching the least, while S-FP modifies it the most. Indeed, S-FM, S-FMM and S-FMMP only modify the train matching in Step 1, integrating the TMP at most with the SMP. As it can be expected, the higher the number of sub-problems the TMP is integrated with, the higher the number of modifications.

The average total delay and number of canceled maintenance operations are reported in columns four and five of Table 8. S-FMMP gives solutions with the longest delay, and in particular with significantly longer delay than S-FMM. In both algorithms, a MILP integrates the TMP and the SMP in Step 1, which then returns the same solution for both algorithms. This solution is not re-assessed in the following steps. Hence, maintenance operations have the same schedule in the final solution of both algorithms. The higher delay in S-FMMP is due to shunting movements between the shunting tracks and the platforms. Indeed, in S-FMMP, as the TAP* is heuristically solved in Step 1, the SRP solved in Steps 2 and 3 has a limited alternatives to find good solutions: the shunting track where the trains' routes begin and finish have to be consistent with the TAP* solution. In S-FMM, the TAP* and the SRP are integrated in Steps 2 and 3: a larger set of possible routes is available and traffic conflicts can be avoided. Although the TAP* is heuristically solved in Step 1 of S-FP, too, this algorithm gives the solutions with the shortest delay in average. Indeed, this algorithm profits from the solution of the TMP in Steps 2 and 3, together with the SMP and the SRP. Therefore, it solves a trade-off between total delay, number of modifications to the initial matching and number of coupling or uncoupling operations, which allows the reduction of delay. S-FM achieves the worst results in terms of number of maintenance operations canceled and quite bad results in terms of total delay. It is the only algorithm in which only the TMP is solved in Step 1. The train matching is then considered fixed in the following. As mentioned in Section 4.1, the MILP formulation for the TMP used by S-FM includes a penalty if the solution does not let enough time for maintenance operations. However, the penalty simply considers operations duration and does not take into account crews or tracks availability, which instead may have an impact on solutions. The average number of operations that need to be canceled to fit the train matching is then slightly higher than for the other algorithms. The solutions found by S-FMM, S-FMMP and S-FP have the same number of maintenance operations cancellations for all instances. While for most of the instances S-FM finds the same numbers, it cancels an additional operation for one instance. It is a daytime instance (D3-2 in Appendix) in which many arrivals are delayed. In this case, S-FM gives a solution which follows the initial matching, but there are not enough crews to carry out internal cleaning and ensure the on-time departures. Therefore, one internal cleaning operation is canceled to avoid a long delay. Differently, S-FMM, S-FMMP and S-FP change the initial matching so that no operation is canceled.

Looking at the shunting movements, whose number and duration are reported in the last two columns of Table 8, S-FMMP and S-FP provide the solutions with fewer shunting movements. These algorithms return solutions with the same number of shunting movements. Indeed, the number of shunting movements is set with a TAP* solution: the more shunting tracks an intermediate train is parked on the more shunting movements are performed. In S-FMMP and S-FP, the TAP* solution is obtained with a heuristic in Step 1. The greedy algorithm for the TAP* favors the minimization of the number of shunting movements. On the contrary, once the SRP is solved taking as input a TAP* solution, the total duration of shunting movements is higher than in algorithms where the TAP* and the SRP are integrated (S-FM and S-FMM).

Fig. 6 shows mean objective function values for different types of scenario.

In the perturbation-free instances, S-FM and S-FMM provide slightly better solutions than S-FMMP and S-FP. Nevertheless, all the algorithms provide solutions without delays. The difference is due to a higher duration of shunting movements, that increases the cost of S-FMMP and S-FP solutions.

In the scenarios with arrival train delays, S-FP gives particularly good results compared to the other algorithms. Its final delays are almost twice as low as the other on average. Moreover, S-FM gives the worst solutions in average. This attests the benefit of integrating the TMP and the SMP in these scenarios. If a train matching is set, then SMP solutions are often of low quality in case of delays. Indeed, the train matching set in S-FM is different from the one in S-FMM. The latter algorithm provides better solutions than S-FMMP in delay scenarios. As the two algorithms use the same solution for the TMP and the SMP, found by a MILP in Step 1, this observation highlights the benefit of integrating the TAP* and the SRP.

Conversely, in track closure scenarios, S-FM provides the best solutions on average. The reason why S-FMM and S-FMMP are less successful may be linked to the SMP solution. Indeed, setting the maintenance schedule might be an issue, since it limits the set of alternative routes for a train. During track-closure periods, alternative routes are crucial to

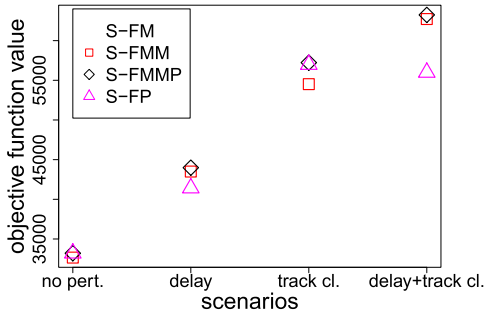

Fig. 6. Mean values of objective function for different types of scenarios.

Table 8
Mean values of objective function components (number of modifications to the planned train matching, of maintenance operations canceled, of coupling and uncoupling operations, of shunting movements, total shunting movements duration).

algorithm	objective	# modif. matching	total delay (sec)	# oper. cancel.	# coupling or uncoupling	# shunt mov.	total shunt. mov. time (sec)
S-FM	48093	1.4	604	0.22	2.9	19.7	16078
S-FMM	48502	1.7	549	0.19	2.9	20.3	16830
S-FMMP	49559	1.7	707	0.19	2.9	18.8	16972
S-FP	47012	2.2	527	0.19	3	18.8	17008

avoid potential traffic conflicts. Solving the TAP* in Step 1 can be detrimental for the same reason. Indeed, S-FMMP gives worse solutions than S-FMM. S-FP manages to partially compensate the early solution of the TAP* by re-assessing the solutions of the other problems. However, this does not allow the complete recovery of solution quality. In Table 9, we propose additional details for these scenarios, concerning the number and the duration of shunting movements. We can observe that S-FMM generates the highest number of shunting movements. Through them, the algorithm avoids conflicts with passing trains, which otherwise would cause departure delays. Conversely S-FP and S-FMMP give solutions that have fewer shunting movements and longer delays.

In scenarios that combine delays and track closures, S-FP provides more better solutions than other algorithms. In particular, they have shortest total delay.

For the eight real-life instances available, we compare the solutions of our algorithms with the ones implemented by yard planners. Fig. 7 depicts this comparison. In general, our algorithms perform at least as well as planners' decisions, and in some cases they do better. As shown in Fig. 7a, S-FMM, S-FMMP and S-FP cancel a maintenance operation less than planners. Moreover, the solutions implemented by planners stick to the initial matching, while our solutions modify it in two instances (Fig. 7b). When analyzing our results, these modifications have been declared efficient by experts of Metz-Ville station. Apart from these indicators which appear in the objective function, solutions differ mostly in terms of routing and maintenance operation schedules. As a consequence of these differences, all our algorithms obtain solutions with delay smaller than or equal to the planners' one (Fig. 7c). The only exception is for instance D2-0 in which S-FM and S-FMM have a total delay 10 s longer than the planners' one, which represents a 0.1% difference.

In summary, we can conclude that there is no algorithm that always outperforms the others, although integrating the TAP* to other problems significantly increases the problem difficulty. However, this increase does not always imply solution quality worsening in the computation time limit considered. Globally, we think that S-FP can be considered the best algorithm, as it achieves the best objective function values overall (Table 8) and in two out of four types of scenarios. In the two remaining types, its average objective function value is 2% and 14% worse than its best competitor S-FM. Instead, when S-FM is not the best, the difference with respect to S-FP is 9% and 10% in two types of scenarios (Fig. 6).

7. Conclusion

In this paper, we proposed four solution algorithms for the G-TUSP, based on the sequential or integrated solution of different groups of sub-problems. We assess their performance on a real case study, observing their computation times and the quality of their solutions. Our experiments show that the TAP* is sub-problem that mostly complicates the G-TUSP. Once this sub-problem tackled, the others can be solved quite easily. However, to successfully solve the TAP*, appropriate solutions of the TMP and the SMP must be provided. Indeed, different instance characteristics may imply different relative performance of the algorithms proposed. In the paper, we assessed these performance when various types of perturbations occur.

Table 9
Shunting movements in track closure scenarios.

algorithm	mean number of shunting movements	mean total duration of shunting movements (sec)
S-FM	21.07	17887
S-FMM	22.36	19649
S-FMMP	19.14	19395
S-FP	19.14	19668

In future research, on the one hand, we will focus on the SRP. Namely, we will work on the definition of its search space, i.e., on the set of alternative routes to be considered for each train. Indeed, the search space shall not be too large, so as to be able to effectively explore it, but it shall be large enough and include high quality solutions, so as to allow the necessary routing flexibility. A first step towards this definition is the identification of relevant performance indicators for routes, following, e.g., the directions pointed out by Riezebos and Van Wezel (2009). For example, a convenient route generates *a priori* few traffic conflicts.

On the other hand, we will investigate different possibilities to improve the performance of the solution of the MILP formulations we proposed, for example through relaxation techniques.

Another research direction will consist in studying different models for the G-TUSP. Specifically, the TAP* and the SMP may be suitable for the use of a discrete time model, as done by Jacobsen and Pisinger (2011). Instead, the SRP rather requires a continuous time model to consider accurately shunting movements capacity utilization. Then, we will investigate ways to link discrete and continuous time models of the different sub-problems.

Finally, a possibility that will be worth assessing is the use of an eigenmodel approach to iteratively solve sub-problems so as to find the overall problem solution. Such approach has been successfully used by Schöbel (2017) to deal with the integration of three tactical problems in the railway planning process: line planning, timetabling and rolling stock rostering. It is reasonable to conjecture that the approach could be promising for the G-TUSP too, provided that sub-problems can be solved fast enough to allow the performance of several iterations in the rather short computation time available in the pre-operational phase.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

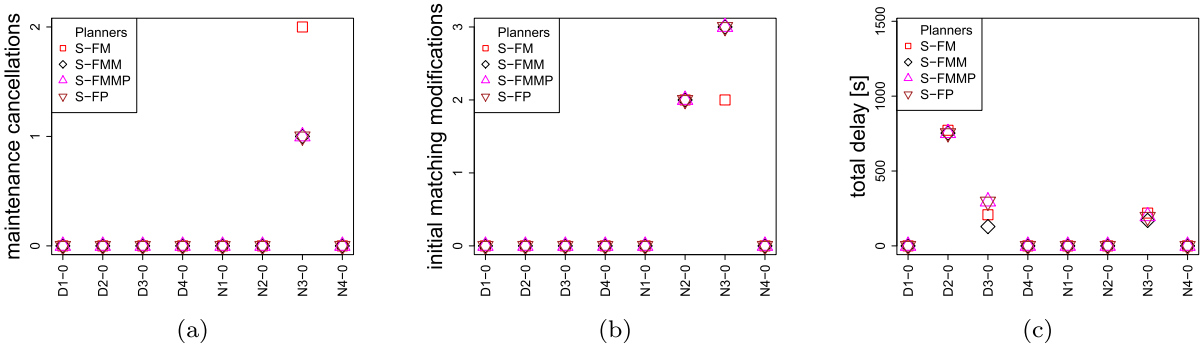

Fig. 7. Comparison of solutions obtained with algorithms and solutions implemented by yard planners in the available real instances.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ejtl.2021.100042.

Appendix. Instances tackled in the experimental analysis

In following table we list the main characteristics of the instances solved in the experimental analysis, presented in Section 6. The first eight instances of the list are real-life scenarios.

Name	Day /Night	Number of train units	$\|T_P\|$	Original disturbance	Infrastructure disturbance added	Number of delays added
D1-0	Day	14	243	None	None	0
D2-0	Day	17	241	T26 Closure	None	0
D3-0	Day	15	243	2 arrival delays	None	0
D4-0	Day	18	242	None	None	0
N1-0	Night	23	162	None	None	0
N2-0	Night	19	165	T26 Closure	None	0
N3-0	Night	26	161	2 arrival delays	None	0
N4-0	Night	24	158	None	None	0
D1-1	Day	14	243	None	None	2
D2-1	Day	17	241	T26 Closure	None	2
D3-1	Day	15	243	2 arrival delays	None	2
D4-1	Day	18	242	None	None	2
N1-1	Night	23	162	None	None	2
N2-1	Night	19	165	T26 Closure	None	2
N3-1	Night	26	161	2 arrival delays	None	2
N4-1	Night	24	158	None	None	2
D1-2	Day	14	243	None	None	4
D2-2	Day	17	241	T26 Closure	None	4
D3-2	Day	15	243	2 arrival delays	None	4
D4-2	Day	18	242	None	None	4
N1-2	Night	23	162	None	None	4
N2-2	Night	19	165	T26 Closure	None	4
N3-2	Night	26	161	2 arrival delays	None	4
N4-2	Night	24	158	None	None	4
D1-3	Day	14	243	None	Track 74 closure	0
D2-3	Day	17	241	T26 Closure	Track 74 closure	0
D3-3	Day	15	243	2 arrival delays	Track 74 closure	0
D4-3	Day	18	242	None	Track 74 closure	0
N1-3	Night	23	162	None	Track 74 closure	0
N2-3	Night	19	165	T26 Closure	Track 74 closure	0
N3-3	Night	26	161	2 arrival delays	Track 74 closure	0
N4-3	Night	24	158	None	Track 74 closure	0

References

Abbink, E., 2006. Intelligent Shunting: Dealing with Constraints (Satisfaction). John Wiley & Sons, Inc, pp. 391–413.

Borndörfer, R., Eßer, T., Frankenberger, P., Huck, A., Jobmann, C., Krostitz, B., Kuchenbecker, K., Moorhagen, K., Nagl, P., Peterson, M., Reuther, M., Schang, T., Schoch, M., Schülldorf, H., Schütz, P., Therolf, T., Waas, K., Weider, S., 2020. Deutsche bahn schedules train rotations using hypergraph optimization. Informs Journal on Applied Analytics in press.

Borndörfer, R., Reuther, M., Schlechte, T., Weider, S., 2011. A hypergraph model for railway vehicle rotation planning. In: OpenAccess Series in Informatics, Proceedings of ATMOS2011 - 11th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems, vol. 20, pp. 146–155.

Cardonha, C., Borndörfer, R., 2009. A set partitioning approach to shunting. Electron. Notes Discrete Math. 35 (Suppl. C), 359–364. LAGOS'09 – V Latin-American Algorithms, Graphs and Optimization Symposium.

Demange, M., Stefano, G.D., Leroy-Beaulieu, B., 2012. On the online track assignment problem. Discrete Appl. Math. 160 (7), 1072–1093.

Di Stefano, G.e., Koči, M.L., 2004. A graph theoretical approach to the shunting problem. Electron. Notes Theor. Comput. Sci. 92, 16–33.

Freling, R., Lentink, R.M., Kroon, L.G., Huisman, D., 2005. Shunting of passenger train units in a railway station. Transport. Sci. 39 (2), 261–272.

Gallo, G., Miele, F.D., 2001. Dispatching buses in parking depots. Transport. Sci. 35 (3), 322–330.

Gilg, B., Klug, T., Martienssen, R., Paat, J., Schlechte, T., Schulz, C., Seymen, S., Tesch, A., 2018. Conflict-free railway track assignment at depots. Journal of rail transport planning & management 8 (1), 16–28.

Haahr, J.T., Lusby, R.M., Wagenaar, J.C., 2017. Optimization methods for the train unit shunting problem. Eur. J. Oper. Res. 262 (3), 981–995.

Haijema, R., Duin, C., Van Dijk, N., 2006. Train shunting: a practical heuristic inspired by dynamic programming. Planning in Intelligent Systems: Aspects, Motivations, and Methods 437–475.

Hoogervorst, R., Dollevoet, T., Maróti, G., Huisman, D., 2020. Reducing passenger delays by rolling stock rescheduling. Transport. Sci. 54 (3), 565–853.

Jacobsen, P.M., Pisinger, D., 2011. Train shunting at a workshop area. Flex. Serv. Manuf. J. 23 (2), 156–180.

Kamenga, F., Pellegrini, P., Rodriguez, J., Merabet, B., Houzel, B., 2019. Train unit shunting: integrating rolling stock maintenance and capacity management in passenger railway stations. In: 8th International Conference on Railway Operations Modelling and Analysis (RailNorrköping *2019*).

Kroon, L.G., Lentink, R.M., Schrijver, A., 2008. Shunting of passenger train units: an integrated approach. Transport. Sci. 42 (4), 436–449.

Lentink, R.M., Fioole, P.-J., Kroon, L.G., van't Woudt, C., 2006. Applying operations research techniques to planning of train shunting. Planning in Intelligent Systems: Aspects, Motivations, and Methods 415–436.

Li, H., Jin, M., He, S., Ye, Z., Song, J., 2017. Optimal track utilization in electric multiple unit maintenance depots. Comput. Ind. Eng. 108 (Suppl. C), 81–87.

Marinov, M., Sahin, I., Ricci, S., Vasic, G., 2013. Railway operations, time-tabling and control. Res. Transport. Econ. 41, 59–75.

Maróti, G., 2006. Operations Research Models for Railway Rolling Stock Planning. PhD thesis. TU Eindhoven.

Pellegrini, P., Marlière, G., Pesenti, R., Rodriguez, J., 2015. RECIFE-MILP: an effective milp-based heuristic for the real-time railway traffic management problem. IEEE Trans. Intell. Transport. Syst. 16 (5), 2609–2619.

Qi, X., Yue, Y., Han, J., Zhou, L., Rakha, H.A., 2017. Integrated modelling and optimization of train scheduling and shunting at complex railway passenger stations. In: Transportation Research Board 96th Annual Meeting.
Riezebos, J., Van Wezel, W., 2009. k-shortest routing of trains on shunting yards. Spectrum 31 (4), 745.
Schöbel, A., 2017. An eigenmodel for iterative line planning, timetabling and vehicle scheduling in public transportation. Transport. Res. C Emerg. Technol. 74, 348–365.
Tomii, N., Zhou, L.J., 2000. Depot shunting scheduling using combined genetic algorithm and pert. Computers in Railways 7, 437–446.
Van Den Broek, J., Kroon, L., 2007. A capacity test for shunting movements. In: Algorithmic Methods for Railway Optimization. Springer, pp. 108–125.
Winter, T., Zimmermann, U.T., 2000. Real-time dispatch of trams in storage yards. Ann. Oper. Res. 96 (1), 287–315.