Railway timetable optimization considering robustness and overtakings

Yan, Fei; Goverde, Rob M.P.

DOI
10.1109/MTITS.2017.8005683

Publication date
2017

Document Version
Accepted author manuscript

Published in
5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017 - Proceedings

Citation (APA)
Yan, F., & Goverde, R. M. P. (2017). Railway timetable optimization considering robustness and overtakings. In *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017 - Proceedings* (pp. 291-296). Article 8005683 IEEE. https://doi.org/10.1109/MTITS.2017.8005683

Railway Timetable Optimization Considering Robustness and Overtakings

Fei Yan
Department of Transport and Planning
Delft University of Technology, Delft, The Netherlands
Email: f.yan@tudelft.nl

Rob M.P. Goverde
Department of Transport and Planning
Delft University of Technology, Delft, The Netherlands
Email: r.m.p.goverde@tudelft.nl

***Abstract*—This paper focuses on optimizing the robustness of a timetable with multiple train lines of different frequencies, where overtakings are also taken into account. An optimization model is considered of a cyclic railway timetable problem where dwell times and running times are variable and overtaking is allowed for relevant stations and each line. Based on the Periodic Event Scheduling Problem, train journey time, robustness and the number of dwell time stretches (which decides whether a train can have overtakings) are proposed as objectives, with corresponding constraints included in the model. This approach is studied in a small network with six stations and proved to be efficient. Six model variants from a different combination of objectives and constraints are compared on robustness, for which a number of robustness indicators are defined.**

I. Introduction

The railway timetable optimization model aims at finding the scheduled departure and arrival times at each station within a given line structure. This paper considers an optimization model of a cyclic railway timetable problem where dwell times and running times are variable and overtaking is allowed for relevant stations and each line. A regular train service to passengers at each period is the main idea of a cyclic timetable. Multiple frequencies might also occur within one period for some train lines. For example, hourly frequencies are one, two, or four in the railway network of the Netherlands. As passenger demand is various and unbalanced, different frequencies that are not multiples of each other are more applicable in some networks, like in China. Regularity constraints corresponding to flexible frequencies need to be designed to keep all trains follow the same path. Due to increasing attention on punctuality and the huge impact of delay propagation, traditional objectives of passenger travel time and train journey time (TJT) are not sufficient to optimize timetables, whereas a robust timetable could have a significant effect on mitigating secondary delays or knock-on delays. Meanwhile, overtakings could impact timetable robustness when train journey times differ on the same corridor. Therefore, robust timetable optimization at the planning level with consideration of overtakings are necessary to be studied.

The Periodic Event Scheduling Problem (PESP) [1] has been successfully applied to solve the macroscopic scheduling of a cyclic timetable in [2]–[4]. This model is based on a periodic event-activity network, where each node represents one event which is an arrival or departure for a train in a certain station along its path. The original PESP is a feasibility problem since periodic schedules only need to satisfy the constraints. Several objectives from passenger and operator aspects were added in [3], and solved by setting up a Mixed Integer Linear Programming (MILP). Illegal overtakings when two trains occupy the same open track section at the same time may occur when variable trip times (constrained by a lower and upper bound) are set in the model. To address this issue, extra dummy nodes are adopted to forbid the conflict in [5] and [6], while a relation of modulo parameters is presented to find a conflict-free timetable in [7]. Both methods were tested in our model, and the latter one is applied whereas it proved to be more efficient.

A lot of achievements in robust timetable optimization have been obtained based on PESP model in recent years. A robustness objective function is proposed in [3] by pulling apart trains, that is, to push the headway to half the cycle time. Both [8] and [9] discussed to improve the robustness by allocating buffer times between two successive trains and time supplements along one train path. While the former proposed a stochastic programming approach, the latter combined stochastic programming and robust optimization to deal with uncertain data, and introduced the concept of *recoverable robustness*. The authors of [10] proposed a three level framework of integrated timetable construction with consideration of feasibility, stability, robustness, efficiency and energy consumption. They combined microscopic and macroscopic models of timetable design, as detailed in [11]. An overview of nominal and robust timetable optimization for both cyclic and non-cyclic patterns is summarized in [12]. An extended PESP model in [6] investigated the maximization of network stability by generating feasible timetables, with optimal train orders and overtakings.

Most robust optimization models mentioned above need an initial feasible timetable, where train orders have been already fixed, which leaves less space for improvements compared with flexible orders. It is proved that a timetable can achieve maximum reliability (robustness) from [13] while equalizing scheduled headways for all trains with the same train type. Our paper aims at computing a robust timetable while determining train orders by spreading the headways as equal as possible. Regularity constraints of lines for different frequencies are proposed to provide regular train service for passengers. Without loss of generality, the number of overtakings might increase while enhancing overall robustness of the timetable, especially in a dense corridor. This trade-off between robustness and overtakings is also discussed in this paper. Our model is a modified PESP with objectives of train journey time, robustness, and the number of dwell time stretches. With respect to verifying the feasibility of our model, we defined several PESP models with different objectives and constraints to test and compare. Moreover, a number of statistic indicators are introduced to assess robustness.

The paper is organized as follows: Section II introduces the problem formulation and the corresponding cyclic railway timetable model variants. Section III illustrates the approach in case studies, and finally Section IV ends the paper with conclusions.

II. Model Description

A. Model Statement

The PESP formulation can be represented by a direct graph $G = (E, A)$, which represents a periodic event-activity network. With a given line plan, the model is associated with a set of train lines L. Each line $l \in L$, defines a stopping pattern $(s_1, ..s_k.., s_N)$ and a frequency f_l within a given time period T. The set E contains departure events at s_1, arrival and departure events (for stops) or through events at s_k, and arrival events at s_N for all frequencies of all lines. The set of activities A which link these events, represents the constraints on process time between a pair of events. For each event i, we determine the scheduled time $\pi_i \in [0, T)$ in a basic period while satisfying the set of constraints A. Due to periodicity, this event would occur at times $\pi_i + p \cdot T$, where $p = 1, 2, ...$ Each process time a_{ij} corresponds to an activity $(i, j) \in A$, where i and j are two consecutive events, which can be distinguished as running time, dwell time, headway time, and regularity interval, and each of them has a lower l_{ij} and upper bound u_{ij} modulo T.

Running activities A_{run} and dwell activities A_{dwell} are generated from the consecutive events of the same train. The lower bound for running time is the minimum running time, which equals the technical running time plus a proper time supplement that covers various train behaviors. The upper bound is the maximum running time that can be accepted by passengers. The minimum time for boarding and alighting of passengers, and the maximum time for passengers waiting at stations represent the lower bound and upper bound of the dwell process. Headway activities A_{infra} are generated between different train events at the same station. The minimal safety interval is the lower bound l_{ij} for headway time, whilst $T - l_{ji}$ is the upper bound to ensure the safety between trains in the reverse order. If the frequency f_l of line l is greater than one, regularity activities A_{reg} are needed to ensure a regular service. Since all trains in a line have the same stop pattern, we predefine a departure sequence of these trains which does not influence the results. $A^l_{\text{reg}}(s_k)$ represents the regularity activities between trains of line l in station s_k. The lower and upper bounds are set to be T/f_l to line l when strict regularity is needed. This ensures regular scheduling of line l. Transfer connections and rolling stock connections are not considered in this paper. All activities are represented by $A = A_{\text{run}} \cup A_{\text{dwell}} \cup A_{\text{infra}} \cup A_{\text{reg}}$.

PESP aims at finding the event times of all π_i, $i \in E$, where all processes

$$a_{ij} = \pi_j - \pi_i + p_{ij} \cdot T \tag{1}$$

satisfy the lower bound l_{ij} and upper bound u_{ij}. The modulo parameter p_{ij} determines the order of event i and j within a defined period T. Here we assume $u_{ij} - l_{ij} \in [0, T]$ and $l_{ij} \in [0, T]$. Then $p_{ij} = 1$ if $\pi_i > \pi_j$, and it is zero otherwise.

B. Model formulation

With the description above, the mathematical timetabling model is designed as follows. First, we introduce a periodic timetable optimization model with objective of train journey time, defined as (*PESP-TJT*):

$$\text{Minimize} \sum_{(i,j) \in A_{\text{run}} \cup A_{\text{dwell}}} \alpha_{ij} \cdot (\pi_j - \pi_i + p_{ij} \cdot T) \tag{2}$$

Subject to

$$l_{ij} \le \pi_j - \pi_i + p_{ij} \cdot T \le u_{ij} \qquad \forall (i,j) \in A \tag{3}$$

$$0 \le \pi_i < T \qquad \forall i \in E \tag{4}$$

$$p_{ij} \in \{0, 1\} \qquad \forall (i,j) \in A \tag{5}$$

$$p_{ij} + p_{i'j'} + p_{ii'} + p_{jj'} = 2 * c_{ii'jj'} \tag{6}$$

$$0 \le c_{ii'jj'} \le 2 \tag{7}$$

$$c_{ii'jj'} \in \mathbb{N} \tag{8}$$

where (6)-(8) hold for all

$$(i, j), (i', j') \in A_{\text{run}}, (i, i'), (j, j') \in A_{\text{infra}}$$

Objective function (2) includes train running times and dwell times, and α_{ij} represents the weight of different processes. We assume it equals one in order to compare travel time, dwell time and time supplement with different models. Constraint (3) ensures that all process times are within the given bounds. Constraint (4) requires periodicity of events by bounding to $[0, T)$. Constraints (6) -(8) guarantee that no illegal overtaking can arise, when the sum of four modulo parameters of related running and infra processes equals zero, two or four, see for details [7].

A strict cyclic pattern means that trains from the same line have the same scheduled event time modulo its own cycle time, described by

$$a_{ij} = T/f_l \qquad \forall (i,j) \in A^l_{\text{reg}} \tag{9}$$

Flexible frequencies are proposed in this paper, so f_l could be any number given from line plan. A parameter θ is introduced to provide a certain deviation in case T/f_l is not integer or to express the tolerance from strict regularity. We assume that all trains from the same line have the same trajectory (train path), which means the same running time between two successive stations and dwell time at each station. So we reformulated the regularity constraints above, with a_{ij} short for $\pi_j - \pi_i + p_{ij} \cdot T$ from (1). *PESP-Reg* is used to represent the new regularity constraints:

$$T/f_l - \theta \leq a_{ij} \leq T/f_l + \theta, \forall (i,j) \in A^l_{\text{reg}}(s_1), l \in L \quad (10)$$

$$a_{mn} = a_{ij},$$
$$\forall (m,n) \in A^l_{\text{reg}}(s_k), (i,j) \in A^l_{\text{reg}}(s_1), k \in [2,N], l \in L \quad (11)$$

In order to improve robustness, a new variable δ_{ij} and parameter ϕ_{ij} are introduced, defined as

$$\phi_{ij} = (u_{ij} + l_{ij})/2 \quad (12)$$

$$\delta_{ij} = |a_{ij} - \phi_{ij}| \quad (13)$$

Parameter ϕ_{ij} is the middle of $[l_{ij}, u_{ij}]$. For a certain process $(i,j) \in A_{\text{infra}}$, if $a_{ij} = \phi_{ij}$, it could provide the best robust solution for event i and j since the two events are distributed as far as possible. We define the sum of δ_{ij} as the headway deviation from half the cycle time (HDHC). Hence, minimization of HDHC to spread train events is proposed:

$$\text{Minimize} \sum_{(i,j) \in A_{\text{infra}}} \delta_{ij} \quad (14)$$

$$-\delta_{ij} \leq a_{ij} - \phi_{ij} \leq \delta_{ij} \qquad \forall (i,j) \in A_{\text{infra}} \quad (15)$$

This constraint is the linearized version of (13) since $a_{ij} - \phi_{ij}$ can be positive or negative. The robust PESP model (*PESP-Rob*) is composed of the foregoing objective function and constraint, and all constraints elaborated in *PESP-TJT*.

The number of overtakings might increase when robustness is appealing to. In order to analyze the trade-off between robustness and overtakings, the number of overtakings are considered to be optimized. We assume the minimal dwell times of all stations are smaller than twice the minimal headway. An extra binary variable y_{ij} is proposed to the dwell time supplement d_{ij}. The upper bound of the dwell activity then becomes

$$u_{ij} = l_{ij} + d_{ij} \cdot y_{ij}$$

Here, d_{ij} is predefined by a maximum passenger waiting time at stations. When y_{ij} equals one, it can be interpreted as a stretch of dwell time, and overtakings could occur at this stop. Otherwise, the dwell time equals the minimal dwell time, and overtaking can not happen as the time is not enough due to infrastructure constraints. Therefore, the minimization of the number of dwell time stretches aslo reduces overtakings. The model can be expressed as:

$$\text{Minimize} \sum_{(i,j) \in A_{\text{dwell}}} y_{ij} \quad (16)$$

$$l_{ij} \leq a_{ij} \leq l_{ij} + d_{ij} \cdot y_{ij} \qquad \forall (i,j) \in A_{\text{dwell}} \quad (17)$$

We define *PESP-Ovt* with these objective and constraints, and the constraints in *PESP-Reg* and *PESP-TT* with (3) for $A \setminus A_{\text{dwell}}$.

When all three objectives (2), (14) and (16) need to be satisfied, traditionally appropriate weights are assigned to deal with the multi-objective problem. Our multi-objective model is formulated as

$$\text{Minimize} \quad \sum_{i=1}^{3} \omega_i \cdot Z_i \quad (18)$$

With the aim of testing the validity of this model, comparisons between some sub-models are needed. Each weight could also stand for whether the objective is considered or not. Zero means it is not considered, and otherwise it is selected. In addition, partitioning the activities of objective TJT and HDHC leads to the same order of magnitude, which makes it easier to assign weights. Each objective is defined as:

$$Z_1 = \sum_{(i,j) \in A_{\text{run}} \cup A_{\text{dwell}}} a_{ij}/(N_{\text{run}} + N_{\text{dwell}}) \quad (19)$$

$$Z_2 = \sum_{(i,j) \in A_{\text{infra}}} \delta_{ij}/N_{\text{infra}} \quad (20)$$

$$Z_3 = \sum_{(i,j) \in A_{\text{dwell}}} y_{ij} \quad (21)$$

Subject to the corresponding constraints with respect to the selected objective(s). N_{run}, N_{dwell} and N_{infra} stand for the number of run activities, dwell activities, and infra activities respectively.

III. Experiments

A. Instances and results

The optimization models are tested using Matlab R2016b, Gurobi version 7.0.2, and the Yalmip toolbox [14]. This section shows case studies of six different optimization models based on the objectives and constraints in Section II.

A small network with five stations is considered. Fig. 1 displays the given line plan, with seven trains in total and the same speed limit. And certain amount of time supplement is allowed in optimization process. Table I depicts the values of the parameters applied in our model. We assume all stations are allowed to have overtakings, and the maximal dwell time is set to be $2h$ (6 min) for all stations which allows only one overtaking, in order to have less dwell time loss. Weight ω_i is set as binary, as objective value of *PESP-Ovt* is also in the same order of magnitude as the other two objectives in our case. The components of each sub-model are depicted in Table II.

Fig. 2 illustrates the computed timetables in time-distance diagrams, and Table III shows the obtained values of objective

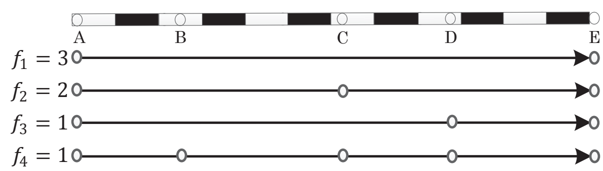

Fig. 1. Line plan

TABLE I
INPUT PARAMETERS OF TIMETABLING MODEL

Parameters	Values
Period cycle time T	60 min
Minimal dwell time	1 or 2 min
Minimal headway time h	3 min
Dwell time supplement d	4 or 5 min
Parameter in regularity constraint θ	1 min

TABLE II
OBJECTIVE AND CORRESPONDING CONSTRAINTS FOR EACH MODEL

Models	*PESP-TJT*	*PESP-Reg*	*PESP-Rob*	*PESP-Ovt*
(A)Min TT	✓			
(B)Min TT+Reg		✓		
(C)Min Rob			✓	
(D)Min TT+Rob	†		✓	
(E)Min TT+Reg+Rob	†	*	✓	
(F)Min TT+Reg+Rob+Ovt	†	*	✓	✓

Note: ✓represents both objective and constraints are considered. † represents only objective is considered, and * only constraints are considered.

function (Obj val), TJT, average TJT (Z_1), HDHC, average HDHC (Z_2), total dwell time (TD), mean dwell time (MD), the number of dwell time stretches (Z_3), the number of overtakings (NO), total time supplement (SupT), and optimization time (Opt time) for the different models. It can be straightly observed that *PESP-Rob* and *PESP-Reg* could help to improve robustness by comparing figures (a) and (b), and (a) and (d) in Fig. 2, and regularity constraints also spread trains even though for the same line.

Both model (C) and (D) achieved the minimum HDHC, but the latter also obtained the minimum TJT of 159 as in model (A). Only minimizing HDHC might gain more robustness but could result in a big increase of TJT, with a maximal running time supplement of 27.3%. Meanwhile, dwell time stretch also occurs even without an overtaking in Model (C) which leads to longer dwell time, see the third departing train in station D with dwell time larger than the minimal dwell time in (c) of Fig. 2. This is because the dwell time stretch could help to spread the headway more equal. When it comes to robustness optimization, the computation time raised, especially for model (C) and (D) with consideration of *PESP-Rob*, see opt time in Table III. Adding regularity constraints reduces computation time comparing model (D) and (E), with only a slight growth of HDHC from 2160 to 2162. Furthermore, *PESP-Ovt* could contribute to cut down overtakings and dwell time stretches without a huge increase of TJT and HDHC. In summary, it is better to combine *PESP-TT* with *PESP-Rob* in order to control journey time consumption and acquire a robust timetable at the same time. Moreover, *PESP-Reg* should be added to the model to decrease optimization time as well as providing regular

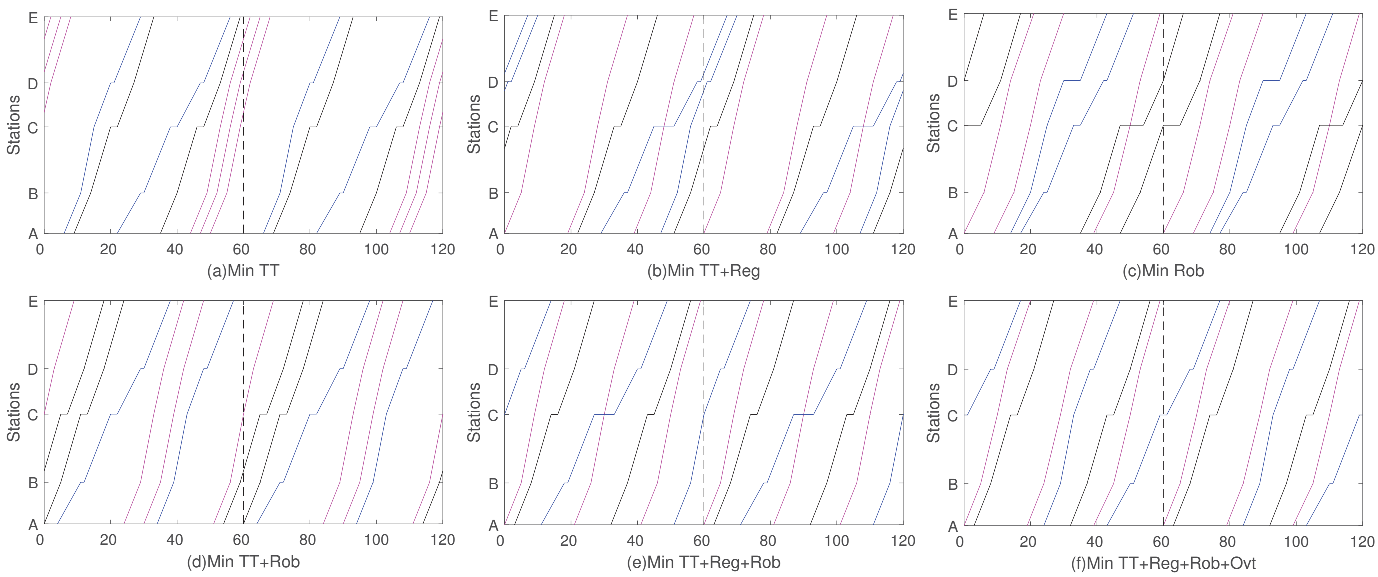

Fig. 2. Time-distance diagram of each obtained timetable for two periods

TABLE III
RESULTS OF DIFFERENT TIMETABLING MODELS

Modles	Obj val [-]	TJT [min]	Z_1 [min]	HDHC [min]	Z_2 [min]	TD [min]	MD [min]	Z_3 [-]	NO [-]	SupT [min]	Opt time [sec]
(A)Min TT	18.82	159	4.68	2376	14.14	9	1.50	0	0	6.5	0.31
(B)Min TT+Reg	18.40	163	4.79	2286	13.61	13	2.17	1	1	10.5	0.41
(C)Min Rob	18.33	186	5.47	2160	12.86	21	3.50	3	1	33.5	3212.85
(D)Min TT+ Rob	17.53	159	4.68	2160	12.86	9	1.50	0	0	6.5	2213.67
(E)Min TT+Reg+Rob	17.66	163	4.79	2162	12.87	13	2.17	1	1	10.5	2.03
(F)Min TT+Reg+Rob+Ovt	17.86	165	4.85	2186	13.01	9	1.50	0	0	12.5	1.30

service, and *PESP-Ovt* could be applied when less overtakings are preferred.

B. Robustness measures

Headway deviation of half the cycle time (HDHC) can only give a general idea of robustness, since it is the sum of all pairs of headway at each station. In practice, robustness is quantified between successive events on the same station. The headway between these successive events is defined as H_{ij}, $(i,j) \in A^{*}_{\text{infra}} \in A_{\text{infra}}$, and can be calculated from the obtained event times. In addition, even when the train order is fixed, it is still hard to assess robustness by one indicator. Therefore, this section introduces several measures to evaluate robustness with headway processes from A^{*}_{infra}. Generally, range, standard deviation (SD), and mean absolute deviation (MAD) of H_{ij} are used to assess robustness. When the number of activities N_{H} in A^{*}_{infra} is known, mean, SD and MAD of headway can be calculated as follows.

$$\bar{H} = T/\sum f_l \tag{22}$$

$$\text{H}_{\text{sd}} = \sqrt{\sum (H_{ij} - \bar{H})^2} \tag{23}$$

$$\text{H}_{\text{mad}} = (\sum |H_{ij} - \bar{H}|)/N_{\text{H}} \tag{24}$$

With the value of H_{sd} and H_{mad} obtained, it is still inapparent how much robustness is achieved for a certain model. In case all trains have the same type, [13] proved that the maximum possible value of SD and MAD can be computed by

$$\text{H}^{\max}_{\text{sd}} = T \cdot \sqrt{\sum f_l - 1}/\sum f_l \tag{25}$$

$$\text{H}^{\max}_{\text{mad}} = 2T \cdot (\sum f_l - 1)/(\sum f_l)^2 \tag{26}$$

In order to make them more comparative, the measures of robustness are converted to a 0-1 scale as follows.

$$\text{Rob}_{\text{sd}} = \text{H}_{\text{sd}}/\text{H}^{\max}_{\text{sd}} \tag{27}$$

$$\text{Rob}_{\text{mad}} = \text{H}_{\text{mad}}/\text{H}^{\max}_{\text{mad}} \tag{28}$$

For the given line plan, $\bar{H}$, N_{H}, $\text{H}^{\max}_{\text{sd}}$ and $\text{H}^{\max}_{\text{mad}}$ are around 8.6 min, 56, 21.0 min and 14.7 min, respectively. Table IV describes all values of the measures explained above, as well as additional measures. The sum of negative headway deviation (NHD) is defined as the headways H_{ij} lower than the mean headway $\bar{H}$ (LMH). The number of LMH (N_{LMH}) is expressed to compute the ratio of N_{LMH} to all activities N_{H}. This ratio indicates the total deviation scale of headways.

$$R_{\text{LMH}} = N_{\text{LMH}}/N_{\text{H}} \tag{29}$$

TABLE IV
ROBUSTNESS MEASUREMENTS

Models	Rob_{sd} [-]	Rob_{mad} [-]	SD [min]	MAD [min]	NHD [min]	N_{LMH} [-]	R_{LMH} [-]	MinH [min]	MaxH [min]	S_{R} [-]	MedH [min]	ModeH [min]	R_{ModeH} [-]	R_{MinH} [-]
(A)Min TT	0.30	0.38	6.38	5.52	-154.43	34	0.61	3	23	0.33	6	3	0.41	0.41
(B)Min TT+Reg	0.24	0.26	5.06	3.78	-105.86	33	0.59	3	19	0.27	8	3	0.20	0.20
(C)Min Rob	0.18	0.20	3.76	3.00	-84.00	28	0.50	3	18	0.25	8.5	9	0.14	0.07
(D)Min TT+ Rob	0.22	0.26	4.65	3.84	-107.43	34	0.61	3	20	0.28	6	6	0.34	0.05
(E)Min TT+Reg+Rob	0.15	0.17	3.16	2.50	-70.00	21	0.38	3	15	0.20	9	9	0.21	0.14
(F)Min TT+Reg+Rob+Ovt	0.21	0.25	4.37	3.64	-101.86	33	0.59	3	18	0.25	8	8	0.14	0.13

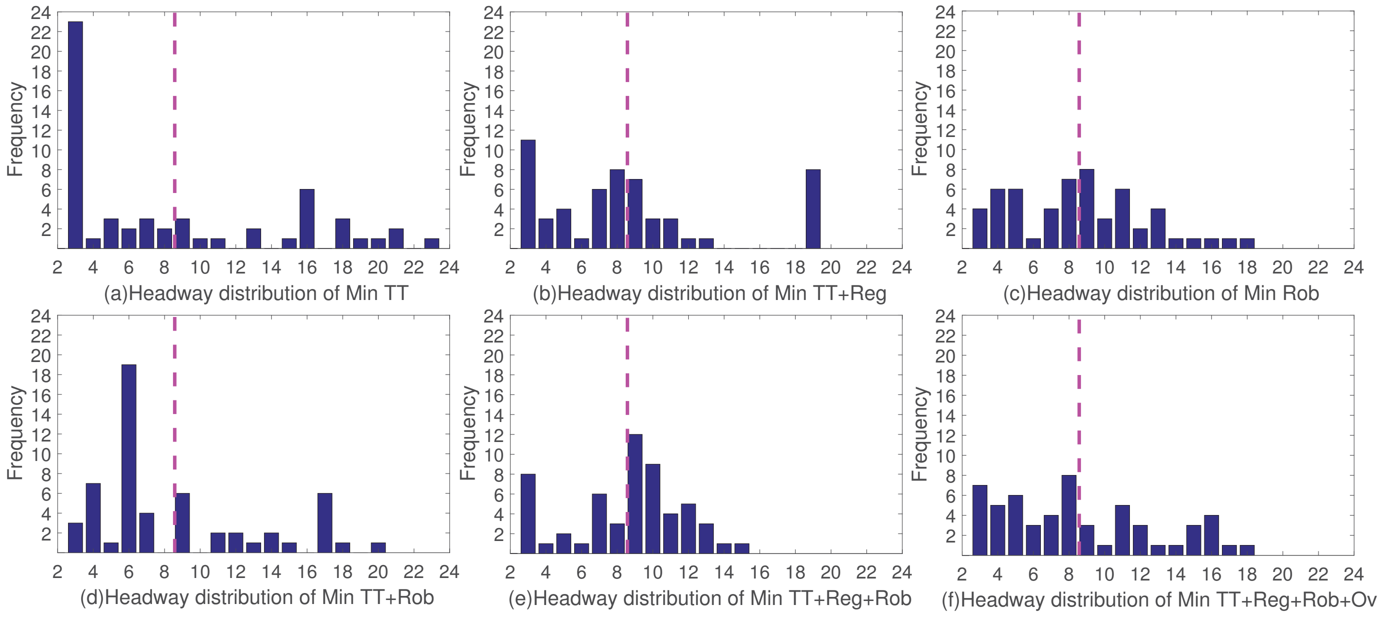

Fig. 3. Headway distributions of timetable for different models

The minimal headway (MinH) and maximal headway (MaxH) are applied to identify the range of headways, and the scale of range (S_{R}) could make this measure more readable. The maximum range can be considered as T for all timetables, and the scaled range is then calculated by

$$S_{\mathrm{R}} = (\mathrm{MaxH} - \mathrm{MinH})/T \tag{30}$$

The median headway (MedH) and mode headway (ModeH) are collected to show an overview of headways distribution within the range, and the corresponding ratio for the number of mode headway (R_{ModeH}) is also derived. The ratio for the number of minimal headway (R_{MinH}) can show the vulnerability of the timetable. A timetable with more numbers of minimal headway is more sensitive to even small delays.

As can be seen from Table IV, model (A) has the worst robustness with a maximum or minimum of all measures. Model (E) can be recognized as the best robustness according to the lowest $\mathrm{Rob}_{\mathrm{sd}}$, $\mathrm{Rob}_{\mathrm{mad}}$, R_{LMH} and S_{R}. With comparison of R_{MinH} between timetable (A) with (D), and (B) with (E), it can be found that *PESP-Rob* could lead to a lower number of minimal headways. This is also verified by model (C) with only 7.14% of R_{MinH} from Table IV. *PESP-Reg* tries to enlarge robustness by all indicators except R_{ModeH} and R_{MinH}, with (A) in contrast with (B), and (D) with (E). The minimization of dwell time stretches leads to a less robustness timetable from almost all aspects of our measures, except for a slight decrease of R_{MinH}. This means that overtakings could help to improve robustness to some extent, as it could lead to less capacity consumption for a pair of trains, and leave more time for other trains.

Fig. 3 shows the headway distribution of each model in histograms with the dashed line representing the position of $\bar{H}$. The radar diagram is designed to analyze overall robustness in Fig. 4, including $\mathrm{Rob}_{\mathrm{sd}}$, $\mathrm{Rob}_{\mathrm{mad}}$, R_{LMH}, S_{R} and R_{MinH} from Table IV. All these values are between 0 to 1, with the smaller the better robustness. Therefore, the enclosed area by lines with the same color is proposed to stand for the overall robustness of each timetable. It can be concluded that timetable (E) is more insensitive to delays compared with others by overall robustness.

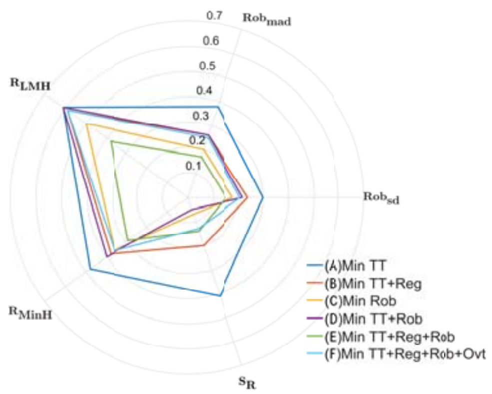

Fig. 4. Robustness measures

IV. Conclusions

This paper presented several extended PESP models with objectives of minimal train journey time, headway deviation of half the cycle time, and the number of dwell time stretches. Regularity constraints for different frequencies of train lines were developed. Robustness objectives and constraints were introduced to obtain a robust timetable. Moreover, a variable dwell time supplement was proposed to reduce dwell time stretches as well as overtakings. By comparison of timetables from six models, it can be summarized that regularity constraints and robustness constraints are both useful to improve robustness, especially, *PESP-Rob* reduces the number of minimal headways. In addition, the benefit of *PESP-Reg* is proved to have less computation time, and *PESP-Ovt* could attribute to have less overtakings but with some robustness loss. The future work is to integrate this timetabling model with line planning, and improve robustness at the line planning level, and multiple overtakings would also be taken into account.

Acknowledgment

This project is partially supported by China Scholarship Council (No.201309110101).

References

[1] P. Serafini and W. Ukovich, "A mathematical model for periodic scheduling problems," *SIAM Journal on Discrete Mathematics*, vol. 2, no. 4, pp. 550–581, 1989.

[2] K. Nachtigall, "Periodic network optimization with different arc frequencies," *Discrete Applied Mathematics*, vol. 69, no. 1, pp. 1–17, 1996.

[3] L. Peeters, *Cyclic railway timetable optimization*. Erasmus University Rotterdam: PhD thesis, 2003.

[4] C. Liebchen, "The first optimized railway timetable in practice," *Transportation Science*, vol. 42, no. 4, pp. 420–435, 2008.

[5] L. G. Kroon and L. W. Peeters, "A variable trip time model for cyclic railway timetabling," *Transportation Science*, vol. 37, no. 2, pp. 198–212, 2003.

[6] D. Sparing and R. M. P. Goverde, "A cycle time optimization model for generating stable periodic railway timetables," *Transportation Research Part B: Methodological*, vol. 98, pp. 198–223, 2017.

[7] X. Zhang and L. Nie, "Integrating capacity analysis with high-speed railway timetabling: A minimum cycle time calculation model with flexible overtaking constraints and intelligent enumeration," *Transportation Research Part C: Emerging Technologies*, vol. 68, pp. 509–531, 2016.

[8] L. Kroon, G. Maróti, M. R. Helmrich, M. Vromans, and R. Dekker, "Stochastic improvement of cyclic railway timetables," *Transportation Research Part B: Methodological*, vol. 42, no. 6, pp. 553–570, 2008.

[9] C. Liebchen, M. Lübbecke, R. Möhring, and S. Stiller, "The concept of recoverable robustness, linear programming recovery, and railway applications," in *Robust and online large-scale optimization*. Springer, 2009, pp. 1–27.

[10] R. M. P. Goverde, N. Bešinović, A. Binder, V. Cacchiani, E. Quaglietta, R. Roberti, and P. Toth, "A three-level framework for performance-based railway timetabling," *Transportation Research Part C: Emerging Technologies*, vol. 67, pp. 62–83, 2016.

[11] N. Bešinović, R. M. P. Goverde, E. Quaglietta, and R. Roberti, "An integrated micro–macro approach to robust railway timetabling," *Transportation Research Part B: Methodological*, vol. 87, pp. 14–32, 2016.

[12] V. Cacchiani and P. Toth, "Nominal and robust train timetabling problems," *European Journal of Operational Research*, vol. 219, no. 3, pp. 727–737, 2012.

[13] M. Carey, "Ex ante heuristic measures of schedule reliability," *Transportation Research Part B: Methodological*, vol. 33, no. 7, pp. 473–494, 1999.

[14] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in matlab," in *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

2017 Models and Technologies for Intelligent Transportation Systems (IEEE MT-ITS 2017)
26-28 June 2017. Naples, Italy

Ant Colony Optimization for train routing selection: operational vs tactical application

Marcella Samà, Andrea D'Ariano, Dario Pacciarelli
Università degli Studi Roma Tre
Dipartimento di Ingegneria,
Sezione di Informatica e Automazione,
via della vasca navale 79, 00146 Rome, Italy
Email: {sama,dariano,pacciarelli}@ing.uniroma3.it

Paola Pellegrini, Joaquin Rodriguez
Université Lille Nord de France,
IFSTTAR, COSYS, LEOST and ESTAS,
Rue Élisée Reclus 20,
59666 Villeneuve d'Ascq, Lille, France
Email: {paola.pellegrini,joaquin.rodriguez}@ifsttar.fr

Abstract—**Railway traffic is often perturbed by unexpected events. To effectively cope with these events, the real-time railway traffic management problem (rtRTMP) seeks for train routing and scheduling methods which minimize delay propagation. The size of rtRTMP instances is strongly affected by the number of routing alternatives available to each train. Performing an initial selection on which routings to use during the solution process is a common practice to simplify the problem. The train routing selection problem (TRSP) reduces the number of routings available for each train to be used in the rtRTMP. This paper describes an Ant Colony Optimization (ACO) algorithm for the TRSP, and analyses its application in two different contexts: at tactical level, based on historical data and with abundant computation time, or at operational level, based on the specific traffic state and with a limited computation time. Promising results are obtained on the instances of the Lille terminal station area, in France, based on realistic traffic disturbance scenarios.**

Index Terms—**Railways; Train Scheduling and Routing; Graph Theory; Ant Colony Optimization.**

I. Introduction

In Europe, railway traffic demand has been steadily increasing in the last decades. Railway infrastructure managers need to face this ever increasing demand ensuring a good quality of service. This, added to the limited space and funds available to build new infrastructure in bottleneck areas, has stimulated in recent years the development of efficient ways for improving the reliability of the traffic. This reliability consists in the capability of running trains precisely following a predefined schedule. Typically, a hierarchical decision-making approach is adopted when planning railway operations, resulting in a series of tractable problems for which artificial intelligence and operations research approaches have been proposed. These problems can be grouped in three levels: strategic, tactical and operational [13].

Train timetables which are in principle feasible may become impossible to respect due to unexpected events disturbing traffic. These unexpected events may find their origin within the railway system (e.g., the delayed departure of a train due to an unexpectedly high number of passengers getting on it at a station) or be exogenous (e.g., a strong wind imposing speed limitation on a portion of the infrastructure). The problems dealing with the creation of new feasible working timetables in real-time to cope with disturbances belong to the operational level [10]. There is no general agreement on the most suitable objective function for operational level problems. However, the minimization of delay propagation is definitely one of the most often used. It takes different forms in the existing literature, as the minimization of the maximum [4] [6][18] or of the total [14][15][17] delay indirectly caused by an unexpected event, or the minimization of the deviation from the timetable [2]. Other used objectives are based on the level of service provided to the passengers [3][8][11][21].

The real-time Railway Traffic Management Problem (rtRTMP) is one of the most important operational level problems. The rtRTMP consists in defining a feasible working timetable in case of disturbances. In particular, it seeks for a suitable train routing and scheduling solution (passing orders and timings). This is necessary in case of detection of conflicting track requests (or simply conflicts) done by multiple trains during disturbed operations. These are time-overlapping requests, due to the fact that at least one of the involved trains is not following its original schedule. In turn, this may be due to an unexpected event either directly (e.g., as previously mentioned, the train departed late from a previous station) or indirectly (e.g., the train had to slow down to let the directly impacted train pass first and is now late). The former is called primary delay, and no action is possible to reduce it. The latter is called secondary delay, and it depends on the rtRTMP solution implemented (e.g., the secondary delays are different depending on the train which has the authorization to pass first on common track portions).

The rtRTMP is NP-Hard. The size of an rtRTMP instance and, in general, the time required to solve it are strongly affected by the number of routing alternatives available to each train [20]. In the literature, different solution approaches limit the number of routing variables, to simplify the solution process. The common practice is to select routings by following infrastructure managers directives. However, these directives are often based on the current (manual) practice, and are then most likely far from being the optimal choices. Moreover, they sometimes include a still very large number of routings.

In [20], we proposed a formalization of the Train Routing Selection Problem (TRSP). It consists in the selection of the suitable subsets of train routings, where the impact of scheduling decisions is only heuristically evaluated. The TRSP is to be solved as a preliminary step of the rtRTMP solution process: the routings selected in the TRSP are the only ones considered in the rtRTMP. Previous attempts to solve the TRSP were based on a-priori [1] or random decisions [16]. Instead, in [20] we proposed an Ant Colony Optimization (ACO) [9] algorithm for this problem. In a thorough experimental analysis, it proved to be very effective in allowing an improvement to the rtRTMP final solution through the appropriate selection of the train routing subsets. In the original paper, we applied the ACO algorithm at the operational level: When conflicts are detected, the ACO algorithm solves the TRSP in real-time based on the specific traffic perturbation to be handled, and the rtRTMP is solved afterwards with the returned routing subsets. This algorithm is named ACO-rtTRSP (real-time TRSP). The time typically considered available for defining a new feasible working timetable is between three and five minutes. In the proposed framework, then, this time has to be shared by ACO-rtTRSP and the rtRTMP solver. However, it is not necessarily required to solve the TRSP at operational level.

This paper investigates when is the best moment to solve the TRSP through our ACO algorithm. The first alternative is to apply ACO-rtTRSP at the operational level, as in [20]. In this case, the routing selection is based on the specific traffic perturbation to be faced, but a limited computation time is available. The second alternative is to apply the ACO algorithm at the tactical level, right after the production of the original timetable. In this case, no specific perturbation is to be tackled, and the routing selection is based on historically observed scenarios, which are likely to well represent future ones. The solution time available to the ACO algorithm can be much longer here than for the ACO-rtTRSP, allowing a possibly better exploration of the search space. The algorithm applied at the tactical level is named ACO-TRSP.

To compare the two possible alternatives, we evaluate the benefit that solving the TRSP at the operational and at the tactical level has on the rtRTMP. To do so, we perform a computational analysis on the French railway infrastructure of the Lille station area. The considered instances represent traffic perturbed by train delays at their entrance in the station area, and different instances with similar characteristics are used as historical data to solve the TRSP at the tactical level.

II. The real-time Railway Traffic Management Problem

As mentioned in the Introduction, the timetable developed at the tactical level includes three main families of information: routing assigned to each train, called *timetable routing*; passing orders of different trains on common resources (track portions); planned passing, arrival and departure times at stations or at other relevant points. However, disturbances may affect the train traffic flows, bringing to the emergence of conflicts and delays, and hence making the original timetable infeasible.

The rtRTMP is the problem of defining a new feasible working timetable by selecting a suitable (possibly optimal) set of routing and scheduling decisions. In this paper, the rtRTMP is modelled microscopically on a railway network. A railway *network* is the part of the infrastructure under study, usually representing a single train dispatching area. A network is formed by block-sections and track-circuits. A *track-circuit* represents the most microscopic part of the infrastructure. For safety reasons, each track-circuit must be used by at most one train at a time. A *block-section* is a sequence of track-circuits between two consecutive signals. Signals, interlocking and Automatic Train Protection (ATP) systems are used to impose safety regulations between trains and control the train traffic by setting up train routings and enforcing speed restrictions on running trains. A train *routing* is a sequence of block-sections that leads from an entry point to an exit point in the network, represented by station platforms or border block-sections. The *running time* of a train on a track-circuit represents the time required by a train to traverse it. A minimum separation is imposed by safety regulation between trains requiring the use of the same tracks (or resources), and can be translated into a minimum *headway time* between the starts of the running times of the two trains on the specific resources. In particular, the headway time is computed considering the route-lock sectional-release interlocking system [12], in which the use of a block-section locks only the block-sections sharing with it a not yet released track-circuit. This interlocking system is the most common in Europe nowadays. The objective function considered is the minimization of the total secondary delays. The rtRTMP is formulated as described in [15] and solved using the RECIFE-MILP algorithm [16].

III. The Train Routing Selection Problem

One of the decisions necessary to solve the rtRTMP is the selection of a routing for each train. This routing must be chosen in a set of available ones. This set contains the timetable routing and its possible alternatives. An *alternative routing* must have the same entry and exit point of the timetable one, as well as stop in the same stations. If an entry or exit point corresponds to a station platform, other platforms in the same station can be considered as alternative entry or exit points. Considering many available alternative routings translates, on the one hand, into a higher flexibility and hence possibly a better solution achievable. On the other hand, the computation time required to properly explore the larger search space and find such a better solution may be long.

The TRSP is the problem of defining for each train a subset of p routings chosen among its alternative ones. The so defined subsets are then to be used as input for the rtRTMP to limit the size of its search space.

We model the TRSP through a *construction graph* $G = (C, L)$ [20]. In the construction graph, each component $c_i \in C$ represents a single alternative routing for a train, and a component exists for each possible alternative routings of all trains. The non-oriented link $l_{ij} \in L$ connects the two coherent components c_i and c_j it refers to. Two components are *coherent* unless they are alternative routings of two different trains which use the same rolling stock, and, if they are such, unless they imply the need for additional local movements to perform the required turnaround, join or split operation. The construction graph is thus formed by n disjoint sets of components, one for each train, where n is the number of trains.

In the TRSP, the cost of a routing assignment to each train is quantified in terms of *potential delay*. This is an estimation of the delay propagation due to the assignment: the delay propagation may be due to the routing assignment in itself (e.g., if the alternative routing requires a longer travel time than the timetable one) and is represented by the cost u_i of selecting component c_i, or it may be due to a combination of routing assignments to different trains (for the emergence of conflicting requests) and is represented by the cost w_{ij} of selecting both components c_i and c_j and thus link l_{ij}. Formally, selecting a routing assignment for each train corresponds to building a clique of cardinality n in the construction graph G.

In Figure III we report three examples feasible train routing assignments for an instance including four trains (t_1, t_2, t_3, t_4), as well as a resulting TRSP solution, in the framed Figure 1(d). In the considered instance, t_1 can use four routings, t_2 and t_4 two and t_3 three. The construction graph G associated with the example has $\{c_1, c_2, c_3, c_4\} \in T_1$, $\{c_5, c_6\} \in T_2$, $\{c_7, c_8, c_9\} \in T_3$, $\{c_{10}, c_{11}\} \in T_4$. All components are coherent, thus $|L| = 44$. For each routing assignment in Figures 1(a), 1(b) and 1(c), the selected components of each feasible routing assignment are colored in black, while the other components are colored in grey. Only the links connecting these components are shown, for ease of visibility. The framed Figure 1(d) shows the resulting TRSP solution (s) for $p = 2$: the two best routing assignments (let them be ra_2 and ra_3) are considered and, for each train, the alternative routings there considered are included in s.

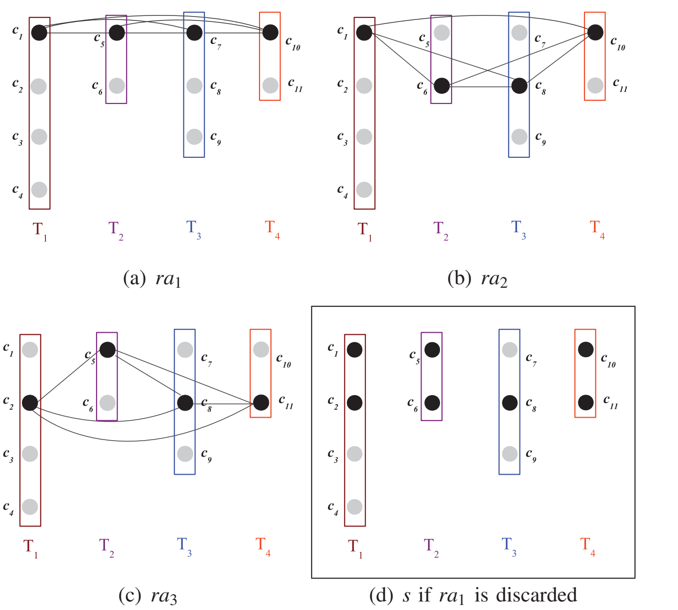

Fig. 1. Three feasible train routing assigments for four trains and one TRSP solution s for $p = 2$

IV. Ant Colony Optimization for the TRSP

The ACO algorithm described here was first proposed in [20]. The proposed algorithm is inspired by the one developed for the maximum clique problem in [22].

ACO is a meta-heuristic inspired by the foraging behaviour of ant colonies. The routing assignments are incrementally constructed by each of the *nAnts* ants of a colony. At each step of the construction process, an ant a selects a new component probabilistically using the random proportional rule, which is based on *pheromone trails* and *heuristic information* (i.e., the colony's shared knowledge and a greedy measure on the quality of a component which may be added to the current partial routing assignment). Once all the ants of the colony have built a routing assignment, the best one is stored and the pheromone trails are updated accordingly. This process is repeated iteratively until the time limit of computation.

Our ACO algorithm works on the construction graph G as follows. For each ant a of the *nAnts* ones composing a colony, the first component $c_i \in C$ is randomly selected and is inserted in the current partial routing assignment ra_a. The set of *candidates* among which the ant can choose the next component $c_j \in C$ includes all the components linked to c_i, i.e., all coherent components. The component c_j is chosen with a probability computed via the random proportional rule, that is based on the following pheromone trails and heuristic information raised to the power of α and β, respectively.

The value of the heuristic information associated to each $l_{ij} \in L$ is $\eta(l_{ij}) = \frac{1}{1+w_{ij}+u_j}$, stating the desirability of selecting c_j after c_i. After each addition of a component c_j in the partial routing assignment, the set of candidates is updated by removing both c_j and all components $c_h \in C : \nexists l_{jh} \in L$, i.e., all components not coherent with c_j. The routing assignment construction process terminates when the set of candidates is empty. A feasible routing assignment ra_a is accepted only if it includes n components, where n is the number of trains.

Among the feasible routing assignments found in the current