



**HAL**  
open science

# Sub-problems interaction analysis based three- level decomposition algorithm for real-time train scheduling and routing problems in railway stations

Lijie Bai, Thomas Bourdeaud 'Huy, Emmanuel Castelain, Qi Zhang

## ► To cite this version:

Lijie Bai, Thomas Bourdeaud 'Huy, Emmanuel Castelain, Qi Zhang. Sub-problems interaction analysis based three- level decomposition algorithm for real-time train scheduling and routing problems in railway stations. EITRT 2017 - Proceedings of Electrical and Information Technologies for Rail Transportation, Oct 2017, Changsha, China. hal-01720322

**HAL Id: hal-01720322**

**<https://hal.science/hal-01720322v1>**

Submitted on 1 Mar 2018

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

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

# Sub-problems interaction analysis based three-level decomposition algorithm for real-time train scheduling and routing problems in railway stations

Lijie Bai, Thomas Bourdeaud'huy, Emmanuel Castelain, Qi Zhang

**Abstract** The management of rail traffic in stations requires careful scheduling to fit to the existing infrastructure, while avoiding conflicts between large numbers of trains and satisfying safety or business policy and objectives. The train scheduling and routing problem studied includes four tasks: scheduling, routing, platforming and conflict resolution. We propose a three-level decomposition method based on the interactional relationship among four tasks mentioned above to enhance the computational efficiency. This method is tested on full-day timetable obtained from the real-world stations.

**Keywords** Train routing; Train scheduling; Conflict resolution; Decomposition method

## 1. Introduction

A railway station with complex entering and leaving routes network is a busy system with increasing patterns of train services that require accurate scheduling and routing to adapt to the limited infrastructures. At peak hours, the infrastructure is operated nearly at capacity margins. When unexpected disturbances and disruptions affect the normal course of daily operations, dispatchers need to regenerate a feasible timetable by rerouting, delaying or even cancelling some trains to reduce the propagation of delay.

[1] has developed a real-time dispatching system, called ROMA, to automatically recover disturbances. ROMA is able to automatically control traffic, evaluating the detailed effects of train reordering [2] and local re-routing [3] actions, while taking into account minimum distance headway between consecutive trains and the corresponding variability of train dynamics[4][5][6]. [7] focus on the real-time CDR (Conflict Detection and Resolution) problem through junctions and proposes a constraint programming formulation for the combined routing and sequencing

Lijie Bai,

Beijing Hua-Tie Information Technology Development Co.  
National Research Center of Railway Intelligence Transportation System Engineering Technology, China Academy of Railway Sciences, Beijing, China

Thomas Bourdeaud'huy, Emmanuel Castelain

LAGIS - Laboratoire d'Automatique, Génie Informatique et Signal. École Centrale de Lille, Cité Scientifique, Villeneuve d'Ascq, France.

Qi Zhang

Signal & Communication Research Institute, China Academy of Railway Sciences, Beijing, China.

e-mail: lijie.bai@hotmail.com

problem. [8] develop two new routing neighborhoods of different size in order to search for more effective routings, and study their structural properties in a tabu search scheme. [9] and [10] treat train platforming problem considered for the case of multiple routes where platform times can vary in a discrete interval. [11] propose an Integer Linear Programming model (ILP) for the TTP in railway node. This paper studies the scheduling and routing problem at an operational level faced by railway station managers to generate a conflict-free non-cyclic timetable which consists of two sets of circulations. One set is made of commercial circulations given by regional levels. The other set corresponds to technical circulations (shunting) added by the railway station managers to prepare or repair the trains. The problem that we address in this paper is NP-hard and can be stated as follows. Given the layout of a railway station, arrival and departure times, as well as the destination and origin of trains, we aim to schedule the technical circulations within the allowable deviations and to route as many as possible trains through the station. The blocking trains should be pointed out if a solution for all trains cannot be found. The solution must ensure that no pair of trains is conflicting over routes and tracks, while allowing the coupling and uncoupling of trains at tracks and respecting their preferences of platforms.

To solve our problem, we update the timetable under rolling-horizon framework to adapt the dynamic railway traffic, as stated in our previous article [12]. To complete the update calculation in short time, we propose sub-problems interaction analysis based three-level decomposition algorithm.

## 2. Problem formalization

In this section, we firstly formalize our scheduling and routing problem in station. After the description of three-level operational mechanism, we focus on the detailed mathematical model of three levels. A railway station contains a set  $L^e$  of lines located outside the home signals at the entrance of the station, a set  $L^i$  of lines next to the platforms and a set of switches. A route  $p \in P^{(l^i, l^e)}$  contains a set of ordered switches  $p = [s_1^p, s_2^p, \dots, s_{|p|}^p]$  which connects a platform line  $l^i$  with a section line  $l^e$ . The traffic in the railway station is defined by a set of trains  $T$ . Every train  $t$  contains a set of ordered movements  $M^t = [m_1^t, m_2^t, \dots, m_{|M^t|}^t] \subset M$ . The index of a movement represents its chronological order, for example  $m_1^t$  occurs before  $m_2^t$ .

**Table 1.** Movements classification

|          | Commercial | Technical |
|----------|------------|-----------|
| Leaving  | $M^{CL}$   | $M^{TL}$  |
| Entering | $M^{CE}$   | $M^{TE}$  |



cancellation is permitted, ensuring that a feasible solution is always obtained. The objective function is to minimize the number of trains cancelled.

### ***3. Three-level decomposition algorithm***

The problem described in our paper consists of three parts of decision: scheduling, routing and cancellation. The cancellation decision is supported by both scheduling and routing decisions. A wise combination of scheduling and routing decisions can efficaciously reduce the number of trains cancelled. However the combination of scheduling and routing decisions represents a huge solutions search space. In order to reduce the calculation complexity, we divide the problem into two parts: scheduling-cancellation sub-problem and routing-cancellation sub-problem, so as to avoid the huge amount of combinations between scheduling and routing variables. At the same time, the correlation between the two sub-problems cannot be ignored. In this case, a three-level decomposition method is designed as a circle mechanism to satisfy the demand of independence and cooperation between scheduling and routing in the resolution process. Level I solve the routing-cancellation sub-problem with the given scheduling decision. Level II is responsible for the smooth cooperation between the two sub-problems. Succeeding the given scheduling decision and the cancellation decision solved in Level I, Level II re-route the trains to maximize the margin of scheduling decisions for trains cancelled in Level I. Level III succeeds the routing decision solved in Level II and re-schedule the trains to minimize the number of trains cancelled.

Three-level decomposition method simplifies the whole problem by three ways. Firstly, the huge amount of combinations between scheduling and routing decisions are avoided. In each sub-problem, the solution search space is narrowed. Secondly, the cooperation relationship between routing and scheduling is recovered by Level II which try to re-route trains and movements with the cancellation decision solved in Level I, so as to maximize the scheduling margin for Level III. Finally, the original cancellation is identified in Level I. We distinguish the cancellation into two types according to the causes: original cancellation and operational cancellation. The original cancellation brought by ideal timetable is caused by the unavoidable conflicting stops on tracks among trains or the overload of commercial movements on station networks. Note that only the actual time of technical movements can be modified in scheduling sub-problem. In the ideal timetable, the duration of stop on tracks is minimal for all trains. In the following steps, modification of actual time for technical movements make the trains' stopping on tracks prolonged and the conflicts on tracks intensified. So at the first iteration, level I has the best chance to allocate tracks and paths separately to trains' stopping and commercial movements with the ideal timetable as parameters. Moreover, the weighted function of Level I prefer to cancel technical movements. Under all these favorable conditions, the original cancellation is identified by Lev-

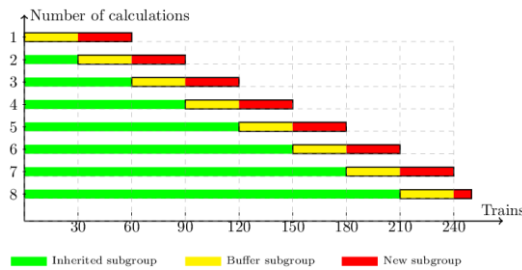
el I at the first iteration and unrecoverable in the following steps. The operational cancellation is caused by the infeasible technical movements. It is possible to absorb the operational cancellation through the following iterations.

#### 4. Experiments and Results

To observe the performance of three-level decomposition algorithm, we solve a full-day timetable in a real-world station in two ways. Firstly, we solve the full-day problem step by step under rolling-horizon framework. In each calculation, the solution obtained by three-level method is provided as an initial solution to the exact method described in our previous article [12]. Maximum iteration of three-level method is 5. Solutions obtained with and without three-level method are compared in Table 3. Secondly, we execute one calculation of three-level method to solve the full-day problem. Maximum iteration of three-level method is 9.

The computational results are obtained by using CPLEX version 12.6 on a 64 bits computer under Linux with Intel i5-2520M CPU at 2.5GHz and 8GB memory RAM. The calculation time is limited to 500 seconds.

A brief introduction of rolling-horizon framework designed in [12] is as follows. The full-day problem of 247 trains is updated by 8 calculations in chronological sequence. An illustrative execution is given in Fig 2. In each calculation, we solve



a problem of 60 trains (one yellow buffer subgroup and one red new subgroup) and take the solution of green inherited subgroup as valid constraints. Inherited subgroup includes the buffer subgroup solved in the last calculation and the trains have already passed through the station.

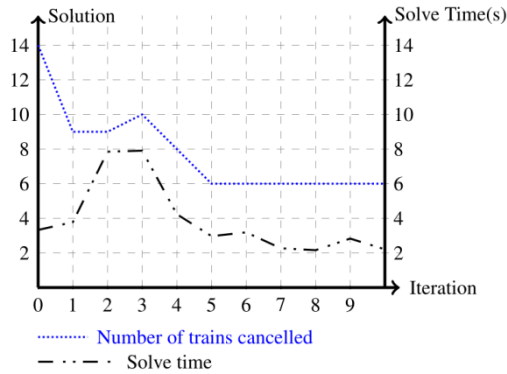
**Fig.2** Rolling-horizon framework.

The one-day timetable is divided into 8 subgroups of trains shown in the first column. The second and third column contains the numbers of technical and commercial movements for the relevant subgroup. The train subgroup can be divided into three groups: inherited group, fixed group and buffer group which are described, in the 5th and 6th columns, by the number of trains and the time interval occupied in minutes. With  $N = 60$ , there is no intersection between the time interval of buffer group and the time interval of inherited group to avoid insolvable potential conflicts with inherited group. The minimum number of trains cancelled solved within 500 seconds and the resolution information are shown in the last two large columns in Table 3. We can see that three-level method greatly reduces

the solve time from 2496s to 83s. At the same time, comparison of solution gap states clearly that three-level decomposition algorithm efficiently improve the quality of solution to optimality as the gap is reduced to 0%. With three-level method, 6 trains, instead of 9 trains, are cancelled in the full-day timetable.

**Table 3.** Recovery of full-day timetable under rolling-horizon framework

| Trains Subgroup | Movements |       | Group Type | Trains  | Time Interval(min) | TiLim=500s with tri-level |       |            | TiLim=500s without tri-level |         |            |
|-----------------|-----------|-------|------------|---------|--------------------|---------------------------|-------|------------|------------------------------|---------|------------|
|                 | Tech.     | Comm. |            |         |                    | Obj                       | GAP   | Solve time | Obj                          | GAP     | Solve time |
| 0-59            | 45        | 78    | Inherited  | -       | -                  | 0                         | 0.00% | 1.56       | 0                            | 0.00%   | 50.23      |
|                 |           |       | Buffer     | 0-29    | 30-406             |                           |       |            |                              |         |            |
|                 |           |       | New        | 30-59   | 322-530            |                           |       |            |                              |         |            |
| 30-89           | 41        | 78    | Inherited  | 0-29    | 30-406             | 2                         | 0.00% | 48.42      | 2                            | 100.00% | 500.05     |
|                 |           |       | Buffer     | 30-59   | 322-530            |                           |       |            |                              |         |            |
|                 |           |       | New        | 60-89   | 438-662            |                           |       |            |                              |         |            |
| 60-119          | 48        | 73    | Inherited  | 0-59    | 30-530             | 4                         | 0.00% | 2.15       | 7                            | 85.71%  | 500.07     |
|                 |           |       | Buffer     | 60-89   | 438-662            |                           |       |            |                              |         |            |
|                 |           |       | New        | 90-119  | 575-795            |                           |       |            |                              |         |            |
| 90-149          | 47        | 76    | Inherited  | 0-89    | 30-662             | 4                         | 0.00% | 2.06       | 7                            | 33.33%  | 500.02     |
|                 |           |       | Buffer     | 90-119  | 575-795            |                           |       |            |                              |         |            |
|                 |           |       | New        | 120-149 | 735-967            |                           |       |            |                              |         |            |
| 120-179         | 40        | 84    | Inherited  | 0-119   | 30-795             | 4                         | 0.00% | 7.65       | 7                            | 0.00%   | 165.09     |
|                 |           |       | Buffer     | 120-149 | 735-967            |                           |       |            |                              |         |            |
|                 |           |       | New        | 150-179 | 892-1133           |                           |       |            |                              |         |            |
| 150-209         | 31        | 92    | Inherited  | 0-149   | 30-967             | 5                         | 0.00% | 7.42       | 8                            | 25.00%  | 500.04     |
|                 |           |       | Buffer     | 150-179 | 892-1133           |                           |       |            |                              |         |            |
|                 |           |       | New        | 180-209 | 1012-1208          |                           |       |            |                              |         |            |
| 180-239         | 32        | 89    | Inherited  | 0-179   | 30-1133            | 6                         | 0.00% | 9.2        | 9                            | 12.50%  | 275.81     |
|                 |           |       | Buffer     | 180-209 | 1012-1208          |                           |       |            |                              |         |            |
|                 |           |       | New        | 210-239 | 1143-1361          |                           |       |            |                              |         |            |
| 210-246         | 22        | 53    | Inherited  | 0-209   | 30-1208            | 6                         | 0.00% | 4.26       | 9                            | 9.37%   | 6.10       |
|                 |           |       | Buffer     | 210-239 | 1143-1361          |                           |       |            |                              |         |            |
|                 |           |       | New        | 240-246 | 1215-1494          |                           |       |            |                              |         |            |



**Fig 3.** Recovery of full-day timetable by three-level method

Then we execute one calculation of three-level method to solve the full-day timetable. The solution is displayed in Fig 3. The best solution found at the 5<sup>th</sup> iteration is 6 trains cancelled. The total solve time is 30s shorter than 83s.

## 5. Conclusion

In this paper, we present the train scheduling and routing problem through a railway station. To ensure the feasibility of timetable, cancellation of trains is permitted to eliminate insolvable conflicts. The timetable adapts the dynamic railway traffic under the rolling horizon framework. Three-level decomposition algorithm based on sub-problems interaction analysis efficiently improves the per-

formance of adapting timetable in railway stations. The complete algorithm can be easily applied on different railway stations.

In further work, more real cases in different railway stations will be collected and studied to evaluate the efficiency of the complete algorithm.

**Acknowledgments** We would like to thank the anonymous reviewers for their helpful and constructive remarks. In addition authors wish to thanks their institutions, Beijing Hua-Tie Information Technology Development Co., National Research Center of Railway Intelligence Transportation System Engineering Technology, Signal & Communication Research Institute of China Academy of Railway Sciences (CARS) and LAGIS of École Centrale de Lille, for their support. This work is supported by Science Foundation for Young Scientists of CARS (Grant no. 2016YJ065).

## Reference

- [1] A. D’Ariano, Improving real-time train despatching: model, algorithms and applications, PhD thesis, TU Delft, 2008.
- [2] A. D’Ariano, D. Pacciarelli, M. Pranzo, A branch and bound algorithm for scheduling trains in a railway network., *Eur J Oper Res* 183(2) (2007a) 643–657.
- [3] A. D’Ariano, F. Corman, D. Pacciarelli, M. Pranzo, Real-time train conflict detection and resolution: global sequencing and local rerouting., van Zuylen HJ (ed) *Proceedings 9th TRAIL congress, selected papers*. Delft University Press (2006b) 77–72.
- [4] A. D’Ariano, T. Albrecht, Running time re-optimization during real-time timetable perturbations., Allan J, Brebbia CA, Rumsey AF, Sciutto G, Sone S, Goodman CJ (eds) *Computers in railways X.WIT*, Southampton, (2006a) 531–540.
- [5] A. D’Ariano, M. Pranzo, I. Hansen, Conflict resolution and train speed co-ordination for solving real-time timetable perturbations., *IEEE Trans Intell Transp Syst* 8(2) (2007b) 208–222.
- [6] A. D’Ariano, M. Pranzo, I. Hansen, Reordering and local rerouting strategies to manage train traffic in real time., *Transportation Science* 42(4) (2008) 405–419.
- [7] J. Rodriguez, A constraint programming model for real-time train scheduling at junctions., *Transportation Research Part B* 41(2) (2007) 231–245.
- [8] F. Corman, A. D’Ariano, D. Pacciarelli, M. Pranzo, A tabu search algorithm for rerouting trains during rail operations., *Transportation Research Part B* 44 (2010) 175–192.
- [9] A. Caprara, Almost 20 years of combinatorial optimization for railway planning: from lagrangian relaxation to column generation., In: Thomas Erlebach and Marco Lübbecke (Eds), *Proceedings of the 10th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization and Systems(ATMOS)*. Schloss Dagstuhl. Germany, ATMOS 2010 1 (2010) 1–12.
- [10] A. Caprara, L. Galli, Solution of the train platforming problem, *Transportation Science* 45(2) (2011a) 246–257.
- [11] V. Cacchiani, F. Furini, M. Kidd, Approaches to a real-world train timetabling problem in a railway node, *Omega* 58 (2016) 97–110.
- [12] Lijie Bai, Thomas Bourdeaud’huy, Emmanuel Castelain. (2015) Automatic one-day timetable generation in busy and complex railway station with cancellation processing: platforming, routing and scheduling. In 15th IFAC Symposium on Information Control in Manufacturing (INCOM 2015). Ottawa, Canada, 11-13 May 2015.(EI: 20160201797448)