Overview of Optimization Models and Algorithms for Train Platforming Problem

Yinggui Zhang, Aliyu Mani Umar and Min An

Abstract In this paper, an overview of recent advances in the research on train platforming problem (TPP) is presented. The TPP is usually the last problem encountered in planning a railway system which occurs after a schedule of trains in a railway network (train timetable) has been determined. It aims to map a given train timetable to an existing station infrastructure. This process is critical as it determines the feasibility of an optimally generated train timetable along a railway line at station(s) to be visited by trains on the timetable. This optimization problem is in most stations solved manually, and it is a time consuming and error-prone process. Several computer programs are now being developed to aid infrastructure managers and train operators as decision support systems in solving this problem. This paper presents some of these solutions. However, due to variations in operating policies of railway industries in different countries, several variants of this problem exist in the literature. These variations could be seen in the solution approach through the importance attached to level of service, safety of operations, capacity utilization, etc. These variations and the various optimization techniques adopted by researchers are also discussed in this paper. Currently, most models and algorithms presented in literature are not ready for use as commercial systems. Integrating such systems into real-life planning and operations is crucial for efficient use of railway systems.

Keywords Train platforming problem · Optimization models · Optimization algorithms · Review

Y. Zhang · A. M. Umar
School of Traffic and Transportation Engineering, Central South University, Changsha 410004, China

A. M. Umar
Department of Civil Engineering, Ahmadu Bello University, Zaria 810107, Nigeria

M. An (✉)
School of Science,EngineeringandEnvironment, University of Salford, Manchester M5 4WT, UK
e-mail: M.An@Salford.ac.uk

1 Introduction

In railway operations, trains operate in a railway network following a systematic predetermined schedule. One of such schedules is the train platforming plan. This essential component of railway operations planning provides information on the routing of trains at stations and platform each train will occupy for a definite period of time. Hence, the train platforming problem aims to solve, for a given train timetable and station topology, the allocation of platform and route to (and from) such platform for each train. This plan is crucial as it validates the feasibility of an optimal train timetable since most macroscopic-modelled timetables only contain an upper bound of the maximum number of trains that can simultaneously be present in a station.

The solution to such problem is usually an easy task when the station has fewer number of tracks and less traffic. However, the problem becomes complicated as the number of platforms and traffic increases; which is mostly the case, as many countries are promoting railway transport over other modes of transportation. As a result, using computer algorithms in solving the TPP becomes necessary as the conventional manual method is tedious and, in some cases, yielding infeasible solutions. Cardillo and Mione [1] highlighted how in a particular case, platforming 242 trains in a station with 16 platforms require 15 working days for an expert planner.

Capacity of stations to handle the TPP is usually determined by the number of platforms, station tracks, and the trains operations (coupling and uncoupling of trains, frequency, arrival and departure times, headway, dwell times, etc.). All these factors are known a priori and are considered in coming up with a station platforming plan. However, the occurrence of a disturbance in real-life railway operations is inevitable and when such happens, an existing platforming plan in most cases becomes infeasible. Hence, the TPP is a problem encountered at three levels of a railway system [2, 3]. The first level (strategic level) involves analysis of future infrastructural capacity requirements of station. The second level (tactical level) is during the timetabling stage; where the feasibility of a generated timetable at stations is determined. Lastly, during real-time operations (operational level) when a rescheduled timetable invalidates an existing train platforming plan.

This paper focuses only on recent optimization models and algorithms for solving train platforming problems at strategic, tactical, and operational levels. However, discussions on what the authors believe are fundamental older models and algorithms are included.

The paper is structured as follows; Sect. 2 gives a background on train timetabling and train platforming and how the two are related. Section 3 presents the different variants of TPP models and algorithms with their performances. Finally, Sect. 4 contains conclusions and suggestions on future research paths.

2 Background

2.1 Train Timetable and Station Infrastructure

In railway transportation, trains move along a network of rail lines (sections) and stations in a systematic way using a series of coordinated signals and communications systems. Due to capacity limitation and the need for a safe operation, these trains move along the network through predetermined schedule defined in a timetable. The timetable provides information such as arrival and departure times at stations to be visited by trains, direction of travel, etc. This information helps in providing a conflict-free operation across the railway network. The train timetable could be cyclic; in which it repeats itself every hour of a day or acyclic; in which trains are operated as per demand period and also repeats itself every day. Because transportation is a derived demand, demand for transportation could be low during some periods of the day and exceptionally high during other periods (peak demand). This makes the latter method of timetabling more accommodating to real-life situations. However, in many countries, the train timetable is periodic [4] technically because it makes the operation and management easier and is also easier for passengers to remember.

An optimum timetable (which is one of the two inputs in a train platforming plan) ensures that there are no conflicts along the sections in a railway network while making efficient use of available resources. This timetable is usually obtained after frequency and stopping patterns (line planning) have been defined [5]. Researchers developed several computer programs to generate these timetables while others provide a conflict-free timetable in the event of a perturbation in the system [6–8]. Because this paper is not focused on train timetabling, we will refer the reader to a review on mathematical models and algorithms involved in railway timetable scheduling [3] and railway timetable rescheduling [9].

The second input in a train platforming plan is the station infrastructure, usually presented in a form of station topology. The optimum train timetable ensures a conflict-free operation across the railway sections. The next task is to ensure a conflict-free operation at station. This is determined with the aid of the station topology. The station topology is a diagrammatical representation of physical elements in the station (platforms, turnouts, switches, track sections, etc.) with nodes and directional lines (Fig. 1).

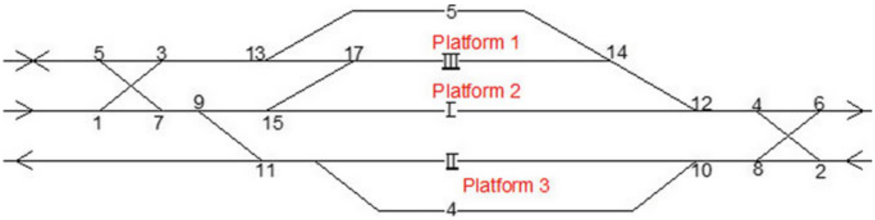

Fig. 1 Topology of a typical passenger railway station

2.2 General Train Platforming Problem and Mathematical Formulation

The assignment of trains to platforms in a station as per timetable schedule is what a platforming problem entails. This assignment has to also define the path that each train will follow to such platform while maintaining operational constraints. Usually, such problem requires the planner (either manual or automatic) to map the train traffic in a given train timetable unto station infrastructure. Over the years, computer programs have been developed to aid dispatchers solve such problem, some of which have been incorporated into real-train operations, example, RFI-Italy [10] and Ocapi-Belgium [11]. Like other railway operations, the platforming problem is mostly solved as a periodic event scheduling problem (PESP) since most train timetables are cyclic. In such problem, event times are confined within [0, C), where "C" represents the cycle length.

Different mathematical formulations were provided by researchers; which will be discussed briefly in section three. However, as a representative example, we will present a description of a general and encompassing mathematical formulation similar to that in Caprara et al. [10]. The station to be considered here (Fig. 1) has (one-way) double lines and a single (two-way) line, 6 platforms (or 3 shared platforms) with several arrival and departure paths. An arrival path, here, is a set of interconnected sections of track and switches a train will follow upon entering the station to its assigned platform. A departure path in this paper, defines the set of interconnected sections of track and switches a train will follow when leaving the station from its assigned platform. In the general version of the problem, we are given a set B of platforms and a set T of trains to be routed to a platform every day of a given time horizon. Moreover, for each train $t \in T$, we are given a collection P_t of possible *patterns*. Each pattern corresponds to a feasible route of train t within the station, including a stopping platform, an arrival path and an arrival time, a departure path and a departure time. Each train must be assigned a pattern that will be repeated every day of the time horizon.

Operational constraints forbid the assignment of patterns to trains if this implies occupying the same platform at the same time, or also using arrival/departure path that intersects at the same time or too close in time. In the general version, this is represented by defining a pattern-incompatibility graph with one node for each train-pattern pair (t, p) with $p \in P_t$, and an edge joining each pair (t_1, p_1), (t_2, p_2) of incompatible patterns. This graph models "hard" incompatibilities that must be forbidden in a feasible solution. However, in the general version, there are also "soft" incompatibilities, generally associated with the use of arrival/departure paths close in time that are admitted but penalized in the objective function.

In case not all trains could be assigned to regular platforms, it is customary to make use of dummy platforms; which are fictitious platforms that we will penalize their use (in the objective function) but may be necessary to obtain a feasible solution. For a strategic train platforming plan, the use of a dummy platform suggests enlarging the station, whereas for a tactical and operational train platforming plan, the use of a dummy platform suggests that not all trains can be platformed at the given instance. When such happens, the options are either to cancel trains, queue-up trains or relax some hard constraints imposed in the model.

The TPP requires the assignment of a pattern $p \in P_t$ to each train $t \in T$ so that no two incompatible patterns are assigned and the objective function defined by the following coefficients is minimized. There is a cost c_b for each platform $b \in B$ that is used in the solution, a cost $c_{t,p}$ associated with the assignment of pattern $p \in P_t$ to train $t \in T$, and a cost ct_1, p_1, t_2, p_2 associated with the assignment of pattern $p_1 \in Pt_1$ to train t_1 and the assignment of pattern $p_2 \in Pt_2$ to train t_2 for $(t_1, t_2) \in T^2$, in case these two patterns have a "soft" incompatibility. Here, $T_2 \subseteq \{(t_1, t_2){:}\ t_1, t_2 \in T, t_1 \neq t_2\}$ denotes the set of pairs of distinct trains whose patterns may have a "hard" or "soft" incompatibility.

3 Train Platforming Models and Algorithms

3.1 Strategic Level Optimization Models and Algorithms

The TPP at this level is typically a station's infrastructure capacity assessment, with a view of determining the adequacy or otherwise of station infrastructure. Zwaneveld et al. [2] approached the routing of trains through stations based on a node-packing approach following their proof of the problem as NP-complete. The algorithm developed, which is based on the formulation of the problem as a node-packing problem, and on the application of preprocessing techniques, heuristics and a branch-and-cut procedure was implemented into the planning system, STATIONS.

Zwaneveld et al. [12] improved on the model and algorithm presented in Zwaneveld et al. [2]. Specifically, the model was improved by incorporating shunting decisions and preferences to allocation of trains to certain desired platforms and routes. The algorithm was improved by extending the preprocessing techniques and also investigating their characteristics with respect to propagation. These improvements proved promising as all the problem instances studied were resolved to optimality within an average computing time of about 1 min.

3.2 Tactical Level Optimization Models and Algorithms

At the tactical stage, it is believed that the platformer has all the organizational details of the railway system to plan for. These details include the train timetable, layout of stations along the line and other enterprises' policies that exist.

While Zwaneveld et al. [12] considered the general routing of trains through stations (which assigns trains to a complete path through a station; platform allocation being part of the task) other researchers solved the problem while putting emphasis on the allocation of platforms to arriving trains. One of such works was carried out by Cardillo and Mione [1]. They modelled the TPP as a graph coloring problem (the k L-list τ coloring problem).

An algorithm was developed based on the formulation of the problem as a graph coloring problem and application of a backtracking and heuristic technique to solve the problem. In one of their reported case studies, a station with 13 tracks (platforms) and 177 trains on a 24-hour cycle took a Linux Pentium at 166 MHz, 115 s of CPU time to yield a solution.

Billionet [13] suggests integer programming as an alternative solution approach to the TPP as formulated by De Cardillo and Mione [1]. The two ILP formulations he described aim to find at each time, whether an integer solution exists or not. These solutions, however, do not provide an optimization result of the TPP. To obtain that, Billionet [13] introduced into the more effective ILP an objective function which maximizes the assignment of trains to a particular platform. A station with up to 200 trains and 14 platforms could be solved using standard and commercially available ILP solver software.

Carey and Carville [14] presented a greedy heuristic solution to the TPP which aims to simulate the practical process of train operations in countries where there are competing train operating companies (TOCs) operating on common lines and stations. To overcome the difficulty in adding up the costs or penalties imposed on deviations from preferred train arrival and departure times and cost of choosing less preferred platform, Carey and Carville [14] introduced “lexicographic” cost functions or decision rules. To resolve conflicts, Carey and Carville [14] delay the trains rather than advance them in an effort to imitate the practice of traditional manual planners (especially in Britain).

The algorithm (which they call A1) proves promising when tested on the busy and complex Leeds station (in the North of England) with 12 main platforms (or 34 sub platforms) and 491 trains daily. The work of Carey and Crawford [15] extends the problem to consider a network of busy complex stations. This is particularly essential because a change in the planned arrival and/or departure time (s), dwell time at a station for a train will propagate to subsequent stations the train will visit especially when adequate buffer time is not available.

Caprara et al. [10] considered minimizing the number of dummy platforms used in the objective function. The model contains a quadratic term which results from the “soft” incompatibility constraints. This complexity in the model is relieved by using a novel linearization method that requires smaller number of variables and leads to a stronger linear programming relaxation instead of the conventional approach of introducing additional variables to represent the product of the original binary variables.

To assess the performance of their branch-and-cut-and-price method, they compared it with the current heuristic method used by Rete Ferroviaria Italiana. In the four cases they studied, their algorithm proved superior at all possible values of dynamic threshold (π) tested.

The ability of a train platforming plan to absorb the inevitable disturbances in railway operations is crucial. This led Dewilde et al. [16] to introduce an approach to improve the robustness in a complex station zone. To do that, they focused on three aspects of the planning; the routing of the trains through the station zone, the timetable at the stations within this zone, and the platform assignments. The algorithm developed has three modules, each to tackle an aspect of the planning.

In the platforming module, platform assignment of all trains is assumed to be fixed, as is the usual approach in solving platforming problems. To save computation time, only relevant candidate platforms are evaluated for a train (when assumption of a fixed platform assignment could not be made). A dominance rule is used to limit the number of candidate new routes and a restriction is placed on the amount of conflicts associated with the new route in comparison with the old one. The process described will yield for each candidate platform change, a solution for all the train platforming at all the stations within the zone. The impact of each change is evaluated using the internal timetabling module and the best platform change is selected if it leads to an improved solution. Such cycle is repeated until the overall algorithm is not able to find an improved version of route, timetable, and platform assignment anymore.

Contrary to De Cardillo and Mione [1] and Billionet [13], Sels et al. [11] dropped the assumption that all routes in the station will require the same time to be traversed by trains. This assumption is impractical, considering the variation in speed limit at different switches, length of routes, train length and speed, etc. Optimality in the mixed-integer linear programming (MILP) model is attained by minimizing the total cost function, which comprises of penalty for assignment of a non-preferred (real) platform and an even higher penalty for assignment of a dummy platform. In the goal function, all hard constraints are forbidden. This is necessary so that more platforming options could be obtained when a preferred platform assignment could not be made.

The authors compared three solvers (CPLEX, Gurobi, and XPRESS) to determine which best solves the MILP model within a reasonable time. The computation times obtained are all satisfactory even without the use of variable reduction techniques. For the 10 station's one-day traffic tested, and results showed that about 30 s are required to platform all trains at the tactical level and below 9 min at the strategic level.

Petering et al. [17] modeled the train timetabling and platforming problems together by a mixed-integer linear programming (MILP) model and consider a single track, unidirectional rail line consisting of an origin, destination, intermediate stations laying between the origin and destination, and a set of parallel sidings (platforms) in each station that accommodate trains stopping in that station.

The MILP model has two parts objective function. The first aims at minimizing the cycle length, while the second minimizes the total journey time of all train-types using linear constraints and a linear objective function. The effectiveness of the model was demonstrated when it solved a large problem instance inspired by the Japanese Shinkansen train in less than an hour using IBM ILOG CPLEX 12.5 solver on a desktop computer with eight 3.4 GHz cores and 16 GB RAM. Due to the complexity of the model and the importance attached to computing times, preprocessing technique is used and this helps in reducing computing time.

3.3 Operational Level Optimization Models and Algorithms

The solution of TPP at the operational level is the most sought-after, since it is at this level that real-time management of operations is involved. To enhance the stability of a train platforming model, Miao et al. [18] present a model that omits the compatibility constraints of resource occupations. This restricts the assignment of only one resource to every operation. Stability according to Miao et al. [18] could be achieved by making the headway times among potential conflicting tasks as rationale as possible. In the two-component objective function, they propose, the first component (which is the primary objective) ensures the stability of the train platforming plan by maximizing the time interval between two adjacent occupations of track and the second component (secondary objective) ensures compatibility of the platform allocation plan in the station. The overall objective function aims to minimize the cost of changing arrival and departure times of trains to return a feasible solution. The stability enhancing train platforming model (SETPM) is solved using an ant colony optimization algorithm.

The assess of the effectiveness of the SETPM, they compared its performance with a model for minimizing resource allocation costs. This is comparison of cost minimization; therefore, the component of the SETPM objective function that measures stability is dropped and a penalty is introduced to ensure a feasible solution. The results of an experiment carried out on a high-speed train station in Changsha reveals that the SETPM is capable of increasing the stability of the train platforming plan by about 37%.

Chakroborty and Vikram [19] presented an optimum solution approach to TPP to take care of the uncertainties that occur during real-life operations. This according to them is necessary as most long-distance trains are often delayed by an hour or more (in their case study, India). This situation leads to some trains queuing up at the station entrance due to unavailability of platforms. They presented a model which takes into account the delay (that happens in real-life operations) and subsequent queuing up of trains as a result of such delays. The model is capable of resolving such problems provided the arrival of trains to stations is known at least an hour in advance. Because this is a solution at operational level, the authors do not want any adjustments to the arrival times of trains (since this will translate to even more delays or impractical advancement) and hence, arrival times of trains are direct inputs in the model (not variables). The key decision variables are the times trains (queue at the station entrance) will enter the station and the allocated platform for each train.

To obtain an optimum assignment, the costs on time a train spent waiting at station entrance, a non-preferred platform assignment and last-minute change to previously (announced) assigned platform are minimized. The MILP formulation using ILOG CPLEX 9.0.0 on a 400 MHz processor and 1 GB RAM is used to solve to optimality various problems related to a busy station in India; with 9 platforms and an average arrival rate of 55 trains per hour (specifically, 110 trains in 2-hour time horizon). In the 10 min computation time, all trains were platformed without any queue at the station entrance.

4 Conclusions and Further Research

In this paper, we discussed the train platforming problem, which is a problem of assigning platforms to arriving trains in a station while satisfying various constraints encountered in railway operations. We presented a general mathematical description of the problem and the various levels of railway system at which this problem is encountered. A great number of papers on TPP tackled the problem at the tactical level with an aim to provide an optimum (or at least a feasible) assignment of platforms to trains. Most models considered the optimum assignment of as many trains as possible to platforms and the unassigned trains will either be rescheduled or cancelled. Other models considered preferences in allocation of certain platforms to certain trains. In formulating such models, it is believed that some operations (in real time) will overlap and lead to infeasibility. Hence, buffer times are introduced to absorb such small discrepancies. However, perturbations in real-life railway operations are unpredictable and, in most cases, render an existing train platforming plan infeasible. This problem is addressed in TPP at operational level. This is perhaps the most demanding, since real-time management of train operations is involved and in the event of a disturbance which invalidates an existing train platforming plan, solutions will be required within short period of time. Unlike in the strategic and tactical levels, computing time for solving TPP at operational level is very important.

The use of combined approach in tackling the problem of perturbations at operational level is seen in most recent works on railway operations planning. This combined approach could involve incorporating the timetabling and platforming plans into one problem and solving the problem all together. Although, most TPP models and algorithms are developed as stand-alone solutions, others could be used as components for a more general system in scheduling a railway network. This approach makes the whole process much efficient and easier to manage.

In further research, more attention should be focused on improving the robustness of railway stations by considering an integrated approach of timetabling and platforming for even larger network of stations. This will ensure the stability of train timetabling and platforming plans to effects of disturbances and disruptions.

Also, the use of some hard constraints limits usable capacity in a station. This could be seen in models where a hard constraint is imposed on the occupation of a route in a station by two trains irrespective of the clearance between them. This is indeed not always true, as liberation points exist in real stations that allow two trains to occupy the same route at a time especially during peak periods or periods where the timetable is rescheduled. Subsequent research should explore the use of these flexible constraints that could improve the capacity of a station while maintaining safety of operations.

Acknowledgements The research is supported by the National Natural Science Foundation of China (Grant No. 71971220) and the Natural Science Foundation of Hunan Province, China (Grant No. 2019JJ50829).

References

1. Cardillo DDL, Mione N (1998) kL-list λ colouring of graphs. Eur J Oper Res 106(1):160–4
2. Zwaneveld PJ, Kroon L, Romeijn H, Salomon M, Dauzere-Peres S, Van Hoesel SPM et al (1996) Routing trains through railway stations: model formulation and algorithms. Transport Sci 1(30):181–194
3. Lusby RM, Larsen J, Ehrgott M, Ryan D (2011) Railway track allocation: models and methods. OR Spectrum 33:843–883
4. Sparing D, Goverde RMP (2017) A cycle time optimization model for generating stable periodic railway timetables. Transp Res Part B 98:198–223
5. Sels P, Cattrysse D, Vansteenwegen P (2016) Automated platforming & routing of trains in all Belgian railway stations. Expert Syst Appl 62:302–316
6. Schrijver A, Steenbeek A (1995) Timetable construction for Railned Amsterdam: CWI
7. Tormos P, Lova A, Barber F, Ingolotti L, Abril M, Salido MA (2008) A genetic algorithm for railway scheduling problems. In: Xhafa F, Abraham A (eds) Metaheuristics for scheduling in industrial and manufacturing applications, Springer, Berlin, Heidelberg, pp 255–76
8. Bettinelli A, Santini A, Vigo D (2017) A real-time solution algorithm for the train rescheduling problem. Transp Res Part B 106:237–265
9. Cacchiani V, Huisman D, Kidd M, Kroon P, Toth P, Veelenturf L et al (2014) An overview of recovery models and algorithms for real-time railway rescheduling. Transp Res Part B 63:15–37
10. Caprara A, Galli L, Toth P (2011) Solution of the train platforming problem. Transport Sci 45 (2):246–257
11. Sels P, Vansteenwegen P, Dewilde T, Cattrysse D, Waquet B, Joubert A (2014) The train platforming problem: infrastructure management perspective. Transp Res Part B 61:55–72
12. Zwaneveld PJ, Kroon LG, Romeijn HE, Salomon M, Dauzere-Peres S, Van Hoesel SP et al (1996) Routing trains through railway stations: model formulation and algorithms. Transport Sci 30(3):181–194
13. Billionet A (2003) Using integer programming to solve the train platforming problem. Transport Sci 2(37):213–222
14. Carey M, Carville S (2003) Scheduling and platforming trains at busy complex stations. Transp Res Part A 37:195–224
15. Carey M, Crawford I (2007) Scheduling trains on a network of busy complex stations. Transp Res Part B 41:159–178
16. Dewilde T, Sels P, Cattrysse D, Vansteenwegen P (2013) Robust railway station planning: An interaction between routing, timetabling and platforming. J Rail Transp Plann Manag 3:68–77
17. Petering MEH, Heydar M, Bergmann DR (2015) Mixed-integer programming for railway capacity analysis and cyclic, combined train timetabling and platforming. Transport Sci, Articles In Advance. 2015:1–18
18. Miao J, Yu Y, Meng L, Yang Z (2012) Stability-oriented optimization of train platforming problem on high speed railway. J Transpn Sys Eng IT 3(12):115–121
19. Chakroborty P, Vikram D (2008) Optimum assignment of trains to platforms under partial schedule compliance. Transp Res Part B 42:169–184