Communications in Combinatorics and Optimization
Vol. xx, No. x (xxxx), pp. 1-24
https://doi.org/10.22049/cco.2026.30275.2391

CCO
Commun. Comb. Optim.

Research Article

Platforming trains in multi-line stations under flexible track utilization policy

Emine Akyol Özer[1,*], Tuğba Saraç[2]

[1]Department of Industrial Engineering, Engineering Faculty, Eskişehir Technical University, Türkiye
*emineakyol@eskisehir.edu.tr

[2]Industrial Engineering Department, Faculty of Engineering and Architecture, Eskişehir Osmangazi University, Eskişehir, Türkiye
tsarac@ogu.edu.tr

Received: 14 January 2025; Accepted: 16 May 2026
Published Online: 26 May 2026

Abstract: In railway stations, there are two track utilization policies: fixed and flexible. Under fixed track utilization, platform tracks are grouped as inbound and outbound parts associated with directions. The trains from the same direction occupy the tracks in each part, and the platforming problem can be considered separately. Under flexible track utilization rule, trains can be assigned to any platform track considering the station layout, which provides more flexibility when platforming. However, it naturally causes a more complex platforming problem. In this paper, we address the train platforming problem in a multi-directional station under flexible track utilization policy. A mixed-integer linear programming formulation is proposed to assign trains to the station's resources without conflict routes. The objective function is to minimize total weighted delays of trains in which weight refers to the importance level of each train. Previously published studies have only been carried out in small stations with limited trains. However, this study considers busy and complex stations, accommodating more than 1000 trains, and unlike the previous studies, a genetic algorithm is proposed to obtain near-optimal solutions in a short time. Computational analyses are conducted on derived test problems. We construct test problems based on planning periods and traffic volume levels. The performances of the mathematical model and the genetic algorithm are presented in terms of both solution quality and solution time.

Keywords: Train platforming problem, flexible track utilization, mathematical programming model, genetic algorithm.

AMS Subject classification: 90B06, 90C11, 90C90

* *Corresponding Author*

1. Introduction

The past decade has seen increasingly rapid advances in the field of railway systems depending on economic development. The scale and complexity of the system may bring a lot of planning problems related to optimization. These problems can be categorized as strategic, tactical and operational level based on the planning horizon. Strategic-level planning problems focus on future infrastructure of railway network. Tactical-level problems include offline problems such as train timetabling, train platforming and rolling stock scheduling. Rescheduling problems on day-to-day basis are classified as operational-level planning problems. We refer readers to reference [9] for railway track allocation problems at different decision levels.
A well-known example of tactical-level problems is train platforming problem (TPP) which is also known as the track allocation problem. TPP basically involves assignment of trains to the station's resources within station providing conflict-free routes. The complexity of the problem depends on not only the number of trains, but also characteristics of the station's resources such as lines, platform tracks and switches. The large scale and high-density railway stations, especially multi-direction railway stations in which switch area is more complicated, make the problem difficult. Furthermore, TPP with assignment restrictions such as train length and dead-end platform tracks becomes extremely challenging. In this paper, we concentrate on TPP with multi-direction lines, complicated switch area and assignment restrictions in a specific railway station.
The main input of TPP is a daily timetable which is output of train timetabling problem and it can be said that these planning problems are strongly interconnected. Several studies ([6],[15]) addressed these integrated problems. The study of [6] deals with scheduling problem of additional trains into a busy timetable considering capacities of the stations. The authors formulated a bi-objective mixed integer linear programming model to minimize the total travel time of the additional trains and minimizes the adjustment on the existing trains at the same time. In another recent study ([15]), the authors considered rescheduling problem of timetables and platform schedules under planned track maintenance. They developed 0-1 integer mathematical model to adjust the timetable and platform schedules in a such manner that deviation from the planned schedules is minimized. However, in these studies, topology of the considered stations is simple and detailed characteristics of the TPP such as multi-lines, dead-end tracks are negligible. Another integrated version of TPP is with train unit shunting problem. [13] consider the routing and scheduling of trains and locomotives in a station. They firstly proposed a 0-1 integer programming model and this model is decomposed with decomposition methods based on Lagrangian relaxation and Alternating Direction Method of Multipliers.
Multi-direction stations can be described as stations in which more than two main lines are merged. Although these stations are very common in the world, especially in European countries, there is relatively small body of literature that directly focused on TPP in the multi-direction stations. [16] deal with TPP in a multiple direction station from passenger and infrastructure perspectives. A multi-objective nonlinear 0-1

integer programming model is developed to balance the usage of arrival–departure tracks and minimize the walking time of passengers in the station. In a follow-up study, [17] proposed a track occupancy plan for a multi-direction station considering rolling stock plan, and resources in the station throat area. An integer linear mathematical model is formulated for the problem, and capacity of station is discussed from the point of view of the station's resources.

There are two station track utilization policies about assignment of trains to the platform tracks: fixed and flexible. The platform tracks in the station are grouped as inbound and outbound parts associated with directions under fixed track utilization rule and tracks in each part serves the trains from the same direction. However, the flexible utilization policy provides assignment of trains to any platform track considering station layout. In recent years, several studies ([17], [15]) pay particular attention to track utilization policies. Although flexible track utilization policy makes the problem more challenging, the results of [15] indicated that better timetables are provided under this policy.

TPP has been widely addressed in the railway optimization literature and many solution approaches have been developed. Some of the most popular approaches are integer programming models, branch and cut approach, and heuristic/metaheuristic methods.When the studies using mathematical models are taken into account, [22] presented a mathematical model formulation based on the node packing problem for routing trains through railway stations. [1] developed two 0-1 integer programming models based on the study of [3]. The computational results indicated that TPP with 200 trains and 14 tracks can be solved with integer-programming technique. [23] improved the model and the algorithm of [22] including shunting decisions and preferences of trains for platforms and routes. [11] considered TPP on the tactical and strategical level and developed a mixed integer programming model. This model is an extended version of [1] accounting for different route durations calculated from maximum route speeds and train lengths. In addition to these studies, graph coloring and node packing formulations are commonly used, see e.g., [5] and [22]. [3] proposed an efficient metaheuristic algorithm based on graph coloring for TPP in which there are six stations with different number of trains and platform tracks. Another approach to handle TPP is the branch and cut algorithm. [22] developed a solution procedure based on a branch-and-cut approach and solution approaches are tested in data from the station of Zwolle in the Netherlands. [2] handled train platforming problem which is a case study from the Italian Infrastructure manager. The authors proposed a pattern incompatibility graph and designed a branch-and-price-and-cut method. One of the widely used solution approaches in the literature is heuristic/metaheuristic algorithms. [7] addressed reallocating trains to platforms at stations in case of slight schedule changes or major disruptions and developed a greedy interchange heuristic algorithm for the problem. Other solution approaches in the literature include constraint programming methods, hybrid metaheuristic algorithms and simulation methods. We refer the readers to [4], [10], [15], and [14] for more details.

In Table 1, we compare all these studies on the topic of TPP in the key dimensions

Table 1. **Literature review**

Study	Track Utilization Policy	Direction	Objective	Solution Method	Problem Size
[22]	-	-	Max number of trains	Mathematical model and BC	18 trains
[23]	-	-	Maximize number of trains	BC	80 trains per hour
[4]	-	-	Min cost	Heuristic algorithm	491 trains
[1]	-	-	Max addition of cuts	IP based on GC	200 trains
[10]	-	-	Min delay of trains	CP	491 trains
[5])	-	-	Min delay of trains	Two alternative graph formulations	80 trains in one hour
[2]	-	-	Min deviation from “desired” times	branch-and-price-and-cut method	237 trains
[11]	-	-	Min penalties	MIP	160 trains
[6]	Fixed	-	Min total travel time and adjustment of the existing trains	Three stage optimization method	70 trains
[16]	Fixed	Multi	Min the variance between the total occupation time of each track and walking time	BINLP and SA-based algorithm	66 trains
[17]	Flexible	Multi	Min total occupation time of resources	MIP	49 trains in two hours
[15]	Flexible	Single	Min cancellation costs and weighted train travel times	BILP, Lagrangian Relaxation and Alternating Direction Method of Multipliers	40 Trains
[20]	-	Single	Min negative impact of delays	MIP	70 trains in six hours
[18]	Flexible	Single	Min total weighted train running cost	BILP	48 trains
[19]	Fixed	Multi	Min balanced buffer time	MIP and GA	49 trains
This paper	Flexible	Multi	Min total weighted delay of trains	MIP and GA	1047 Trains

of track utilization policy, direction, objective function, solution method and problem size. As can be seen from the Table 1, there is a clear trend towards routing of trains in the multi-direction stations under track utilization policy. In this paper, we consider TPP in a multi-direction station under flexible track utilization policy. To the best of our knowledge, there is only one study, namely that of [17], that focuses on track allocation problem in the multi-direction stations under flexible track utilization policy as we do. However, compared to [17], the considered problem in our study is obviously more complex in terms of planning period, traffic density and station layout. In addition to this, they proposed a mixed integer programming model to obtain solutions for two hours. This problem is at least NP-hard problem, where an optimal solution cannot be found in reasonable time, as the problem size increases. So, we proposed a genetic algorithm to solve large scale TPP.

This study makes the following three contributions to the state-of-the-art in the train platform problem:

- We addressed TPP under flexible track utilization policy in a busy and complex multi-direction station. Platform tracks, lines, and switches are considered in this model. Furthermore, unlike previous studies, we also paid attention to the unidirectional platform tracks which are called as dead-end tracks.
- We present a new mixed integer linear programming model for assignment of trains to the station's resources without conflict routes. The aim is to minimize total weighted delays of trains in which weight refers the importance level of each train.
- A genetic algorithm is designed to quickly obtain the near-optimal solutions for test problems that are derived based on planning periods and traffic volume levels.

2. Problem description and mathematical model

In this section, a detailed problem description and MILP formulation is given.

2.1. Problem description

Stations in railway networks can be generally separated into two categories according to the number of railway lines traversed: single-line stations and multi-line stations. Single-line stations are only traversed by one railway line while multi-line station is intersection point of more than one railway line. For better explanation, we give an illustration of a railway network in Figure 1. Stations C, D, and F are located on the blue railway line and Stations A, B and E belong to red railway line. Station X is a connection station of these two railway lines, so it is a multi-line station.

The most important resource of a multi-line railway station is switches. The trains occupy some group of switches to arrive at/depart from a platform track. They are shared resource for trains since only one train can operate a switch at a time.

In Figure 2, a detailed layout of a multi-line station is presented. The right side (R) of Station X is linked to Station B and D with double track. On the left side (L) of the considered station, the railway lines from Station C are equipped with quadruple track, while those from Station A are double-track. The trains go from the in-route over the platform to the out-route. The in-route train 3 from Station A arrives at platform track 6J by passing through switches (1, 2, 7, 8) and the switches (25, 27, 28, 11) are on the out-route of this train. Similarly, the in-route train 2 coming from Station C line 1 must pass switches (9, 7, 6, 5, 11, 17) and arrives at platform track 3S. If train 2 and train 3 arrive at the station simultaneously, one of these must wait outside the station since both of them use the switch 7. The situation where at least one switch is shared for adjacent in-route or out-route trains can be defined as route conflict and it causes the train delays.

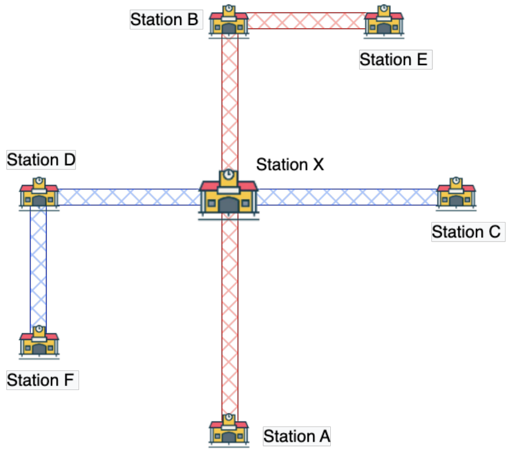

Figure 1. **Railway network**

Under flexible track utilization rule, the in-route and out-route trains can be assigned to any platform tracks connected with their directions. In Figure 2, the trains coming from Station B can be dwell at any platform track based on this rule. However, when fixed track utilization rule is applied, platform tracks are classified according to their direction and the trains from the same direction can be only assigned to tracks in each class. So, assignment of trains running between station B and station D to a platform track can not be realized. Therefore, it is reasonable to handle track allocation problem in multi-line stations under flexible track utilization policy.
In real life, railway stations are more complex and each platform track is not standard. Standard platform tracks are easily accessible from both directions. However, some platform tracks are dead-end. In Figure 2, track 1S can not serve to the Train 5 since it runs from Station D to Station C. Similarly, Train 2, 3 and 4 can not be assigned to platform track 1J. Therefore, we take the assignment limitation of platform tracks into account.

2.2. Mathematical model formulation

In the considered TPP in a multi-direction station, we assume that an ideal timetable and detailed station infrastructure are provided and all parameters are known and given in integer values. The sets, parameters, and variables used in the mathematical formulations are given as follows:
Sets
N : Set of trains, indexed by i,j
M : Set of platform tracks, indexed by l,m
R : Set of switches, indexed by r
K : Set of signals, indexed by k, $k^{'}$, $k^{''}$

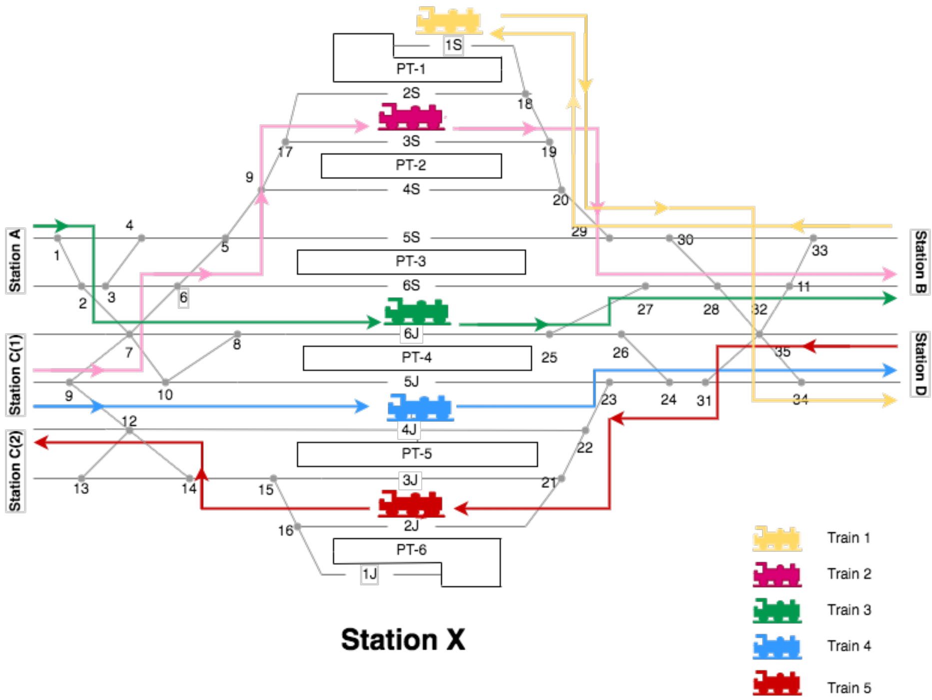

Figure 2. **An example of multi-lines station**

Parameters

α_j : weight of train j

h_j^l : running time for train j between signal and platform on the left side

h_j^r : running time for train j between signal and platform on the right side

p_j^a : planned arrival time of train j to the entry signal

p_j^d : planned departure time of train j from platform

v_j : dwell time of train j

b_{jl} : 0–1 parameter, equal to 1 if there exists a suitable infrastructure-based route (track and switch connectivity) from the arrival and departure tracks of train j to platform track l, 0 otherwise

γ_{jlr}^a : 0–1 parameter, equal to 1 if in-route train j use switch r to reach platform track l, 0 otherwise

γ_{jlr}^d : 0–1 parameter, equal to 1 if out-route train j use switch r to reach platform track l, 0 otherwise

e_j: in-route and out-route side parameter for train j

H_j: total running time on switch area for train j

β_j^a : entry signal of train j

β_j^d : exit signal of train j

δ_j : running time for an in-route train j
t^h, t^s, t^p : minimum headway time for signal, switch and platform
M : A big enough positive integer

Decision variables
s_{jk} : actual departure time of train j from signal k
T_j : delay of train j
a_j^d : actual departure time of train j from platform
w_j^s : waiting time of train j at entry signal
w_j^p : waiting time of train j at platform
x_{jl} : 0–1 binary train assignment variables, equal to 1 if train j is assigned to platform track l, 0 otherwise
y_{ij}^p : 0–1 binary platform-conflict decision variables, equal to 1 if train j arrives to the platform before train i, 0 otherwise
y_{ijk}^s : 0–1 binary signal-conflict decision variables, equal to 1 if train j arrives to the signal k before train i, 0 otherwise
$y_{ij}^{m'}$: 0–1 binary switch-conflict decision variables for in-route/in-route, equal to 1 if train i uses the switch before train j, 0 otherwise
$y_{ij}^{m''}$: 0–1 binary switch-conflict decision variables for in-route/out-route, equal to 1 if train i uses the switch before train j, 0 otherwise
$y_{ij}^{m'''}$: 0–1 binary switch-conflict decision variables for out-route/out-route, equal to 1 if train j uses the switch before train i, 0 otherwise

In this paper, signal index k refers to entry/exit signal in the each double track. As illustrated in Figure 3, entry signal of train j and exit signal of train i are signal $k^{'}$. The actual departure time of train j from entry signal $k^{'}$ ($s_{jk'}$) is sum of planned arrival time of train j to the entry signal (p_j^a) and waiting time of train j at entry signal (w_j^s). After the running time between signal and platform (h_j^r), it arrives to the platform and stays for dwell time (v_j). Actual departure time of train j from platform is bigger than planned departure time of train j from platform. The difference of them refers to delay of train j and in this case, it is naturally equal to w_j^s. At the time $s_{jk''}$, train j leaves the station. After a while, train i arrives to the entry signal at the time p_i^a and the actual departure time of train j from entry signal $k^{''}$ ($s_{jk''}$) is equal to p_i^a value, since train does not wait at entry signal (w_i^s=0). However, this train waits on the platform for w_i^p in addition to dwell time. Similar to train j, train i is delayed and T_i value is equal to w_i^p in this case. As can be seen from Figure 3, arrival and departure direction of train q is same and $s_{qk''}$ should be calculated for arrival and departure case. This train arrives and departs to/from station on time. So, w_q^s, w_q^p and T_q values are zero.
The running time of an in-route/out-route train on the switch area (signal/platform - platform/signal) depends on not only train speed but also arrival/departure side of train. To achieve more compact formulation, we purify the sides of station from the model. Based on in-route/out-route side of trains (e_j), we determine the running time on the switch area(H_j) and running time between entry signal and platform for

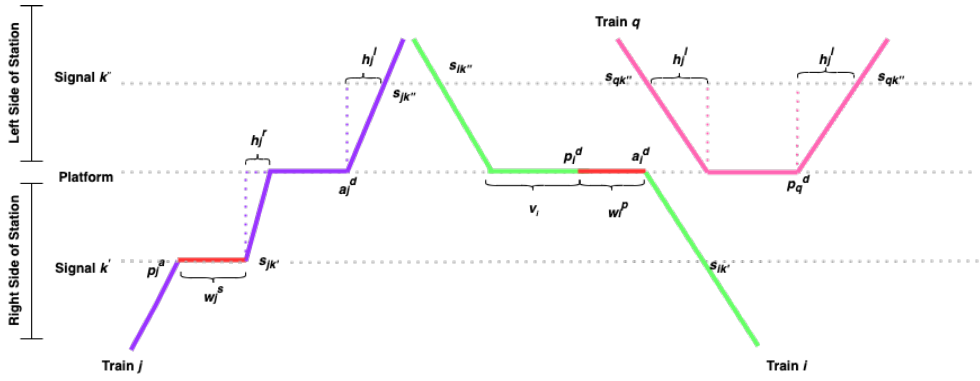

Figure 3. Time space network

in-route train (δ_j). These parameters' values are given as follows:

$$e_j = \begin{cases} 1, & \text{if in-route side of train } j \text{ is left and out-route side of train } j \text{ is left (L-L)} \\ 2, & \text{if in-route side of train } j \text{ is left and out-route side of train } j \text{ is right (L-R)} \\ 3, & \text{if in-route side of train } j \text{ is right and out-route side of train } j \text{ is left (R-L)} \\ 4, & \text{if in-route side of train } j \text{ is right and out-route side of train } j \text{ is right (R-R)} \end{cases}$$

$$H_j = \begin{cases} 2h_j^l, & e_j = 1 \\ h_j^l + h_j^r, & e_j = 2 \text{ or } e_j = 3 \\ 2h_j^r, & e_j = 4 \end{cases}$$

$$\delta_j = \begin{cases} h_j^l, & e_j = 1 \text{ and } e_j = 2 \\ h_j^r, & e_j = 3 \text{ or } e_j = 4 \end{cases}$$

The objective (2.1) of the proposed TPP model is the minimization of the total weighted delays in which weight is associated with train priorities. Note that delay of a train can occur at the two station points: entry signal and platform.

$$Minz = \sum_{j=1} \alpha_j T_j \tag{2.1}$$

The proposed model includes following constraints:

(I) Calculation Constraints

These group of constraints (2.2)-(2.5) are formulated for calculating the values of decision variables. The actual departure time of a train from signals is expressed in constraints (2.2) and (2.3). For an in-route train, it is equal to sum of planned arrival time of the train to the entry signal and waiting time at this location. Dwell time, running times on switch area and waiting time at platform should be also included for calculation of actual departure time of a train from exit signal.

$$s_{jk} = p_j^a + w_j^s \qquad \forall j, k, \quad \beta_j^1 = k \tag{2.2}$$

$$s_{jk} = p_j^a + w_j^s + v_j + w_j^p + H_j \qquad \forall j, k, \quad \beta_j^2 = k \tag{2.3}$$

The constraint (2.4) is formulated for actual departure time of a train from platform. Based on this decision variable, delays of trains are calculated in constraint (2.5).

$$a_j^d = s_{jk} + w_j^p + v_j + \delta_j \qquad \forall j, k, \quad \beta_j^1 = k \tag{2.4}$$

$$T_j \geq a_j^d - p_j^d \qquad \forall j \tag{2.5}$$

(II) Assignment Constraints

Equations (2.6) and (2.7) are platform assignment constraints. Constraint (2.6) ensures that each train must be assigned to a platform track and this track should be met platform eligibility conditions.

$$\sum_l x_{jl} = 1 \qquad \forall j \tag{2.6}$$

$$x_{jl} \leq b_{jl} \qquad \forall j, l \tag{2.7}$$

(III) Conflict Resolution Constraints for Platform, Signal and Switch
To present route conflicts, we primarily focus on the station's resources where trains overlap and we determine three resources on the station: Platform, Signal and Switch.

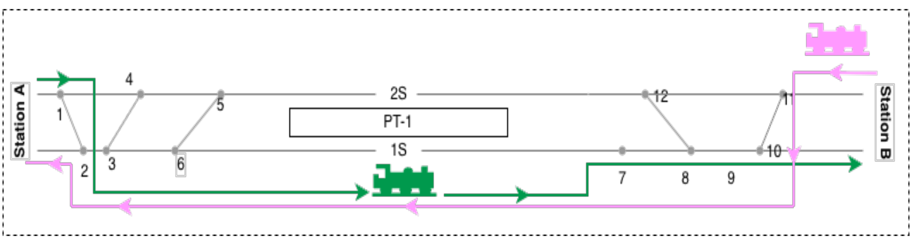

Figure 4. **Illustration of a platform conflicts**

$$a_j^d + t^p \leq s_{ik} + \delta_i + M(3 - x_{jl} - x_{il} - y_{ij}^p) \qquad \forall i, j, k, l \quad i \neq j, \quad \beta_i^1 = k \tag{2.8}$$

$$a_i^d + t^p \leq s_{jk} + \delta_j + M(2 - x_{jl} - x_{il} + y_{ij}^p) \qquad \forall i,j,k,l \quad i \neq j, \quad \beta_j^1 = k \tag{2.9}$$

$$y_{ij}^p + y_{ji}^p = 1 \qquad \forall i,j \quad i \neq j \tag{2.10}$$

In Equation (2.8)-(2.10), conflict resolution constraints for platform are expressed. A platform cannot be used by two trains at the same time, and this restriction is illustrated in Figure 4. In addition to this, there should be a minimum headway time for platform between trains. These constraints guarantee that platform usage time of trains which are assigned to same platform can not overlap.

$$s_{jk} - s_{ik} \geq \delta_i + t^s \qquad \forall i,j,k \quad i < j, \quad \beta_i^1 = k, \quad \beta_j^1 = k, \quad p_i^a \leq p_j^a \tag{2.11}$$

$$s_{jk} - s_{ik} \geq \delta_i + t^h - M(y_{ijk}^s) \qquad \forall i,j,k \quad i \neq j, \quad \beta_i^2 = k, \quad \beta_j^2 = k \tag{2.12}$$

$$s_{ik} - s_{jk} \geq \delta_j + t^h - M(1 - y_{ijk}^s) \ \forall i,j,k \qquad i \neq j, \qquad \beta_i^2 = k, \qquad \beta_j^2 = k \tag{2.13}$$

$$y_{ijk}^s + y_{jik}^s = 1 \qquad \forall i,j,k \quad i \neq j, \qquad \beta_i^2 = k, \qquad \beta_j^2 = k \tag{2.14}$$

One of the resources that cause route conflict is signal. Constraints (2.11)-(2.14) represent signal resolution constraints. If two consecutive trains enter to the station from signal k, first train departs from the signal based on first come first serve principle and this restriction is illustrated in Figure 5. Constraint(2.11) ensures that a signal cannot be used by two trains at the same time. Similarly, for the trains which are potentially overlapping in time at the same exit signal, we enforce separation of their signal usage time intervals with the help of Big-M method in Constraints (2.12) and (2.13). Note that, if train i and j are using two different signals, no time separation is enforced between them. Constraint (2.14) ensures that only one of the two trains arrives at the signal first.

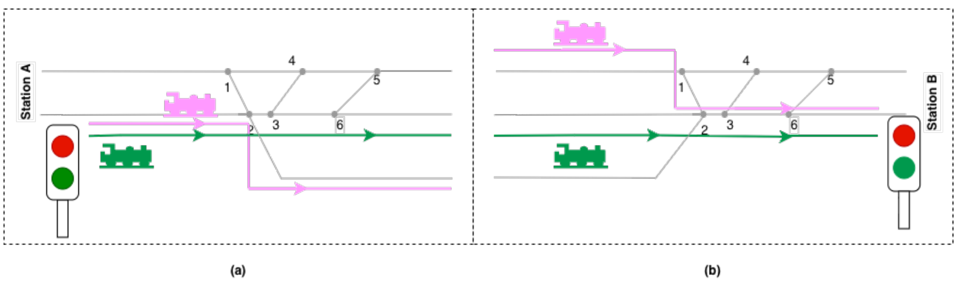

Figure 5. **Illustration of signal conflicts**

Similar to platform and signal constraints, we have to formulate conflict resolution constraints for switch to avoid that two trains using the same switch at any given time. These are given in the set of constraints (2.15) - (2.23). Unlike other conflict resolution constraints, we should consider in which route(In/Out) they encounter in switch area. So, separation of their switch usage time intervals are formulated based on trains' route: (i) In-route / In-route, (ii) In-route / Out-route, (iii) Out-route / Out-route. These conflicts are illustrated in Figure 6.

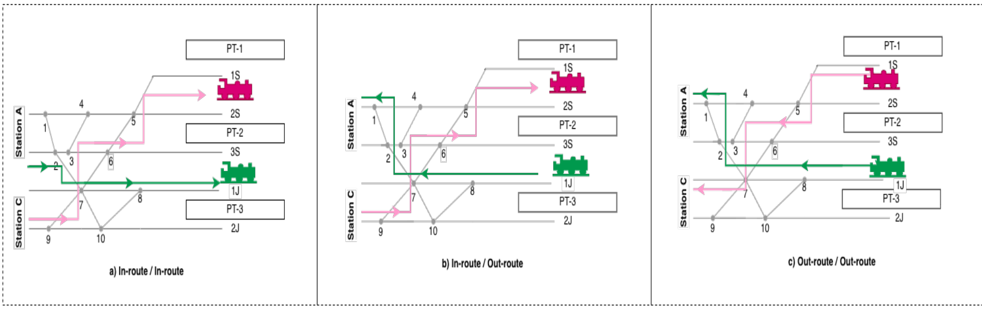

Figure 6. **Illustration of switch conflicts**

(i) In-route / In-route : For two in-route trains that arrive from different signals and passing a switch in common, we enforce separation of their switch usage time intervals by using Constraints (2.15) - (2.17).

$$s_{im} - s_{jk} \geq \delta_j + t^s - M(y_{ij}^{m'}) - M(2 - x_{jl} - x_{iq}) \quad \forall i, j, l, q, k, m \quad i \neq j,\ l \neq q,$$
$$k \neq m, \quad \beta_i^1 = m, \quad \beta_j^1 = k, \quad \sum_r \gamma_{jlr}^1 \gamma_{iqr}^1 \geq 1 \tag{2.15}$$

$$s_{jk} - s_{im} \geq \delta_i + t^s - M(1 - y_{ij}^{m'}) - M(2 - x_{jl} - x_{iq}) \quad \forall i, j, l, q, k, m \quad i \neq j,\ l \neq q,$$
$$k \neq m, \quad \beta_i^1 = m, \quad \beta_j^1 = k, \quad \sum_r \gamma_{jlr}^1 \gamma_{iqr}^1 \geq 1 \tag{2.16}$$

$$y_{ij}^{m'} + y_{ji}^{m'} = 1 \qquad \forall i, j \quad i \neq j \tag{2.17}$$

(ii) In-route / Out-route : To prevent conflict in switch area for an in-route train i and an out-route train j, constraints (2.18) - (2.21) are used. These are formulated based on track assignment of these trains. More specifically, if the trains are assigned to different platform tracks, constraints (2.18) and (2.19) guarantee conflict-free routes. The upper bound of sum of s_{jk} and t^s comes before the lower bound of s_{im} or upper bound of sum s_{im}, δ_i and t^s comes before the lower bound of s_{jk}. In case of assignment to same platform track, the outbound train j should leave firstly and after minimum headway time for platform, in-route train i moves towards the platform. The constraint (2.21) ensures that only one of the two trains arrives at the switch first.

$$s_{jk} - s_{im} + t^s \leq M(y_{ij}^{m''}) + M(2 - x_{jl} - x_{iq}) \quad \forall i,j,l,q,k,m \quad i \neq j, \quad l \neq q,$$
$$k \neq m, \quad \beta_j^2 = k, \quad \beta_i^1 = m, \quad \sum_r \gamma_{jlr}^2 \gamma_{iqr}^1 \geq 1 \tag{2.18}$$

$$\sum_{m=1,\beta_i^1=m} s_{im} + \delta_i + t^s \leq a_j^d + M(1 - y_{ij}^{m''}) + M(2 - x_{jl} - x_{iq}) \quad \forall i,j,l,q \quad i \neq j,$$
$$l \neq q, \quad \sum_r \gamma_{jlr}^2 \gamma_{iqr}^1 \geq 1 \tag{2.19}$$

$$s_{jk} - s_{im} + t^p \leq M(2 - x_{jl} - x_{il}) \quad \forall i,j,l,k,m \quad i \neq j, \quad m \neq k, \quad \beta_j^2 = k,$$
$$\beta_i^1 = m, \quad \sum_r \gamma_{jlr}^2 \gamma_{ilr}^1 \geq 1 \tag{2.20}$$

$$y_{ij}^{m''} + y_{ji}^{m''} = 1 \qquad \forall i,j \quad i \neq j \tag{2.21}$$

(iii) Out-route / Out-route : This is the case when two outbound trains depart from different signals and they use a switch in common. So, in constraints (2.22) and (2.23), we impose that their usage time intervals do not overlap. We also leave a minimum headway time for switch between each subsequent trains. The constraint (2.24) ensures that only one of the two trains arrives at the switch first.

$$\sum_{m=1,\beta_i^2=m} s_{im} + t^s \leq a_j^d + M(y_{ij}^{m'''}) + M(2 - x_{jl} - x_{iq}) \quad \forall i,j,l,q \quad i \neq j, \quad l \neq q,$$
$$\sum_r \gamma_{jlr}^2 \gamma_{iqr}^2 \geq 1 \tag{2.22}$$

$$\sum_{k=1,\beta_j^2=k} s_{jk} + t^s \leq a_i^d + M(1 - y_{ij}^{m'''}) + M(2 - x_{jl} - x_{iq}) \quad \forall i,j,l,q \quad i \neq j,$$
$$l \neq q, \quad \sum_r \gamma_{jlr}^2 \gamma_{iqr}^2 \geq 1 \tag{2.23}$$

$$y_{ij}^{m'''} + y_{ji}^{m'''} = 1 \qquad \forall i,j \quad i \neq j \tag{2.24}$$

Finally, the domains of the decision variables are given in group of constraints (2.25)-(2.29).

$$s_{jk} \geq 0 \qquad \forall j, k \tag{2.25}$$

$$T_j,\ a_j^d,\ w_j^s,\ w_j^p \geq 0 \qquad \forall j \tag{2.26}$$

$$x_{jl} \in \{0,1\} \qquad \forall j, l \tag{2.27}$$

$$y_{ijk}^s \in \{0,1\} \qquad \forall i, j, k \tag{2.28}$$

$$y_{ij}^p,\ y_{ij}^{m'},\ y_{ij}^{m''},\ y_{ij}^{m'''} \in \{0,1\} \qquad \forall i, j \tag{2.29}$$

3. Genetic algorithm

The genetic algorithm (GA), firstly proposed by [8], is a population-based meta-heuristic algorithm inspired by Charles Darwin's theory of natural evolution. GA has a powerful search capability and is comprehensively used in various railway optimization problems such as train platforming problem ([20]) and train timetabling problem ([12]).The procedure of GA involves generation of initial population, fitness calculation procedures, selection, crossover and mutation. A brief description of the our GA implementation procedure is presented in the following section.
Initial Population Procedure: GA maintains a set of feasible solutions, called population, where each potential solution is represented by a chromosome. Chromosomes of the our GA are an n bit string where n is equal to number of trains. All the genes of the chromosome have values between 1 and m (number of platform tracks). An example of the our chromosome encoding scheme is shown in Figure 7.

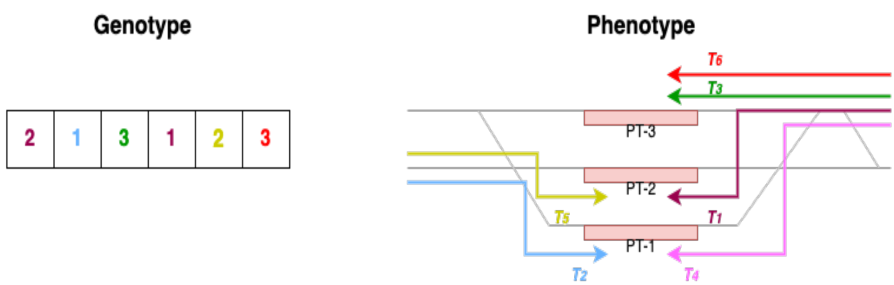

Figure 7. **Chromosome encoding scheme**

As can be seen from the left side of Figure 7, the chromosome denotes a platform track assignment plan for six trains in a small station with three tracks. For instance, the value of the 2nd gene is equal to 1. This means that train 2 is assigned to platform track 1. All the track assignments of trains based on this chromosome are shown in right side of Figure 7.
Initial population is generated by a purely random assignment of trains to the tracks. The solution representation guarantees that ensure assignment constraints in constraint (2.6). However, constraint (2.7) should be also assured to prevent infeasible solutions. Therefore, gene values are generated based on platform eligibility conditions of related train.
Fitness Function Procedure: The fitness value of a chromosome is defined as the objective function of the problem which is equal to total weighted delay in our TPP. The chromosome representation gives the value of the decision variable x_{jl}, which gives information about a particular train's platform track assignment. However, the values of the remaining decision variables are still unknown. For instance, train 2 and train 4 are assigned to platform track 1 in Figure 7 and we have no information on which train will use the platform track first. So, we determine first train based on first come first serve rule (FIFO) in case of route conflicts. Similar to conflict resolution constraints for platform, signal and switch (2.8)-(2.23), waiting times, delays of trains can be calculated based on this information. Therefore, the fitness function evaluates all potential switch, signal, and platform track conflicts, and assigns waiting times to these trains. This mechanism provides operationally feasible schedule.

Selection: The selection phase determines which chromosomes in the current generation will be chosen for reproduction offspring. One of the selection methods is tournament selection. This operator randomly selects k individuals from the population each time and chooses the one with the best fitness value. This continues until the number of selected chromosomes is equal to the population size. [21] proved that the tournament selection is more efficient than other methods in terms of convergence and computational complexity. In this paper, we adopted the widely used tournament selection strategy with tournament size k=4. In addition, elitism strategy is also used to prevent loss of the best chromosomes. The number of elitist chromosomes is equal to the top 5% of chromosomes in the population.

Crossover: Crossover combines the genetic information of two parents to generate new children. The proposed GA has a uniform crossover operator. In a uniform crossover, the chromosome does not divide into segments, we treat each gene separately based on a binary mask formed by 0/1 digits. In the mask, "0" means that the gene's donor is parent 1 for child 1. Otherwise, the donor is parent 2. A sample crossover is given in Figure 8. 2nd gene of the binary mask is equal to "1". In this case, child 1 gets the 2nd gene of parent 2 and gene of child 2 is copied from parent 1. Similarly, the 5th gene value of the binary mask is "0", child 2 gets the 5th gene of parent 2. As can be seen from the figure, only genes in the same point of each chromosome are changed, so eligibility constraints are still satisfied.

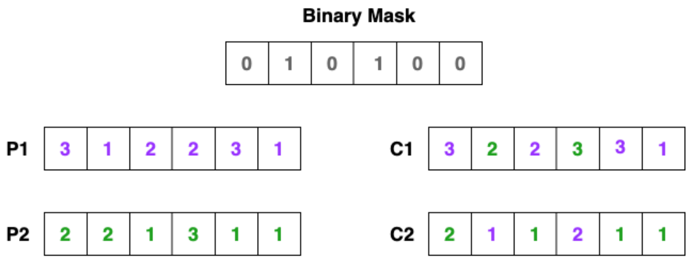

Figure 8. **A sample crossover**

Mutation: Selection and crossover produce different chromosomes. However, they may not be a sufficient variety of chromosomes. So, a mutation operator can help converge to the optimal solution. In our GA, a random number is generated for each gene. If the random number is smaller than the mutation rate, the value of the gene is generated based on the initial population generation mechanism so that platform eligibility conditions are considered. The value of the gene does not change if the random number is greater than the mutation rate.

4. Computational results

In this section, we present the results of the proposed mathematical model and GA for derived test instances based on a railway station in Europe. The proposed MILP model is solved with GAMS/CPLEX solver and the proposed GA is coded in Go programming language. All instances in this paper are tested on a personal Windows computer with i7-3610QM @ 2.3GHz CPU and 8.0GB RAM.

4.1. Test problems

Station layout is an essential part of TPP. To derive test instances for train platforming problem, we use a virtual station structure considering busy and complex railway station in Europe. It is comprised of 9 platforms and 16 platform tracks. Three of them are unidirectional and can not be accessed by both sides of the station. The rest of them are bidirectional and can be occupied by all trains. There are 5 main lines within station and equipped with double track. The platform tracks and lines are connected with more than 100 railway switches.

Test problems are constructed as small size (S), medium size (M) and large size (L) based on planning period. t refers to the length of planning period. For S, M and L size test instances, t values are 1 hour (60 min), 8 hours (480 min) and 24 hours (1440 min), respectively. We assumed that each size test problem has three traffic volume level (t^v) with different number of trains : low (t^v=1), medium (t^v=2) and heavy(t^v=3). In Table 2, planning periods, traffic volume levels and range for number of trains (n^t) in each level are summarised. The number of trains follows the uniform distribution and of course, it is an integer. For instance, small size problems with heavy traffic volume uniformly distributed between 40 and 50 trains.

Table 2. **Parameters of problem sizes**

Problem Size	t	t^v	Range for n^t
		Low	U[10,20]
Small	60 min	Normal	U[25,35]
		Heavy	U[40,50]
		Low	U[95,175]
Medium	480 min	Normal	U[200,280]
		Heavy	U[305,385]
		Low	U[390,630]
Large	1440 min	Normal	U[600,840]
		Heavy	U[810,1050]

The planned arrival time of trains to the entry signal (p^a_j) is also determined based on length of planning period and traffic volume level. It is calculated as follows and rnd is a uniform random number between 0 and 1.

$$p^a_j = \begin{cases} p^a_1 = 0, \quad p^a_{j+1} = p^a_j + \frac{t^v}{n^t}, & t^v = 1 \\ p^a_1 = 0, \quad p^a_{j+1} = p^a_j + (rnd\frac{t^v}{n^t/2}), & t^v = 2 \quad or \quad t^v = 3 \end{cases}$$

In the test problems, trains are categorized as three groups: international trains (type I), regional trains (type II) and local trains (type III). We assumed that % 20 of number of trains are belongs to type I, % 50 of them are type II and the rest of them are type III. The some of the related parameters to the trains are generated depending on train types. These are given in Table 3:

Table 3. **Related parameters with train type**

	Type I	Type II	Type III
	%94, v_j=10		%70, $v_j \sim$ U[2,5]
v_j	%3, $v_j \sim$ U[20,25]	$v_j \sim$ U[7,11]	%10, $v_j \sim$ U[7,8]
	%3, $v_j \sim$ U[50,100]		%20, $v_j \sim$ U[10,20]
α_j	3	2	1
h^l_j	1	1	1
h^r_j	1	1	1

The minimum headway parameter for signal is set to two minutes, while minimum headway parameter for switch and platform are assumed to be one minutes. The planning departure time of trains (p_j^d) are calculated as sum of p_j^a, δ_j and v_j.

4.2. Test results

Randomly generated test problems are solved with both the proposed mathematical model and GA with a time limit of 7200 seconds. GA was performed with 10 repetitions. The results of small-sized problems are given in Table 4, the results of medium-sized problems are given in Table 5, and the results of large-sized problems are given in Table 6.

Table 4. **Results of small-sized problems**

Problem	n	GAMS			GA				Imp.%
		z	t	$rgap$	z_{min}	z_{mean}	σ	t_{mean}	
S-1-1	11	1	0.31	0.00	1	1.00	0.00	6	0
S-1-2	14	0	1.06	0.00	0	0.00	0.00	7	0
S-1-3	12	0	0.44	0.00	0	0.00	0.00	7	0
S-1-4	16	0	0.80	0.00	0	0.00	0.00	8	0
S-1-5	19	2	1.33	0.00	2	2.00	0.00	9	0
S-2-1	26	19	8.11	0.00	19	19.00	0.00	12	0
S-2-2	33	33	46.03	0.00	41	41.00	0.00	14	-20
S-2-3	30	10	56.88	0.00	12	14.67	4.62	14	-17
S-2-4	31	29	269.39	0.00	29	36.33	9.45	18	0
S-2-5	26	5	44.52	0.00	6	6.00	0.00	12	-17
S-3-1	41	70	7200.00	0.36	46	50.67	4.16	301	52
S-3-2	48	221	7200.00	0.80	99	110.67	10.12	400	123
S-3-3	46	56	7200.00	0.16	59	63.67	4.16	351	-5
S-3-4	42	99	7200.00	0.28	75	78.33	3.06	334	32
S-3-5	43	62	7200.00	0.44	37	43.33	5.51	319	68
			2428.59					14.47	14.4

Table 4 consists of four parts. In the first part, the names of the test problems and number of trains(n), in the second part the details of the results obtained with GAMS, in the third part the results obtained with GA, and in the last part, the percent improvement values (imp.%) are given. The GAMS part consists of three columns. The objective function value (z) is given in the first column, the solution time (t) in seconds in the second column, and the relative gap ($rgap$) in the last column. The $rgap$ value shows the objective function value how close the optimum is as a percentage, and if it is equal to zero, it means that the optimum has been reached. The GA part consists of four columns. In the first column, the most successful of the ten solutions of a test problem obtained by GA (z_{min}) is given. The average of these values (z_{mean}) in the second column, the standard deviation in the third column, and the average of solution time in seconds (t_{mean}) in the last column are given. imp.% value, given in the last part of the table, shows how successful the GA results are compared to GAMS in percentage. If this value is negative, it means GAMS, if it is positive, it means that GA finds a more successful solution. Percent improvement values have been calculated using the formula in Equation (4.1).

$$imp.\% = ((z - z_{min})100)/z_{min} \tag{4.1}$$

As can be seen from the *rgap* column of Table 4, the optimum solutions of 10 out of 15 small-sized test problems and feasible solutions for the others have been obtained within the time limit by GAMS/Cplex. Although GA lagged behind GAMS for a few tests, it achieved significant success both by finding optimal solutions to 7 problems, their optimal solutions are known, and by improving GAMS results by an average of 14.47%. The fact that the mean objective function values of the GA are close to the minimum value and the small standard deviation shows that the algorithm can reach successful solutions for small-sized problems with less repetition. Moreover, the average solution time of GA is shorter than GAMS.

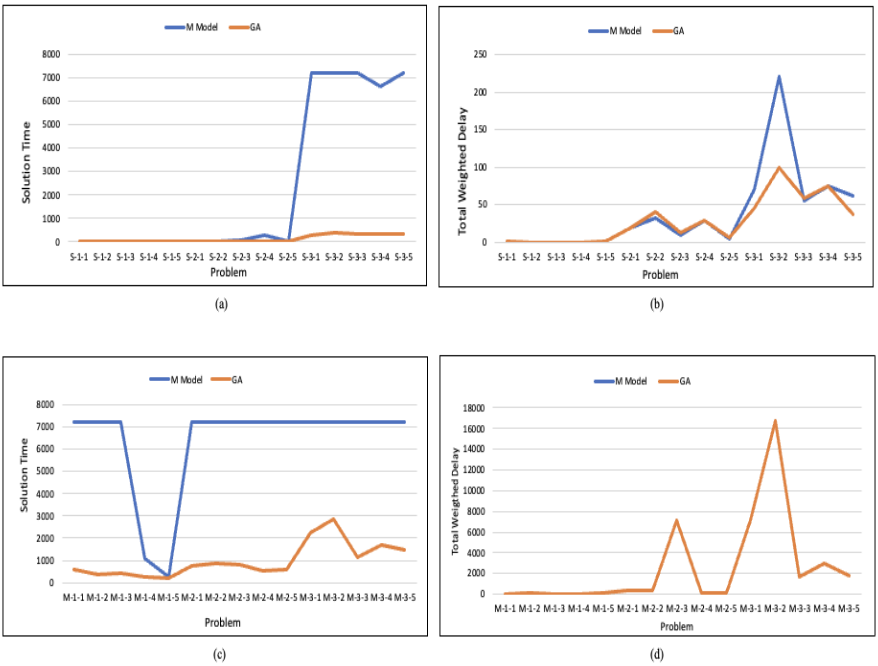

Figure 9. **Comparison of M Model and GA, (a) Comparison of Running Speeds for Small-sized Problems, (b) Comparison of Obj. Value for Small-sized Problems, (c) Comparison of Running Speeds for Medium-sized Problems, (d) Comparison of Obj. Value for Medium-sized Problems.**

Although the structure of Table 5 is similar to Table 4, since a significant part of the problems could not be solved with GAMS/CPLEX, the last part containing the improvement values is not included in it. Table 5 reveals that GAMS/Cplex was unable to find feasible solutions for all but 2 out of the 15 medium-sized test problems within the given time limit. These solutions are optimum, as indicated by the *rgap* column. On the other hand, GA was able to produce optimal solutions for the same

2 problems, as well as feasible solutions for all the others. Additionally, GA had a faster average solution time compared to GAMS. In order to better demonstrate the effectiveness of GA in terms of both solution time and quality, Figure 4.2 includes graphs displaying the results for the small and medium-sized problems.

Table 5. Results of medium-sized problems

Problem	n	GAMS			GA			
		z	t	$rgap$	z_{min}	z_{mean}	σ	t_{mean}
M-1-1	165	-	7200	-	28	30.00	2.65	572
M-1-2	130	-	7200	-	75	76.67	2.08	347
M-1-3	159	-	7200	-	40	40.00	0.00	433
M-1-4	114	4	1098	0	4	4.00	0.00	253
M-1-5	105	74	285	0	74	74.00	0.00	195
M-2-1	247	-	7200	-	305	364.00	57.11	752
M-2-2	270	-	7200	-	364	391.33	35.16	850
M-2-3	254	-	7200	-	7160	7167.33	6.43	808
M-2-4	204	-	7200	-	169	171.67	3.06	539
M-2-5	213	-	7200	-	181	182.00	1.00	572
M-3-1	371	-	7200	-	7140	7864.67	721.03	2268
M-3-2	381	-	7200	-	16693	18200.67	1399.04	2841
M-3-3	309	-	7200	-	1724	1857.67	162.82	1147
M-3-4	346	-	7200	-	2942	3440.00	464.68	1712
M-3-5	331	-	7200	-	1812	2460.00	569.00	1484

In TPP, simply the number of trains is not enough to accurately assess the effectiveness of a proposed solution approach. What really matters is the number of trains per unit of time, or the traffic density, which can greatly impact the complexity and feasibility of the problem. For example, while GAMS/Cplex cannot reach optimal solutions in 7200 time limit for small-sized test problems with heavy traffic density (S-3), optimum solutions can be obtained for medium-sized test problems with low traffic density (M-1-4,M-1-5) despite having more trains.

The test problems in Table 4 and Table 5 were divided into two groups: those for which the optimal solution was obtained within the time limit, and those for which only a feasible solution was obtained. For both groups, paired t-tests were conducted using the results of the GA and the mathematical model. For the problems that only feasible solutions could be obtained, the paired t-test indicates a statistically significant difference between GA and the mathematical model ($t = -3.316$, $p = 0.029$), suggesting that the GA provides notably better performance on the larger and more complex instances. In contrast, for the problems that the optimal solution is known, the paired t-test results ($t = 1.372$, $p = 0.197$) show no statistically significant difference between GA and the mathematical model, as both methods produced identical or very similar results on most instances.

In Table 6, the results of large-sized problems are presented. GAMS/Cplex was unable to solve any of the large-sized problems within the time limit, so its results are not included. However, GA was able to produce feasible solutions for all of the test problems within a reasonable amount of time, as shown in the table.The large-

Table 6. Results of large-sized problems

Problem	n	GA			
		z_{min}	z_{mean}	σ	t_{mean}
L-1-1	412	140	142.33	2.08	290
L-1-2	403	1634	1684.67	49.08	299
L-1-3	504	555	579.00	25.63	388
L-1-4	478	119	121.33	4.04	371
L-1-5	619	1413	1513.33	103.65	646
L-2-1	818	8429	9815.33	1355.62	1249
L-2-2	633	1131	1148.00	19.97	677
L-2-3	702	3029	3125.00	134.71	822
L-2-4	757	5032	5351.67	334.92	929
L-2-5	643	1227	1316.33	107.75	671
L-3-1	834	8239	8726.33	490.02	1076
L-3-2	895	16716	18916.00	2685.08	1625
L-3-3	894	23654	28353.00	5022.41	1984
L-3-4	1047	89501	91079.67	1536.41	4062
L-3-5	840	14281	14496.00	216.52	1389

sized problems used in our experiments, containing 403 to 1,047 trains, reflects traffic volumes observed in major European and Asian railway stations, where traffic demand fluctuates significantly depending on the time of day and season. By successfully generating feasible solutions in a reasonable time across all these traffic levels, it can be said that the proposed GA is suitable for real-world railway planning problems.
In summary, tests have shown that GA can successfully solve problems that GAMS/C-plex fails. This success has been demonstrated by not only finding feasible solutions for all test problems, but also having a reasonable standard deviation and solution time.

5. Conclusion

In this paper, we have addressed the train platforming problem in a multi-direction station under a flexible track utilization policy. A mixed-integer linear programming (MILP) formulation was proposed to assign trains to the station's resources without conflicting routes. The objective function aimed to minimize total weighted delays of trains, considering the importance level of each train. Due to the problem being NP-hard, a genetic algorithm (GA) was also proposed to obtain near-optimal solutions within a reasonable time frame.
Our computational analyses were conducted on test problems generated by considering traffic volume levels and planning periods, which distinguishes this study from previous works. The performance of the mathematical model and the genetic algorithm was assessed in terms of both solution quality and solution time.
Based on our findings, the following conclusions were reached: (1) The MILP formulation is effective in solving the train platforming problem under a flexible track utilization policy. It provides optimal solutions for small and medium-sized instances.

However, for larger instances, the computational time required to obtain an optimal solution may be prohibitively long. (2) The genetic algorithm can provide near-optimal solutions within a relatively short time for larger instances. It is a suitable alternative when computational time is a critical factor, especially for real-time or time-sensitive applications. (3) The flexible track utilization policy offers more efficient platforming and potential improvements in train punctuality. Nevertheless, the increased complexity of the problem should be considered when implementing this policy in practice.
Future studies could focus on refining the genetic algorithm by incorporating advanced operators, improving the population diversity, and exploring hybrid algorithms combining MILP and GA approaches for more efficient problem-solving. It would also be beneficial to investigate the impact of various traffic volume levels and planning periods on the performance of the proposed models to provide better insights into real-world applications. By addressing the train platforming problem under a flexible track utilization policy, this study contributes to the ongoing efforts to improve railway operations and enhance the overall efficiency of transportation systems.

Acknowledgments: This work is supported by Eskişehir Technical University Scientific Research Project Commission under grant 19ADP006. The authors are deeply grateful for the financial support from Eskişehir Technical University. We also would like to thank Halilcan Ugur for his generous support to increase the quality and clarity of this paper significantly.

Conflict of Interest: The authors declare that they have no conflict of interest.

Data Availability: Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

References

[1] A. Billionnet, *Using integer programming to solve the train-platforming problem*, Transportation Science **37** (2003), no. 2, 213–222. https://doi.org/10.1287/trsc.37.2.213.15250.

[2] A. Caprara, L. Galli, and P. Toth, *Solution of the train platforming problem*, Transportation Science **45** (2011), no. 2, 246–257. https://doi.org/10.1287/trsc.1100.0366.

[3] D.D.L. Cardillo and N. Mione, *k L-list λ colouring of graphs*, European J. Oper. Res. **106** (1998), no. 1, 160–164.

[4] M. Carey and S. Carville, *Scheduling and platforming trains at busy complex stations*, Transportation Research Part A: Policy and Practice **37** (2003), no. 3, 195–224. https://doi.org/10.1016/S0965-8564(02)00012-5.

[5] F. Corman, R.M. Goverde, and A. D'Ariano, *Rescheduling dense train traffic over complex station interlocking areas*, Robust and online large-scale optimization: models and techniques for transportation systems **5868** (2009), 369–386. https://doi.org/10.1007/978-3-642-05465-5_16.

[6] Y. Gao, L. Kroon, L. Yang, and Z. Gao, *Three-stage optimization method for the problem of scheduling additional trains on a high-speed rail corridor*, Omega **80** (2018), 175–191. https://doi.org/10.1016/j.omega.2017.08.018.

[7] R. García-Ródenas, M.L. López-García, L. Cadarso, and E. Codina, *An efficient greedy heuristic for the real-time train platforming problem*, Computers and Operations Research **164** (2024), 106525. https://doi.org/10.1016/j.cor.2023.106525.

[8] J.H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*, MIT press, 1992.

[9] R.M. Lusby, J. Larsen, M. Ehrgott, and D. Ryan, *Railway track allocation: models and methods*, OR spectrum **33** (2011), no. 4, 843–883. https://doi.org/10.1007/s00291-009-0189-0.

[10] J. Rodriguez, *A constraint programming model for real-time train scheduling at junctions*, Transportation Research Part B: Methodological **41** (2007), no. 2, 231–245. https://doi.org/10.1016/j.trb.2006.02.006.

[11] P. Sels, P. Vansteenwegen, T. Dewilde, D. Cattrysse, B. Waquet, and A. Joubert, *The train platforming problem: The infrastructure management company perspective*, Transportation Research Part B: Methodological **61** (2014), 55–72. https://doi.org/10.1016/j.trb.2014.01.004.

[12] Z. Yao, L. Nie, and Z. He, *A genetic algorithm for heterogeneous high-speed railway timetabling with dense traffic: The train-sequence matrix encoding scheme*, Journal of Rail Transport Planning and Management **23** (2022), 100334. https://doi.org/10.1016/j.jrtpm.2022.100334.

[13] B. Zhang, Y. Zhang, A. D'Ariano, T. Bosi, G. Lu, and Q. Peng, *Optimal platforming, routing, and scheduling of trains and locomotives in a rail passenger station yard*, Transportation Research Part C: Emerging Technologies **152** (2023), 104160. https://doi.org/10.1016/j.trc.2023.104160.

[14] H. Zhang, M. Guo, L. Hu, and G. Lu, *A multi-agent simulation based train platforming research for facilitating passenger transfer in a high-speed railway station*, Simulation Modelling Practice and Theory **130** (2024), 102856. https://doi.org/10.1016/j.simpat.2023.102856.

[15] Q. Zhang, R.M. Lusby, P. Shang, and X. Zhu, *Simultaneously re-optimizing timetables and platform schedules under planned track maintenance for a high-speed railway network*, Transportation Research Part C: Emerging Technologies **121** (2020), 102823. https://doi.org/10.1016/j.trc.2020.102823.

[16] Q. Zhang, X. Zhu, and L. Wang, *Track allocation optimization in high-speed railway stations from infrastructure management and service perspectives*, Measurement and Control **51** (2018), no. 7-8, 243–259.
https://doi.org/10.1177/0020294018785187.

[17] ———, *Track allocation optimization in multi-direction high-speed railway stations*, Symmetry **11** (2019), no. 4, 459.
https://doi.org/10.3390/sym11040459.

[18] Q. Zhang, X. Zhu, L. Wang, and S. Wang, *Simultaneous optimization of train timetabling and platforming problems for high-speed multiline railway network*, Journal of Advanced Transportation **2021** (2021), no. 1, 1–16.
https://doi.org/10.1155/2021/6679008.

[19] Y. Zhang, R. Hu, Q. Zeng, Y. Wang, Y. Liu, and S. Huang, *Optimal train platforming with shunting operations for multidirectional passenger stations: A case study of guangzhou station*, Mathematics **11** (2023), no. 14, Article number: 3136.
https://doi.org/10.3390/math11143136.

[20] Y. Zhang, Q. Zhong, Y. Yin, X. Yan, and Q. Peng, *A fast approach for reoptimization of railway train platforming in case of train delays*, Journal of Advanced Transportation **2020**, no. 1, 560952.
https://doi.org/10.1155/2020/5609524.

[21] J. Zhong, X. Hu, J. Zhang, and M. Gu, *Comparison of performance between different selection strategies on simple genetic algorithms*, Proc. Int. Conf. on Computational Intelligence for Modelling, Control and Automation (CIMCA-IAWTIC'06), IEEE, 2005, pp. 1115–1121.

[22] P.J. Zwaneveld, L.G. Kroon, H.E. Romeijn, M. Salomon, S. Dauzere-Peres, S.P. Van Hoesel, and H.W. Ambergen, *Routing trains through railway stations: Model formulation and algorithms*, Transportation Science **30** (1996), no. 3, 181–194.
https://doi.org/10.1287/trsc.30.3.181.

[23] P.J. Zwaneveld, L.G. Kroon, and S.P. Van Hoesel, *Routing trains through a railway station based on a node packing model*, European J. Oper. Res. **128** (2001), no. 1, 14–33.
https://doi.org/10.1016/S0377-2217(00)00087-4.