applied sciences

MDPI

Article

Real-Time Path Planning of Driverless Mining Trains with Time-Dependent Physical Constraints

Xiaojiang Ren *, Hui Guo, Kai Sheng and Guoqiang Mao

Guangzhou Institute of Technology, Xidian University, Guangzhou 510555, China
* Correspondence: xjren@xidian.edu.cn

Abstract: While the increased automation levels of production and operation equipment have led to the improved productivity of mining activity in open pit mines, the capacity of the mine transport system has become a bottleneck. The optimisation of the mine transport system is of great practical significance to reduce the production and operation cost and improve the production and organizational efficiency of mines. In this paper, we first formulate a multiobjective optimisation problem for mine railway scheduling by introducing a set of mathematical constraints. As the problem is NP-hard, we then devise a mixed integer programming-based solution to solve this problem and develop an online framework accordingly. We finally conduct test cases to evaluate the performance of the proposed solution. Experimental results demonstrate that the proposed solution is efficient and able to generate a train schedule in a real-time manner.

Keywords: mine railway; train scheduling; mixed integer programming; intelligent transportation system

check for updates

Citation: Ren, X.; Guo, H.; Sheng, K.; Mao, G. Real-Time Path Planning of Driverless Mining Trains with Time-Dependent Physical Constraints. *Appl. Sci.* **2023**, *13*, 3729. https://doi.org/10.3390/app13063729

Academic Editor: Yosoon Choi

Received: 11 February 2023
Revised: 6 March 2023
Accepted: 12 March 2023
Published: 15 March 2023

1. Introduction

Open pit mines have a large, complex system, and the optimisation of their railroad transportation system is an important part of it. The optimisation of the railroad transportation system can play a crucial role in reducing production and operation costs, as well as in improving the production and organisational efficiency of mines. In recent years, the mining industry has been continuously improving its production and operation equipment, leading to increased automation levels and improved productivity. However, despite these advancements, the bottleneck problems of the mine transport system have started to emerge, particularly with regard to the traditional routing optimisation planning models and algorithms that rely on a static road network analysis method.

Much research work, including mixed integer programming and dynamic programming, has been conducted to solve urban railroad transportation optimisation problems in the literature [1–6]. Compared with urban railway networks, mine railway networks have a limited amount of tracks, most of which are bidirectional single tracks. This leads to some particular problems, as follows.

- Track allocation: Given most tracks are bidirectional single tracks, the transportation capacity is limited by the number of receiving and departure tracks, the number of siding/meetpoints, and the length of single tracks. Moreover, the capacity is also restricted by track maintenance.
- Train conflict: For a bidirectional single-track, there are mainly two conflicts between trains. First, a predefined time difference is necessary to ensure safe operation when trains running in the same direction. Secondly, trains in opposite directions cannot be in the same segment at any time.

Given the problems stated above, the mine railroad scheduling problem is a complex optimisation problem, and the results under different decisions are far from each other. With the purpose of the less running time for all trains and maximizing the total amount of rolling stock, this paper proposes an integrated model by which to optimize train timetable and track allocation. The main contributions of this paper are listed as follows.

- We first formulate a multiobjective optimisation problem for mine railway scheduling by introducing a set of mathematical constraints.
- As the problem is NP-hard, we then devise a MIP-based solution to solve this problem in a real-time manner.
- We finally conduct test cases to demonstrate the validity and effectiveness of the solution.

In the following sections, Section 2 presents a review on the railway scheduling and train timetabling problems. Section 3 introduces the system model and describes the mining railroad optimisation problem. Section 4 provides the mathematical formulation for the objective and the set of constraints. Section 5 presents a description of all the modules that constitute the developed solution, in order for all the readers to get an understanding of the entire process. Section 6 presents a summary of the result for the tests carried out to evaluate the proposed solution. Section 7 is dedicated to the conclusions.

2. Related Works

An enormous number of studies have been conducted on railway scheduling and train timetabling problems [7–12]. The research work can be classified into three main categories: the train scheduling and rescheduling problem, the periodic and nonperiodic timetabling problem, and the passenger train and freight train timetabling problem [13].

Train scheduling is an offline problem that determines the arrival and departure times for trains at each station before the schedule is executed, e.g., [14–18]. With a planned schedule, rescheduling is a real-time problem that aims to determine detailed train movements and timetables to minimize train deviations, e.g., [19–22]. Sanat et al. [17] studied a train scheduling problem in a large national railway network, and presented two flexible heuristics based on a mixed integer program formulation for local optimisation to improve infrastructure utilization. Wang et al. [18] studied the integration of train scheduling and rolling stock circulation planning under time-varying passenger demand for an urban rail transit line and proposed three approaches to solve the resulting multiobjective mixed-integer nonlinear programming problem to deliver both an irregular train schedule (i.e., departure and arrival times of all train services) and a rolling stock circulation plan (including entering/exiting depot operations of rolling stocks and connections between train services) simultaneously. Bersani et al. [14] formalized demand-oriented scheduling and rescheduling models in order to propose a dynamic timetable, and proposed a min–max method with which to address operational constraints related to train capacity, train speed limits, train transfers, possible conflict in the track section use, with the main objective to minimize the travel time. Wang et al. [20] proposed a train rescheduling optimisation model in the case of the vehicle breakdown on a metro line. Efficient rescheduling strategies including flexible short-turning and adding backup trains in particular are formulated into the model. Zhu and Goverde [22] proposed a timetable rescheduling model to handle unexpected disruptions, where flexible stopping (i.e. skipping stops and adding stops) and flexible short-turning (i.e. full choice of short-turn stations) are innovatively integrated with three other dispatching measures: retiming, reordering, and cancelling.

Periodic timetabling requires that most or all train paths repeat in time with a certain period (e.g., 12 h), e.g., [23–27]. However, as it becomes difficult to obtain effective periodic schedules when dealing with interruptions or conflicts (e.g., track maintenance), a nonperiodic timetable becomes more appropriate, e.g., [28,29]. Huang et al. [23] integrated stop planning, service planning, and scheduling in a periodic timetabling problem and modelled it as a mixed integer linear programming formulation to minimize the average travel delay of passengers. They then developed a genetic algorithm supported by a scheduling heuristic to solve the problem for better scalability and efficiency. Polinder et al. [24] considered a robust periodic timetabling problem—the problem of designing a periodic timetable that can easily be adjusted in case of small periodic disturbances—and developed a solution method for a parameterised class of uncertainty regions. Sparing et al. [25]

used the minimum cycle time of the periodic timetable as an indicator for stability, and defined an optimization problem with this minimum cycle time as the objective function to be minimized. They then proposed an optimization method to find a feasible periodic timetable that also ensures maximum stability for heterogeneous railway networks, which is capable of handling flexible train orders, running and dwell times, and overtaking locations. Schiewe et al. [26] studied a periodic timetabling problem in public transportation planning, and developed an exact preprocessing method for reducing the problem size and a heuristic reduction approach in which only a subset of the passengers is considered. It provides upper and lower bounds on the objective value, such that it can be adjusted with respect to quality and computation time.

Because railways provide both passenger and freight services, there are naturally passenger train and freight train timetabling problems, e.g., [30–36]. Mu and Dessouky [33] introduced two mathematical formulations to cope with the rapidly increasing freight demand for railway transportation, and presented several heuristics that can significantly reduce the solution time of the exact method carried out by CPLEX, yet produce a satisfactory solution quality. Bešinović et al. [31] introduced the integrated passenger and freight train timetable adjustment problem which handles both passenger as well as freight trains and developed a mixed integer linear programming model to simultaneously retime, reroute, and cancel trains in the network. Sato et al. [35] studied freight train locomotive rescheduling when dealing with a disrupted situation in the daily operations in Japan, and solved the problem by changing the assignment of the locomotives to all the trains and considering their periodic inspections. Ozturk et al. [34] considered a new concept for freight transport within the borders of a city with an urban rail transit system, and presented a decision support framework for the problem of urban freight movement by rail together with mathematical methods for the optimal distribution of goods. Given a transportation network with fixed routes for passenger trains and a set of freight trains (requests), Borndörfer et al. [30] addressed a freight train routing problem, and calculated a feasible route for each freight train, such that the sum of all expected delays and all running times is minimal. Ursavas and Zhu [36] studied decision-making problems that railway infrastructure managers face in a rail network with dedicated tracks and shared-use corridors, and analyzed the consolidation strategy for shared-use corridors, where the track serves passenger and freight.

These research lines are viewed from different perspectives but are not truly independent of each other. For example, passenger train scheduling is usually a periodic timetabling problem. Li et al. [13] studied a nonperiodic freight train scheduling problem, in which a schedule is planned for freight trains and can be different during different periods of the day. The proposed method considers car flow transfer between consecutive trains and the shipment delivery time requirements. A tabu search algorithm is developed that extends the applicability of the proposed optimisation method for large-scale problems. Experiments on real-world instances of the Menghua railway indicate the effectiveness of the proposed algorithm.

3. Problem Description

Figure 1 shows a sample layout of mining railway network, where trains start from station A, travel via the tracks, load mine from load-out G, and return back to station A.

The mining railroad optimisation problem is to determine the best feasible timetable for a set of trains in order to load as many mines as possible. The related constraints are to satisfy restrictive operational constraints (e.g., track capacity, travel speed, safe distance, etc.) and to avoid possible conflicts when using each single track.

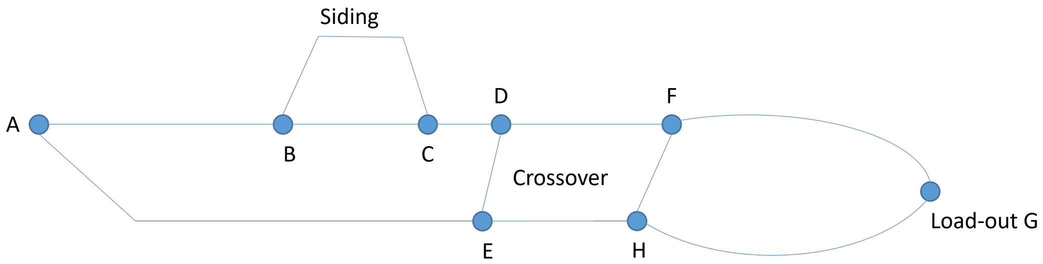

Figure 1. Railway network sample.

3.1. System Model

We define a physical network $G^* = (S, W)$ with a set of nodes S and a set of links W. The set S consists of switch stations ($S_u \subset S$), and load-outs ($S_l \subset S$) in the physical network. $W = \{(i, j)\}$ is the set of links in physical network, where (i, j) stands for link connecting node $i \in S$ to node $j \in S$. The set W consists of mainline tracks ($W_u \subset W$), siding tracks ($W_w \subset W$) and crossovers ($W_o \subset W$) in the physical network. For each link (i, j), let

- d_{ij} be the capacity of (i, j) where capacity is the maximum number of trains that can stand on the link (i, j) at any point of time;
- $f(i, j)$ be the travel time from i to j, where f is the speed profile-based function; and
- $f(j, i)$ be the travel time from j to i, where f is the speed profile-based function.

In order to model the business operation of mine loading, for each load-out $i \in S_l$, let P_i be the average loading time for a train.

Let $Q = \{0, 1, \cdots, |Q| - 1\}$ be the set of time instants, where time is discretized into discrete time instants of length g minutes. For instance, if we take g = 5 then a period of 1 h would be represented by discrete set $Q = \{0, 1, 2, \cdots, 12\}$ in our model.

3.2. Train Model

Let $M = \{1, 2, \cdots, m\}$ be the set of real trains travelling in the physical network G^*. For each train $m \in M$, we will get the following set of inputs.

- $loSeq^m$: Sequence of scheduled load-outs to visit;
- $depTime^m$: Scheduled departure time; and
- $depLoc^m$: Scheduled departure location.

As we have a cyclical network and the train changes direction (turns back) after going to a load-out, in order to model this behaviour we will break down the train journey into different parts. Each part will be represented by a different model train. Thus, if a train m goes to n load-outs, its whole journey will be represented by a set of $n + 1$ model trains, called T_m, where each model train represents a specific segment of train m's journey. Accordingly, one business rule will be added so that any model train can depart only after its predecessor has finished its journey.

For example, considering the case of a train m starting from station A and going to two load-outs G1 and G2, its journey will be modelled as follows in Table 1.

Table 1. Model train example.

Model Train	Departure Node	Destination Node	Predecessor Train
t_{m1}	A	G1	null
t_{m2}	G1	G2	t_{m1}
t_{m3}	G2	A	t_{m2}

From now on, we will use the term "train" to refer to model train only, unless specified otherwise. Thus, for each train $t \in T$, we will have the following set of information.

- $depQ_t$: Scheduled departure time;
- dep_t: Scheduled departure location node;
- $dest_t$: Final destination node of the train, after which it will disappear from the network; and
- t_{pre}: Predecessor train of train t.

3.3. Time–Space Network

We here consider a time–space network $G = (N, A)$, where N denotes the node set and A denotes the arc set. For each node $i \in N$, let $Outb_i \subseteq A$ be the set of all the outbound arcs of node i, and $Inb_i \subseteq A$ be the set of all the inbound arcs of node i.

Given a physical network G^*, we then can construct the time–space network G as follows:

- For each load-out in $s \in S_l$, we replace it with two separate node s_{in} (entry node) and s_{out} (exit node), and add a load-out link (s_{in}, s_{out}) into W. Let $W_l \subset W$ be the set of load-out links. By splitting the entry and exit point nodes for each load-out, we ensure that trains are staying at load-out for the required loading time. At the same time, we replace the related inbound links $(i, s) \in W$ to (i, s_{in}) and the related outbound links $(s, j) \in W$ to (s_{out}, j). Accordingly, we have $f(i, s_{in}) = f(i, s)$, $f(j, s_{out}) = f(j, s)$, and $f(s_{in}, s_{out}) = P_s$, where P_s indicates the required loading time.
- For each link $(i, j) \in W$ with capacity $d_{ij} > 1$, we break down the link into smaller links by adding $\lfloor d_{ij} - 1 \rfloor$ dummy nodes into S and replacing the link (i, j) with $\lfloor d_{ij} - 1 \rfloor$ dummy links (with proportional travel time) as well. Note that after this operation, the capacity of each arc is 1. This operation guarantees that on every arc at any point of time only one train travels on that arc, which will in turn guarantee that trains maintain a safe headway separation while travelling in the same direction.
- For each siding track $(i, j) \in W_w$, we add a siding node i' to S and replace the siding track (i, j) with two separate arcs (i, i') and (i', j). Let $S_w \subset S$ be the set of siding nodes. Accordingly, we have $f(i, i') = f(i', j) = f(i, j)/2$ and $f(j, i') = f(i', i) = f(j, i)/2$. For each link $(i, j) \in W$, let $Iden(i, j)$ be the set of identical arcs in A.
- For each node $s \in S$, we add $|Q|$ corresponding nodes $\{s^0, \ldots, s^{|Q|-1}\}$ in N. For each train $t \in T$, we add corresponding virtual source node s_0^t in $N_s \in N$, where s_0^t represents the source node where train t departs. We also add one sink s_1 into N, where s_1 represents the sink node where every train terminates its journey.
- For each link $(i, j) \in W$, we add following transit arcs into $Iden(i, j) \subset A$:

$$(i^k, j^{k+f(i,j)}),\ for\ all\ k = 0, \ldots, |Q| - 1\ and\ k + f(i, j) \leq |Q| - 1$$

  and

$$(j^k, i^{k+f(j,i)}),\ for\ all\ k = 0, \ldots, |Q| - 1\ and\ k + f(j, i) \leq |Q| - 1.$$

- For each link $(i, i^{'}) \in W_w$, we then update $Iden(i, i^{'})$ as follows:

$$Iden(i, i^{'}) = Iden(i^{'}, j) = Iden(i, i^{'}) \cup Iden(i^{'}, j).$$

- For each siding node $i \in N$, we add the following waiting arcs into $A_w \subset A$. In this way, we allow trains to wait/dwell on sidings:

$$(i^k, i^{k+1}),\ for\ all\ k = 0, \ldots, |Q| - 2.$$

- For any load-out $s \in S_l$ with loop capacity cap_s, we add $(cap_s - 1)$ waiting arcs between (s_{in}^k, s_{out}^k), for all $k = 0, \ldots, |Q - 2|$. In this way, we allow trains to wait/dwell in the loops.

- For each station node $i \in S_u$, we add an arc from that node to the sink, signifying that this allows train cancellation in case of deadlock (i.e., we can cancel a train with a very high penalty). Thus, we add these train disappearing arcs into $A_d \subset A$:

$$(i^k, s_1),\ for\ all\ k = 0, \ldots, |Q| - 1.$$

- For each train $t \in T$, we add an arc from the source to the scheduled starting node of that train, and we add these starting arcs into $A_s \subset A$:

$$(s_0^t, j^k),\ for\ all\ k \geq depQ_t,\ and\ j\ is\ the\ scheduled\ starting\ node(i.e.,\ j = dep_t).$$

- In all the above cases, whenever an arc $(i, j) \in A$ is added to the time–space network, we also add this arc to the outbound arc set of node $i(i.e.,\ Outb_i)$ and to the inbound arc set of node $j(i.e.,\ Inb_j)$.

4. Mathematical Model

According to the definitions made in the previous section, the objective and the set of constraints are now presented in a formal manner.

4.1. Attributes

For each $(i^k, j^l) \in A$ and a given train $t \in T$, there exists a cost attribute $c^t_{i^k j^l}$ as follows:

$$c^t_{i^k j^l} = \begin{cases} \gamma * (l - k), \forall t \in T, \forall (i^k, j^l) \in A - A_w - A_d - A_s \\ \alpha * (|Q| - \lfloor k/(60/g) \rfloor), \forall t \in T, \forall (i^k, i^{k+1}) \in A_w \\ \rho * time_{left},\ if\ dest_t = i\ or\ k = Q, \forall t \in T, \forall (i^k, s_1) \in A_d \\ M,\ if\ dest_t \neq i, \forall t \in T, \forall (i^k, s_1) \in A_d \\ \beta * (l - depQ_t), \forall t \in T, \forall (s_0^t, j^l) \in A_s, \end{cases}$$

where (i^k, j^l) denotes an arc in the time–space network representing possible movement (of a train) starting from node i at time k and terminating at node j at time l. The cost $\alpha * (|Q| - \lfloor k/(60/g) \rfloor)$ implies that we give weighted penalties for waiting/dwelling, where α is the constant parameter. The cost $\beta * (l - depQ_t)$ implies the penalty for late departure, where β is a constant parameter. The cost $\rho * time_{left}$ implies that we try to route trains close to the destination. The cost $\gamma * (l - k)$ implies that we try to route trains along the shortest-time path.

4.2. Variables

For each $(i^k, j^l) \in A$ and a given train $t \in T$, define the following binary decision variable:

$$x^t_{i^k j^l} = \begin{cases} 1, if\ train\ t\ will\ travel\ on\ arc(i^k, j^l)\ during\ k\ to\ l \\ 0, otherwise. \end{cases}$$

4.3. Objective

We here adopt the weighted-sum method to optimize multiple objectives simultaneously, where each objective is assigned a weight that represents its relative importance. The weighted-sum method then finds a single solution that is optimal with respect to the weighted sum of the objectives. Given the above cost attributes and variables, we then define the objective as follows:

$$Min \sum_{t \in T} \sum_{(i^k, j^l) \in A} c^t_{i^k j^l} * x^t_{i^k j^l}.$$

4.4. Constraints

In particular, we formulate the related constraints as follows.

- Flow conservation constraint: These constraints algebraically state that the sum of the flow through arcs directed toward a node plus that node's supply, if any, equals the sum of the flow through arcs directed away from that node plus that node's demand, if any. We have

$$\sum_{(s_0^t,j)\in Outb_{s_0^t}} x^t_{s_0^t j} = 1, \forall t \in T$$

$$\sum_{(i,s_1)\in Inb_{s_1}} x^t_{is_1} = 1, \forall t \in T$$

$$\sum_{(i,j)\in Outb_i} x^t_{ij} - \sum_{(j,i)\in Inb_i} x^t_{ji} = 0, \forall i \in N - N_s - \{s_1\}, \forall t \in T.$$

- Node capacity constraint: This constraint ensures that for each node excluding s_1, trains can be held safely without any collision or deadlock at any time point. We have

$$\sum_{t\in T} \sum_{(j,i)\in Inb_i} x^t_{ji} \le 1, \forall i \in N - \{s_1\}.$$

- Arc capacity constraint: This constraint implies that for any track, load-out in the network, at any time instant $q \in Q$, there can only be, at maximum, one train traveling or staying. We have

$$\sum_{t\in T} \sum_{(i'^k,j'^l)\in Iden(i,j)\&\{k\le q-1, l\ge q\}} x^t_{i'^k j'^l} \le 1, \forall q \in Q, \forall (i,j) \in W.$$

- Train departure constraint: This constraint implies that any train t with predecessor not null, which represents the return trip of a real train, has to depart at the time when its predecessor train completes its journey. We have

$$x^t_{s_0^t dep_t^q} - x^{t_{pre}}_{dest_t^q s_1} = 0, \forall q \in [depQ_t, Q], \forall t \in T\&t_{pre} \ne null.$$

5. Solution

The traditional approach to solve the above MIP problem is to use an offline scenario, which optimises the train schedule for a predefined time window. However, this approach is not suitable for the online scenario, where real-time updates are needed to keep up with the dynamic nature of the entire system. In the online scenario, the solving time often turns out to be unacceptable. For example, it may take several days to generate a train schedule for the next 24 h. It is not feasible to wait several days to generate a train schedule. Therefore, a more efficient solution framework is needed to handle the real-time requirements of the online scenario. We here propose a solution framework for the online scenario, which aims to address the limitations of the traditional approach by finding a balance between computational efficiency and optimisation accuracy.

In particular, we adopt the rolling-window approach. In this approach, a "rolling window" of time is defined, and the train scheduling problem is modeled as a MIP over this time horizon. The window is then moved forward in time, and the MIP is solved again for the new time period, taking into account the updated information. This solution framework can continuously monitor the train system, incorporate new information, and generate updated schedules in real time. This helps ensure that the trains are operating optimally, even in the face of unexpected events or changes.

As shown in Figure 2, it runs iteratively. The cycle length is predefined (e.g., every 5 min), which is decided as per the business requirements (e.g., problem size, algorithm running time, etc.). For each cycle, it contains 4 steps as follows.

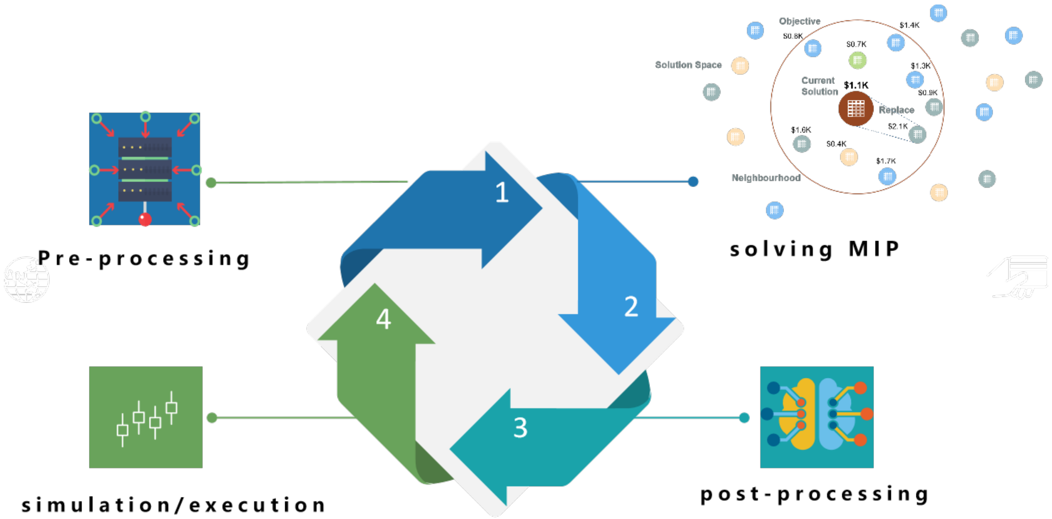

Figure 2. Solution framework.

5.1. Preprocessing

In the preprocessing step, we use the current status as input and formulate the above MIP problem, in which complexity changes exponentially when variable size varies. To further reduce the running time, we adopt the following approaches to reduce the variables.

- Initializing variables only for arcs on which train will travel: for each train, we only consider the reasonable arcs according to its destination. For example, in Figure 3 the train *t* is heading for load-out A. We then can set all the unreasonable variables (i.e., those variables associated to the arcs heading for load-out C) to 0.
- Removing variables corresponding to unrealistic movements: consider the train *t* in Figure 4 going towards load-out, which is not allowed to move on track F. We then fix variables corresponding to such invalid movements to 0.

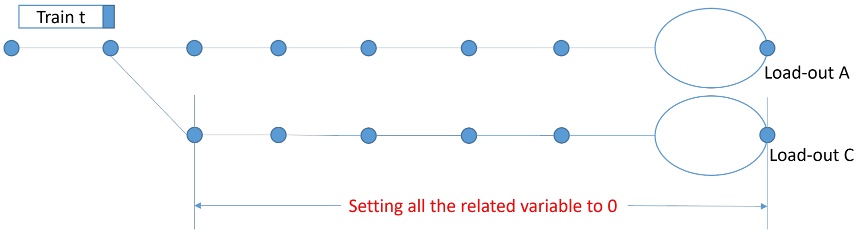

Figure 3. Preprocessing sample 1.

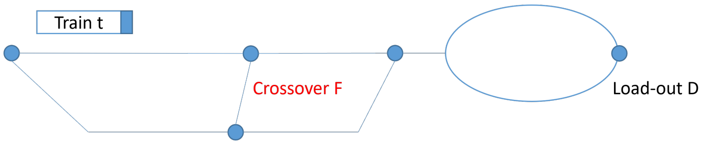

Figure 4. Preprocessing sample 2.

5.2. Solving MIP

In this step, we solve the formulated MIP problem for the coming scheduling window via mathematical optimisation solver (e.g., Gurobi, Xpress, Cplex, etc.). The choice of a proper scheduling window size depends on the specific requirements and constraints of the problem, as well as the available computational resources. To better utilize the solver, we also integrate more strategies as follows.

- Warm start: supply hints to help the solver find an initial solution, which consists of pairs of variables and values, known as a warm start. The hints may come from soft business rules or human experience.

- Solver tuning: use offline data to tune the solver, and adopt the solver setting for the online running. As shown in Table 2 , we tuned parameters for both Gurobi and Xpress.

Table 2. Solver tuning.

Solver	Parameters Tuned
Gurobi	Cuts, CutPasses, PrePasses, Method
Xpress	CoverCuts, CutStrategy, GomCuts, LNPBest, Presolve, PreProbing, etc.

5.3. Postprocessing

The postprocessing step is a crucial part of the optimisation solution. After getting the MIP results, we then need to translate the math-style results into a schedule of trains' movements. This process involves combining individual trains with their related predecessor trains to get complete movements. As shown in Table 1, we need to combine the results of t_{m1}, t_{m2}, t_{m2} to get the schedule of train m.

5.4. Simulation/Execution

In this step, we validate the schedule further through simulation and add more details, such as maintenance allocation. This step helps ensure that the schedule is accurate and feasible before it is executed. Finally, we communicate the detailed schedule to the trains, triggering an actual execution. This involves updating the trains' schedules in real-time so that they can follow the optimal path and arrive at their destinations as efficiently as possible. This step is critical for ensuring that the solution is implemented correctly and that the trains' movements are optimised as planned.

6. Experiments and Results

In this section, we evaluate the performance of the proposed solution.

6.1. Experimental Environment Setting

We here consider the sample network in Figure 1, which has track capacity shown in Figure 5 and is further revised in Figure 6. As per Section 3.3, we then construct the time-space network of a 50-min time window in Figure 7, for which the time instant length is 5 min.

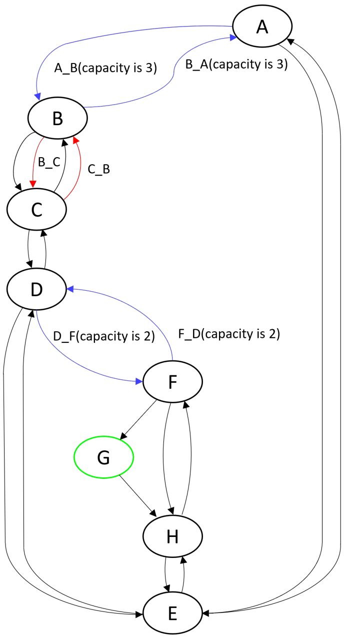

Figure 5. Physical network with capacity.

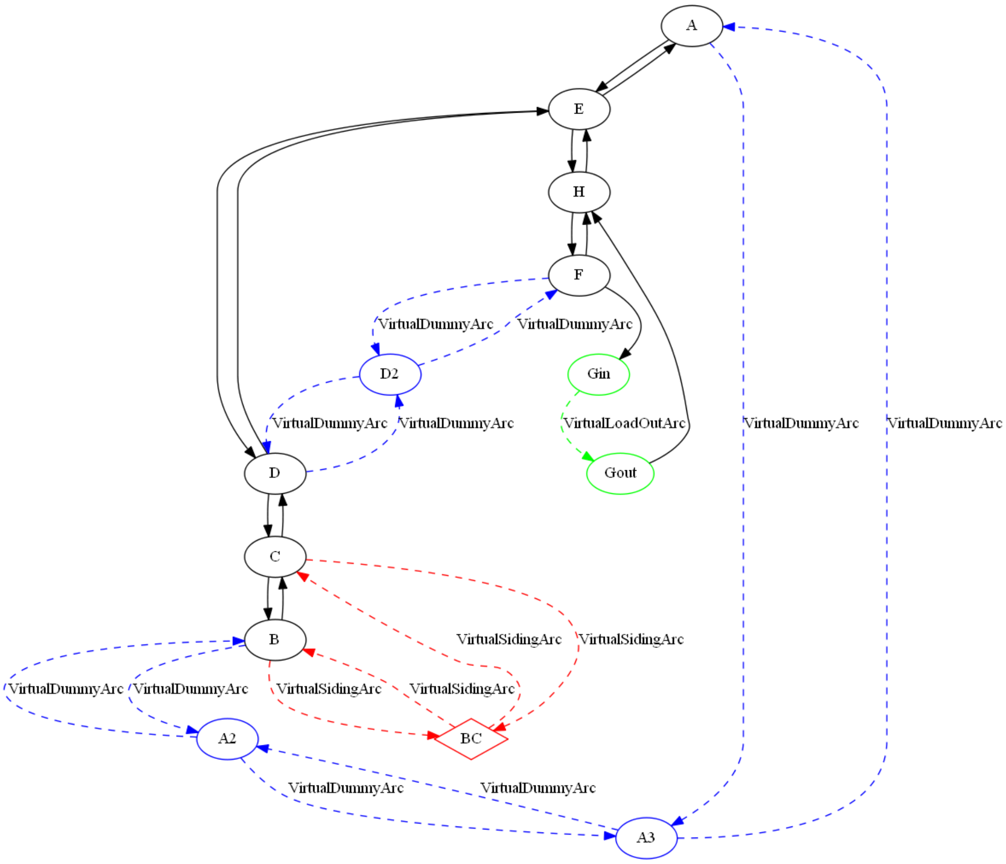

Figure 6. Further revised network.

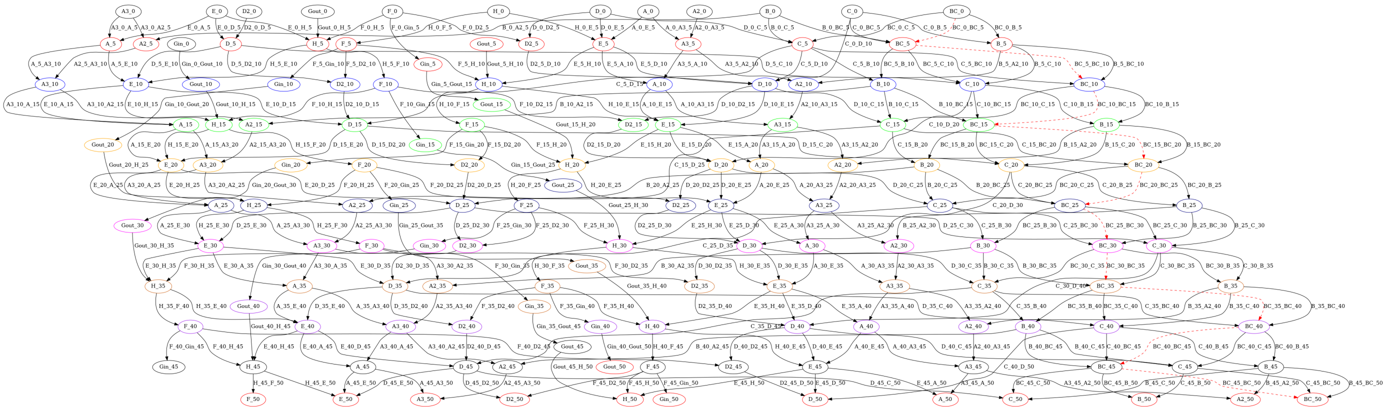

Figure 7. Constructed time–space network.

We then evaluate the proposed solution via the following test cases. From case 1 to case 4, we vary trains to schedule, generate the train schedule, and demonstrate the solution in a time–space network.

6.2. Test Case 1

Given a single train info in Table 3, we are able to get the solution in less than one minute, where the train schedule for train Mtest01 is demonstrated via green-path in Figure 8.

Table 3. Case 1: Train information.

Train Name	Departure Node	Departure Time	Destination Node
Mtest01	A	0	G

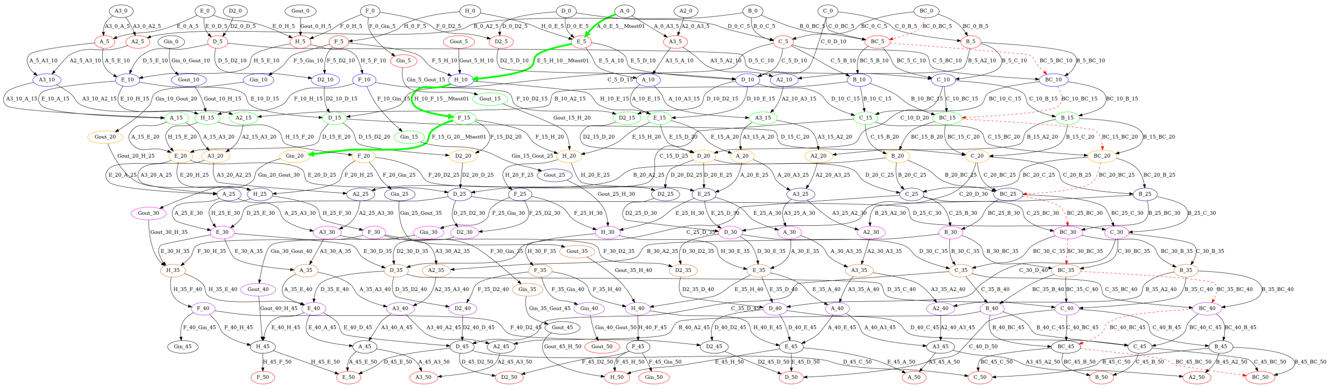

Figure 8. Case 1: Train schedule.

6.3. Test Case 2

We have two trains to schedule in Table 4, where train Mtest01 goes toward load-out G and train Mtest02 returns from load-out G. We are able to get the solution in less than one minute, where the train schedule for train Mtest01/Mtest02 is demonstrated via green-path/blue-path, respectively, in Figure 9.

Table 4. Case 2: Train information.

Train Name	Departure Node	Departure Time	Destination Node
Mtest01	A	0	G
Mtest02	G	0	B

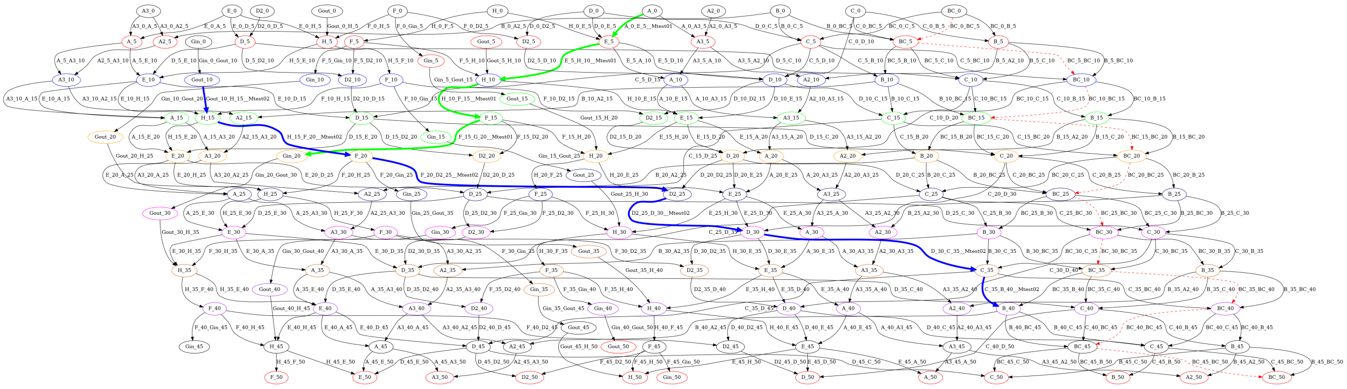

Figure 9. Case 2: Train schedule.

6.4. Test Case 3

We have three trains to schedule in Table 5. Both train Mtest01 and Mtest03 go toward load-out G from different start stations. Train Mtest02 goes back from load-out G with a departure time constraint. We are able to get the solution in less than one minute, and the train schedule is demonstrated in Figure 10. In particular, the schedules for trains Mtest01, Mtest02, and Mtest03 are demonstrated by the green path, blue path, and orange path, respectively. The detailed train schedules are listed in Table 6.

Table 5. Case 3: Train information.

Train Name	Departure Node	Departure Time	Destination Node
Mtest01	A	0	G
Mtest02	G	10	B
Mtest03	B	0	G

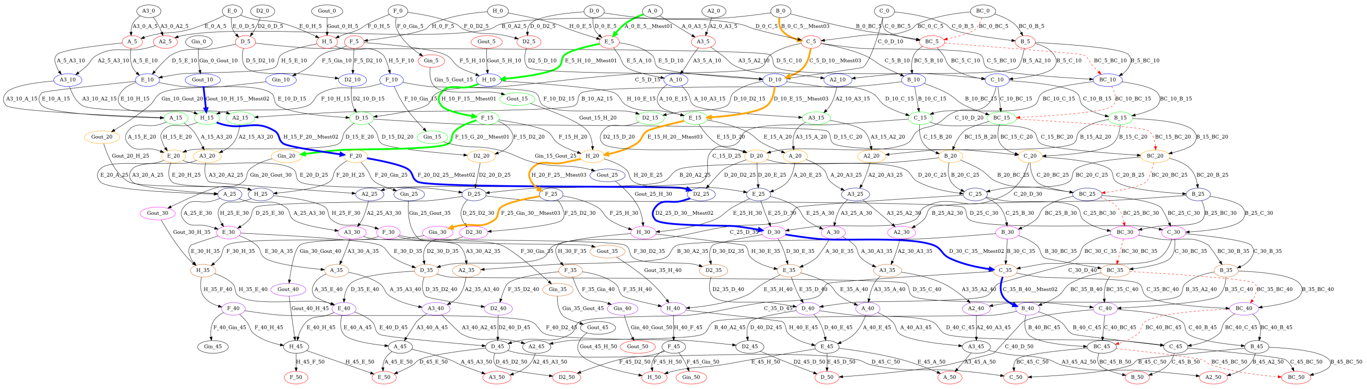

Figure 10. Case 3: Train schedule.

Table 6. Case 3: Train schedule.

Time–Train	Mtest01	Mtest02	Mtest03
0	A		B
5	E		C
10	H	G	D
15	F	H	E
20	G	F	H
25			F
30		D	G
35		C	
40		B	

6.5. Test Case 4

Given the three sets of trains information in Table 7, both train Mtest01 and Mtest03 go toward load-out G with the same start station; train Mtest02 goes back from load-out G with departure time constraint. Similar to case 3, we are able to get the solution in less than one minute, and demonstrate the train schedule in Figure 11, where the schedules for trains Mtest01, Mtest02, and Mtest03 are shown via the green path, blue-path, and orange path, respectively. The detailed train schedules are listed in Table 8.

Table 7. Case 4: Train information.

Train Name	Departure Node	Departure Time	Destination Node
Mtest01	A	0	G
Mtest02	G	10	B
Mtest03	A	0	G

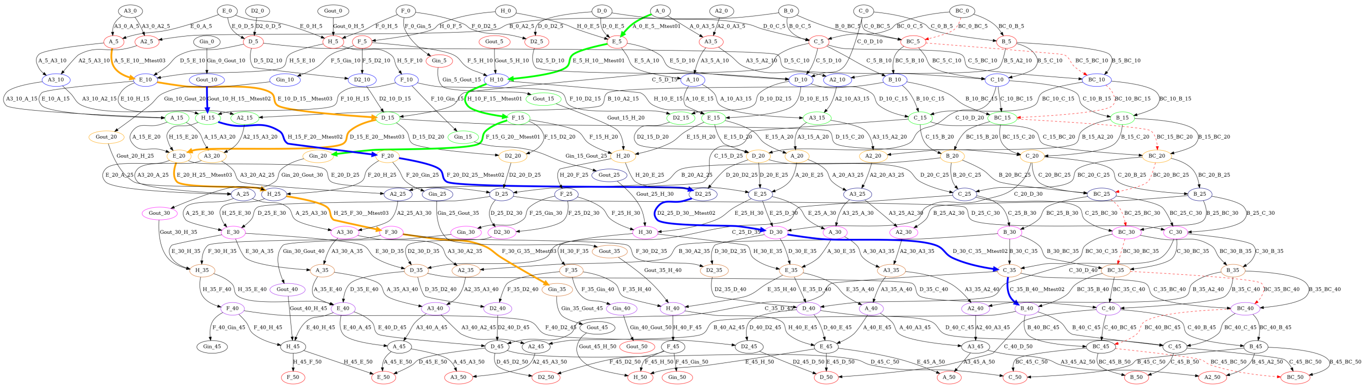

Figure 11. Case 4: Train schedule.

Table 8. Case 4: Train Schedule.

Time-Train	Mtest01	Mtest02	Mtest03
0	A		
5	E		A
10	H	G	E
15	F	H	D
20	G	F	E
25			H
30		D	F
35		C	G
40		B	

Overall, from test case 1 to 4, it is shown that with more trains gradually added, we are able to route trains properly.

7. Conclusions

This work studied an online mining railway optimisation problem. We first formulated the problem as a mixed integer programming problem and developed a novel online solution to address it. The solution was then evaluated through various test cases, which showed its effectiveness in solving the problem. It is worth mentioning that this model serves as an initial integrated optimisation model for the scheduling of mine railways and the allocation of station tracks. However, it still has limitations and room for further improvement. One limitation is that the scheduling window size needs to be carefully chosen, as choosing too large a window can result in computational complexity, while choosing too small a window can limit the ability to incorporate long-term scheduling decisions. Another limitation is that the proposed solution may not be suitable for very fast-paced real-time scenarios in which there is a need to make scheduling decisions on a second-to-second basis, as the time required to generate a schedule may be too long for the required response time.

Given the promising results of this study, further research will focus on the following several aspects:

- to consider more practical constraints, such as ensuring that the first-come-first-serve policy is upheld near load-out areas, which will increase the real-world applicability of the solution and make it more useful for mining companies; and
- to develop an effective heuristic algorithm to increase the efficiency of the solution. By improving the computational efficiency of the algorithm, this will make the solution more suitable for large-scale, real-world applications.

Author Contributions: Conceptualization, X.R.; Methodology, X.R., K.S. and G.M.; Software, X.R.; Validation, X.R. and H.G.; Investigation, X.R. and H.G.; Data curation, H.G.; Writing—original draft, X.R.; Writing—review and editing, X.R. and H.G.; Supervision, K.S. and G.M.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant U21A20446; Guangzhou Applied Basic Research Program under Grant 202201011786.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are grateful to anonymous reviewers for their thorough and most helpful comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ghasempour, T.; Heydecker, B. Adaptive railway traffic control using approximate dynamic programming. *Transp. Res. Part C Emerg. Technol.* **2020**, *113*, 91–107. [CrossRef]
2. Hou, Z.; Zhou, M.; Ning, L.; Dong, H. Automatic Train Regulation with Time Tuning and Holding Control under the Saturated Passenger Demand Condition. In Proceedings of the 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 22–24 May 2021.
3. Wu, Y.; Yang, H.; Zhao, S.; Shang, P. Mitigating unfairness in urban rail transit operation: A mixed-integer linear programming approach. *Transp. Res. Part B Methodol.* **2021**, *149*, 418–442. [CrossRef]
4. Xu, P.; Feng, G.; Cui, D.; Dai, X.; Zhang, Q.; Chen, F. Timetable Mapping Model and Dynamic Programming Approach for High-speed Railway Rescheduling. In Proceedings of the 2020 39th Chinese Control Conference (CCC), Shenyang, China, 27–29 July 2020.
5. Yin, J.; Andrea, D.; Wang, Y.; Xun, J.; Su, S.; Tang, T. Mixed-Integer Linear Programming Models for Coordinated Train Timetabling with Dynamic Demand. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019.
6. Yin, J.; Tang, T.; Yang, L.; Gao, Z.; Ran, B. Energy-efficient metro train rescheduling with uncertain time-variant passenger demands: An approximate dynamic programming approach. *Transp. Res. Part B* **2016**, *91*, 178–210. [CrossRef]
7. Cacchiani, V.; Furini, F.; Kidd, M.P. Approaches to a real-world train timetabling problem in a railway node. *Omega* **2016**, *58*, 97–110. [CrossRef]
8. Harrod, S. Modeling network transition constraints with hypergraphs. *Transp. Sci.* **2011**, *45*, 81–97. [CrossRef]
9. Kang, L.; Meng, Q. Two-phase decomposition method for the last train departure time choice in subway networks. *Transp. Res. B Methodol.* **2017**, *104*, 568–582. [CrossRef]
10. Kang, L.; Wu, J.; Sun, H.; Zhu, X.; Gao, Z. A case study on the coordination of last trains for the Beijing subway network. *Transp. Res. B Methodol.* **2015**, *72*, 112–127. [CrossRef]
11. Li, Z.; Mao, B.; Bai, Y.; Chen, Y. Integrated optimization of train stop planning and scheduling on metro lines with express/local mode. *IEEE Access* **2019**, *7*, 88534–88546. [CrossRef]
12. Qi, J.; Li, S.; Gao, Y.; Yang, K.; Liu, P. Joint optimization model for train scheduling and train stop planning with passengers distribution on railway corridors. *J. Oper. Res. Soc.* **2018**, *69*, 556–570. [CrossRef]
13. Li, S.; Lv, H.; Xu, C.; Chen, T.; Zou, C. Optimized Train Path Selection Method for Daily Freight Train Scheduling. *IEEE Access* **2020**, *8*, 40777–40790. [CrossRef]
14. Bersani, C.; Estil-Les, M.A.D.C.; Sacile, R. Min-Max Approach for High-Speed Train Scheduling and Rescheduling Models. *IEEE Access* **2021**, *9*, 41042–41051. [CrossRef]
15. Liu, R.; Li, S.; Yang, L. Collaborative optimization for metro train scheduling and train connections combined with passenger flow control strategy. *Omega* **2020**, *90*, 101990. [CrossRef]
16. Niu, H.; Zhou, X.; Gao, R. Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: Nonlinear integer programming models with linear constraints. *Transp. Res. Part B* **2015**, *76*, 117–135. [CrossRef]
17. Sanat, R.; Dutt, T.; Chandrababu, A.; Abhilasha, A.; Prasanna, G.N.S. Optimizing Schedule of Trains in Context of a Large Railway Network. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018.
18. Wang, Y.; D'Ariano, A.; Yin, J.; Meng, L.; Tang, T.; Ning, B Passenger demand oriented train scheduling and rolling stock circulation planning for an urban rail transit line. *Transp. Res. Part B Methodol.* **2018**, *118*, 193–227. [CrossRef]
19. Kang, L.; Wu, J.; Sun, H.; Zhu, X.; Wang, B. A practical model for last train rescheduling with train delay in urban railway transit networks. *Omega* **2015**, *50*, 29–42. [CrossRef]

20. Wang, Z.; Su, S.; Su, B.; Tang, T. A metro train rescheduling approach considering flexible short-turning and adding backup trains strategies during disruptions in the case of the vehicle breakdown. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021.
21. Xu, W.; Xie, B.; Zhang, X.; Ning, L. Timetable Rescheduling Considering Headway: A Genetic Algorithm Combined with the Benchmark Time Method. In Proceedings of the 2020 IEEE 5th International Conference on Intelligent Transportation Engineering (ICITE), Beijing, China, 11–13 September 2020.
22. Zhu, Y.; Goverde, R. Railway timetable rescheduling with flexible stopping and flexible short-turning during disruptions. *Transp. Res. Part Methodol.* **2019**, *123*, 149–181. [CrossRef]
23. Huang, P.; Kung, C.; Lin, C. An Integrated Formulation and Optimization for Periodic Timetabling of Railway Systems. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021.
24. Polinder, G.; Breugem, T.; Dollevoet, T.; Maróti, G. An adjustable robust optimization approach for periodic timetabling. *Transp. Res. Part B Methodol.* **2019**, *128*, 50–68. [CrossRef]
25. Sparing, D.; Goverde, R. A cycle time optimization model for generating stable periodic railway timetables. *Transp. Res. Part B Methodol.* **2017**, *98*, 198–223. [CrossRef]
26. Schiewe, P.; Schöbel, A. Periodic Timetabling with Integrated Routing: Toward Applicable Approaches. *Transp. Sci.* **2020**, *54*, 1714–1731. [CrossRef]
27. Zhong, J.-H.; Shen, M.; Zhang, J.; Chung, H.S.-H.; Shi, Y.-H.; Li, Y. A Differential Evolution Algorithm with Dual Populations for Solving Periodic Railway Timetable Scheduling Problem. *IEEE Trans. Evol. Comput.* **2013**, *17*, 512–527. [CrossRef]
28. Cacchiani, V.; Toth, P. Nominal and robust train timetabling problems. *Eur. J. Oper. Res.* **2012**, *219*, 727–737. [CrossRef]
29. Harrod, S. A tutorial on fundamental model structures for railway timetable optimization. *Surv. Oper. Res. Manag. Sci.* **2012**, *17*, 85–96. [CrossRef]
30. Borndörfer, R.; Klug, T.; Schlechte, T.; Fügenschuh, A.; Schang, T.; Schülldorf, H. The Freight Train Routing Problem for Congested Railway Networks with Mixed Traffic. *Transp. Sci.* **2016**, *50*, 408–423. [CrossRef]
31. Bešinović, N.; Widarno, B.; Goverde, R.M.P. Adjusting freight train paths to infrastructure possessions. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020.
32. Kang, L.; Zhu, X.; Sun, H.; Puchinger, J.; Ruthmair, M.; Hu, B. Modeling the first train timetabling problem with minimal missed trains and synchronization time differences in subway networks. *Transp. Res. B Methodol.* **2016**, *93*, 17–36. [CrossRef]
33. Mu, S.; Dessouky, M. Scheduling freight trains traveling on complex networks. *Transp. Res. B Methodol.* **2011**, *45*, 1103–1123. [CrossRef]
34. Ozturk, O.; Patrick, J. An optimization model for freight transport using urban rail transit. *Eur. J. Oper. Res.* **2018**, *267*, 1110–1121. [CrossRef]
35. Sato, K.; Fukumura, N. Real-time freight locomotive rescheduling and uncovered train detection during disruption. *Eur. J. Oper. Res.* **2012**, *221*, 636–648. [CrossRef]
36. Ursavas, E.; Zhu, S.X. Integrated Passenger and Freight Train Planning on Shared-Use Corridors. *Transp. Sci.* **2017**, *52*, 1376–1390. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.