A Formulation of MIP Train Rescheduling at Terminals in Bidirectional Double-Track Lines with a Moving Block and ATO

Kosuke Kawazoe[1]
Faculty of Science and Engineering, Waseda University, Shinjuku, Tokyo, Japan

Takuto Yamauchi
Faculty of Science and Engineering, Waseda University, Shinjuku, Tokyo, Japan

Kenji Tei
Faculty of Science and Engineering, Waseda University, Shinjuku, Tokyo, Japan

Abstract

When delays in trains occur, train schedules are rescheduled to reduce the impact. Despite many existing studies of automated train rescheduling, this study focuses on automated rescheduling considering a moving block and Automatic Train Operation (ATO). This study enables such automated rescheduling by formalizing this problem as a mixed integer programming (MIP) model. In previous work, the formulation was achieved for unidirectional single-track railway lines. In this paper, we aim to achieve the formulation for bidirectional double-track lines. Specifically, we propose a formulation of constraints about trains' running terminal stations. To evaluate our automated rescheduling approach, we implemented an MIP model consisting of a combination of the new constraints with the previous MIP model. We demonstrated the feasibility of our approach by applying it to a bidirectional double-track line with eight delay scenarios. We also evaluate the delay reduction and computation overhead of our approach by comparing it with a baseline with these eight scenarios. The results show that the total delay of all trains from our approach reduced from 20% to 30% than one from the baseline. On the other hand, the computation time increased from less than 1 second to a minimum of about 20 seconds and a maximum of about 1600 seconds.

2012 ACM Subject Classification Applied computing → Transportation

Keywords and phrases Train rescheduling, Mixed integer programming, ATO, Moving block

Digital Object Identifier 10.4230/OASIcs.ATMOS.2022.10

1 Introduction

In railway operation, when trains are delayed due to accidents, troubles, or congestions, the train schedule needs to be often reconstructed as a temporary schedule to reduce the impact of the delay [4] [13]. This is called train rescheduling. Automated train rescheduling attracts attention from the industry because manual scheduling is cumbersome and error-prone [4]. There are many studies for the practical application of automated rescheduling [4]. Especially, recent studies deal with automated rescheduling considering new types of train systems: a moving block [10] [9] and Automatic Train Operation (ATO) [14]. A moving block is a new railway safety system using radio communications. It is said to be effective in the reduction of delays, especially when a small delay such as several minutes occurs in busy lines with short train intervals [10]. Recently, this moving block started to spread combined with ATO [10] [9], especially in metropolitan busy lines.

[1] Corresponding author

22nd Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2022).
Editors: Mattia D'Emidio and Niels Lindner; Article No. 10; pp. 10:1–10:18
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The purpose of this study is to formulate a mixed integer programming (MIP) rescheduling model [1] in bidirectional double-track railway lines with both a moving block and ATO. The novelty of this study is that currently MIP approach has not been applied to rescheduling considering both the systems in bidirectional double-track lines. On the other hand, the MIP method has the advantage that it is always possible to get the solution to minimize desired indices among possible solutions keeping all of described constraints [1]. It can be thought that this advantage is important in the rescheduling in busy lines with short train intervals where a moving block and ATO would be implemented.

Recently, Kawazoe et al. [5] proposed a formulation of an MIP rescheduling model considering both a moving block and ATO. However, their formulation deals with only unidirectional single-track lines. This limits the applicability of the automated rescheduling because most industrial scenes require double-track lines and bidirectional train operation.

In this paper, we extend this work to deal with bidirectional double-track lines with both a moving block and ATO. Specifically, we formulate additional constraints for trains' running around terminal stations. To formulate such constraints, we newly assume a moving block on a bidirectional double-track line with ATO operation, which is based on the model of Hou et al. [4] Then, we classify seven patterns based on the positional relationships of the trains around terminal stations, and formulate constraints for each pattern. Furthermore, we implemented the MIP model including those new constraints, and demonstrated the feasibility of our approach by applying it to a bidirectional double-track line with eight delay scenarios by using the CPLEX solver. We evaluated the delay reduction and the calculation time of our approach by comparing it with a baseline constructed by extending Hou et al. [4]'s model to support bidirectional double-track lines. The results show that generated schedules of our approach reduced the total delay from 20% to 30% than one from the baseline. On the other hand, the computation time increased from less than 1 second to a minimum of about 20 seconds and a maximum of about 1600 seconds.

There are two contributions of this paper as follows. First, it makes us get the train rescheduling solution that minimizes the total delay of all trains in bidirectional double-track lines with both a moving block and ATO among possible solutions keeping all of the constraints we described. Second, it makes us confirm the rescheduling model with the moving block reduced the total delay from 20% to 30% than one from the baseline without a moving block in bidirectional double-track lines with ATO.

This paper is organized as follows. Section 2 describes the previous work of automated train rescheduling. Section 3 describes a moving block. Next, Section 4 describes assumptions about our formulation. Then, Section 5 describes the formulation of of trains' running around terminal stations according to the assumptions in Section 4. Section 6 describes the implementation of the MIP model including the constraints in Section 5 and evaluation of it. After that, Section 7 describes discussions about the results of the evaluation. Furthermore, Section 8 describes related studies, and Section 9 describes our conclusions and future work.

2 Automated Train Rescheduling

We introduce previous studies of automated train rescheduling following two ways of classification. The first classification is based on the method of finding answers to rescheduling. In previous studies, there are two mainstream methods using a search [2]. The first one is using metaheuristics, which includes greedy search [3] and genetic algorithm [11]. The second one is using MIP [1] [13]. Metaheuristics has the disadvantage that it is not always possible to obtain solutions that minimize the desired indices. This is because they do not

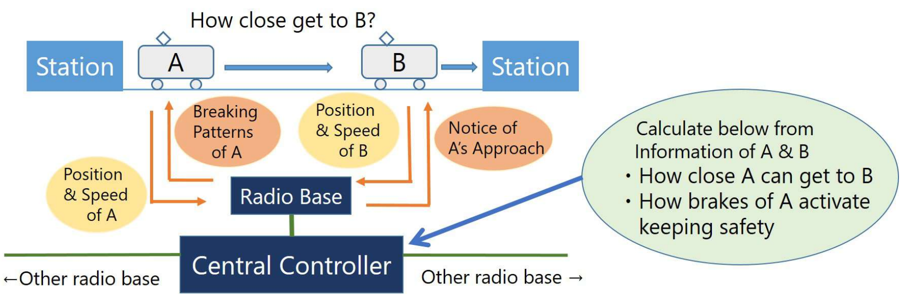


Figure 1 An example of radio communications on CBTC.

do exhaustive searches [1]. On the other hand, the MIP method uses a specialized solver for MIP (a MIP solver) and performs an exhaustive search. Therefore, it is always possible to solutions to minimize desired indices, for example, a total delay of all stations and all trains among possible solutions satisfying all of the constraints [1]. In addition to these two methods, there are also studies using the graph theory [2] [7]. However, this method also has to consider the order to decide the values of variables. Therefore, it takes more time compared with the MIP method when solving the same scale problem. Furthermore, in recent years, there are studies using data-driven machine learning [15] or reinforcement learning [18]. However, in these methods, it is not certain to always get the solutions that minimize the object index as same as metaheuristics.

The second classification is based on the consideration of new railway systems. Recent studies deal with automated rescheduling considering new types of train systems. Among these studies, we focus on the previous studies considering ATO and a moving block. There are studies [11] [17] [12] considering Communications-Based Train Control (CBTC) [10], which is a standard using a moving block. CBTCs in all of these three studies are assumed to have the function of ATO. Therefore, these three studies consider both those two systems at the same time. However, each of these uses metaheuristics [11], a unique decision algorithm [17], or a simulation software specialized in railway operation [12], so they did not use the MIP method. On the other hand, Kawazoe et al. [5] proposed an MIP model that considers both a moving block and ATO. This model is based on the model of Hou et al. [4] Before this [5], there were rescheduling studies considering only either ATO [4] or a moving block [16]. However, Kawazoe et al. [5] assumed a unidirectional single-track line in their model. The rescheduling with both the systems in bidirectional multiple track lines is not formulated as an MIP model in previous work.

3 Moving Block

A moving block is a new railway safety system using radio communications. Each train gets the information of the preceding train to prevent collisions. The most widespread international standard using a moving block is CBTC [10]. Fig. 1 shows the example of radio communications on CBTC. Train A and B in Fig. 1 sequentially send the central controller the information of each train's position or speed through the nearest radio base. From this information, the central controller calculates how close A can get to B and how the brakes of A activate keeping safety. The calculation results are sent to A sequentially, and if A gets too close to B, the brakes of A will activate and A will stop automatically to keep safe and prevent collisions.

Table 1 Symbols of Description.

	Sets, elements, and constants		Variables
$i \in M$	Stations set $M = \{1, 2, ..., i, ..., \vert M\vert\}$	$ta_{i,j}$	Actual arrival time at i of j (s)
$j \in N$	Trains set $N = \{1, 2, ..., j, ..., \vert N\vert\}$	$td_{i,j}$	Actual departure time at i of j (s)
$k \in K$	Sections set $K = \{1, 2, ..., k, ..., \vert N\vert - 1\}$	$\varepsilon^l_{k,j}$	Whether level is l of j in k (0-1)
$l \in P$	ATO levels set $P = \{1, 2, ..., l, ..., \vert P\vert\}$		
$R^l_{k,j}$	ATO running time with l in k (s)		

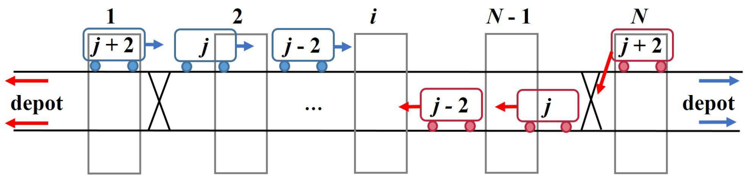


Figure 2 Perspective of the assumed railway line.

The conventional signal system needs to leave room for the distance between two trains. This is because it keeps safety considering that it cannot detect the positions of all trains online and the error of braking distance. On the other hand, a moving block can detect the safe distance not to crash into preceding train online using position information by radio communications. Therefore, a moving block can shorten distances between trains keeping safety, compared with the conventional system. For this reason, a moving block is said to be effective in reducing delays, especially by several minutes in busy railway lines with short intervals of trains [9].

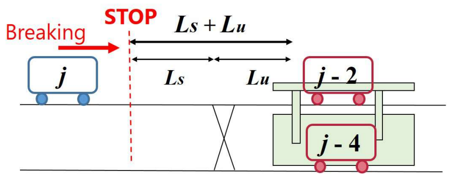
Some standards of CBTC have the function of ATO [10]. Moreover, some of the other standards, for example, ATACS [9] developed by East Japan Railway Company in Japan, are also assumed to be combined with ATO in the future. Therefore, the number of lines with both a moving block and ATO will increase from now on.

4 Assumptions

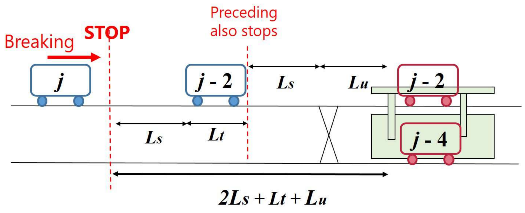
In this section, we describe the perspective, ATO, and the moving block of the railway line we assumed in this study. In addition, Table 1 shows the symbols used in assumptions and constraints this section and in Section 5. These include the sets and these elements, and the decision variables of MIP based on the model of Hou et al. [4] mentioned in Section 2.

4.1 Perspective and ATO

In this study, we assume the railway line as a metropolitan busy subway line with bidirectional double tracks. Fig. 2 shows the perspective of the line. It has $|M|$ stations. We assume $|N|$ train services in this line, and each has a train number. The "up" forward trains whose train numbers are odds go from station 1 to $|M|$. The "down" forward trains whose train numbers are evens, go from $|M|$ to 1. The section between Station i and $i + 1$ is Section k. Therefore, $i = k$. Every station has two platforms. Station 1 and $|M|$ are terminal stations. Some trains which arrived at terminal stations come from depots. Some trains in terminal stations become "turnaround trains" and go back as the counter-forward trains. We describe the turnaround train of j as $r(j)$. The train number $r(j)$ is decided up to j in advance and never changed by rescheduling. The other trains go back to depots as Fig. 2 shows.



(a) Approaching when (TA) in terminal sections.



(b) Approaching when (TB) in terminal sections.

Figure 3 Assumed approaches due to the assumed the moving block at terminal sections.

ATO is the system to automate trains' running. In this study, we assume every train has ATO on it. ATO has pre-programmed P speed patterns for each section. Each pattern in P patterns is called ATO level "l", and the smaller the value of "l", the shorter the running time of the section. In each section, each train follows one ATO level selected by staff on the train and runs the section automatically. In this study, $\epsilon^l_{k,j}$ is the boolean decision variable of rescheduling, which represents whether the ATO level is l of Train j in Section k or not. These assumptions of ATO are based on Hou et al. [4]

4.2 Moving Block

In this study, if a train approaches the preceding train, the moving block will activate automatic brakes for the train to prevent collisions. This follows the scheme of CBTC [10] we described in Section 3. However, these braking behaviors are different up to sections and the position relationship of trains. In this paper, we focus on the sections with terminal stations. Therefore, we divide the runs of trains into two large patterns below with reference to Kawazoe et al. [5] and for each pattern, we assume how a train can approach the preceding train and how the moving block auto brakes work.

- Pattern (TA): In the sections with terminals (We call them "terminal sections" hereafter), we assume when a train departs from Station $N-1$ (up) or 2 (down), after the preceding arrived at Terminal $|N|$ (up) or 1 (down). As Fig. 3a, in terminal sections, there are railroad switches to change track L_u (m) in front of the terminal to the other and turn around in the opposite direction. In the example of Fig. 3a, if the following train j stops and waits covering or running over a railroad switch, the preceding train $j-4$ will not be able to turn around and j will stay unable to reach the terminal. Therefore, in this pattern, if two trains are dwelling at the terminal, the following train will have to stop in front of the switch. Moreover, to ensure that the entire train stops in front of the switch, the train has to set the stop target L_s (m) before the switch. L_s means the minimum limit of the distance to spare from the stop target, which keeps the safety of trains even if considering the braking distance error of auto brakes [10] [9]. In summary, as Fig.3a shows, totally the following can approach up to $L_s + L_u$ (m) in front of the terminal.
- Pattern (TB): In terminal sections, we assume when a train departs from Station 2 or $N-1$ before the preceding arrived at Terminal 1 or $|N|$. In this pattern, the preceding train also might stop as (TA). If so, in the example of 3b, the following train j can approach up to more L_s (m) in front of the preceding $j-2$ stopping as (TA) and wait for $j-6$. In addition, the limited distance to spare L_s is also applied to the stop target in front of the stopping preceding train. In other words, j can approach up to more $L_s + L_t$ (m) in front of the stopping position of (TA). L_t means the constant length of one train. Therefore, totally j can approach up only to $2L_s + L_t + L_u$ (m) in front of the terminal,

as Fig. 3b shows. In addition, in this study, we make two simplifications in Pattern (TB). First, we assume that train j always only can approach up to just $2L_s + L_t + L_u$ (m) behind the terminal, when the preceding $j-2$ is still on the way at the time j has to start deceleration to stop with auto brakes. This is not up to whether the much earlier trains $j-4$ or $j-6$ are still dwelling at the terminal at that time. Furthermore, in Pattern (TB), two trains are running in one section at the same time. The second simplification is that the number of trains running close together in one section is limited to two in this study. We will formalize this assumption as a constraint in Section 5.

In the moving block, when the preceding train is stopping at the next station, the following train approaches up to the limited distance, i.e., L_s (m) in this study, in front of the preceding. Thereby the following can arrive at the next station immediately after the preceding departs. The patterns above are based on this idea.

5 Constraints Description

In this study, we propose to formulate constraints about trains' running in terminal sections in bidirectional double-track lines with both a moving block and ATO, which we assumed in Section 4. This allows trains to run with close distances between each other in terminal sections, and turn around in the opposite direction at terminal stations. In this section, we describe the formulation of running constraints based on the two patterns described in Section 4.2.

5.1 Pattern (TA)

In Pattern (TA), j has to decide whether it stops with $L_s + L_u$ in front of the terminal as Fig.3a shows, due to the railroad switch. If we set the time of brake application as $t_{T1A(k,l)}$, we can further divide (TA) into these three cases for j below. "Beyond the switch" means in the terminal or in the section between the railroad switch and the terminal.

- (TA1): No trains are beyond the switch at $t_{T1A(k,l)}$
- (TA2): 1 train is beyond the switch at $t_{T1A(k,l)}$
- (TA3): 2 trains are beyond the switch at $t_{T1A(k,l)}$

In (TA1), j need not use the moving block brakes, and just takes ATO running time $R^l_{k,j}$ to the terminal. We can formulate this as Constraint (1) (up) or (2) (down).

$$ta_{N,j} - td_{N-1,j} = \max(\varepsilon^l_{N-1,j} R^l_{N-1,j}) \tag{1}$$

$$ta_{1,j} - td_{2,j} = \max(\varepsilon^l_{1,j} R^l_{1,j}) \tag{2}$$

On the other hand, in (TA2), $j-2$ or $r(j-2)$ is still beyond the switch. If $j-2$ does not turn around and go to the depot, j need not use brakes to stop in front of the terminal. This is because j can immediately enter the platform that is not the one where $j-2$ is stopped. Therefore, if $r(j-2)$ does not exist, j will just take ATO running time R^l_k to the terminal as same as (TA1), i.e., the constraints for j are Constraint (1) and (2). However, if $j-2$ returns in (TA2), the constraint has to be made to prevent the collision of j and $r(j-2)$. In this model, if $r(j-2)$ already has departed from the terminal at $t_{T1A(k,l)}$, j shall wait for that $r(j-2)$ passes the switch before arriving at the terminal (TA2-1). We set the time that j stops as $t_{T2A(up,l)}$ or $t_{T2A(down,l)}$. Moreover, after j starts moving again, j has to runs $L_s + L_u$ and more L_t to arrive at the station. We set the constant time j runs L_s, L_u and L_t as T_{LsLtLu}, which is not up to j and j's ATO level l. Hence, when we formulate them, if in (TA2-1) j has to satisfy Constraint (3) (up) or (4) (down). t_{sd} and t_{su} are the time constants which it takes from leaving each terminal to passing the switch. Otherwise, i.e., if $r(j-2)$ is still dwelling at the terminal at $t_{T1A(k,l)}$, j shall arrive at the terminal before $r(j-2)$ departs from the terminal (TA2-2). Hence, when we formulate them, in (TA2-2) j has to satisfy Constraint (1) and (5) (up), or (2) and (6) (down).

$$ta_{N,j} \geq \max(td_{N,r(j-2)} + t_{sd}, td_{N-1,j} + \varepsilon^l_{N-1,j} t_{T2A(up,l)}) + T_{LsLtLu} \tag{3}$$
$$ta_{N,j} \geq \max(td_{1,r(j-2)} + t_{su}, td_{2,j} + \varepsilon^l_{1,j} t_{T2A(down,l)}) + T_{LsLtLu} \tag{4}$$
$$ta_{N,j} \leq td_{N,r(j-2)} \tag{5}$$
$$ta_{1,j} \leq td_{1,r(j-2)} \tag{6}$$

Moreover, in (TA3), $j-4$ or $r(j-4)$ is also still beyond the switch in addition to $j-2$ or $r(j-2)$. In this pattern, j has to stop with $L_s + L_u$ to the terminal and the wait for the departure of $r(j-4)$. If $r(j-4)$ does not exist, j can start to move again to the terminal immediately after $j-4$ departs from the terminal to the depot. Therefore, when we formulate them, if $r(j-4)$ does not exist, in (TA3) j has to satisfy (7) (up) or (8) (down). On the other hand, if $r(j-4)$ exists, j shall wait for that $r(j-4)$ passes the switch before arriving at the terminal. Hence, when we formulate them, if $r(j-4)$ exists, in (TA3) j has to satisfy Constraint (9) (up) or (10) (down). Furthermore, if $j-2$ returns as $r(j-2)$, j shall arrive at the terminal before $r(j-2)$ departs from the terminal as same as (TA2-2). Hence, j also has to satisfy Constraint (5) (up) or (6) (down) as same as (TA2-2).

$$ta_{N,j} \geq \max(td_{N,j-4}, td_{N-1,j} + \varepsilon^l_{N-1,j} t_{T2A(up,l)}) + T_{LsLtLu} \tag{7}$$
$$ta_{1,j} \geq \max(td_{1,j-4}, td_{2,j} + \varepsilon^l_{1,j} t_{T2A(down,l)}) + T_{LsLtLu} \tag{8}$$
$$ta_{N,j} \geq \max(td_{N,r(j-4)} + t_{sd}, td_{N-1,j} + \varepsilon^l_{N-1,j} t_{T2A(up,l)}) + T_{LsLtLu} \tag{9}$$
$$ta_{1,j} \geq \max(td_{1,r(j-4)} + t_{su}, td_{2,j} + \varepsilon^l_{1,j} t_{T2A(down,l)}) + T_{LsLtLu} \tag{10}$$

5.2 Pattern (TB)

In Pattern (TB), first of all, j does not leave $N-1$ or 2 until the two earlier train $j-4$ arrives at the terminal $|N|$ or 1. This is by our assumption that the number of trains running close together in one section is limited to two, mentioned in Section 4.2. We can describe this as Constraint (11) below.

$$td_{i,j} > ta_{i,j-4} \tag{11}$$

Based on the above, in this pattern, j has to decide whether it stops with $2L_s + L_t + L_u$ in front of the terminal as Fig.3b shows. If we set the time of brake application as $t_{T1B(k,l)}$, we can further divide (TB) into these four cases for j up to positions of trains at $t_{T1B(k,l)}$ below.

- (TB1): $j-2$ has already arrived at the terminal, and no others are beyond the switch
- (TB2): $j-2$ has already arrived at the terminal, and another train is beyond the switch
- (TB3): $j-2$ is still on the way to the terminal, but no others are beyond the switch
- (TB4): $j-2$ is still on the way to the terminal and another train is beyond the switch

In (TB1), j shall arrive at the terminal before $r(j-2)$ departs from the terminal, and j need not use brakes to stop in front of the terminal. This is because j can immediately enter the platform that is not the one where $j-2$ is stopped as same as cases of (TA2). Therefore, in (TB1), j can run the ATO running time, i.e., Constraint (1) or (2). Furthermore, if $r(j-2)$ exists, in (TB1), j also has to satisfy Constraint (5) (up) or (6) (down) as same as (TA2-2).

In (TB2), the situation is the same as (TA3), i.e., $j-4$ or $r(j-4)$ is beyond the switch in addition to $j-2$ dwelling at the terminal. Therefore, in (TB2), the constraints which train j has to satisfy branch conditionally as same as (TA3). Hence, if $r(j-4)$ does not exist, j has to satisfy Constraint (7) (up) or (8) (down). Otherwise, i.e., $r(j-4)$ exists, j has to satisfy Constraint (9) (up) or (10) (down). Furthermore, if $r(j-2)$ exists, j also has to satisfy Constraint (5) (up) or (6) (down).

However, in (TB3), $j-2$ is still on the way to the terminal at $t_{T1B(k,l)}$. Therefore, j need use brakes to stop with $2L_s + L_t + L_u$ in front of the next station. This is because of the simplification we mentioned when we describe Pattern (TB) in Section 4.2. we set the time that j stops as $t_{T2B(up,l)}$ or $t_{T2B(down,l)}$. Then, after j wait for arrival of $j-2$ at the terminal, j starts moving again, and j has to runs $2L_s + L_t + L_u$ and more L_t to arrive at the station. We set the constant time j runs $2L_s + 2L_t + L_u$ as $T_{2Ls2LtLu}$, which is not up to j and j's ATO level l. Hence, when we formulate them, if in (TB3) j has to satisfy Constraint (12) (up) or (13) (down). Furthermore, if $r(j-2)$ exists, in (TB3), j also has to satisfy Constraint (5) (up) or (6) (down) as same as (TB2) and so on.

$$ta_{N,j} \geq \max(ta_{N,j-2}, td_{N-1,j} + \varepsilon^l_{N-1,j} t_{T2B(up,l)}) + T_{2Ls2LtLu} \tag{12}$$
$$ta_{1,j} \geq \max(ta_{1,j-2}, td_{2,j} + \varepsilon^l_{1,j} t_{T2B(down,l)}) + T_{2Ls2LtLu} \tag{13}$$

Finally, in (TB4), $j-4$ or $r(j-4)$, and sometimes also $j-6$ or $r(j-6)$ is also still beyond the switch in addition to the situation of (TB3). In this pattern, j has to stop with $2L_s + L_t + L_u$ to the terminal and wait for both of the departure of $j-4$ or $r(j-4)$, and the arrival of $j-2$. If $r(j-4)$ does not exist, j can start to move again to the terminal immediately after $j-4$ departs to the depot and $j-2$ arrives at the terminal. when we formulate them, if $r(j-4)$ does not exist, in (TB4) j has to satisfy Constraint (14) (up) or (15) (down). On the other hand, if $r(j-4)$ exists, j shall wait for that $r(j-4)$ passes the switch before arriving at the terminal. Hence, when we formulate them, if $r(j-4)$ exists, in (TB4) j has to satisfy Constraint (16) (up) or (17) (down). Furthermore, if $r(j-2)$ exists, j also has to satisfy Constraint (5) (up) or (6) (down) as same as (TB3) and so on.

$$ta_{N,j} \geq \max(ta_{N,j-2}, td_{N,j-4}, td_{N-1,j} + \varepsilon^l_{N-1,j} t_{T2B(up,l)}) + T_{2Ls2LtLu} \tag{14}$$
$$ta_{1,j} \geq \max(ta_{1,j-2}, td_{1,j-4}, td_{2,j} + \varepsilon^l_{1,j} t_{T2B(down,l)}) + T_{2Ls2LtLu} \tag{15}$$
$$ta_{N,j} \geq \max(ta_{N,j-2}, td_{N,r(j-4)} + t_{sd}, td_{N-1,j} + \varepsilon^l_{N-1,j} t_{T2B(up,l)}) + T_{2Ls2LtLu} \tag{16}$$
$$ta_{1,j} \geq \max(ta_{1,j-2}, td_{1,r(j-4)} + t_{su}, td_{2,j} + \varepsilon^l_{1,j} t_{T2B(down,l)}) + T_{2Ls2LtLu} \tag{17}$$

6 Evaluation

In this section, we implemented a new MIP model with new constraints we formulated in Section 5. This model is the rescheduling model considering both the moving block and ATO in bidirectional double-track lines, and we call this “our model” below. Then, we executed our model on the MIP solver CPLEX. We conducted experiments with our model to answer two research questions.

RQ1 How much is the total delay reduced by our constraints of the moving block?
As we mentioned in Section 3, a moving block is said to be effective in reducing delays due to running with close distances between trains. We evaluate how this effect can be seen in bidirectional double-track lines with ATO. To do this, in Section 6.2 we compared the total delay of all trains at all stations of the solution of our model with the model of the baseline without a moving block in eight different delay scenarios (Experiment 1).

RQ2 How long does it take to run the model?
We evaluate the calculation time of our model to get solutions. To do this, in Section 6.3 we measured the calculation time for the eight delay scenarios (Experiment 2).

6.1 Experiment Setting

In this subsection, we describe the model implementation, and the line data to be applied in the experiments. In addition, for experiments, we used a PC with Intel Core i7-7500U, 8GB RAM, and Windows 10 64-bit version. In this PC, we use CPLEX Optimization Studio 12.10.0 Academic Edition (IBM) as a MIP solver. We call this CPLEX below.

For experiments, we implemented our model with new constraints mentioned in Section 5. As for the objective function, We set T_{delay} as that of our model in order to evaluate RQ1. T_{delay} is the sum of the delay time of arrival and departure compared with the original schedule for all trains at all stations. We write the arrival and departure time of the original schedule as $Td_{i,j}$ and $Ta_{i,j}$, and describe T_{delay} as Equation (18).

$$T_{delay} = \sum_{i,j} (ta_{i,j} - Ta_{i,j}) + (td_{i,j} - Td_{i,j}) \tag{18}$$

As for the constraints in our model, they are combinations of the parts of constraints based on Hou et al. [4] and Kawazoe et al. [5], and new constraints which we formulated in Section 5. Specifically, the constraints consisting the model are as follows.

- Constraint (1)-(17) for trains' running in terminal sections we formulated in Section 5.
- Order constraints of arrival and departure, and constraints about running of trains in not-terminal sections. These constraints are based on the model of Kawazoe et al. [5], which we described as Constraint (19)-(27) in Appendix B.1 and B.2.
- The other constraints about selecting ATO levels, dwelling time, and relationship between the original schedule and the rescheduled schedule. These constraints based on the model of Hou et al. [4], which we described as Constraint (28)-(32) in Appendix B.3.

To implement this model, we used OPL [6] which is developed to be specialized to describe input models of CPLEX. In addition, only when the implementation of our model, we defined true or false valuables which judge the pattern branches of each train in each section, and made the solver also outputs the solutions of those variables. Thereby we can identify into which pattern from (TA1) to (TB3) in Section 5 each train branches at each section from the rescheduling solution we got. Furthermore, we also implemented the baseline model without considering a moving block to be compared with our model in experiments. The objective function of the baseline is also T_{delay}, and the constraints of that are changed to be applied to bidirectional double-track lines as it is from constraints of Hou et al. [4]

Next, we describe the data of the railway line to be applied below. In the experiments, we applied the models to an imaginary metropolitan subway line with four stations, i.e., $|M|$=4 and $|K|$=3. Within the assumed time period, we assumed ten services running in the line, i.e., $|N|$=10, whereas we assumed six physical trains. Therefore, we assumed that four of them each run once as a turnaround train in the opposite direction as $r(1) = 8$, $r(3) = 10$, $r(2) = 7$, and $r(4) = 9$. We set the original schedule with no delays as follows; the intervals of arrivals and departures between two consecutive trains are 155 seconds at all stations. In addition, the number of ATO levels $|L|$ is 5 in each section. In the original schedule, the ATO level of every train is set to l=2 in all sections. As for the settings of the other detailed constants, we describe them in Appendix C. Furthermore, we inputted the delay information of each delay scenario for every model execution. Each delay information consists of a set of three values: the station where the primary departure delay caused d_l, the train whose departure primary delayed d_t, and t_d which is the amount of the departure delay time of d_t at d_l.

Table 2 T_{delay} (s), its reduction rate, and execution time (s) of delay scenarios.

Scenario		$T_{delay}(s)$			Exec. time (s)	
No.	(d_l,d_t,t_d)	Baseline	Our model	Reduction (%)	Baseline	Our model
1	(1,1,200)	4022	3222	19.9	0.13	88
2	(2,1,200)	3776	2868	24.0	0.09	84
3	(1,3,200)	3906	2810	28.1	0.09	33
4	(2,3,200)	3430	2445	28.7	0.11	21
5	(1,1,400)	16854	11948	29.1	0.08	1639
6	(2,1,400)	15463	10462	32.3	0.06	1083
7	(1,3,400)	-	10608	-	-	83
8	(2,3,400)	-	8183	-	-	43

6.2 Experiment 1: Delay Reduction

In Experiment 1 (referred to as "EX1" hereafter), we ran our model and the baseline on CPLEX with the same eight delay scenarios. From this experiment, we evaluated our model to answer RQ1. We set eight delay scenarios as $d_l = 1$ or 2, $d_t = 1$ or 3 and $t_d = 200$ or 400 (s). This setting is to evaluate the difference of solutions between if d_l is a terminal station and ones otherwise, and also between if d_t is the first train and otherwise in the assumed time period. In addition, we want to evaluate the effect of a moving block for a short delay such as a several minutes. Also considering this, we set the value of t_d as 200 (s) and 400 (s) in order to evaluate the difference of solutions up to the amount of the first delay.

We show the result of EX1 as the left side of Table 2. In Table 2, we wrote down the values of T_{delay} gotten from two models, and the reduction rate from the baseline to our model in each delay scenario. As for from Scenario No. 1 to No. 6, we could got the solution from both of two models, and the reduction rates are from about 20% to over 30% in all of them. As for Scenario No. 7 and No. 8, we could got the rescheduling solution only from our model. Moreover, with same t_d, when d_l is 1 and d_t is 1, the reduction rate is a little smaller than otherwise. On the other hand, with same d_l and d_t, The larger t_d, the smaller rate. In addition, we have confirmed that the rescheduled train schedule and solutions of decision variables got from each model satisfy all constraints of each model in all delay scenarios in which we could get the solutions.

6.3 Experiment 2: Calculation Time

In Experiment 2 (referred to as "EX2" hereafter), we measured and compared the CPLEX calculation time of our model and the baseline to get the solution. In this EX2, we used the same eight delay scenarios as EX1 in order to evaluate the difference in the calculation time between different d_l, d_t, and t_d. In addition, we used the average time displayed on the API of CPLEX after execution as calculation time. This is because, due to the nature of CPLEX, there is a slight variation in computation time per execution. We show the result of EX2 on the right side of Table 2. The time values of Scenario No. 5 and No. 6 are the averages of five executions, and the ones of the others are averages of ten executions. As Table 2, although the baseline takes less than 1 second to get the solution in all six scenarios from No. 1 to No. 6, our model takes more than 20 seconds at least to get the solution in the same scenarios. Moreover, with same t_d, when d_l is 1 and d_t is 1, the calculation time is longer than otherwise. On the other hand, with same d_l and d_t, the larger t_d, the longer calculation time. Especially, It took more than 1000 seconds to get the solution in No. 5 and

No. 6.averages of ten executions. Although the baseline takes less than 1 second to get the solution in all of six scenarios from No. 1 to No. 6, our model takes more than 20 seconds at least to get the solution in the same scenarios. Moreover, with same t_d, when d_l is 1 and d_t is 1, the calculation time is longer than otherwise. On the other hand, with same d_l and d_t, the larger t_d, the longer calculation time. Especially, It took more than 1000 seconds to get the solution in No. 5 and No. 6.

7 Discussions

First, we conclude RQ1: how much the total delay is reduced by our constraints of the moving block. In six scenarios from No.1 to No.6 in which we got solutions from both two models in EX1, we confirmed the reduction of total delay for about from 20% to 30% in our model compared with the baseline. Seeing the detailed branching patterns of their solutions, no less than one train run branching into (TB), i.e., patterns of running with especially close distances with the preceding train, in the terminal section in all of the six scenarios. All arrivals and departures of trains at the terminal after running with (TB) of our model were about 30 to 60 seconds earlier than ones of the baseline. Therefore, it can be said that running with closer distances with the preceding train branching into such assumed patterns made the departures and arrivals earlier due to the moving block we assumed. In the discussions above, we confirmed the delay reduction we saw in the six scenarios is the effect of our model with the moving block. Furthermore, even in No. 7 and No. 8 our model could get solutions whereas the baseline could not. The reason for this could be that the baseline did not allow some trains to select one ATO level in some sections due to constraints keeping long distances with each other, whereas our model allowed all trains to run with more close distances with each other, and every train could select one ATO level in each section. In addition, from the result of EX1, it can be said the larger number of delayed trains, the smaller the reduction rate. It can be thought this is because, if the number of trains or sections to be considered for delay reduction is large, the reduction will need to be more spread out at each train and section. Moreover, it can be also said the larger the first delay, the smaller the reduction rate. It can be thought this is because, if the first delay is larger, our model can enlarge the range of reduction of each train's delay in each section.

Next, we conclude RQ2: how long it does take to run the model. We confirmed our model takes more than 20 seconds at least to get the solution in all scenarios in EX2, although the baseline takes less than 1 second in each. The reason for this could be the difference of the number of decision variables whose solutions have to be got with the solver. The baseline has less than 500 decision variables in all scenarios, whereas our model has over 9000 decision variables. This is because our model includes variables representing which pattern each train branches into in each section, as we mentioned in Section 6.1. In addition, from the result of EX2, it can be said the larger number of delayed trains, the longer the calculation time. It can be thought this is because, if the number of delayed trains becomes larger, the number of sections in which the solver has to decide which pattern the successors of such trains branch into will increase. Moreover, it can be also said the first delay is larger, the longer the calculation time. The reason for this could be that the range of solution candidates for rescheduled arrival and departure times will be enlarged if the first delay becomes larger. As for No. 5 and No. 6, it can be thought that the combination of these two factors caused the calculation time to rise exponentially. These two values of calculation time are more than 10 minutes larger than t_d. Thus, refining the design of our model to reduce the number of variables and shorten calculation time is a future challenge.

8 Related Work

From the discussions of Section 7, if we apply our model to a real railway line with a larger size than the sizes of the imaginary small line we used in experiments in Section 6, it is expected that it takes more calculation time than ones in Table 2. Thereby we introduce the study [8] as related work. This study is to divide a rescheduling problem in a large railway line by sections or time periods and solve it as a superposition of smaller MIP problems. Combining this method with our model, it can be thought that it is possible to apply our model to a real large railway line keeping calculation time shorter.

9 Conclusions

The purpose of this study is to formalize train rescheduling considering both a moving block and ATO in bidirectional double-track railway lines as a MIP model. To achieve this purpose, we proposed to formulate constraints for trains' running in terminal sections of a bidirectional double-track line. We implemented an MIP model by integrating our constraints with the models proposed in the previous study [4] [5]. We demonstrated the feasibility of our approach by applying it to a bidirectional double-track line with eight delay scenarios. In these scenarios, generated schedules of our approach reduced the total delay from 20% to 30% than one from the baseline, whereas the computation time rose from less than 1 second to about 20 seconds at least. In future work, We will refine the design of our model of this study to reduce the number of variables and shorten calculation time and apply it to larger railway lines and longer time periods than the experiments we did. Furthermore, our assumption of the moving block has two simplifications mentioned in Section 4.2. Therefore, removing them and incorporating assumptions of more complex dynamics of a moving block into the model is another future challenge.

References

1 Valentina Cacchiani, Dennis Huisman, Martin Kidd, Leo Kroon, Paolo Toth, Lucas Veelenturf, and Joris Wagenaar. An overview of recovery models and algorithms for real-time railway rescheduling. *Transportation Research Part B: Methodological*, 63:15–37, 2014. `doi:10.1016/j.trb.2014.01.009`.

2 Wei Fang, Shengxiang Yang, and Xin Yao. A survey on problem models and solution approaches to rescheduling in railway networks. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):2997–3016, 2015. `doi:10.1109/TITS.2015.2446985`.

3 Zhenhuan He. Research on improved greedy algorithm for train rescheduling. In *2011 Seventh International Conference on Computational Intelligence and Security*, pages 1197–1200, 2011. `doi:10.1109/CIS.2011.265`.

4 Zhuopu Hou, Hairong Dong, Shigen Gao, Gemma Nicholson, Lei Chen, and Clive Roberts. Energy-saving metro train timetable rescheduling model considering ato profiles and dynamic passenger flow. *IEEE Transactions on Intelligent Transportation Systems*, 20(7):2774–2785, 2019. `doi:10.1109/TITS.2019.2906483`.

5 Kosuke Kawazoe, Takuto Yamauchi, Kenji Tei, Norio Tomii, and Shinichi Honiden. Applying mip train traffic rescheduling model with automatic train control to moving block systems (japanese edition). *Journal of Information Processing (Japanese Edition)*, 63(3), 2022. `doi:10.20729/00217476`.

6 Philippe Laborie, Jérôme Rogerie, Paul Shaw, and Petr Vilím. Ibm ilog cp optimizer for scheduling. *Constraints*, 23(2):210–250, 2018. `doi:10.1007/s10601-018-9281-x`.

7 Leonardo Lamorgese, Carlo Mannino, Dario Pacciarelli, and Johanna T. Krasemann. *Train Dispatching*. Springer, 2018. `doi:978-3-319-72153-8_12`.

8 Leonardo Lamorgese, Carlo Mannino, and Mauro Piacentini. *Advances and Trends in Optimization with Engineering Applications*. Society for Industrial and Applied Mathematics, 2017. `doi:10.1137/1.9781611974683.ch6`.

9 N. Miyaguchi, D. Uchiyama, I. Inaba, Y. Baba, and N. Hiura. The radio-based train control system atacs. In *WIT Transactions on The Built Environment*, volume 155, pages 175–183, 2015. `doi:10.2495/CRS140151`.

10 Robert D. Pascoe and Thomas N. Eichorn. What is communication-based train control? *IEEE Vehicular Technology Magazine*, 4(4):16–21, 2009. `doi:10.1109/MVT.2009.934665`.

11 Juliette Pochet, Sylvain Baro, and Guillaume Sandou. Supervision and rescheduling of a mixed cbtc traffic on a suburban railway line. In *2016 IEEE International Conference on Intelligent Rail Transportation (ICIRT)*, pages 32–38, 2016. `doi:10.1109/ICIRT.2016.7588547`.

12 Juliette Pochet, Sylvain Baro, and Guillaume Sandou. Automatic train supervision for a cbtc suburban railway line using multiobjective optimization. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, 2017. `doi:10.1109/ITSC.2017.8317670`.

13 Kei Tamura, Keisuke Sato, and Norio Tomii. A train timetable rescheduling mip formulation with additional inequalities minimizing inconvenience to passengers. *The IEICE Transactions on Information and Systems*, 97(3):393–404, 2014.

14 Yihui Wang, Bing Ning, Fang Cao, Bart De Schutter, and Ton J.J. van den Boom. A survey on optimal trajectory planning for train operations. In *Proceedings of 2011 IEEE International Conference on Service Operations, Logistics and Informatics*, pages 589–594, 2011. `doi:10.1109/SOLI.2011.5986629`.

15 Chao Wen, Ping Huang, Zhongcan Li, Javad Lessan, Liping Fu, Chaozhe Jiang, and Xinyue Xu. Train dispatching management with data- driven approaches: A comprehensive review and appraisal. *IEEE Access*, 7:114547–114571, 2019. `doi:10.1109/ACCESS.2019.2935106`.

16 Peijuan Xu, Dawei Zhang, Jingwei Guo, Dan Liu, and Hui Peng. Integrated train rescheduling and rerouting during multidisturbances under a quasi-moving block system. *Journal of Advanced Transportation*, 2021:1–15, April 2021. `doi:10.1155/2021/6652531`.

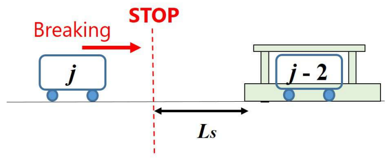
17 Jing Xun, Bin Ning, Ke-Ping Li, and Tao Tang. An optimization approach for real-time headway control of railway traffic. In *2013 IEEE International Conference on Intelligent Rail Transportation Proceedings*, pages 25–31, 2013. `doi:10.1109/ICIRT.2013.6696262`.

18 Darja Šemrov, R. Marsetič, Marijan Zura, Ljupco Todorovski, and Aleksander Srdić. Reinforcement learning approach for train rescheduling on a single-track railway. *Transportation Research Part B Methodological*, 86:250–267, April 2016. `doi:10.1016/j.trb.2016.01.004`.

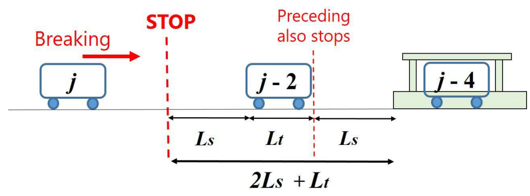
A Assumptions In On-the-way Sections

In this appendix section, we describe the assumptions of the model of Kawazoe et al. [5] to describe constraints in Appendix B.1, which we used to implement our model in Section 6.1. In the model of Kawazoe et al. [5], they divided the runs of trains in the unidirectional single-track lines into two large patterns below, and for each pattern, they assumed how a train can approach the preceding train and how the moving block auto brakes work. We rewrite and describe these two patterns according to assumptions in Section 4.1.

- Pattern (OA): In the on-the-way sections, we assume when the train departs from the station after the preceding arrived at the next. As Fig. 4a shows, in this pattern, the following train can approach up to L_s (m) behind the stopping at the next.
- Pattern (OB): In on-the-way sections, we assume when the train departs from the station before the preceding arrived at the next. In this pattern, the following train can approach up to L_s (m) behind the preceding as same as (OA). However, in this pattern, the preceding also might stop as (OA) to prevent the collision with one earlier train. If so,



(a) Approaching when (OA) in on-the-way sections.



(b) Approaching when (OB) in on-the-way sections.

Figure 4 Assumed approaches due to the assumed the moving block in on-the-way sections.

totally the following can approach up to just $2L_s + L_t$ (m) behind the next station, as Fig. 4b shows. In addition, Kawazoe et al. [5] made the same simplification as (TA). In the example of 4b, they assumed that train j always only can approach up to just $2L_s + L_t$ (m) behind the next station, when the preceding $j - 2$ is still on the way at the time j has to start deceleration to stop with auto brakes. This is not up to whether the one earlier $j - 4$ is still stopping at the next station then.

B Constraints Based On Previous Models

In this appendix section, we describe the constraints based on previous models of [4] and [5]. These constraints are combined with our new constraints in Section 5, in order to implement our model in Section 6.1.

B.1 Running Constraints in On-the-way Sections

In Section 5, we formulated constraints for trains' running at terminal sections in bidirectional double-track lines. On the other hand, Kawazoe et al. [5] formulated constraints of trains' running in unidirectional single-track lines, which are based on their moving block assumptions we mentioned in Appendix A. For on-the-way sections in our model, we use constraints rewritten from these constraints in their model according to our assumptions in Section 4.1. We describe those rewritten constraints following each of the two patterns mentioned in Appendix A.

B.1.1 Pattern (OA)

In Pattern (OA), the train j departs from the station after the preceding $j - 2$ arrived at the next station. In this pattern, j has to decide whether it applies the brakes to stop with L_s to the next as Fig.4a shows. Kawazoe et al. [5] set the time of brake application as $t_{1A(k,l)}$, and they further divided (OA) into these two cases for j below.

- (OA1): $j - 2$ already leaves the next station at $t_{1A(k,l)}$
- (OA2): $j - 2$ is still stopping at the next at $t_{1A(k,l)}$

In (OA1), j need not use brakes to stop in front of the next station. Therefore, j just takes ATO running time R^l_k to the next station. This is as the same as (TA1) in Section 5.1, and this can be formulated as Constraint (19) (up) or (20) (down).

$$ta_{i+1,j} - td_{i,j} = \varepsilon^l_{k,j} R^l_{k,j} \tag{19}$$
$$ta_{i,j} - td_{i+1,j} = \varepsilon^l_{k,j} R^l_{k,j} \tag{20}$$

On the other hand, in (OA2), j need use brakes to stop with L_s in front of the next station, and wait for the departure of $j - 2$. j can arrive at next station after both of that j

stops at once and the departure of $j-2$. They set the time j stops as $t_{2A(k,l)}$. Furthermore, after j starts moving again, j has to runs L_s and the length of j itself L_t to arrive at the station. They set the constant time j runs L_s and L_t as T_{LsLt}, which is not up to j and j's ATO level l. Putting all of them together, in (OA2) j has to satisfy Constraint (21) (up) or (22) (down).

$$ta_{i+1,j} \geq \max(td_{i+1,j-2}, td_{i,j} + \varepsilon^l_{k,j} t_{2A(k,l)}) + T_{LsLt} \tag{21}$$
$$ta_{i,j} \geq \max(td_{i,j-2}, td_{i+1,j} + \varepsilon^l_{k,j} t_{2A(k,l)}) + T_{LsLt} \tag{22}$$

B.1.2 Pattern (OB)

In Pattern (OB), j departs from the station before $j-2$ arrived at the next station. In this pattern, j has to decide whether it applies the brakes to stop with $2L_s + L_t$ to the next as Fig.4b shows. Kawazoe et al. [5] set the time of brake application as $t_{1B(k,l)}$, and they further divided (OB) into these two cases for j below.

- (OB1): $j-2$ already arrived at the next station at $t_{1B(k,l)}$
- (OB2): $j-2$ is still on the way to the next at $t_{1B(k,l)}$

In (OB1), j need not use brakes at $t_{1B(k,l)}$. j just has to use brakes to stop as same as (OA2) at $t_{1B(k,l)}$, in order to wait for the departure of $j-2$ from the next station. Therefore, in (OB1), the constraint which j has to satisfy is the same one as (OA2), i.e., Constraint (21) or (22).

On the other hand, in (OB2), j need use brakes to stop with $2L_s + L_t$ in front of the next station. This is because of simplification of Kawazoe et al. [5] we mentioned when we describe Pattern (OB) in Appendix A. If $j-4$ which is the preceding of $j-2$ is still on the way to the next station at $t_{1B(k,l)}$, j has to wait the departures of both of $j-4$ and $j-2$. $j-2$ will arrive at the next station after $j-4$ leave there. At the same time, j starts to move, runs $L_s + L_t$ and stops again to wait for $j-2$ as same as (OA2). Putting all of them together, in (OB2) j has to satisfy Constraint (23) (up) or (24) (down).

$$ta_{i+1,j} \geq \max(td_{i+1,j-4} + T_{LsLt}, td_{i+1,j-2}) + T_{LsLt} \tag{23}$$
$$ta_{i,j} \geq \max(td_{i,j-4} + T_{LsLt}, td_{i,j-2}) + T_{LsLt} \tag{24}$$

B.2 Order Constraints

Here, we describe the constraints which keeps orders of trains. These constraints are also rewritten from the constraints in the model of Kawazoe et al. [5] according to assumptions in Section 4.1, which has double tracks and bidirectional operation.

$$td_{i,j} > td_{i,j-2} \tag{25}$$
$$ta_{i,j} > ta_{i,j-2} \tag{26}$$

If i is not the terminal station, then

$$ta_{i,j} > td_{i,j-2} \tag{27}$$

Constraint (25) and (26) mean the arrival and departure of each train is earlier than one of its preceding train at any station in the lines. Constraint (27) means the arrival of each train at any station excluding the terminal is earlier than the departure of its preceding train from that station.

B.3 Other Constraints

Here, we describe the other constraints to use in our model mentioned in Section 6.1. These constraints are rewritten from the constraints in the model of Hou et al. [5] according to assumptions in Section 4.1, which has double tracks and bidirectional operation.

First, any train must satisfy the one ATO level in each section. This can be formulated with $\epsilon^l_{k,j}$ as below.

$$\sum_l \epsilon^l_{k,j} = 1 \tag{28}$$

Next, any train must satisfy the range of the dwelling time length at each station. Using the maximum allowed dwelling time constants Dw^{max}_i and the minimum ones Dw^{min}_i, this dwelling time range can be formulated as these constraints below.

$$td_{i,j} - ta_{i,j} \geq Dw^{min}_i \tag{29}$$

If i is not d_l or j is not d_t, then

$$td_{i,j} - ta_{i,j} \leq Dw^{max}_i \tag{30}$$

Finally, any train cannot depart from and arrive at each station before the original time schedule $Td_{i,j}$, $Ta_{i,j}$. This can be formulated as these constraints below.

$$ta_{i,j} \geq Ta_{i,j} \tag{31}$$

$$td_{i,j} \geq Td_{i,j} \tag{32}$$

In addition, they added these constraints to minimize the deviation between the original timetable and the rescheduled timetable [4].

C Details Of Experiment Setting

In this appendix section, we describe the details of the experiment setting we described in Section 6.1.

C.1 ATO and Moving Block

Here, we describe detailed assumptions and values of constants about ATO and the moving block to implement our model and do experiments in Section 6. If Train j runs the whole of Section k following ATO Level l without the moving block auto brakes, it takes $R^l_{k,j}$ (s) as Constraint (19) and (20). we set the values of ATO running time $R^l_{k,j}$ as Table 3 shows. Furthermore, we set the simple profile of this ATO running in our experiments as follows.

- After departure from any station, any train accelerates with a constant rate r_{ac}=1.0 (m/s^2) in any ATO level. Each ATO level has a unique ATO maximum speed $V^{max}_{k,l}$ in each section. The smaller the number of "l", the larger the value of $V^{max}_{k,l}$. Any train accelerates until the speed of the train reach $V^{max}_{k,l}$.
- After the acceleration, any train keeps $V^{max}_{k,l}$ until it approaches the next station, unless the moving block automatic brakes are activated as defined in Section 4.2.
- When any train approaches the next station, it slows down at a constant deceleration rate r_{de}=1.0 (m/s^2) and stops at the next station in any ATO level.

Table 3 Values of $R^l_{k,j}$ (s).

k	$l=1$	$l=2$	$l=3$	$l=4$	$l=5$
1	63	73	83	93	118
2	105	115	125	135	160
3	123	133	143	153	178

Table 4 Values of x_k (m) and $V^{max}_{k,l}$ (m/s).

		$V^{max}_{k,l}$				
k	x_k	$l=1$	$l=2$	$l=3$	$l=4$	$l=5$
1	839	19.12	14.29	11.78	10.12	7.60
2	1564	17.97	15.75	14.10	12.80	10.46
3	1649	15.31	13.83	12.65	11.66	9.80

In this setting, the following quadratic equation holds for $V^{max}_{k,l}$ using $R^i_{k,j}$ and the constant length of each section x_k.

$$x_k = \frac{V^{max\,2}_{k,l}}{2r_{ac}} + \frac{V^{max\,2}_{k,l}}{2r_{de}} + V^{max}_{k,l}(R^i_{k,j} - \frac{V^{max}_{k,l}}{r_{ac}} - \frac{V^{max}_{k,l}}{r_{de}}) \tag{33}$$

We can set the values of $V^{max}_{k,l}$ by solving this equation. We shows the setting of x_k and $V^{max}_{k,l}$ in Table 4. In addition, the value setting of $R^i_{k,j}$ and x_k is with reference to the experiment setting of Hou et al. [4]

Moreover, we set the moving block deceleration rate as Constant r_{mbs}=1.0 (m/s^2). We also set the constant lengths L_s, L_t, and L_u as all 120 (m). From these constants and $V^{max}_{k,l}$ above, we can calculate and set the time of starting brake application of the moving block to stop with a unique distance in front of the next station up to each of Pattern (TA), (TB), (OA), and (OB) in Section 5 and B.1. For example, when Train j runs in Pattern (TA2-1) in Section 5.1, j has to stop with $Ls + Lu$ (m) in front of the terminal and wait for $r(j-2)$. In this situation, when j can start braking application after its speed reaches $V^{max}_{k,l}$, the time from departure from the last station to starting brake application is $t_{T1A(k,l)}$, and we can calculate and set $t_{T1A(k,l)}$ as follows.

$$t_{T1A(k,l)} = \frac{V^{max}_{k,\,l}}{r_{ac}} + \frac{x_k - (L_s + L_u) - \frac{(V^{max}_{k,\,l})^2}{2r_{mbs}} - \frac{(V^{max}_{k,\,l})^2}{2r_{ac}}}{V^{max}_{k,\,l}} \tag{34}$$

On the other hand, when j has to start braking application before its speed reaches $V^{max}_{k,l}$, we can calculate and set $t_{T1A(k,l)}$ as follows.

$$t_{T1A(k,l)} = \sqrt{2(x_k - (L_s + L_u))\frac{r_{mbs}}{r_{ac}(r_{mbs} + r_{ac})}} \tag{35}$$

Table 5 Values of Dw^{max}_i (s) and Dw^{min}_i (s).

i	1	2	3	4
Dw^{max}_i	105	90	90	100
Dw^{min}_i	40	25	25	40

These calculations are with reference to Kawazoe et al. [5] Similarly, we can calculate and set the values of $t_{T1B(k,l)}$ in Pattern (TB), $t_{1A(k,l)}$ in Pattern (OA), and $t_{1B(k,l)}$ in Pattern (OB). Furthermore, from $t_{T1A(k,l)}$, we can also calculate and set $t_{T2A(k,l)}$ in Constraint (3) as follows. In addition, this $t_{T2A(k,l)}$ is the time when a train stops with $L_s + L_u$ (m) in front of the terminal.

$$t_{T2A(k,l)} = t_{T1A(k,l)} + \frac{V_{k,l}^{max}}{r_{mbs}} \tag{36}$$

Similarly, we can calculate and set the values of $t_{T2B(k,l)}$ in Pattern (TB) and $t_{2A(k,l)}$ in Pattern (OA).

In addition to them, we set the time constants of T_{LsLtLu}=35 (s) used in Constraint (3), $T_{2Ls2LtLu}$ =45 (s) used in Constraint (12), and T_{LsLt}=25 (s) used in Constraint (21). Furthermore, we also set the times of t_{sd} and t_{su} used in Constraint (3) and (4). They are the times which it takes from leaving each terminal to passing the switch. For simplification, in our experiments, we set them as constants which it takes from leaving each terminal to passing the switch in ATO level 5, i.e., the level with the lowest $V_{k,l}^{max}$. Therefore, it can be said that the train has already passed the switch whatever its ATO level it is when t_{sd} or t_{su} passed after departure from the terminal. With this simplification, we calculated and set the times as t_{sd} = 37.55 (s) and t_{su} = 48.00 (s).

C.2 Other Constants

Here, we show the other values of constants to use to implement our model and do experiments in Section 6. They are Dw_i^{max} and Dw_i^{min}, which are constants of the range of dwelling time at stations. We show the setting of Dw_i^{max} and Dw_i^{min} in Table 5. The value setting them is with reference to the experiment setting of Hou et al. [4]