ISSN 0005-1179, Automation and Remote Control, 2020, Vol. 81, No. 6, pp. 955–966. © Pleiades Publishing, Ltd., 2020.
Russian Text © The Author(s), 2020, published in Avtomatika i Telemekhanika, 2020, No. 5, pp. 91–105.

TOPICAL ISSUE

Rescheduling Traffic on a Partially Blocked Segment of Railway with a Siding

Y. Zinder[*,a], **A. A. Lazarev**[**,***,b], **and E. G. Musatova**[**,c]

[*] *University of Technology, Sydney, Australia*
[**] *Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia*
[***] *National Research University Higher School of Economics, Moscow, Russia*
e-mail: [a] *Yakov.Zinder@uts.edu.au,* [b] *jobmath@mail.ru,* [c] *nekolyap@mail.ru*

Received July 2, 2019
Revised October 15, 2019
Accepted November 28, 2019

Abstract—The paper presents a polynomial-time algorithm for rescheduling traffic when one track of a double-track railway becomes unavailable, the remaining track has a siding, and there are two categories of trains—priority trains such as passenger trains and ordinary trains such as the majority of freight trains. The presented algorithm minimises the negative effect, caused by the track blockage, first for the priority trains and then for the ordinary trains on the set of all schedules optimal for the priority trains.

Keywords: single-track railway, dynamic programming, rescheduling, polynomial-time algorithm

DOI: 10.1134/S0005117920060016

1. INTRODUCTION

Disruptions such as accidents or track damages often lead to the temporary blockage of a track of a double-track railway segment. In such situations, it is necessary to schedule traffic in both directions on the remaining track with the goal to minimise the impact of the blockage. Railway rescheduling remained an area of active research over several decades [2]. The paper contributes to this research a polynomial-time dynamic programming based optimisation procedure for the case when the remaining track has a siding and there are two categories of trains—priority trains such as passenger trains and ordinary trains such as the majority of freight trains.

More specifically, consider a double-track railway between two points, A and B, where one track is blocked and all trains must be rescheduled on the remaining track. The available track has a siding, i.e. a segment, allowing a train to pass another train. In this case, this another train must be stationary in the siding, whereas the passing train cannot stop in the siding. At most one train can be stationary in the siding at a time. All trains have the same constant speed. The time required for a train to cover the distance between point $s \in \{A, B\}$ and the siding will be denoted by p_s. Without loss of generality, assumed that $p_A \geqslant p_B$.

For the safety reason, two trains can not arrive at the siding simultaneously. Let β be the minimal time between such arrivals. For the same reason, for each end point of the considered segment of the track, any two arrival times at this point, as well as any two departure times, and any pair of arrival time and departure time must differ at least by β. In contrast, two trains can leave the siding simultaneously if they move in different directions. The assumption that all these safety requirements have the same minimal time β simplifies the presentation but is not essential. It is also assumed that $\beta < p_B$.

The set of trains is comprised of two categories—the priority trains and the ordinary trains. For each $s \in \{A, B\}$, let $\bar{s} = \{A, B\} \setminus \{s\}$. Denote by N_s^p and N_s^o the sets of priority and ordinary trains that are to travel along the track in the direction from point s to point $\bar{s}$. It is assumed that, for each $s \in \{A, B\}$, $N_s^p \cup N_s^o \neq \emptyset$, because otherwise the rescheduling problem does not exist.

The original timetable, designed under the assumption that both tracks are operational, allocates to each train $j \in N_s^\alpha$ a time window $[r_j^{s,\alpha}, d_j^{s,\alpha}]$ within which j should pass through the considered double-track segment of the railway network. According to this initial timetable, one track is allocated to all trains, priority and ordinary, moving from A to B, and another track is allocated to all trains, priority and ordinary, travelling in the opposite direction, from B to A. Therefore, for any two trains $j \in N_s^\alpha$ and $j' \in N_s^{\alpha'}$, moving from s to $\bar{s}$, the corresponding time windows satisfy the condition

$$r_j^{s,\alpha} \neq r_{j'}^{s,\alpha'} \quad \text{and} \quad r_j^{s,\alpha} > r_{j'}^{s,\alpha'} \ \text{implies} \ d_j^{s,\alpha} > d_{j'}^{s,\alpha'}. \tag{1}$$

As a result of rescheduling, each train $j \in N_s^\alpha$ is assigned the point in time $S_j^{s,\alpha}$ when this train should enter the remaining railway track, which will be referred to as its departure time from s, and the point in time $C_j^{s,\alpha}$ when this train should leave the track, which will be referred to as its arrival time at $\bar{s}$. Any such new schedule must satisfy the following condition, imposed by the initial time windows:

(s1) for each $s \in \{A, B\}$, each $\alpha \in \{p, o\}$ and each train $j \in N_s^\alpha$, the departure time $S_j^{s,\alpha}$ of this train satisfies the inequality $S_j^{s,\alpha} \geqslant r_j^{s,\alpha}$.

The goal is to find a schedule that minimises the objective function

$$\max_{s\in\{A,B\}} \max_{j\in N_s^p} \left[C_j^{s,p} - d_j^{s,p}\right] \tag{2}$$

and that minimises the objective function

$$\sum_{s\in\{A,B\}} \sum_{j\in N_s^o} \left[C_j^{s,o} - r_j^{s,o}\right] \tag{3}$$

on the set of all schedules that are optimal for (2).

The objective function (3) is only one of the possible measures of the impact of the blockage on the ordinary trains. Thus, the optimisation procedure below can be easily modified to the case when instead of (3) is used

$$\max_{s\in\{A,B\}} \max_{j\in N_s^o} \left[C_j^{s,o} - d_j^{s,o}\right]. \tag{4}$$

There exist a number of publications on scheduling and rescheduling on a single-track railway (some references can be found in [2, 3]). Among them, the publications [1, 4, 5] are the most closely related to this paper. Similar to this paper, [1] is concerned with rescheduling when the original schedule for the fully operational double-track segment of the railway network specifies a time window for each train and considers several categories of trains. In contrast to this paper, [1] assumes that there is no siding and reflects the existence of different types of trains by introducing weights in the objective function.

Another closely related publication is [5], which is concerned with the optimisation of the ordered objective functions, but in contrast to this paper, assumes that all trains are available simultaneously (such situation may occur after the complete blockage of the considered segment of the railway). Furthermore, [5] assumes that there is no siding.

Finally, [4] is concerned with scheduling on a single-track railway which has a siding, but similar to [5] assumes that all trains are available simultaneously. Furthermore, in contrast to the optimisation procedure below, which is designed for two categories of trains and the ordered objective functions, all algorithms, presented in [4], have been designed for a single objective function and assume that all trains have the same type. Despite the mentioned above significant differences between [4] and this paper, several results on the structure of optimal schedules, obtained in [4], allow a straightforward generalisation to the case when the original schedule specifies for each train a time window. Furthermore, the main insight of [4] that there exists an embedded Bellman decision process associated with the departure times of certain trains remains valid for this paper, although the set of states and the implementation of the general scheme of dynamic programming are different.

2. EXPRESSES AND NON-EXPRESSES

Consider an arbitrary schedule for the track that remained available after the blockage. As in [4], a train that does not stop at the siding will be referred to as an express, whereas a train that stops at the siding will be called a non-express. The notions of express and non-express are unrelated to the notions of priority and ordinary trains. In other words, any priority train as well as any ordinary train can be either an express or a non-express depending on the schedule. The set of all expresses that pass the same non-express will be called a batch. For any point in time t, a train $j \in N_s^\alpha$ will be called active at t if

$$S_j^{s,\alpha} \leqslant t \leqslant C_j^{s,\alpha}.$$

If trains $j \in N_s^\alpha$ and $j' \in N_{\bar{s}}^{\alpha'}$, i.e. two trains that are moving in the opposite directions, are active simultaneously at some point in time, then they are moving towards each other at the point in time $\max[S_j^{s,\alpha}, S_{j'}^{\bar{s},\alpha'}]$. Consequently, these two trains must be an express and a non-express that this express passes at the siding.

Since, at any point in time, at most one train can be stationary at the siding, if train $j \in N_s^\alpha$ passes any train j' that travels in the same direction as j, i.e. from s to $\bar{s}$, then, in the time interval $[S_j^{s,\alpha}, C_j^{s,\alpha}]$, there are no active trains, moving from $\bar{s}$ to s. Hence, instead of waiting for j, the train j' can leave the siding at least at the point in time $S_j^{s,\alpha} + p_s - \beta$ which can only improve the value of the objective function because this function is nondecreasing.

Since both objective functions, (2) and (3), are nondecreasing, the discussion above implies that without loss of generality it suffices to consider only schedules that in addition to (s1) satisfy the conditions:

(s2) For each $s \in \{A, B\}$ and each $\alpha \in \{p, o\}$, for any two trains $j \in N_s^\alpha$ and $j' \in N_s^\alpha$, the inequality $r_j^{s,\alpha} > r_{j'}^{s,\alpha}$ implies $S_j^{s,\alpha} > S_{j'}^{s,\alpha}$.

(s3) For any non-express, there exists at least one express that passes this non-express.

(s4) All expresses that pass the same non-express travel in the opposite direction to the direction of this non-express.

(s5) A non-express leaves the siding simultaneously with the last express that passes this non-express.

Two expresses, travelling in the opposite directions, cannot be active simultaneously. Furthermore, two expresses, travelling in the same direction, say from s to $\bar{s}$, can depart from s only at points in times that are at least β apart. Hence, all departure times of the expresses are different.

Several factors determine the minimal possible difference between two consecutive departure times of expresses. If, as in [4], all trains are available simultaneously, these factors are the directions in which the expresses travel and the situations at the siding when these expresses arrive at it. All possible combinations of these factors are specified in [4] by associating with each express a pair (s, b), referred to as its type. Here, s indicates that the express travels in the direction from s to $\bar{s}$, whereas b reflects the situation at the siding and assumes the following values:

- 0, if the express goes through an empty siding;
- 1, if the express is part of a batch and is not last in this batch;
- 2, if the express is the last in a batch.

Let $i \in N_s^\alpha$ and $i' \in N_{s'}^{\alpha'}$ be two consecutive expresses of types (s, b) and (s', b') respectively. Let the departure time of i be equal to t. Assume that $b \neq 1$, $b' \neq 0$, and that express i' passes some non-express $g' \in N_{\bar{s}'}^\gamma$. According to [4], the departure time of express i determines the following earliest possible departure time of non-express g':

$$\hat{\tau} = \begin{cases} t + p_A + p_B + \beta, & \text{if } \ s = s' \\ t + \beta, & \text{if } \ s \neq s', \ b = 0 \\ t + 2p_s + \beta, & \text{if } \ s \neq s', \ b = 2. \end{cases} \tag{5}$$

Indeed, if $s = s'$, then g' traverses the track from $\bar{s}$ to s. By virtue of $b \neq 1$, trains g' and i cannot be active simultaneously. Hence, g' can depart from $\bar{s}$ only β after the arrival of i, which arrives at $\bar{s}$ at $t + p_A + p_B$. If $s \neq s'$, then both, i and g', move from s to $\bar{s}$. Therefore, if $b = 0$, then the departure times of g' and i can differ only by β, whereas if $b = 2$, then g' can leave s only β after the arrival at s of the train that i passes in the siding. The latter leaves the siding at the same time as i, that is at $t + p_s$, and after that needs p_s time units to reach s.

In contrast to [4], which assumes that all trains are at the respective endpoints of the track simultaneously, this paper is concerned with the rescheduling problem which takes into account the time windows assigned to trains by the initial timetable. So, there exists one more restriction on the earliest possible departure time imposed by the train's original time window. Hence, the earliest possible departure time of g' from the corresponding endpoint of the track is

$$\tau = \max\left\{\hat{\tau}, r_{g'}^{\bar{s}',\gamma}\right\}. \tag{6}$$

If the time window for i' is not considered, then, according to [4], the departure times t and τ (if train g' exists) determine the following earliest departure time of i':

$$\hat{t} = \begin{cases} t + \beta, & \text{if } s = s' \text{ and } b = 1 \\ t + \beta, & \text{if } s = s' \text{ and } b = b' = 0 \\ \max\{t + 2p_s + \beta, \tau + p_{\bar{s}'} + \beta - p_{s'}\}, & \text{if } s = s', \ b = 2, \ b' \neq 0 \\ t + 2p_s + \beta, & \text{if } s = s', \ b = 2, \ b' = 0 \\ \tau + p_{\bar{s}'} + \beta - p_{s'}, & \text{if } s = s', \ b = 0, \ b' \neq 0 \\ t + p_A + p_B + \beta, & \text{if } s \neq s', \ b' = 0 \\ \max\{t + p_A + p_B + \beta, \tau + p_{\bar{s}'} + \beta - p_{s'}\}, & \text{if } s \neq s', \ b' \neq 0. \end{cases} \tag{7}$$

Taking into account the restriction, imposed by the time window for i', its earliest possible departure time is

$$\max\left\{\hat{t}, r_{i'}^{s',\alpha'}\right\}. \tag{8}$$

Let express i of type (s, b) have the earliest departure time among all expresses. Let g be a train which departs earlier than i. Then, g is a non-express. This, by virtue of (s3) and (s4), implies that g departs from $\bar{s}$ and is stationary in the siding when i goes through the siding. Hence, i is the first train from s. Furthermore, since any express can pass at most one non-express, g is the first train from $\bar{s}$.

For each $s \in \{A, B\}$ and each $\alpha \in \{p, o\}$, let n_s^α be the cardinality of the set N_s^α and number all trains $j \in N_s^\alpha$ in the decreasing order of $r_j^{s,\alpha}$ or equivalently in the decreasing order of $d_j^{s,\alpha}$. Let the first express in the sequence of expresses has type (s, b) and is a train of category α which passes a non-express of category γ (if it exists). Then, taking into account the condition (s2), the departure time of this first express is

$$t = \begin{cases} r_{n_s^\alpha}^{s,\alpha}, & \text{if } \ b = 0 \\ \max\left\{ r_{n_s^\alpha}^{s,\alpha}, r_{n_{\bar{s}}^\gamma}^{\bar{s},\gamma} + p_{\bar{s}} + \beta - p_s \right\}, & \text{if } \ b \neq 0. \end{cases} \tag{9}$$

Using the same reasonings as above, it is easy to see that if express i of type (s, b) has the latest departure time among all expresses and g is a train which departs later than i, then i is the last train from s, g is the last train from $\bar{s}$, and g is stationary in the siding when i goes through the siding.

Let i and i' be two consecutive expresses of types (s, b) and (s', b'), respectively, where the departure time of i' is greater than the departure time of i. As has been shown above, the types of these expresses together with the departure time of i completely determine the earliest possible departure time of i'. Furthermore, if i passes a non-express g and $b = 2$, then (s, b) and the departure time of i completely determine the time when g arrives at s; and if i' passes a non-express g' and $b \neq 1$, then the departure time of i and the types of these two expresses completely determine the earliest departure time of g' from $\bar{s}'$. In addition, as has been shown above, the type of the first express completely determines the earliest departure time of this express and the earliest departure time of the train which this express passes in the siding if such train exists. These observations permit to construct the desired schedule by considering only the departure times of expresses and by assigning to each train the earliest possible departure time from the corresponding departure point.

3. MINIMISATION OF THE MAXIMAL LATENESS

This section shows how to find the optimal value of the objective function (2). The objective function (2) involves only one category of trains, and therefore this optimisation problem is similar to that in [4] with one essential difference—in contrast to [4], the considered rescheduling problem takes into account time the windows, assigned to the trains prior to the occurrence of the blockage. This necessitates the inclusion of time into the definition of a state, which in turn changes the implementation of the general dynamic programming framework for the maximum lateness problem.

Consider an arbitrary schedule (recall that this section is concerned only with the priority trains and the existence of all ordinary trains is ignored) and an arbitrary express in this schedule. Let (s, b) be the type of this express and t be its departure time. The number of priority trains that enter the considered railway track at or after the point in time t and that traverse the track from s to $\bar{s}$ will be denoted by k_s. Let $k_{\bar{s}}$ be the cardinality of the set of priority trains each of which is either the train that the considered express passes in the siding, or that traverses the track from $\bar{s}$ to s and departs from $\bar{s}$ at or after t, or both. The departure of the considered express is associated with the tuple (t, k_A, k_B, s, b). According to the terminology of dynamic programming, this tuple will be referred to as a state. It is easy to see that if the departure of an express is associated with state (t, k_A, k_B, s, b), then this express belongs to the set N_s^p and, by virtue of (s2) and the way in which the trains were numbered, is the train number k_s.

State (t, k_A, k_B, s, b) allows to compute

$$C_{k_s}^{s,p} - d_{k_s}^{s,p} = t + p_A + p_B - d_{k_s}^{s,p}$$

and, in the case of $b = 2$,

$$C_{k_{\bar{s}}}^{\bar{s},p} - d_{k_{\bar{s}}}^{\bar{s},p} = t + 2p_s - d_{k_{\bar{s}}}^{\bar{s},p}.$$

Denote

$$L(t, k_A, k_B, s, b) = \begin{cases} \max\left\{t + p_A + p_B - d_{k_s}^{s,p}, t + 2p_s - d_{k_{\bar{s}}}^{\bar{s},p}\right\}, & \text{if } b = 2 \\ t + p_A + p_B - d_{k_s}^{s,p}, & \text{otherwise.} \end{cases}$$

Each schedule induces the sequence of states where the states are listed in the increasing order of the corresponding departure times. Consider all schedules which sequences of states contain (t, k_A, k_B, s, b). Let $F(t, k_A, k_B, s, b)$ be the minimal value of

$$\max_{s\in\{A,B\}} \max_{j\in\{1,\ldots,k_s\}} \left[C_j^{s,p} - d_j^{s,p}\right]$$

on the set of all these schedules. Denote by $\Omega(t, k_A, k_B, s, b)$ the set of all states such that each of them immediately follows (t, k_A, k_B, s, b) in at least one of the above mentioned sequences.

According to the Section 2, the last express is also the last train traversing the railway track in the respective direction. Furthermore, after the departure of this express, at most one train can traverse the track in the opposite direction and the last express passes this train in the siding. Therefore, if state (t, k_A, k_B, s, b) is associated with the departure of the last express, then k_x in (t, k_A, k_B, s, b) is

$$k_x = \begin{cases} 1, & \text{for } x = s \\ 0, & \text{for } x = \bar{s} \text{ and } b \neq 2 \\ 1, & \text{for } x = \bar{s} \text{ and } b = 2. \end{cases} \tag{10}$$

This implies that

$$F(t, k_A, k_B, s, b) = L(t, k_A, k_B, s, b). \tag{11}$$

If in some schedule, which sequence of states contains (t, k_A, k_B, s, b), the express $k_s \in N_s^p$ is not the last express in this schedule, then it is not the last express in all schedules which sequences of states contain (t, k_A, k_B, s, b). Hence

$$F(t, k_A, k_B, s, b) = \max\left\{L(t, k_A, k_B, s, b), \min_{(t',\hat{k}_A,\hat{k}_B,s',b')\in\Omega(t,k_A,k_B,s,b)} F(t', \hat{k}_A, \hat{k}_B, s', b')\right\}. \tag{12}$$

As has been discussed in Section 2, the first express is also the first train traversing the railway track in the respective direction and its departure time can be computed using (9). Let X be the set of all states associated with all possible choices of the first express and its type. Then, the optimal value of (2) can be written as

$$\min_{(t,n_A^p,n_B^p,s,b)\in X} F\left(t, n_A^p, n_B^p, s, b\right). \tag{13}$$

So, taking into account (11), (12), and (13), the optimal value of (2) can be obtained by dynamic programming. Further details are provided in Section 5.

4. MINIMISATION OF THE TOTAL TIME IN SYSTEM

This section is concerned with the problem of minimising (3) on the set of all schedules that are optimal for (2). Let F^* be the optimal value of (2). Any schedule is optimal for (2) if and only if

$$C_i^{s,p} \leqslant d_i^{s,p} + F^*, \quad \text{for each } s \in \{A, B\} \text{ and each } i \in N_s^p. \tag{14}$$

In other words, it is necessary to find a schedule that has the smallest value of (3) among all schedules satisfying the condition (14). Therefore, in this section, only schedules that satisfy (14) will be considered.

In contrast to Section 3, which was concerned only with the priority trains, now all trains are considered. So, more information is associated with the departure of each express and the definition of a state is changed accordingly. Now a state is a tuple $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$, where, as in Section 3, t and the pair (s, b) are the departure time and the type of the corresponding express, but in addition, α is the category of this express and γ is the category of the train (if such train exists) that this express passes in the siding. If such train does not exist, then γ assumes any value from $\{p, o\}$. Whether or not the express passes a train, $k_{\bar{s}}^{\gamma}$ is the cardinality of the subset of $N_{\bar{s}}^{\gamma}$ that is comprised of all trains that arrive at s after t. Each other k_x^u in $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ is the number of trains that depart from x at or after the point in time t. In particular, by virtue of (s2), state $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ is associated with the departure of the train from N_s^{α} which number is k_s^{α}.

If $\alpha = o$, then state $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ provides information for computing

$$C_{k_s}^{s,o} - r_{k_s}^{s,o} = t + p_A + p_B - r_{k_s}^{s,o},$$

and if $\gamma = o$ and $b = 2$, then this state also allows to compute

$$C_{k_{\bar{s}}}^{\bar{s},o} - r_{k_{\bar{s}}}^{\bar{s},o} = t + 2p_s - r_{k_{\bar{s}}}^{\bar{s},o}.$$

Hence, the contribution to the value of the objective function (3) of the express, which departure is associated with $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$, and the train that this express passes in the siding (if such train exists) is

$$R(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma) = \begin{cases} 2t + 3p_s + p_{\bar{s}} - r_{k_s}^{s,o} - r_{k_{\bar{s}}}^{\bar{s},o}, & \text{for } \alpha = o, b = 2, \gamma = o \\ t + p_A + p_B - r_{k_s}^{s,o}, & \text{for } \alpha = o, b = 2, \gamma \neq o \\ t + p_A + p_B - r_{k_s}^{s,o}, & \text{for } \alpha = o, b \neq 2 \\ t + 2p_s - r_{k_{\bar{s}}}^{\bar{s},o}, & \text{for } \alpha \neq o, b = 2, \gamma = o \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Being based on the same concepts of dynamic programming, the optimisation procedures for (2) and (3) have many similarities despite of the several important difference—the different objective functions; the condition (14); and the different sets of trains. Furthermore, the optimisation approach, described in this section, is applied only after the conclusion on the minimisation of (2). Therefore, the use of the same notations Ω and F below will not cause any confusion but rather will stress the similarity and will facilitate the presentation in Section 5.

Consider all schedules, which induced sequences of states contain some state $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$. Recall that in this section only the schedules satisfying (14) are considered. As before, in each induced sequence of states, the states are listed in the increasing order of the corresponding departure times. Denote by $\Omega(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ the set of all states such that each of them immediately follows $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ in at least one of the above mentioned sequences. If $k_A^o + k_B^o > 0$, then define $F(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ as the minimal value of

$$\sum_{s \in \{A,B\}} \sum_{j \leqslant k_s^o} \left[C_j^{s,o} - r_j^{s,o} \right] \tag{16}$$

on the set of the considered schedules. If $k_A^o + k_B^o = 0$, then assume that

$$F\left(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\right) = 0.$$

In what follows, it is convenient to use, for any $\alpha \in \{p, o\}$, the notation $\bar{\alpha} = \{p, o\} \setminus \{\alpha\}$. If state $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ is associated with the departure of the last express, then, according to Section 2, k_x^u in $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ is

$$k_x^u = \begin{cases} 1, & \text{for } x = s \text{ and } u = \alpha \\ 0, & \text{for } x = s \text{ and } u = \bar{\alpha} \\ 0, & \text{for } x = \bar{s} \text{ and } u = \bar{\gamma} \\ 0, & \text{for } x = \bar{s}, u = \gamma \text{ and } b \neq 2 \\ 1, & \text{for } x = \bar{s}, u = \gamma \text{ and } b = 2. \end{cases} \tag{17}$$

This implies that

$$F\left(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\right) = R\left(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\right). \tag{18}$$

If the express, associated with $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$, is not the last express in the schedule, then

$$F\Big(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\Big) = R\Big(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\Big)$$
$$+ \min_{\left(t', \hat{k}_A^p, \hat{k}_A^o, \hat{k}_B^p, \hat{k}_B^o, s', b', \alpha', \gamma'\right) \in \Omega\left(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\right)} F\left(t', \hat{k}_A^p, \hat{k}_A^o, \hat{k}_B^p, \hat{k}_B^o, s', b', \alpha', \gamma'\right). \tag{19}$$

Denote by Y the set of all states that are the first in at least one sequence of states induced by schedules, satisfying (14). Then, the minimal value of (3) on the set of all schedules optimal for (2) is

$$\min_{\left(t, n_A^p, n_A^o, n_B^p, n_B^o, s, b, \alpha, \gamma\right) \in Y} F\Big(t, n_A^p, n_A^o, n_B^p, n_B^o, s, b, \alpha, \gamma\Big). \tag{20}$$

Observe that (13) and (20) are quite different. Indeed, in order to list all states in X, it suffices to know only n_A^p, n_A^p, $r_{n_A^p}^{A,p}$, and $r_{n_B^p}^{B,p}$, whereas the enumeration of all states in Y may require a significantly more sophisticated procedure. This procedure is presented in the next section.

5. DYNAMIC PROGRAMMING

The optimisation procedure below requires a set T of points in time that contains all departure times of expresses in all schedules where each express departs as early as possible for an express of this type. One such set is described below.

Taking into account (5)–(9), it is easy to see that the departure time of any express is

$$t = r_i^{s,\alpha} + m_1 p_A + m_2 p_B + m_3 \beta \tag{21}$$

for some $\alpha \in \{p, o\}$, $s \in \{A, B\}$, $i \in N_s^\alpha$, and some integers m_1, m_2 and m_3. Denote

$$n = n_A^p + n_B^p + n_A^o + n_B^o.$$

Observe that s and α in (21) are not necessarily the same as the corresponding parameters of the considered express. If, for two consecutive expresses, α, s and i in (21) remain the same, then, according to (7) and (9), m_1 and m_2 can not increase by more than 3 and can not decrease by more than 1, whereas m_3 can not increase by more than 2. Thus, m_1 and m_2 do not exceed $3n$ and are not less than $-n$, whereas m_3 does not exceed $2n$. Consequently, all possible departure times of the expresses belong to the set

$$\Big\{t|\ t \geqslant 0,\ t = r_i^{s,\alpha} + m_1 p_A + m_2 p_B + m_3 \beta,\\ i \in N_s^\alpha,\ s \in \{A, B\},\ \alpha \in \{p, o\},\ m_1 \in \{-n, \dots, 0, 1, \dots, 3n\},\\ m_2 \in \{-n, \dots, 0, 1, \dots, 3n\}, m_3 \in \{0, 1, \dots, 2n\}\Big\}, \tag{22}$$

which cardinality is $O(n^4)$.

Consider the minimisation of (2) and assume that $n_A^p > 0$ and $n_B^p > 0$, because otherwise the minimisation of (2) is trivial. Since this problem involves only priority trains, following the approach adopted in Section 3, only the priority trains can be considered. Hence, instead of (22) only its subset

$$\Big\{t|\ t \geqslant 0,\ t = r_i^{s,p} + m_1 p_A + m_2 p_B + m_3 \beta,\ i \in N_s^p,\ s \in \{A, B\},\\ m_1 \in \{-(n_A^p + n_B^p), \dots, 0, 1, \dots, 3(n_A^p + n_B^p)\},\\ m_2 \in \{-(n_A^p + n_B^p), \dots, 0, 1, \dots, 3(n_A^p + n_B^p)\},\\ m_3 \in \{0, 1, \dots, 2\,(n_A^p + n_B^p)\}\Big\} \tag{23}$$

can be taken as T.

The optimisation procedure, presented in this section, obtains the optimal value of (2) by generating a sequence of the sets $V_1, \dots, V_{n_A^p + n_B^p}$. Hence, the number of sets in this sequence is equal to the number of trains, because, as in Section 3, only priority trains are considered. Each set is comprised of tuples (t, k_A, k_B, s, b), which are candidates for being a state in an optimal schedule for (2).

Each set V_k, $1 \leqslant k \leqslant n_A^p + n_B^p$, contains only candidates satisfying the condition

$$k_A^p + k_B^p = k.$$

Thus, by virtue of (10), all candidates for being the state associated with the last express are only in the sets V_1 and V_2. The set V_1 contains only such candidates. More specifically, this set is comprised of all tuples $(t, 1, 0, A, 0)$, where $t \in T$ and $t \geqslant r_1^{A,p}$, and all tuples $(t, 0, 1, B, 0)$, where $t \in T$ and $t \geqslant r_1^{B,p}$.

The set V_2 contains all candidates for being the state, associated with the last express and satisfying the equality $k_A + k_B = 2$, but may also contain other candidates satisfying this equality. The subset of V_2 of all candidates for being the last state is comprised of all tuples $(t, 1, 1, A, 2)$, where $t \in T$ and $t \geqslant r_1^{A,p}$, and all tuples $(t, 1, 1, B, 2)$, where $t \in T$ and $t \geqslant r_1^{B,p}$.

The sets $V_1, \dots, V_{n_A^p + n_B^p}$ are generated one by one in the increasing order of their indices. After the inclusion of all candidates for being the state, associated with the last express, for each set the tuples are examined one by one. To be included in set V_k, a tuple (t, k_A, k_B, s, b), satisfying $k_A + k_B = k$, must possess several properties. These properties include:

(p1) $k_s \leqslant n_s^{s,p}$ for $s \in \{A, B\}$;

(p2) $t \in T$ and $t \geqslant r_{k_s}^{s,p}$;

(p3) if $b \neq 0$, then $k_{\bar{s}} \geqslant 1$;

(p4) if $b = 1$, then $k_s \geqslant 2$;

(p5) if $k_A^p = n_A^p$ and $k_B^p = n_B^p$, then t is the same as computed by (9).

Furthermore, the currently considered tuple (t, k_A, k_B, s, b), which satisfies (p1)–(p5), is included in the set $V_{k_A+k_B}$ only if there exists a tuple $(t', \hat{k}_A, \hat{k}_B, s', b')$, which has been already included in one of the previously generated sets and has t' that can be computed, using (5)–(8); has $\hat{k}_A$ and $\hat{k}_B$ satisfying (24) and (25) below; and satisfies (26) below:

$$\hat{k}_s = k_s - 1; \tag{24}$$

$$\hat{k}_{\bar{s}} = \begin{cases} k_{\bar{s}} - 1, & \text{if } b = 2 \\ k_{\bar{s}}, & \text{otherwise;} \end{cases} \tag{25}$$

$$\text{if } b = 1, \text{ then } s = s' \text{ and } b' \neq 0. \tag{26}$$

For every tuple (t, k_A, k_B, s, b) that possesses the required properties and therefore belongs to $V_{k_A+k_B}$, denote by $W(t, k_A, k_B, s, b)$ the set of all tuples $(t', \hat{k}_A, \hat{k}_B, s', b')$ in the previously generated sets such that t' is computed by (8); $\hat{k}_A$ and $\hat{k}_B$ satisfy (24) and (25); and (26) holds. If (t, k_A, k_B, s, b) is selected as a state, then

$$W(t, k_A, k_B, s, b) = \Omega(t, k_A, k_B, s, b).$$

Therefore, taking into account (13), the optimal value of (2) is

$$\min_{(t, n_A, n_B, s, b) \in V_{n_A^p + n_B^p}} f(t, k_A, k_B, s, b),$$

where f is defined similar to the definition of F in (11) and (12), that is, by assigning to each candidate (t, k_A, k_B, s, b) for being the state of the last express

$$f(t, k_A, k_B, s, b) = L(t, k_A, k_B, s, b), \tag{27}$$

and by assigning to each other (t, k_A, k_B, s, b) in $V_2 \cup \ldots \cup V_{n_A^p + n_B^p}$

$$f(t, k_A, k_B, s, b) = \max \left\{ L(t, k_A, k_B, s, b), \min_{(t', \hat{k}_A, \hat{k}_B, s', b') \in W(t, k_A, k_B, s, b)} f(t', \hat{k}_A, \hat{k}_B, s', b') \right\}. \tag{28}$$

Taking into account the cardinality of T (see (23)), the complexity of this optimisation procedure is $O((n_A^p + n_B^p)^6)$.

The procedure that constructs a schedule that has the smallest value of (3) among all schedules that are optimal for (2) is similar to the minimisation of (2) with one important difference: in order to ensure (14), each tuple must possess some additional properties (see (g6) and (g7) below). According to this procedure, n sets $V_1, \ldots, V_n$ are generated one by one in the increasing order of their indices. Analogously to the above, set V_k, $1 \leqslant k \leqslant n$, contains only tuples $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$, satisfying

$$k_A^p + k_A^o + k_B^p + k_B^o = k.$$

Each tuple in the sets $V_1, \ldots, V_n$ is a candidate for being a state in the desired schedule. To be included in a set a tuple $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ must possess several properties. The first group of properties

(g1) $k_s^\varepsilon \leqslant n_s^{s,\varepsilon}$ for $\varepsilon \in \{p, o\}$ and $s \in \{A, B\}$;
(g2) $r_{k_s}^{s,\alpha} \leqslant t$ and $t \in T$;
(g3) if $b \neq 0$, then $k_{\bar{s}}^\gamma \geqslant 1$;
(g4) if $b = 1$, then $k_s^\alpha + k_s^{\bar{\alpha}} \geqslant 2$;

(g5) if $k_A^p = n_A^p$, $k_A^o = n_A^o$, $k_B^p = n_B^p$, and $k_B^o = n_B^o$, then t is the same as computed by (9)

is analogous to (p1)–(p5). The second group

(g6) if $\alpha = p$, then $t + p_A + p_B \leqslant d_{k_s^p}^{s,p} + F^*$;

(g7) if $\gamma = p$ and $b = 2$, then $t + 2p_s \leqslant d_{k_{\bar{s}}^p}^{\bar{s},p} + F^*$

guarantees that the constructed schedule will satisfy (14).

In order for $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ to be selected as a candidate for a state, there must exist already selected $(t', \hat{k}_A^p, \hat{k}_A^o, \hat{k}_B^p, \hat{k}_B^o, s', b', \alpha', \gamma')$ such that

$$\hat{k}_s^\alpha = k_s^\alpha - 1, \qquad \hat{k}_s^{\bar{\alpha}} = k_s^{\bar{\alpha}} \qquad \text{and} \qquad \hat{k}_{\bar{s}}^{\bar{\gamma}} = k_{\bar{s}}^{\bar{\gamma}}, \tag{29}$$

$$\hat{k}_{\bar{s}}^\gamma = \begin{cases} k_{\bar{s}}^\gamma - 1, & \text{if } \ b = 2 \\ k_{\bar{s}}^\gamma, & \text{otherwise.} \end{cases} \tag{30}$$

The equalities (29) and (30) play the same role as (24) and (25) previously. More specifically, a tuple $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$, which satisfies (g1)–(g7), is included in the set $V_{k_A^p + k_A^o + k_B^p + k_B^o}$ only if there exists an already included in one of the previously generated sets $(t', \hat{k}_A^p, \hat{k}_A^o, \hat{k}_B^p, \hat{k}_B^o, s', b', \alpha', \gamma')$ such that t' is the same as computed using (5)–(8); $\hat{k}_A^p$, $\hat{k}_A^o$, $\hat{k}_B^p$, $\hat{k}_B^o$ satisfy (29) and (30); and (26) holds. Denoting by $W(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ the set of all such tuples $(t', \hat{k}_A^p, \hat{k}_A^o, \hat{k}_B^p, \hat{k}_B^o, s', b', \alpha', \gamma')$, the minimal value of (3) on all set of all schedules that are optimal for (2) can be written as

$$\min_{(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma) \in V_n} f(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma),$$

where f is defined as follows. For each candidate $(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma)$ for being the state of the last express,

$$f(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma) = R(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma).$$

For each element of $V_2 \cup \ldots \cup V_n$ which is not a candidate for the state of the last express

$$f\Big(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\Big) = R\Big(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\Big)$$
$$+ \min_{\left(t', \hat{k}_A^p, \hat{k}_A^o, \hat{k}_B^p, \hat{k}_B^o, s', b', \alpha', \gamma'\right) \in W\left(t, k_A^p, k_A^o, k_B^p, k_B^o, s, b, \alpha, \gamma\right)} f\left(t', \hat{k}_A^p, \hat{k}_A^o, \hat{k}_B^p, \hat{k}_B^o, s', b', \alpha', \gamma'\right).$$

Taking into account the cardinality of T (see (22)), the time complexity of this optimisation procedure is $O(n^8)$.

6. CONCLUSION

The polynomial-time algorithms presented in this paper aim at the reduction of the direct impact of blockage of one track of a double-track railway segment measured by the maximum lateness for the priority trains and the total (and therefore average) time in system for the ordinary trains. The directions of further extension of the presented approach may include the minimisation of other measures of the impact of blockage; the reduction of the impact of blockage on a broader part of the railway network; the planning of maintenance requiring possession of some parts of the railway network; and the development of fast approximation algorithms.

ACKNOWLEDGMENTS

This work was supported by Russian Railways and Russian Foundation for Basic Research, project no. 17-20-01107.

REFERENCES

1. Brucker, P., Heitmann, S., and Knust, S., Scheduling Railway Traffic at a Construction Site, *OR Spectrum*, 2002, vol. 24, no. 1, pp. 19–30.

2. Cacchiani, V., Huisman, D., Kidd, M., Kroon, L., Toth, P., Veelenturf, L., and Wagenaar, J., An Overview of Recovery Models and Algorithms for Real-Time Railway Rescheduling, *Transport. Res., Part B*, 2014, vol. 63, pp. 15–37.

3. Lusby, R.M., Larsen, J., Ehrgott, M., and Ryan, D., Railway Track Allocation: Models and Methods, *OR Spectr.*, 2011, vo. 33, no. 4, pp. 843–883.

4. Zinder, Y., Lazarev, A.A., Musatova, E.G., and Tarasov, I.A., Scheduling the Two-Way Traffic on a Single-Track Railway with a Sidin, *Autom. Remote Control*, 2018, vol. 79, no. 3, pp. 506–523.

5. Zinder, Y., Lazarev, A., Musatova, E., Tarasov, I., and Khusnullin, N., Two-Station Single Track Scheduling Problem, *IFAC-PapersOnLine*, 2016, vol. 49, no. 12, pp. 231–236.

This paper was recommended for publication by F.T. Aleskerov, a member of the Editorial Board