



Planning Algorithms in the Decision-Making Support System for Logistic Problems

S. E. Vlasov¹, N. V. Starostin², and A. E. Timofeev²(✉)

¹ Scientific Research Institute for System Analysis Russian Academy of Sciences Moscow,
9 60-letiya Octyabrya Street, Moscow 117312, Russian Federation

² National Research Lobachevsky State University, 23 Gagarina Avenue,
Nizhny Novgorod 603950, Russian Federation
alexey.timofeev@itmm.unn.ru

Abstract. The development of the railway infrastructure, as well as the intensification of exploitation of railways, made it necessary to schedule the motion of trains to prevent their collision and blocking. Railway transport logistic problems aimed at providing the organization of work of rolling stock in the framework of a railway infrastructure are considered. Planning and organization of railway traffic is a key factor to providing safe and effective functioning of the entire railway transportation system. The currently used approach, based on target plans, is not always successful in practice, especially in emergencies resulting from the fluctuation of traffic capacities of a railway infrastructure. A mathematical model of rolling stock logistics is constructed, accounting for all the key specific features of the railway infrastructure. In the framework of the constructed mathematical model, optimization problems for scheduling the traffic of locomotives and trains for an assigned planning interval are formulated. Solution algorithms, assessing time complexity, are presented for all the formulated problems.

Keywords: Traffic scheduling · Railway infrastructure · Development graph · Frontal algorithm

1 Introduction

The development of the railway infrastructure (RW), as well as the intensification of exploitation of railways, made it necessary to schedule the motion of trains to prevent their collision and blocking. A RW system is divided into contact systems, which, in their turn are subdivided into sections. Each section is supervised by a dispatcher, who is in charge of operational planning of RW traffic. Sections are subdivided into stages covering the distance between couples of individual stations. A stage, in its turn, is subdivided into block sections, which are basic railway elements.

A block section is an undividable section (~1800 m), where not more than one train can be present. The limits of block sections (blocks) are equipped with traffic lights. Block sections are characterized by speed limits (terrain relief, profile, crossroads, switches etc.).

The available traffic dispatching methods are based on preliminarily constructed traffic target plans with safety intervals, the so-called ‘threads’. Such plans are made up in modeling the motion of virtual trains with generalized characteristics. The priority task of a dispatcher is to assign through trains or made-up trains to threads. The motion of a train according to a thread is a guarantee against collisions and blockings within the stations of that thread. The aim is to run through a section the assigned number of trains at minimal cost.

The drawbacks of this approach are locality of the solution, as well as failure of the working system in emergencies. The local character of the dispatching technologies makes it possible to assess the situation within only one section (or maximum two neighboring sections). Choosing optimal solutions for separate sections may lead to system failures within the framework of the entire RW. In an emergency, target plans become either outdated or inapplicable (as in the case of security infraction at the block sections). The dispatcher loses an effective instrument of planning and operational control.

The present paper considers models and methods of traffic planning, overcoming the above-described drawbacks. The proposed approach can be used in decision-making support systems [1, 2] in the field of passenger and freight RW traffic scheduling.

2 Modeling the Motion of Trains

2.1 A Graph of a RW Structure and Its Time Development

Let $G(V, A, d)$ be a graph of a RW structure, where V is a set of nodes (junctions of the railways of a block section), $|V| = n$; $A \subseteq V^2$ is a set of arcs (the railways of a block sections, the orientation determines the direction of a railway), $|A| = m$; $d: A \rightarrow N$ is mapping determining the length of the corresponding railways (meters).

Let $t \in \{0, \dots, T\}$ be planning tacts. Let $U = \{u_0, \dots, u_L\}$ be possible choices of the motion speed of objects (a transfer from a continuous model to a discrete one). It is assumed that on set U the order is defined: $u_0 = 0$ (minimal speed), u_L is maximal speed of an object, for the others $u_i < u_{i+1}$.

$S(u_i, u_j) \rightarrow N$ is distance function (meters) showing the distance covered by a RW object during one planning tact on condition that the object started moving at speed u_i and finished moving at speed u_j . It will be assumed that mapping S is defined on a set of the following form:

$$\begin{aligned} Z = & \{(u_0, u_1), (u_1, u_2), \dots, (u_i, u_{i+1}), \dots, (u_{L-1}, u_L)\} \cup \\ & \{(u_0, u_0), (u_1, u_1), \dots, (u_i, u_i), \dots, (u_L, u_L)\} \cup \\ & \{(u_1, u_0), (u_2, u_1), \dots, (u_{i+1}, u_i), \dots, (u_L, u_{L-1})\}. \end{aligned} \quad (1)$$

In this set, the first subset defines acceleration models, the second one defines models of uniform motion, the third one defines deceleration models. Function $S: Z \rightarrow N$ will be called a model of the motion of the object.

Note: the technology of defining map S imposes certain constraints on U (or versions u_i) of speed modes. Transfer from any admissible version of speed to an ‘adjacent’ version of speed (both lower and higher) $u_i \rightarrow u_{i\pm 1}$ must take less than one tact.

The position of the object in the space of the RW structure will be denoted by $\rho = (a, r, u)$, where $a \in A$ corresponds to an arc (defines the railway and the direction of motion); $r \in [0, 1]$ is real number defining the relative position of the object on the railway (0 is the beginning, 1 is the end of the way); $u \in U$ is motion speed.

A couple ‘position-time’ $(\rho, t) \in \Pi$ defines the location of the object in time and space of the RW structure, where $\Pi = A \times [0, 1] \times U \times \{0, \dots, T\}$ is a set of various admissible locations in the RW structure space and a set of planning tacts.

Binary asymmetric relation (Π, E) , $E \subseteq \Pi^2$ will be defined on set Π . The special feature of the graph E is that it contains only couples of the form: $E \in ((\rho_i, t), (\rho_j, t + 1)) \iff$ position ρ_j is achieved from ρ_i in the period of time $[t, t + 1]$.

Digraph $Q(\Pi, E)$ will be called time development of the graph of the RW structure. The special feature of digraph Q is that it is $(T + 1)$ -partite, as one tact does not contain any connections, there are only arcs from the previous tact to the following.

Obviously, the structure of digraph $Q(\Pi, E)$ will depend on motion model $S(u_i, u_j)$. Different objects of the RW have different motion models, so $Q(\Pi, E) = Q(\Pi, E(S))$. In what follows, this will be assumed, but, to simplify the notation, $Q(\Pi, E)$ will be used and called ‘development’.

2.2 A Model of Cost of Travel of RW Objects

Positive real map $C_{RW-object}: Z \rightarrow R^+$ will be introduced, which defines generalized cost of travel of an object within one-time tact (taking into account that the object is either accelerating or travelling uniformly, or decelerating). In practice, generalized costs are rather difficult to describe. Having a dynamic model of a travelling train, it is possible to use, for instance, approach [3]. Statistic data with subsequent approximation can also be used [4].

Evidently, the motion dynamics of an object can change as a function of its mass, RW profile, climatic conditions etc. Formally, all the factors affecting cost and dynamics of a travelling object can be subdivided into two types. The first type is connected with the notion of ‘train’: a particular ‘locomotive-stock’ couple determines the superposition of costs $C_{train}(C_{locomotive}, C_{stock})$. The second one is connected with the notion of ‘railway block section’ – a model of cost of travel of a particular object along a particular section is defined as a superposition of costs (C_{train}, C_{way}) . Here, C_{way} is understood as a kind of ‘fine’ function introducing corrections into the landscape of the model of costs of a RW object travelling along a particular RW section.

Thus, the model of costs will be:

$$C(C_{train}(C_{locomotive}, C_{stock}), C_{way}(e)), \quad (2)$$

where $e \in E$.

In what follows, the abbreviated form $C(e)$ of the model of costs will be used to simplify the formal notation.

2.3 Optimal Trajectory of the Motion of the Object

Let $\tau = (\tau_0, \tau_1, \dots, \tau_K)$ be route in development $Q(\Pi, E)$, where $\tau_i = (\rho_i, t_i) \in \Pi, i = 0..K$ and $(\tau_{j-1}, \tau_j) \in E, j = 1..K$. The special feature of route τ is that a RW-object needs K tacts to cover the route: $t_K - t_0 = K$. This is connected with the main special feature of development Q - $(T + 1)$ -partite asymmetric graph.

Any route in development $Q(\Pi, E)$ will be called the motion trajectory of an object.

The problem of seeking the fastest motion trajectory of a RW-object in terms of development $Q(\Pi, E)$ is formulated as follows: find the shortest route $\tau = (\tau_0, \tau_1, \dots, \tau_K)$ from initial position ρ_a to final position ρ_b , taking into accounting that the motion begins at time tact t_0 :

$$K(\tau^*) = \min_{\tau=(\rho_a, t_0), \dots, (\rho_b, t_K)} K(\tau), \tag{3}$$

where $K(\tau)$ is number of arcs of route τ .

Here, τ^* is optimal route from the viewpoint of the number of intermediate arcs of development $Q(\Pi, E)$. Let τ^* be minimal length trajectory.

The problem of seeking ‘the cheapest’ motion trajectory of a RW-object in terms of development $Q(\Pi, E)$ is formulated as follows: find route $\tau = (\tau_0, \tau_1, \dots, \tau_K)$ with the minimal cost from initial position ρ_a to final position ρ_b , taking into account that the motion begins at time tact t_0 :

$$F(\tau^*) = \min_{\tau=\tau_0, \dots, \tau_K} \sum_{i=1}^K C(\tau_{i-1}, \tau_i), \tag{4}$$

where $\tau_0 = (\rho_a, t_0), \tau_K = (\rho_b, t_K)$.

Here, τ^* is route of the minimal weight on the weighted digraph of development $Q(\Pi, E, C)$. Let τ^* be minimal cost trajectory.

2.4 Accounting for Arrival/Departure Time Constraints

As a rule, in practical RW-planning problems there are additional constraints connected with arrival/departure times.

The meaning of a constraint on departures is that an object can be kept at an initial point for some time, but not longer than an assigned time. Formally, it means that it is necessary to exclude the motion trajectories which contain vertices $R^- = \{(\rho_0, t_i) \in \Pi \mid t_i > t^-\}$, where ρ_0 is initial position, t^- is latest time of departure from the initial position.

The meaning of a constraint on arrivals is that an object cannot arrive at a final point earlier than an assigned time. Formally, it means that it is necessary to exclude the motion trajectories containing vertices $R^+ = \{(\rho_K, t_i) \in \Pi \mid t_i < t^+\}$, where ρ_K is final position, t^+ is earliest time of arrival in a final position.

Thus, accounting for the above-described constraints involves assuming a new development of the graph: $Q'(\Pi', E') = Q(\Pi, E) \setminus Q(R^-, R^+)$.

2.5 Accounting for Speed Constraints

Some of RW-sections can have additional constraints on the speed of RW-objects travelling along them. Consider such a case using an individual railway.

Let a railway be modeled by a pair of arcs $(v_a, v_b) \in A$ and $(v_b, v_a) \in A$ of graph $G(V, A)$ of the RW-structure. Let this railway have speed constraint $u \leq u_{max}$. Consider development $Q(\Pi, E)$. Among all vertices $(\rho_i, t_i) \in \Pi$, those are noted, for which the speed constraints are violated: $O = O(v_a, v_b, u_{max}, \Pi) = \{(a, r, u), t) \in \Pi \mid u > u_{max} \text{ and } a = (v_a, v_b) \text{ or } a = (v_b, v_a)\}$.

A new development of the graph will be obtained by excluding the subgraph formed at the vertices of set $O : Q'(\Pi', E') = Q(\Pi, E) \setminus Q(O)$. New development Q' rules out the presence of inadmissible trajectories from the viewpoint of the violation of speed constraints on RW-sections.

In the case of several RW-sections with speed constraints, the technology of arriving at a new development looks like successive use of the above-described operation for each route with speed constraints.

2.6 Accounting for Contact System Constraints

Each locomotive in the framework of a general RW structure has its own substructure (a contact system), in the framework of which it can travel. Contact systems cross each other on the level of junction stations. A general RW structure is assumed to be a combination of contact systems:

$G(V, A, d) = G_1(V_1, A_1, d_1) \cup \dots \cup G_s(V_s, A_s, d_s)$ – the graph of the RW structure is represented in the form of a combination of the subgraphs of the contact systems, where s is number of contact systems. Then the development of a RW structure is also represented as a combination of the developments:

$$Q(\Pi E) = Q_1(\Pi_1, E_1) \cup \dots \cup Q_s(\Pi_s, E_s), \tag{5}$$

where $\Pi_i = A_i \times [0, 1] \times U \times \{0, \dots, T\}, i = 1..s$.

In this case, the problem of seeking the optimal motion trajectory of a RW-object is formulated not for the entire development $Q(\Pi, E)$, but only for its part $Q_i(\Pi_i, E_i) \subseteq Q(\Pi, E)$, where i is number of the contact system of the locomotive of a RW-object.

3 The Algorithm of Seeking the Optimal Motion Trajectory of an Object and Its Computational Complexity

The task of seeking the optimal motion trajectory of a RW-object corresponds to the task of seeking on a weighted digraph a development $Q(\Pi, E, C)$, which does not contain any loops or arcs having negative weights. The shortest route is sought from the initial vertex to any vertex of the set of finite vertices. Problems of seeking the shortest route on a graph are solved exactly in effective time/memory under condition of the finiteness of the graph. The problem is that the number of vertices of the graph of development $Q(\Pi, E, C)$ is infinitely large (a continuum), as $\Pi = A \times [0, 1] \times U \times \{0, \dots, T\}$.

To solve the problem of seeking the minimal weigh route on a finite graph having no loops or arcs with negative weighs, classical Dijkstra's algorithm [5] can be used. Its computational complexity is defined as: $O(N \times \log N + M)$, where N is number of vertices of a graph, M is number of arcs of a graph. The specific feature of the algorithm is that, at each iteration, it 'widens' the front – an ordered set of vertices. The front width can increase and decrease. The wider the front, the longer it takes to modify it, and the longer is the iteration of the algorithm. For the case of an infinite graph of development $Q(\Pi, E, C)$, there is a non-zero probability of an indefinitely widening front, which is fraught the danger of exhausting the memory and, consequently, failure of the algorithm.

To overcome this difficulty, a modification of algorithm [6] is proposed, which involves a mechanism of 'checking the vertices for identity'. It will be assumed that vertex $(a_1, r_1, u_1, t_1) \in \Pi$ and vertex $(a_2, r_2, u_2, t_2) \in \Pi$ are 'identical' if conditions $a_1 = a_2, u_1 = u_2, t_1 = t_2$ are simultaneously satisfied. The front will be formed only of the vertices which are not 'identical' pairwise. In this way, the front width will always be limited by value $m \times L \times T$, where m is number of arcs of the graph of the RW structure, L is number of choices of the velocity of the objects, T is number of planning tacts.

Then the modification of algorithm [6] will have the following form (see description of algorithm ROUTE on the pseudocode).

Designations:

- (a_0, r_0, u_0, t_0) – initial position
- *FINISH* – set of final positions
- $c[a, u, t]$ – cost of trajectory (returned result of work of the algorithm)
- $p[a, u, t]$ – trajectory positions (returned result of work of the algorithm)
- *ORDER* – order of vertices of the development ordered according to cost $c[a, u, t]$
- $C(\tau_i, \tau_j)$ – cost of travel from position τ_i to τ_j

Pseudocode of algorithm ROUTE:

Assign $c[a_0, u_0, t_0] \leftarrow 0, p[a_0, u_0, t_0] \leftarrow (a_0, r_0, u_0, t_0)$

For all $(a_i, 0, u_i, t_i) \in \Pi$ different from $(a_0, 0, u_0, t_0)$

assign $c[a_i, u_i, t_i] \leftarrow \infty$

Cycle till *ORDER* $\neq \emptyset$

Extract $(a_i, r_i, u_i, t_i) \leftarrow \text{ORDER}$

If $(a_i, u_i, t_i) \in \text{FINISH}$ then EXIT – solution is found

If not, for all $(a_j, r_j, u_j, t_j) \in \Pi$ such that $((a_i, r_i, u_i, t_i), (a_j, r_j, u_j, t_j)) \in E$ check condition:

If $c[a_j, u_j, t_j] > c[a_i, u_i, t_i] + C((a_i, r_i, u_i, t_i), (a_j, r_j, u_j, t_j))$ then:

change $c[a_j, u_j, t_j] \leftarrow c[a_i, u_i, t_i] + C((a_i, r_i, u_i, t_i), (a_j, r_j, u_j, t_j))$

change $p[a_j, u_j, t_j] \leftarrow p[a_i, u_i, t_i], (a_j, r_j, u_j, t_j)$

put into *ORDER* $\leftarrow (a_j, r_j, u_j, t_j)$ according to cost $c[a_j, u_j, t_j]$

In this case, the computational complexity of the modified algorithm will be described as:

$$O(N \times \log N + M), \quad (6)$$

where $N = m \times L \times T$, $M = m \times (3L - 2) \times T$. Here, the value of M is obtained as the number of arcs in a $(T + 1)$ -partite digraph of a development consisting exclusively of the vertices which are not pairwise 'identical'. The memory requirements are defined as:

$$O(m \times L \times T). \quad (7)$$

4 Accounting for Thread Constraints and the Traffic Scheduling Algorithm Within Threads

Each thread defines a system of constraints on the arrival/departure times of the trains for the stations (see p. 2.4). Not only one thread (a system of constraints), but several alternative choices of threads can be defined for a train. In this case, the problem of choice arises: choose the 'best' route for the train from the admissible ones. An admissible choice implies a route of a train, for which the requirements of the thread are satisfied. Various strategies can be used to make 'the best' choice. For instance, it seems 'reasonable' to use the following order on the set of admissible choices (rule 1): choice A is preferable to choice B if for choice A a larger number of requirements are satisfied, as compared with choice B . An example of another strategy is the following rule (rule 2): choice A is preferable to choice B if choice A arrives in the next station earlier than choice B . This choice is 'reasonable, as, if a train remains at a station for a long time, it exclusively uses the RW-infrastructure of the station, preventing other RW-objects from using this resource.

Lexicography can be introduced on a set of rules of preference. For example, all the choices are ordered first according to rule 2, then, on the sets of comparable choices, ordering according to rule 1 is assumed. A scheme which establishes an order among the sets of admissible routes of a RW-object will be designated by ϕ , where $\phi(\tau_i, \tau_j) \leftrightarrow \tau_i \preceq \tau_j$, and τ_i and τ_j are routes.

The scheme of planning the traffic within threads will be as follows: for a given RW-object, all the choices of the threads are considered; for each thread, the problem of finding the optimal route is formulated (see p. 3); among all the routes 'the best' one is chosen. Each problem of finding the best route for the object is formulated, based on the following conditions: the current position of the object is taken to be its initial position; a thread defines the system of constraints on the arrival/departure times of the trains for the stations (see p. 2.4). The pseudocode of planning algorithm TRAINROUTE within threads is given below.

Designations:

- (a_0, r_0, u_0, t_0) – initial position
- *THREADS* – set of systems of constraints on arrival/departure times (of threads)
- (R_i^-, R_i^+) – the earliest and the latest times of arrival in the assigned point
- D – set of admissible routes
- ROUTE – algorithm for determining the optimal route of an object
- τ – admissible route according to a thread
- ϕ – scheme of choosing the best route

Pseudocode of algorithm TRAINROUTE:

It is assumed that $D := \emptyset$

Choose thread $((R_1^-, R_1^+), \dots, (R_h^-, R_h^+)) \leftarrow THREADS$

The trajectory for a thread starts from current position $\tau \leftarrow (a_0, r_0, u_0, t_0)$

For the current position (a_0, r_0, u_0, t_0) defines the next requirement (R_i^-, R_i^+)

Cycle until $i \leq h$

Assign $FINISH \leftarrow (R_i^-, R_i^+)$

Algorithm ROUTE finds the trajectory from the final position τ in position $FINISH$

If admissible trajectory τ_i is found, change $\tau \leftarrow \tau_i$

Otherwise abort cycle

Change $D \leftarrow D \cup \{\tau\}$

If $D = \emptyset$ then there is no solution, EXIT.

According to rule ϕ ‘the best’ solution among D is chosen, EXIT.

It can be seen from the pseudocode that algorithm TRAINROUTE consists in multiply executing algorithm ROUTE. For each thread, an admissible route is formed successively: algorithm ROUTE tries to complete a trajectory complying with the next constraint. If one assumes that the system of constraints in the threads concerns only the junction stations, then the upper estimate of the execution time of one iteration of the external cycle of algorithm TRAINROUTE is defined by the following expression:

$$O(h \times N' \times \log N' + M'), \quad (8)$$

where $N' = m' \times L \times T$, $M' = m' \times (3L - 2) \times T$, $m' = \max\{|A_1|, \dots, |A_s|\}$ (number of arcs of the largest contact system). Then (it is assumed that the number of constraints in all the threads is equal to h) the following estimate of the computational complexity of algorithm TRAINROUTE is obtained:

$$O(|THREADS| \times h \times N' \times \log N' + M'). \quad (9)$$

The memory requirement is described as: $O(m' \times L \times T)$.

5 The Model of Routing the Rolling Stock, Accounting for the Distribution of Locomotives Over the Contact Systems

Let I is a set of locomotives; S is a set of rolling stocks to be transported over a RW structure; W is set of stations of the RW-structure $W \subseteq V$. For each stock $s \in S$ the ‘technology’ of its travel along the RW is known, which is described by a chain of stations $(w^s_0, \dots, w^s_{d(s)})$, where $w^s_0 \in U$ is initial station of stock s ; $w^s_{d(s)} \in U$ is final station; $w^s_a \in W$, $a = 1..(d(s) - 1)$ are intermediate stations; $d(s)$ is number of parts of the route the stock has to pass before arriving in the final point.

In what follows, actions enabling stock s to cover a part of the route from station w^s_{a-1} to station w^s_a will be called operation w^s_a . J will denote a set of all operations for all stocks. Each operation $j \in J$ involves processes of maneuvering the locomotive, making up a train and travelling to the assigned station. The following designations are

introduced: $s(j)$ is stock; $w(j)$ are stations of arrival; $f(j)$ are stations of departure; $p(j)$ is operation directly preceding operation $j \in J$, so $s(j) = s(p(j))$, $f(p(j)) = w(j)$.

Decomposition of a RW-structure into contact systems assumes that not every locomotive $i \in I$ can execute operations $j \in J$. $\varphi(j)$ will denote a set of locomotives of the contact system of the RW section between stations $f(j)$ and $w(j)$, i.e. $\varphi(j) \subseteq I, j \in J$.

Let $t(i, j)$ define time required for executing operation $j \in J$ by locomotive $i \in I$. It is evident that $t(i, j)$ not only depends on operation j and locomotive i , but on a large number of other factors, such as: status, profile and workload of the RW-structure; the starting tact of the motion of the train; location and routs of other trains etc. Function $t(i, j)$ involves constraints on threads; constraints on the impossibility for several trains to be at the same time at the same place. To evaluate function $t(i, j)$, it is possible to use algorithm TRAINROUTE.

The varied parameters of the mathematical model here are $|J|$ -dimensional integer-valued vectors x and y , the components of which define starting tacts and locomotive numbers for executing corresponding operations.

5.1 Constraints of the Mathematical Model

Natural constraint on variables:

$$x_j \in \{0, \dots, T\}, y_j \in I, j \in J. \tag{10}$$

Interrelation of operations, determining canonicity of a system model and providing a constraint on using the locomotives of the contact system:

$$x_j \geq x_{p(j)} + t(y_j, p(j)), y_j \in \varphi(j), j \in J. \tag{11}$$

Conditions defining the impossibility of simultaneously using the same locomotive with different stocks:

$$x_j \geq x_k + t(y_k, k) \text{ or } x_k \geq x_j + t(y_j, j), k, j \in \{l | y_l = i, l \in J\}. \tag{12}$$

5.2 Accounting for Speed Constraints

It is required to minimize the cost of realizing RW traffic according to the routes and threads of the trains. The notion of ‘costs’ arises only when the routes of the objects within the RW-structure become certain – only then it will be possible to evaluate what resources are required for their motion. To move a rolling stock from one point to another, it is, in a general case, necessary to: move a locomotive to the stock (maneuvering); to make up a train and to provide its travelling to an assigned point (traffic). Couple (x_j, y_j) defines locomotive y_j for executing operation j , which begins at time tact x_j and finishes at time tact $x_j + t(y_j, j)$. Also, period $t(y_j, j)$ takes into account the times of maneuvering, making up the train and its travelling.

The ‘cost’ of realization of operation j will be assumed to mean generalized cost (j, x_j, y_j) including ‘cost’ of the realization of the maneuvering trajectory and ‘cost’ of

the realization of the motion trajectory. The earlier described model $C(C_{train}, C_{way})$ can be used as ‘cost’ function.

As a result, the problem of minimization of the cost function of the realization of the traffic of trains is obtained:

$$\sum_{j \in J} COST(j, x_j, y_j) \rightarrow_{(x,y)} \min. \quad (13)$$

6 The Frontal Planning Algorithm and Computing Costs

First, a ‘greedy’ tact-by-tact scheme of working of the planning algorithm will be considered. It consists in that the algorithm begins planning from tact 0 to tact T , solving the local problem of allocation of the locomotives to the stocks for each tact. The task is to allocate a set of free locomotives $L \subseteq I$ to a set of made-up and awaiting stocks S . Matrix of ‘costs’ $R = (r_{ij})|L| \times |S|$ is formed. Each element r_{ij} defines the ‘cost of allocating the i -th locomotive to j -th stock. According to matrix R , the best allocation, from the viewpoint of minimizing the total ‘cost’, is chosen.

In general, the problem in question can be formulated in terms of the matrix-mapping model [7]. The problem of planning work of an individual locomotive can be formulated in terms of the travelling salesman problem [8]. In general, the problem in question can be formulated in the framework of the resource allocation model [9–13], as well as a multi-index transport problem [14–18]. In a general case, the above problems pertain to the class of NP-completeness problems [19, 20].

On the other hand, there exists a problem that allocating one locomotive to one stock involves travelling within the RW structure and results in changing the traffic capacity of the RW-infrastructure and, consequently, the coefficients in matrix R for other allocations. Because of this, the ‘greedy’ algorithm is introduced, which chooses the minimal element in matrix R and executes the corresponding allocation.

In this case, the ‘greedy’ tact-by-tact planning scheme is as follows: successively, for each tact, beginning with 0: a matrix of ‘cost’ of allocating the free locomotives to the awaiting stocks is constructed; the best allocation is chosen (the classical allocation problem is solved) and is fixed on the development of the RW-structure (excluding trajectory elements to prevent collision of the trains). However, the computation cost of such a scheme of solving the planning problem appears to be excessively high. Thus, assuming that there is only one contact system, all the stocks are made up at 0 time tact and all the locomotives are available beginning from time tact 0. Then, the estimation formula of the execution time will be expressed as:

$$O(T \times |L| \times |S| \times |THREADS| \times h \times N \times \log N + M). \quad (14)$$

To minimize the cost of computation (large value of T), planning can be done not by tacts, but by events in the RW system. An ‘event’ can mean two facts: 1) at a certain tact, a locomotive has become available in the RW system; 2) at a certain tact, an awaiting stock has appeared in the RW system. In this case, for each stock in the system (in the limit) there may be as many events, as many requirements (two-side constraints) are

assigned in the threads. For the locomotives, the facts of their becoming available are also connected with the fact of the realization of the requirements for the stocks. Then, for a 'greedy' planning scheme of events, the following estimation of the computation cost is obtained:

$$\begin{aligned} & O((h \times |S|) \times |L| \times |S| \times |THREADS| \times h \times N \times \log N + M) \\ & = O(|L| \times |S|^2 \times |THREADS| \times h^2 \times N \times \log N + M) \end{aligned} \quad (15)$$

It is noteworthy that estimate (15) is the upper one, and in practical work is, for the most part, determined by how large the sets of available locomotives and of awaiting stocks are at the occurrence time of the events. As a rule, the sets are small enough (for the most part, one locomotive determines 2–3 allocation choices to the stocks of the contact system). Moreover, in reality, a RW-infrastructure and a system of threads are coordinated so that there is no need to consider all the threads in algorithm TRAINROUTE – it is enough to check the realizability of the 2–3 nearest threads and choose the best alternative. As a result, the 'practically expected' estimate of the execution time of the algorithm is obtained:

$$O(|S| \times h \times N \times \log N + M). \quad (16)$$

7 Conclusion

The article considers the problem of RW stock traffic planning. A model of the motion of trains is proposed, which is based on the graph-development of the RW structure in time, accounting for speed and contact system constraints. Based on the proposed model of cost of moving of RW-objects, an optimization problem is formulated, and algorithms of optimal routing of an object, with and without accounting for constraints of target planning of RW traffic are described. A model is introduced and a problem is formulated for planning the entire rolling stock in the framework of an assigned planning interval, taking into account the allocation of locomotives to contact systems. A frontal algorithm of solving such a problem is proposed. The computation costs of all the introduced algorithms are estimated.

Acknowledgments. The authors thank their colleagues from the Research Institute for Mechanics, Nizhniy Novgorod Lobachevski State University, for their help in preparing this article. The work is financially supported by the Federal Targeted Program for Research and Development in Priority Areas of Development of the Russian Scientific and Technological Complex for 2014–2020 under the contract No. 14.578.21.0246 (unique identifier RFMEFI57817X0246).

References

1. Basalin, P.D., Bezruk, K.V.: Hybrid intellectual decision-making support system architecture. *Neurocomputers* **8**, 26–34 (2012)

2. Basalin, P.D., Timofeev, A.E.: Fuzzy models for the functioning of the rule-based hybrid intelligent learning environment. *Int. J. Open Inf. Technol.* **7**(2), 49–55 (2019)
3. Korotchenko, A.G., Smoryakova, V.M.: On a method of construction of numerical integration formulas. In: *AIP Conference Proceedings*, vol. 1776: Numerical Computations: Theory and Algorithms, pp. 090012-1–090012-4 (2016)
4. Ljung, L., Pflug, G., Walk, H.: *Stochastic approximation and optimization of random systems*. Birkhauser, Berlin (1992)
5. Dijkstra, E.W.: A note on two problems in connexion with graphs. *Numerische Mathematic* **1**(1), 269–271 (1959)
6. Starostin, N.V., Balashov, V.V.: The use of hypergraphs for solving the problem of orthogonal routing of large-scale integrated circuits with an irregular structure. *J. Commun. Technol. Electron.* **53**(5), 589–593 (2008)
7. Starostin, N.V., Pankratova, M.A.: Mapping of the graph of a parallel program to the graph of a computing system. In: *Proceedings of the International Conference Numerical Computations: Theory and Algorithms*, pp. 107–108 (2013)
8. Starostin, N.V., Klyuev, I.V.: Iteration multilevel method for the travelling salesman problem. In: *Communications in Computer and Information Science 466. 11th Joint Conference on Knowledge-Based Software Engineering, CCIS, Volgograd, 17–20 September 2014*, pp. 477–482 (2014)
9. Prilutskiy, M.Kh.: Multicriterion allocation of homogeneous resource in hierarchical systems. *Autom. Telemechanics Moscow* **2**, 139–146 (1996)
10. Prilutskii, M.Kh.: Multicriterial multi-index resource scheduling problems. *J. Comput. Sci. Int.* **46**(1), 83–87 (2007)
11. Prilutskii, M.Kh., Kostyukov, V.E.: Optimization models of gas and gas condensate processing. *Autom. Remote Control* **72**(8), 345–349 (2012)
12. Prilutskii, M.Kh.: Optimal planning for two-stage stochastic industrial systems. *Autom. Remote Control* **75**(8), 1384–1392 (2014)
13. Prilutskii, M.K.: Optimal management of double standing stochastic production systems. *Autom. Remote Control* **5**, 69–82 (2018)
14. Afraimovich, L.G., Prilutskii, M.Kh.: Multiindex optimal production planning problems. *Autom. Remote Control* **71**(10), 2145–2151 (2010)
15. Afraimovich, L.G.: Multi-index transport problems with decomposition structure. *Autom. Remote Control* **73**(1), 118–133 (2012)
16. Afraimovich, L.G., Prilutskii, M.Kh.: Multicommodity flows in tree-like system works. *J. Comput. Syst. Sci. Int.* **47**(2), 214–220 (2008)
17. Afraimovich, L.G.: Multiindex transportation problems with 2-embedded structure. *Autom. Remote Control* **74**(1), 90–104 (2013)
18. Afraimovich, L.G., Katerov, A.S., Prilutskii, M.: Multi-index transportation problems with 1-nested structure. *Autom. Remote Control* **77**(11), 1894–1913 (2016)
19. Garey, M.R., Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman & Co., New York (1979)
20. Prilutskii, M.Kh., Vlasov, S.E.: Multicriterion product planning problems. Lexicographic schemes. *Inf. Technol.* **7**, 61–66 (2005)