DOI: 10.17323/2587-814X.2022.2.7.20

Reduction of dimension in the problem of optimal management of a freight cars fleet using unmanned locomotives

Fedor A. Belousov
E-mail: sky_tt@list.ru

Nerses K. Khachatryan
E-mail: nerses-khachatryan@yandex.ru

Ivan V. Nevolin
E-mail: i.nevolin@cemi.rssi.ru

Central Economics and Mathematics Institute, Russian Academy of Science
Address: 47, Nakhimovsky Prospect, Moscow 117418, Russia

Abstract

This paper considers the problem of optimal management of a fleet of freight cars by a transport railway operator. The solution to this problem is an optimal plan, which is a timetable for the movement of freight and empty railway cars, following which the transport operator will receive the maximum profit for the estimated period of time. This problem is reduced to the problem of linear programming of large dimension. Unlike the works of other authors on this topic, which mainly deal with methods of numerical solution of the corresponding linear programming problems, this article focuses on an algorithm that allows one to reduce their dimensionality. This can be achieved by excluding from the calculation those routes that obviously cannot be involved in the solution, or whose probability of participation in the final solution is estimated as extremely low. The effectiveness of the proposed modified algorithm was confirmed both on a model example (several stations, a short planning horizon) and on a real example (more than 1 000 stations, a long planning horizon). In the first case, there was a decrease in the dimension of the problem by 44%, while in the second – by 30 times.

Keywords: railway freight transportation, optimal plan, optimal management of the fleet of cars, linear programming, theory of schedules, operations research, unmanned locomotives

Citation: Belousov F.A., Khachatryan N.K., Nevolin I.V. (2022) Reduction of dimension in the problem of optimal management of a freight cars fleet using unmanned locomotives. *Business Informatics*, vol. 16, no. 2, pp. 7–20. DOI: 10.17323/2587-814X.2022.2.7.20

Introduction

One of the most popular modes of transport for cargo in the Russian Federation is rail. Publications devoted to railway logistics can be divided into the following main groups according to the type of tasks studied:

- railway network infrastructure design tasks;
- railway planning tasks;
- tasks of managing the fleet of locomotives and wagons.

In the first group, works [1–4] can be distinguished. The second group, in particular, is represented by the tasks of forming the timetable of freight trains, as well as the tasks of forming freight flows [5, 6]. One of the approaches to the formation of cargo traffic is presented in the works of Khachatryan and Beklaryan [7–16]. These articles present macroscopic dynamic models in which the process of organizing railway freight transportation is the formation of freight traffic based on the interaction of neighboring stations. These models make it possible to predict dynamics of station congestion and flows arising on the railway network by using a given procedure for organizing cargo traffic. Several configurations of sections of the railway network are considered. The first one is an extended section of the railway line which is characterized by an infinite number of stations in both directions, and also characterized by the absence of hub stations. This configuration of the transport network is suitable for describing transnational transportation (for example, transportation along the Trans-Siberian railway with a length of more than 9.000 km). The second configuration defines the movement of cargo traffic through a closed chain of stations. The third is characterized by a finite number of stations and determines the movement of cargo traffic between two hub stations.

The presented work is devoted to the problem of optimal management of a fleet of railway freight cars. Railway transport operators are faced with the task of optimal management of a fleet of freight cars to maximize profit. Such management is carried out, on the one hand, on the basis of wagons' dislocation, on the other – on the basis of requests for the transportation of goods. Requests are submitted by customers. Each request specifies stations of departure and destination, the volume of cargo transported, expressed in wagons and the rate that the customer is going to pay for each wagon of transported cargo. In addition to the rate that the customers pay to the transport operator for the provision of wagons, they also pay to Russian Railways for the transportation of loaded wagons. The costs of transporting empty wagons are covered by the transport operator. From the entire list of requests, the transport operator selects those that are most profitable for it to execute. Any selected requests can be completed either in full or in part. In accordance to dislocation of wagons and the available list of requests, creation of a wagon management plan implies preparation of a timetable for the movement of loaded and empty wagons, taking into account known time standards. Thus, the task is to find the optimal plan to manage the fleet of freight cars for a certain period of time (as a rule, the plan is drawn up for a month). In [17] a multi-commodity flow model defined on a space-time graph is presented, and an algorithm that allows reducing it to a linear programming problem is proposed.

The problems with a close formulation were previously considered in [18, 19]. In these articles, the authors focus on the methods of numerical solution of the resulting linear programming problem. In particular, they talk about the use of the column generation method [20, 21] and modification of this method [22, 23], which is based on the Danzig–Wulff decomposition [24]. A similar problem was considered in an earlier article [25]. The model presented in this paper was developed at the request of one of the largest transport operators in Latin America. Its peculiarity is that the freight transportation plan is drawn up taking into account the previously known schedule of locomotives, whereas in this article and in the works [18, 19], the time of movement of cars on each of the routes is determined solely based on the standards of the Russian Railways (i.e., the schedule of locomotives in our case is unknown and it is not important for us; this is an internal matter of Russian Railways).

In contrast to these works, the authors of this article do not focus their attention on methods of numerical solution of linear programming problems, but offer methods for constructing space-time graphs that serve as the basis for setting both the objective function and the constraints of a linear programming problem having a smaller dimension. At the same time, the type of the objective function and constraints does not change. Thus, in this paper, a modification of the algorithm described in [17] is proposed which makes it possible to significantly reduce the dimension of the model. Before proceeding to its description, let us present the statement of the linear programming problem itself. To do this, the following notations are going to be introduced:

N – number of stations involved in planning;

T – planning horizon, measured in days; for simplicity one month is taken as the length of the planning horizon in this work (i.e. $T = 30$ or 31);

t – the discrete parameter responsible for time is measured in days and takes values $t = 1, 2, \dots, T$;

$C = \{C_{ij}\}_{i,j=1}^{N}$ – ($N \times N$)-matrix, which elements characterize the tariff set by Russian Railways for an empty run of one wagon from station i to station j;

$\Theta 1 = \{\Theta 1_{ij}\}_{i,j=1}^{N}$ – ($N \times N$)-matrix, which elements characterize the time (in days) of movement of loaded wagons from station i to station j in accordance with Russian Railways standards (time is rounded to a larger integer);

$\Theta 2 = \{\Theta 2_{ij}\}_{i,j=1}^{N}$ – ($N \times N$)-matrix, which elements characterize the time (in days) of movement of empty wagons from station i to station j in accordance with Russian Railways standards (time is rounded to a larger integer);

$P = \{P_{ij}\}_{i,j=1}^{N}$ – ($N \times N$)-matrix, which elements characterize the rate specified by the customer in the request for transportation of one loaded wagon from station i to station j;

$\bar{Q} = \{\bar{Q}_{ij}\}_{i,j=1}^{N}$ – ($N \times N$)-matrix, which elements characterize the number of loaded wagons specified in the corresponding request for cargo transportation from station i to station j. All elements of the matrix take non-negative;

$\bar{S}^0(t) = \{\bar{S}_i^0(t)\}_{i=1}^{N}$ – vector of dimension N, that characterizes the initial location of wagons on day t, the i-th element of this vector equals to the number of wagons that arrived at station i at time $t \in \{1, \dots, T\}$. All values of this vector take non-negative integer values.

Then the above linear programming problem takes the form

$$PC^T \cdot K \to \max_{K \ge 0}, \tag{1}$$

subject to

$$(A_{out} - A_{in}) \cdot K = S_0, \tag{2}$$

$$A_Q \cdot K \le Q, \tag{3}$$

where

K – is a vector the first part of which is responsible for freight routes, the second part corresponds to empty routes, in fact this vector is a transportation plan;

PC – is a vector, the first part of which is responsible for freight rates, the second part corresponds to costs for empty rates, the product of $PC^T \cdot K$ gives a profit that transport operator gets for planning horizon;

A_{out} – is a matrix that takes into account the outgoing routes from each station;

A_{in} – is a matrix that takes into account the incoming routes to each station;

S_0 – is a vector of the initial distribution of wagons by stations and by time;

A_Q – is a matrix such that the product $A_Q \cdot K$ shows the volume in wagons that must be executed for each of the requests in accordance with the solution K;

Q – is a vector of the volume of orders (in wagons) specified in the requests.

Constraint (2) is a balance constraint, i.e. its implementation guarantees that in each period the number of cars entering the station will coincide with the number of cars leaving. The fulfillment of constraint (3) guarantees that the volume of executed cargo routes will not exceed the volumes specified in the requests.

The algorithm presented in [17] gives the dimension of the problem (1)–(3) equals to TN^2, i.e. the number of elements in the vectors K and PC is $2TN^2$. The dimension of the matrices A_{out} and A_{in} turns out to be $TN \times 2TN$, the dimension of the matrix A_Q is $N^2 \times 2TN^2$, the dimension of the vector S_0 is TN, the dimension of Q is N^2.

1. Algorithms for the generation of matrices and vectors for the problem (1)–(3)

The most noticeable reduction in the dimension of the problem (1)–(3) can be achieved by taking into account only freight routes in vector , the use of which will lead to partial or full execution of orders. In other words, reducing the dimension of the problem can be organized by removing from the vector those cargo routes that in any case will not be involved. In addition to vector , the corresponding transformations must be performed in all other vectors and matrices of the problem (1)–(3) so that they are all consistent.

In this section, an algorithm for generating all matrices and vectors of reduced dimension for the problem (1)–(3) is described. In addition to removing unnecessary cargo routes from consideration, the algorithm also provides for the possibility of excluding empty routes selected for any reason from the calculation.

PC, Q, S_0 vectors and dynamic lists

Let us introduce new variables, *routes_from_station_cargo*, *routes_to_station_cargo* and *routes_from_station_empty*, *routes_to_station_empty*, which are dynamic lists with elements taking integer values. The elements of these variables are responsible for the numbers of those routes that are going to be taken into account in the calculation. The *routes_from_station_cargo* list contains numbers of outgoing stations for each of the considered cargo routes; *routes_to_station_cargo* contains the corresponding numbers of incoming stations for the same cargo routes. Similarly, *routes_from_station_empty* contains outgoing station numbers for empty routes; *routes_to_station_empty* contains incoming station numbers for corresponding empty routes. If we take into account all possible routes, as is done in [17], then the number of elements contained in the new variables is going to be equal to N^2. However, these variables were introduced so that fewer routes could be taken into account, thereby reducing the dimension of the problem.

Let us fill in the variables *routes_from_station_cargo* and *routes_to_station_cargo*. We will take into account only those routes that are in the requests; therefore, if there is a request from station i to station j, that is, $P_{ij} > 0$ then add the element i to the variable *routes_from_station_cargo* on the right, and add the element j to the variable *routes_from_station_cargo* on the right. Simultaneously with each addition of elements to the variables *routes_from_station_cargo* and *routes_to_station_cargo*, we will compose a vector p by sequentially adding elements P_{ij}

from below, and we will also compose a vector Q by sequentially adding elements $\bar{Q}_{ij}$ from below characterizing the volumes of the corresponding applications. Thus, at each iteration, the dimension of the vectors p and Q, as well as the variables *routes_from_station_cargo* and *routes_to_station_cargo* increases by one.

It can be seen that the resulting vector Q takes into account only cargo routes, and not all possible routes, as was in [17]. Due to this, the dimension of the vector Q is reduced from N^2 to N_{cargo}.

It is assumed that for each pair of stations i and j, there can be no more than one request from station i to station j. If there are two requests for a pair of stations i and j from i to j, then in this case a duplicate of station м is created, let us call it $\hat{i}$, and in the variables *routes_from_station_cargo* and *routes_to_station_cargo* not only i and j are added, but also $\hat{i}$ and j, respectively. In the case of two bids, the vector p is filled with the corresponding bids in the same sequence as the variables *routes_from_station_cargo* and *routes_to_station_cargo*. Communication between station i and its duplicate $\hat{i}$ is instantaneous and free of charge. At the same time, there is no back route from station i to station $\hat{i}$. All incoming routes are directed to station i, one can get to station $\hat{i}$ only through station i. This is done so that cyclic flows from station i to $\hat{i}$ and back do not appear in solutions. The case when there may be more than two requests from station i to station j is not considered in this article – this is a separate topic, the disclosure of which we will leave for subsequent works in this direction.

Similarly, the *outes_from_station_empty* and *routes_to_station_empty* variables are filled with station numbers only of those routes that were selected according to some criteria. Simultaneously with filling in the variables *routes_from_station_empty* and *routes_to_station_empty*, by analogy with the vector p, a new vector $\tilde{C}$ is filled by adding elements C_{ij} from below (the cost of an empty run from station i to station j). The order of adding elements to vector $\tilde{C}$ corresponds to the order of adding elements to the variables *routes_from_station_empty* and *routes_to_station_empty*. Thus, at each iteration, the dimension of the vector $\tilde{C}$, as well as the variables *routes_from_station_empty* and *routes_to_station_empty* increases by one.

One way to reduce the dimension due to empty routes is to remove from consideration those empty routes, the arrival stations in which are not departure stations for any of the requests for loaded routes. The idea is that there is no need to come to such stations by empty routes, since cars can only leave there by other empty routes, which is unlikely to be optimal. The exception is routes from these stations to themselves, it is better to leave such routes in numerical calculation, since either wagons from the previous period arrive at these stations, or they are the final destination for some loaded routes. Here it is necessary to make a reservation that this method of removing empty routes from consideration is justified, provided that the time and financial costs of empty runs from an arbitrary station A to an arbitrary station B are always no more than when carrying out two consecutive empty runs from A to some station C and from C to B. In practice, this condition is fulfilled, so the exclusion of such double empty routes from consideration does not lead to a deterioration in the target indicators of the solutions obtained.

Using N_{cargo} we denote the dimension of vector p, which coincides with the number of elements in the variables *routes_from_station_cargo* and *routes_to_station_cargo*; using N_{empty} we denote the dimension of the vector $\tilde{C}$, which also coincides with the number of elements in the variables *routes_from_station_empty* and *routes_to_station_empty*. In other words, N_{cargo} characterizes the number of all cargo routes corresponding to the list of requests; N_{cargo} characterizes the number of all possible empty routes that are taken into account when searching for the optimal plan.

Let us construct a vector *PC* of a smaller dimension compared to a similar vector in [17]. To do this, we perform $T-1$ consecutive concatenation of the vector p so that we get a vector of dimension $T \cdot N_{cargo}$. Next, we also add the vector $\tilde{C}$ to the resulting vector by sequential concatenation $T-1$ times. Assign the value of the resulting vector to the vector *PC*; its dimension of this vector is $T \cdot (N_{cargo} + N_{empty})$. The new dimension of the problem (1)–(3) is also equal to $T \cdot (N_{cargo} + N_{empty})$. Obviously, the elements of vector $\bar{K}$ correspond to the same routes that correspond to both the rates and costs of vector *PC*.

The algorithm for creating vector S_0 described in [17] will remain unchanged. Namely, the system of vectors $\bar{S}^0(t)$, $t \in \{1, ..., T\}$ is transformed into a vector S_0 by sequential concatenation of vectors corresponding to each moment of time. With this concatenation, the first N elements of the new vector correspond to the vector $\bar{S}^0(1)$, the next N elements correspond to the vector $\bar{S}^0(2)$, etc. Thus, the dimension of the vector S_0 equals to TN.

A_{in} and A_{out} matrices

Each of the A_{in} and A_{out} matrices consists of two parts. The first part is responsible for cargo routes, the second for empty ones. These matrices are sparse matrices; any nonzero element in them takes a single value. Let us construct A_{out} matrix, which is responsible for outgoing routes originating from each station. At the zero iteration, the A_{out} matrix is a zero matrix of size $T \cdot N \times T \cdot (N_{cargo} + N_{empty})$.

Let us denote by *Index_cargo_out*[1] a dynamic list of indexes $k \in \{1, ..., N_{cargo}\}$, for which *routes_from_station_cargo*[k] = 1 In other words, the variable *Index_cargo_out*[1] contains the numbers of those freight routes among the first N_{cargo} elements of vector K, the starting point for which is station 1. For an arbitrary station $i \in \{1, ..., N\}$, the interpretation of the variable *Index_cargo_out*[i] is similar.

To account for outgoing routes from station 1 in the first time period, it is necessary for all $k \in$ *Index_cargo_out*[1] elements $A_{out}[1, k]$ to be assigned the value 1. To account for outgoing routes from station 1 in the second time period, it is necessary to assign value 1 to the elements $A_{out}[1 + N, k + N_{cargo}]$, $k \in$ *Index_cargo_out*[1]. N is added to the first component of the coordinates of the A_{out} matrix, since the period in the vector S_0 is equal to N. In other words, the first N elements in this vector are responsible for N stations in the first time period, and the next N elements are responsible for the same N stations in the second time period, etc. N_{cargo} elements are added to the second component of coordinates of the A_{out} matrix, since the period in the first part of the *PC* vector responsible for the rates of loaded runs is equal to N_{cargo}. In other words, the first N_{cargo} elements in the first part of the *PC* vector are responsible for routes starting in the first time period; the next N_{cargo} elements are responsible for the same routes, but starting in the second time period, and so on. In other words, the first N_{cargo} elements in the first part of the *PC* vector are responsible for routes starting in the first time period; the next N_{cargo} elements are responsible for the same routes, but starting in the second time period, and so on. Continuing this logic further, it is clear that all elements of matrix A_{out} with coordinates $[1 + (t-1) \cdot N, k + (t-1) \cdot N_{cargo}]$, $k \in$ *Index_cargo_out*[1] have to be assigned to value 1. Thus, to obtain the A_{out} matrix, it is necessary for each station $i \in \{1, ..., N\}$ to create a dynamic list *Index_cargo_out*[i] with such numbers $k \in \{1, ..., N_{cargo}\}$ for which *routes_from_station_cargo*[k] = i. Further, for all $i \in \{1, ..., N\}$ for which *Index_cargo_out*[i] $\neq 0$, elements of A_{out} matrix with coordinates $[i + (t-1) \cdot N, k + (t-1) \cdot N_{cargo}]$, $k \in$ *Index_cargo_out*[i], $t \in \{1, ..., T\}$ have to be assigned to value 1. The first part of the A_{out} matrix responsible for cargo routes has been formed. Similarly, the second part of this matrix is formed, which is responsible for empty routes. To do this, for each station

$i \in \{1, ..., N\}$, other *Index_empty_out*[i] lists are formed with the following numbers $k \in \{1, ..., N_{empty}\}$, for which *routes_from_station_empty*[k] = i. In other words, the variable *Index_cargo_out*[i] contains the numbers of those empty routes among the N_{empty} elements of the vector K following $T \cdot N_{cargo}$ elements, the starting point for which is station i.

Further, for all $i \in \{1, ..., N\}$ for which *Index_empty_out*[i] $\neq 0$, elements of A_{out} matrix with coordinates $[i + (t-1) \cdot N, k + (t-1) \cdot N_{empty} + T \cdot N_{cargo}]$, $k \in$ *Index_empty_out*[i], $t \in \{1, ..., T\}$ have to be assigned to value 1. Since the first $T \cdot N_{cargo}$ elements in vector K are responsible for cargo routes, the rest are responsible for empty ones. Then in the case of considering empty routes in the second component of the A_{out} matrix, there is an additional term $T \cdot N_{cargo}$. After performing all the described operations, the construction of the A_{out} matrix is completed.

At the next stage, the A_{in} matrix is formed. This matrix is responsible for the incoming routes to each station. At the zero iteration A_{in} matrix is a zero matrix of size $T \cdot N \times T \cdot (N_{cargo} + N_{empty})$. To form this matrix, we additionally need information about the travel time for each of the routes, that is, the values of the matrices $\Theta 1$ and $\Theta 2$. Denote by *Index_cargo_in*[1] a dynamic list of those indexes $k \in \{1, ..., N_{cargo}\}$ of the variable *routes_to_station_cargo*[k] for which *routes_to_station_cargo*[k] = 1. For an arbitrary station $i \in \{1, ..., N\}$, the interpretation of the variable *Index_cargo_in*[1] is similar. Then, in order to account for incoming routes to station 1, departures on which are carried out in the first time period, elements of A_{in} matrix with coordinates $[1 + \Theta 1[routes\_from\_station\_cargo[k], 1] \cdot N, k]$, $k \in$ *Index_cargo_in*[1] have to be assigned to value 1. Similarly, to account for incoming routes to station 1, departures on which are carried out in the time period $t \in \{1, ..., T\}$, elements of A_{in} matrix with coordinates $[1 + (\Theta 1[routes\_from\_station\_cargo[k], 1] + t - 1) \cdot N, k + (t-1) \cdot N_{cargo}]$, $k \in$ *Index_cargo_in*[1] have to be assigned to value 1.

For an arbitrary station $i \in \{1, ..., N\}$ dynamic variables *Index_cargo_in*[i] are compiled from those indexes $k \in \{1, ..., N_{cargo}\}$ of the variable *routes_to_station_cargo*[k] for which *routes_to_station_cargo*[k] = i. For all $i \in \{1, ..., N\}$ for which *Index_cargo_in*[i] $\neq 0$, elements of A_{in} matrix with coordinates $[i + (\Theta 2[routes\_from\_station\_cargo[k], i] + t - 1) \cdot N, k + (t-1) \cdot N_{cargo}]$, $k \in$ *Index_cargo_in*[i], $t \in \{1, ..., T\}$ have to be assigned to value 1.

The first part of the A_{in} matrix relating to loaded routes is constructed. It remains to construct the second part of this matrix relating to empty routes. Denote by *Index_empty_in*[i] a dynamic list of those indexes $k \in \{1, ..., N_{cargo}\}$ of the variable *routes_to_station_empty*[k] for which *routes_to_station_empty*[k] = i. For all stations $i \in \{1, ..., N\}$ for which *Index_empty_in*[i] $\neq 0$, elements of A_{in} matrix with coordinates$[i + (\Theta 2 [routes_from_station_cargo[k], i] + t - 1) \cdot N, k + (t-1) \cdot N_{empty} + T \cdot N_{cargo}]$, $k \in$ *Index_empty_out*[i], $t \in \{1, ..., T\}$ have to be assigned to value 1.

A_Q matrix

The A_Q matrix is needed to calculate the volume of completed requests, so only loaded routes are taken into account when calculating this indicator. This means that in the matrix A_Q, which has dimension $N_{cargo} \times T \cdot (N_{cargo} + N_{empty})$, the last $T \cdot N_{empty}$ columns consist exclusively of zero elements, non-zero elements are found only in the first $T \cdot N_{cargo}$ columns. At the zero iteration, we take the zero matrix as the A_Q matrix.

Since the first $T \cdot N_{cargo}$ elements of vector K are ordered with the period N_{cargo}, i.e. the first N_{cargo} elements are responsible for loaded routes outgoing in the first time period, the next N_{cargo} elements are responsible for the same routes outgoing in the second time period, etc. Then in the first row of A_Q matrix T units are written, the first of which is put on the first position, the next to the position $N_{cargo} + 1$, the next to the position $2N_{cargo} + 1$, etc. In other words, in the first row of A_Q matrix, T elements are assigned to the unit starting from the first element and then with the period N_{cargo} elements. In the next row of the A_Q matrix, the unit is also assigned T elements with the period N_{cargo}, but starting from the second element of the second row. In the third row of the A_Q matrix, the algorithm is repeated, but the unit is assigned elements starting from the third element of the third row. This continues until the last line of N_{cargo}. As a result, if we consider the first N_{cargo} rows and the first N_{cargo} columns, we get a unit matrix, if we consider the next N_{cargo} columns, we also get a unit matrix, etc. If we consider the first $T \cdot N_{cargo}$ columns of the matrix A_Q, we will see T sequentially composed unit matrices of dimension $N_{cargo} \times N_{cargo}$, the remaining columns of the matrix are zero.

This section provides algorithms for the construction of all components of the problem (1)–(3) – objective function and constraints. It is shown that the dimension of both the vector of variables of the objective function and the constraint matrices is noticeably reduced. We will demonstrate this both on a model example (several stations, a short planning horizon) given in [17] and on a real example (more than 1,000 stations, long planning horizon).

2. Reducing the dimensionality of the problem on model and real examples

Here is a statement of the model example from [17]. The number of stations is 4 ($N = 4$), the planning horizon is 3 days ($T = 3$). The list of received requests consists of five items, which are shown in *Table 1.*

Based on the list of requests, it is necessary to make two matrices – a matrix of rates P, the elements of which are written in conventional units, and a matrix of request volumes $\bar{Q}$:

$$P = \begin{pmatrix} 0 & 0 & 2.9 & 0 \\ 1.1 & 0 & 2.3 & 0 \\ 0 & 1.9 & 0 & 2.1 \\ 0 & 0 & 0 & 0 \end{pmatrix}; \ \bar{Q} = \begin{pmatrix} 0 & 0 & 3 & 0 \\ 5 & 0 & 4 & 0 \\ 0 & 7 & 0 & 6 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Next, we will give the travel time of both loaded and empty routes in the form of the values of the matrices $\Theta 1$ and $\Theta 2$:

$$\Theta 1 = \begin{pmatrix} 0 & 2 & 1 & 2 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 2 \\ 1 & 2 & 1 & 0 \end{pmatrix}; \ \Theta 2 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 \\ 1 & 2 & 1 & 1 \end{pmatrix}.$$

Recall that the diagonal elements of the matrix $\Theta 2$ are equal to one. This is due to the fact that if the cars need to be left at the station until the next day, then this is equivalent to the

Table 1.

List of requests for cargo transportation in the model example

No.	Departure station	Destination station	Volume of requests (in wagons)	Rate (in conditional units)
1	1	3	3	2.9
2	2	1	5	1.1
3	2	3	4	2.3
4	3	2	7	1.9
5	3	4	6	2.1

fact that they are, as it were, sent from this station to itself on a one-day run.

The values of the Russian Railways tariffs for empty runs, as well as the rates expressed in conventional units, are characterized by the values of the elements of the matrix C:

$$C = \begin{pmatrix} 0 & 1.9 & 1.3 & 1.9 \\ 1.2 & 0 & 1.8 & 0\,9 \\ 1.1 & 1.2 & 0 & 1.6 \\ 1.3 & 1.5 & 1.2 & 0 \end{pmatrix}.$$

As part of this task, it is assumed that cars can stay at stations until the next day for free, so the diagonal elements of the matrix C are zero.

The initial distribution of wagons is characterized by the following vectors:

$$\bar{S}^0(1) = \begin{pmatrix} 0 \\ 2 \\ 1 \\ 3 \end{pmatrix}; \quad \bar{S}^0(2) = \begin{pmatrix} 5 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

During the period $t = 3$, the wagons do not arrive, which is equivalent to the zero vector $\bar{S}^0(3)$.

Dimension of the problem

The dimension of the problem presented in [17] is $2TN^2 = 96$. We calculate dimension when solving the same problem using the algorithm presented in this paper. The proposed algorithm gives the dimension $T \cdot (N_{cargo} + N_{empty})$. Therefore, in order to calculate it, it is necessary to know the values of the parameters N_{cargo} and N_{empty}. To determine N_{empty}, it is necessary to understand which empty routes are planned to be included in the calculation, which are not. We exclude from consideration those empty routes, the arrival stations in which are not departure stations for any loaded runs from clients' requests. There is one such station and this is station four. We will remove from consideration all empty routes in which the destination is station 4. We will leave only the route from 4 to 4 (the car remains at the station until the next period). It turns out that empty routes from 1 to 4, from 2 to 4 and from 3 to 4 are removed from consideration. We get that $N_{empty} = N^2 - 3 = 13$. As for N_{cargo}, its value is equal to the number of requests, i.e. in our case $N_{cargo} = 5$. Thus, the dimension when using the new algorithm turns out to be equal to $T \cdot (N_{cargo} + N_{empty}) = 54$. It turns out that specifically for this example the dimension of the problem has decreased by about 44%. Separately, we note that in practice, empty routes can be excluded from the calculation for other reasons, for example, empty routes can be ignored, the tariff for which is higher than a certain threshold value. Therefore, in real problems, it is possible to achieve an even greater reduction in dimension compared to $2TN^2$.

As an example, we can consider the problem of finding the optimal plan which was solved in practice for $N = 1126$ stations, with a planning period of $T = 30$ days and the number of requests for cargo transportation equal to 1616. The dimension of the problem when solving it by the algorithm from [17] is $TN^2 = 76072560$. To determine the dimension of the problem, which is obtained using the algorithm presented in this paper, one needs to calculate N_{cargo} and N_{empty}. Obviously, $N_{cargo} = 1616$, which corresponds to the number of requests. To calculate N_{empty}, it is necessary to remove from consideration all empty routes in the direction to stations that do not appear in requests as departure stations. In addition, empty routes with tariffs exceeding 50 000 rubles are removed from consideration. As a result, the number of empty routes that should be taken into account in the calculation equals to $N_{empty} = 82058$, i.e. approximately 6.5% of all possible empty routes take part in calculation; the number of all empty routes is equal to $N^2 = 1267877$. As a result, the dimension of the problem is equal to $T \cdot (N_{cargo} + N_{empty}) = 2510220$, i.e. it decreases by about 30 times.

Linear programming

One can write out the linear programming problem (1)–(3) for the model example. To do this, we determine the values of the matrices A_{in}, A_{out} and A_Q, as well as the vectors PC, S_0, Q. After that, we solve this problem and compare the resulting solution with the solution from [17].

Dynamic lists*routes_from_station_cargo*, *routes_to_station_cargo* and *routes_from_station_empty*, *routes_to_station_empty* are the following:

routes_from_station_cargo = {1, 2, 2, 3, 3},

routes_to_station_cargo = {3, 1, 3, 2, 4},

routes_from_station_empty = {1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4, 4},

routes_to_station_empty = {1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 4}.

One can make up the vectors p and c:

$p = (2.9 \quad 1.1 \quad 2.3 \quad 1.9 \quad 2.1)^T$

$c = (0 \quad 1.9 \quad 1.3 \quad 1.2 \quad 0 \quad 1.8 \quad 1.1 \quad 1.2 \quad 0 \quad 1.3 \quad 1.5 \quad 1.2 \quad 0)^T$.

The PC vector is obtained by sequential concatenation of the resulting vectors:

$PC = (p^T, p^T, p^T, c^T, c^T, c^T)^T$.

Vector Q, which is responsible for the volume of requests, takes the following form:

$Q = (3 \quad 5 \quad 4 \quad 7 \quad 6)^T$.

The vector S_0, which characterizes the initial distribution of wagons by time and by stations, takes the form:

$S_0 = (0 \quad 2 \quad 1 \quad 3 \quad 5 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0)^T$

We get the matrices A_{in}, A_{out} and A_Q. Since these matrices are sparse matrices, and nonzero elements can only take single values, therefore, as in [17], one can write these matrices in a sparse format, specifying the coordinates of the elements taking value 1. Let us write out the coordinates of unit elements of A_Q matrix, the dimension of which is equal to $N_{cargo} \times T \cdot (N_{cargo} + N_{empty}) = 5 \times 54$ (hereafter the numbering of rows and columns begins with one):

(1, 1), (2, 2), (3, 3), (3, 4), (4, 5), (1, 6), (2, 7), (3, 8), (4, 9), (5, 10), (1, 11), (2, 12), (3, 13), (4, 14), (5, 15).

Coordinates of unit elements of A_{in} matrix are the following:

(7, 1), (5, 2), (11, 3), (10, 4), (12, 5), (11, 6), (9, 7), (5, 16), (6, 17), (7, 18), (5, 19), (6, 20), (7, 21), (5, 22), (6, 23), (7, 24), (5, 25), (10, 26), (7, 27), (8, 28), (9, 29), (10, 30), (11, 31), (9, 32), (10, 33), (11, 34), (9, 35), (10, 36), (10, 37), (9, 38), (11, 40), (12, 41).

The list of coordinates of the unit elements of A_{ou}matrix:

(1, 1), (2, 2), (2, 3), (3, 4), (3, 5), (5, 6), (6, 7), (6, 8), (7, 9), (7, 10), (9, 11), (10, 12), (10, 13), (11, 14), (11, 15), (1, 16), (1, 17), (1, 18), (2, 19), (2, 20), (2, 21), (3, 22), (3, 23), (3, 24), (4, 25), (4, 26), (4, 27), (4, 28), (5, 29), (5, 30), (5, 31), (6, 32), (6, 33), (6, 34), (7, 35), (7, 36), (7, 37), (8, 38), (8, 39), (8, 40), (8, 41), (9, 42), (9, 43), (9, 44), (10, 45), (10, 46), (10, 47), (11, 48), (11, 50), (12, 51), (12, 52), (12, 53), (12, 54).

The dimension of the matrices A_{in} and A_{ou} is $T \cdot N \times T \cdot (N_{cargo} + N_{empty}) = 12 \times 54$.

Let us write out the solution that was obtained by using MatLab. Since vector K, consisting of $T \cdot (N_{cargo} + N_{empty}) = 54$ elements, also mainly consists of zero elements, we write out values of only non-zero elements:

$K_3 = 2$; $K_4 = 1$; $K_6 = 3$; $K_{13} = 2$; $K_{14} = 4$; $K_{15} = 6$; $K_{26} = 1$; $K_{28} = 2$; $K_{31} = 2$; $K_{40} = 3$.

One can write out the same solution in a more understandable format of matrices $K1(t)$ and $K2(t)$, which are $(N \times N)$-matrices. Elements of these matrixes characterize the number of loaded ($K1(t)$) and empty ($K2(t)$) wagons sent from station i to station j at the time $t \in \{1, \ldots, T\}$.

$$K1(1) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}; \; K1(2) = \begin{pmatrix} 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix};$$

$$K2(1) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2 \end{pmatrix}; \; K1(3) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 4 & 0 & 6 \\ 0 & 0 & 0 & 0 \end{pmatrix};$$

$$K2(2) = \begin{pmatrix} 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 \end{pmatrix}; \; K2(3) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Graphically, this solution is shown in *Fig. 1*. The width of each lane in this picture characterizes the number of wagons sent in a given direction.

The target value of profit, calculated according to the rule $PC^T \cdot K$, is equal to 32.3.

If we compare the obtained solution with the solution from [17], it is clear that they differ, but the values of the target functional expressing the final profit are the same. Thus, the comparison of solutions to the same problem shows that the problem has at least two different solutions.

Conclusion

This article is a continuation of the work [17]. It presents a modified algorithm for solving the problem of optimal management of a fleet of freight cars. The essence of the proposed approach is to exclude from the calculation those loaded or empty routes about which it is known in advance that they either will not be involved in the final solution or the probability of these routes appearing in the solution is estimated as very low. The model example presented in the paper shows that the use of

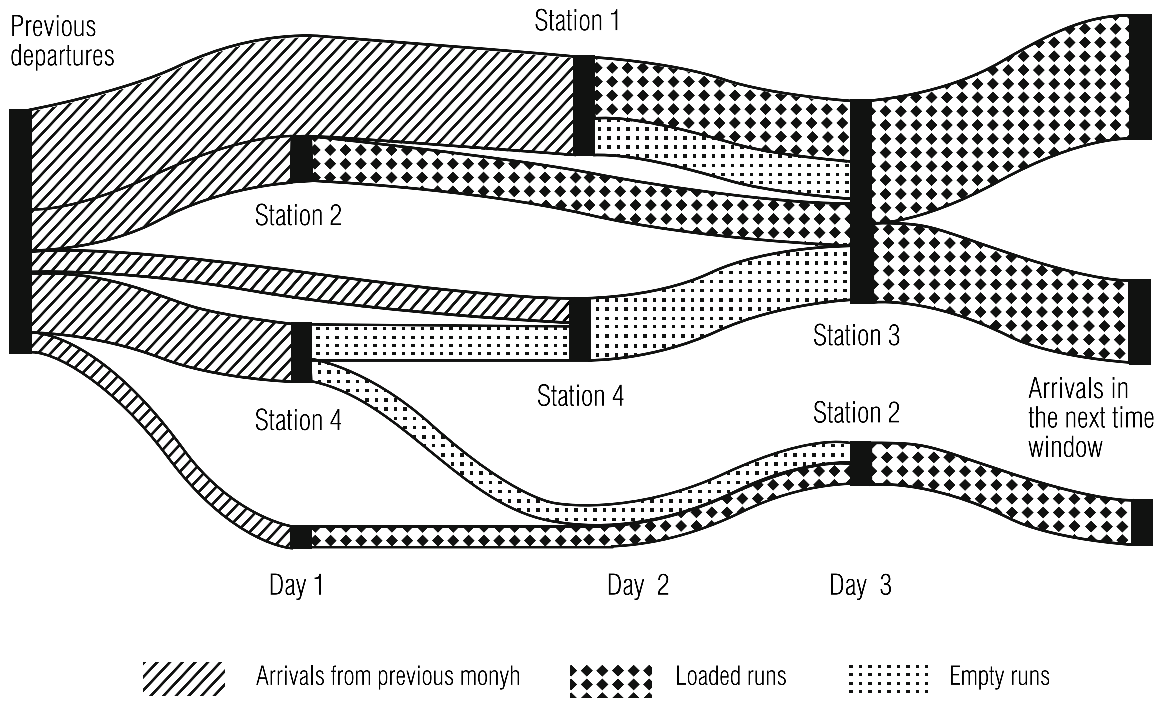

Fig. 1. Schematic representation of the resulting solution.

the modified algorithm leads to a reduction in dimension by about 44%. In practice, as a rule, there is a much more noticeable decrease in dimensionality, in particular because the exclusion of an even larger number of empty routes from the calculation due to additional features (for example, to exclude too expensive, too long empty routes). So, in the problem mentioned in the previous section, which was solved on the basis of real data, the use of an improved algorithm leads to a thirty-fold reduction in dimension compared to the algorithm from [17].

Separately, we note that the potential of methods that allow us to significantly reduce the dimension for transport problems has not been fully exhausted. It can be shown that the space-time graph that is being constructed within the framework of the presented approach can be reduced even more without loss in the quality of final solutions (reducing the space-time graph will obviously lead to a decrease in the dimension of the transport problem). To do this, one can divide all stations into three categories. The first category includes stations to which wagons arrive from the previous period and which do not participate in requests for cargo transportation either as departure stations or as destination stations. The second category includes stations that appear in the requests as destination stations, but not as departure ones, and the third category includes the remaining stations, that is, the stations indicated in the requests as departure ones. For the first category of stations, one can build outgoing empty routes exclusively for those days in which cars arrive from previous period (previous month) only to stations of the third category. In other words, as soon as wagons get to these stations, they are immediately sent by an empty run to the stations from which requests for cargo transportation can be executed. For stations of the second category, incoming empty routes are not built, but only outgoing empty routes are built in stations of the third category. For stations of the third category, a full-fledged space-time graph with incoming and outgoing empty routes is being built. The description of the specified algorithm may become the subject of one of the following articles in this direction. Reducing the space-time graph, and hence the dimension of the transport problem, can be achieved by other more subtle methods. For example, when constructing a space-time graph for stations of the second and third types, it is possible to additionally take into account from what earliest moment in time wagons may begin to appear in these stations and not to build a graph for the corresponding stations until this moment of model time. In this case, in the struggle to reduce the dimension, the only payment is an even greater complication of the algorithms for the formation of matrices and vectors for the problem (1)–(3), which in turn increases the probability of errors when creating such algorithms.

In addition to efforts to further optimize algorithms for the formation of matrices and vectors, another direction for the development of this type of tasks is modernization of the formulation of the optimal management problem of the fleet of freight cars in order to take into account more restrictions. In the current version, the transport problem is of exclusively scientific interest, but not practical in any way. For railway transport operators, who are the main customers of such models, it is important to be able to take into account a sufficiently large number of factors, among which is the possibility to take into account various types of wagons, the prohibition for some types of wagons to enter certain territories, accounting of sediment stations, restrictions on the minimum or maximum number of wagons that must move in the specified directions during the planning horizon etc. The study of the problems described above may be the subject of future research. ■

Acknowledgments

The reported study was funded by RFBR, project number 19-29-06003.

References

1. Higgins A., Ferreira L., Kozan E. (1995) Modeling single-line train operations. Transportation Research Record, vol. 1489, pp. 9–16.
2. Kraay D., Barker P., Chen B. (1991) Optimal pacing of trains in freight railroads: model formulation and solution. Operations Research, vol. 39, no. 1, pp. 82–99. https://doi.org/10.1287/opre.39.1.82
3. Ferreira L., Murray M. (1997) Modelling rail track deterioration and maintenance: current practices and future needs. Transport Reviews, vol. 17, no. 3, pp. 207–221. https://doi.org/10.1080/01441649708716982
4. LeBlanc L. (1976) Global solutions for a nonconvex nonconcave rail network model. Management Science, vol. 23, no. 2, pp. 131–139. https://doi.org/10.1287/mnsc.23.2.131
5. Lazarev A., Musatova E., Kvaratskheliya A., Grafov E. (2012) Schedule theory. Problems of transport system management. Moscow: MSU (in Russian).
6. Lazarev A., Musatova E., Grafov E., Kvaratskheliya A. (2012) Schedule theory. Problems of railway planning. Moscow: ICS RAS (in Russian).
7. Beklaryan L., Khachatryan N. (2006) Traveling wave type solutions in dynamic transport models. Functional Differential Equations, vol. 13, no. 2, pp. 125–155.
8. Beklaryan, L., Khachatryan N. (2013) On one class of dynamic transport models. Computational Mathematics and Mathematical Physics, vol. 53, no. 10, pp. 1649–1667. https://doi.org/10.7868/S0044466913100037
9. Khachatryan N. (2013) Dynamic model of organization of cargo transportation with a limited capacity of the distillation ways. Business Informatics, vol. 26, no. 4, pp. 62–68.
10. Khachatryan N., Akopov A. (2017) Model for organizing cargo transportation with an initial station of departure and a final station of cargo distribution. Business Informatics, vol. 39, no. 1, pp. 25–35. https://doi.org/10.17323/1998-0663.2017.1.25.35
11. Khachatryan N., Akopov A., Belousov F. (2018) About quasi-solutions of traveling wave type in models for organizing cargo transportation. Business Informatics, vol. 43, no.1, pp. 61–70. https://doi.org/10.17323/1998-0663.2018.1.61.70
12. Khachatryan N.K., Beklaryan G.L., Borisova S.V., Belousov F.A. (2019) Research into the dynamics of railway track capacities in a model for organizing cargo transportation between two node stations. Business Informatics, vol. 13, no. 1, pp. 59–70. https://doi.org/10.17323/1998-0663.2019.1.59.70
13. Beklaryan L., Khachatryan N., Akopov A. (2019) Model for organization cargo transportation at resource restrictions. International Journal of Applied Mathematics, vol. 32, no. 4, pp. 627–640. https://doi.org/10.12732/ijam.v32i4.7
14. Khachatryan N., Beklaryan L. (2021) Study of flow dynamics in the model of cargo transportation organization along a circular chain of stations. Economics and Mathematical Methods, vol. 57, no. 1, pp. 83–91. https://doi.org/10.31857/S042473880013024-5
15. Khachatryan N. (2020) Study of flow dynamics in the model of cargo transportation organization between node stations. International Journal of Applied Mathematics, vol. 33, no. 5, pp. 937–949. https://doi.org/10.12732/ijam.v33i5.14
16. Khachatryan N. (2021) Modeling the process of cargo transportation between node stations. International Journal of Applied Mathematics, vol. 34, no. 6, pp. 1223–1235. https://doi.org/10.12732/ijam.v34i6.12
17. Belousov F.A., Nevolin I.V., Khachatryan N.K. (2020) Modeling and optimization of plans for railway freight transport performed by a transport operator. Business Informatics, vol. 14, no. 2, pp. 21–35. https://doi.org/10.17323/2587-814X.2020.2.21.35
18. Sadykov R., Lazarev A., Shiryaev V., Stratonnikov A. (2013) Solving a freight railcar flow problem arising in Russia. Proceedings of the 13th Workshop on Algorithmic Approach for Transportation Modelling, Optimization, and Systems (ATMOS'13), Leibniz, Germany, 5 September 2013, pp. 55–67. https://doi.org/10.4230/OASIcs.ATMOS.2013.55

19. Lazarev A., Sadykov R. (2014) Management problem of railway cars fleet. Proceedings of the XII All-Russian Meeting on Management Issues (VSPU 2014), ICS RAS, Moscow, Russia, 16–19 June 2014, pp. 5083–5093 (in Russian).
20. Desaulniers G., Desrosiers J., Solomon M. (2005) Column generation. New York: Springer. https://doi.org/10.1007/b135457
21. Lübbecke M. (2011) Column generation. Wiley Encyclopedia of Operations Research and Management Science. John Wiley & Sons. https://doi.org/10.1002/9780470400531.eorms0158
22. Frangioni A., Gendron B. (2009) 0–1 reformulations of the multicommodity capacitated network design problem. Discrete Applied Mathematics, vol. 157, no. 6, pp. 1229–1241. https://doi.org/10.1016/j.dam.2008.04.022
23. Sadykov R., Vanderbeck F. (2013) Column generation for extended formulations. EURO Journal on Computational Optimization, vol. 1, nos. 1–2, pp. 81–115. https://doi.org/10.1007/s13675-013-0009-9
24. Dantzig G., Wolfe P. (1960) Decomposition principle for linear programs. Operations Research, vol. 8, no. 1, pp. 101–111. https://doi.org/10.1287/opre.8.1.101
25. Fukasawa R., Aragão M.P., Porto O., Uchoa E. (2002) Solving the freight car flow problem to optimality. Electronic Notes in Theoretical Computer Science, vol. 66, no. 6, pp. 42–52. https://doi.org/10.1016/S1571-0661(04)80528-0

About the authors

Fedor A. Belousov

Cand. Sci. (Econ.);

Senior Researcher, Laboratory of Dynamic Models of Economy and Optimization, Central Economics and Mathematics Institute, Russian Academy of Science, 47, Nakhimovsky Prospect, Moscow 117418, Russia;

E-mail: sky_tt@list.ru

ORCID: 0000-0002-3040-3148

Nerses K. Khachatryan

Cand. Sci. (Phys.-Math.);

Deputy Director of CEMI RAS for Scientific Work, Central Economics and Mathematics Institute, Russian Academy of Science, 47, Nakhimovsky Prospect, Moscow 117418, Russia;

E-mail: nerses-khachatryan@yandex.ru

ORCID: 0000-0003-2495-5736

Ivan V. Nevolin

Cand. Sci. (Econ.);

Leading Researcher, Laboratory of Experimental Economics, Central Economics and Mathematics Institute, Russian Academy of Science, 47, Nakhimovsky Prospect, Moscow 117418, Russia;

E-mail: i.nevolin@cemi.rssi.ru

ORCID: 0000-0002-8462-9011