Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 55-10 (2022) 2557–2562

Comparison of mathematical programming models for optimization of transshipment point seaport - railway ⋆

Egor Grishin * **Nikolay Pravdivets** * **Nikolai Morozov** * **Alexander Lazarev** * **Dmitry Korovkin** ** **Ilia Tyulenev** **

* *V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences 65 Profsoyuznaya street, Moscow 117997, Russia (e-mail: grishin.em16@physics.msu.ru).*
** *Lomonosov Moscow State University, GSP-1, Leninskie Gory, Moscow, 119991, Russia (e-mail: korovkin.dm19@physics.msu.ru).*

Abstract: Sea transport holds the first place in the total number of freight shipments of international transportation. Rail transport takes more than 87% of domestic freight traffic and is increasing annually. In particular, Russian Railways deals with scheduling in international multimodal transport. Sea port-railway transshipment points have a key role in the realization of such transportation. This paper considers the complex problem of unloading the arriving vessels and the formation of trains with the objective function of minimizing the total weighted delivery time of cargo to the destination point and minimizing the cost of forming trains. Two models (binary and integer) were developed and compared. The comparison was provided on pseudo real data which corresponds to Far-East Railway with numbers of vessels and berths comparable with presented in literature. The models were run using Gurobi optimizer.

Keywords: berth allocation problem, railways, discrete optimization, mathematical programming, transshipment point.

1. INTRODUCTION

Supply chains disruptions caused by the pandemic force logistic companies to look for the expenses cut options. One of these is to develop new efficient scheduling algorithms and methods for logistics automation.

Multimodal transportation is one engaging two or more different types of transport, such as water, road or air transport. A harbors terminal considered in this paper consists of seaside represented by a set of berths with loaders and a landside with rails. Such harbors play an essential role in international multimodal transportation since vast amount of international cargo are marine while domestic ones are railway. Especially in counties with huge area covered by railroads like Russian Federation, Australia or Canada.

In general, the problem formulation can be given as follows. A port and its berths are known. We are given a planning horizon, during which we have to define a berth for unloading for all incoming vessels. Each berth can unload only certain types of cargo. A port consists of several operational facilities: berths (seaside), where vessels are unloaded (or loaded); a cargo terminal (yard), which is a buffer area for cargo waiting for further transportation; a loading terminal (landside), where cargo is loaded on land transport and sent to destinations. After unloading the vessels, all cargo has to be assigned to trains for delivery to the destinations.

Planning for harbor seaside is known as Berth allocation problem (BAP) or Berth scheduling problem and shown to be NP-hard by Lim (1998). The BAP objective is to schedule every vessel on a certain berth in order to minimize tardiness, expenses or waiting time. BAPs are usually solved with optimization packages or some heuristic or metaheuristic algorithms. Planning for harbor landside is a Train Formation Problem (TFP). The TFP objective is to obtain railcars assignation and trains' departure schedule in order to minimize tardiness, expenses or waiting time.

Al-Refaie and Abedalqader (2021) present two approaches to constructing a model for the berth allocation problem. The first approach is for normal operation of the berth, and the second one is for the emergency situation for handling as many vessels as possible in the shortest time. Experiments were carried out at real data examples with six vessels and a single wharf for the first model and for 13 vessels for the second one. The LINGO 11.0 optimizer was used to solve the problem, which took 48 hours to solve the first example.

Correcher et al. (2019b) deal with ports with complex layouts that impose certain constraints on the arrival and departure of vessels. A mixed integer linear programming model and a heuristic algorithm are proposed. A heuristic algorithm is presented with an exact solution obtained

⋆ The reported study was funded by RFBR, Sirius University of Science and Technology, JSC Russian Railways and Educational Fund "Talent and success", project 20-38-51010.

10.1016/j.ifacol.2022.10.094

using the CPLEX MILP optimizer for randomly generated examples from 10 to 100 vessels.

Correcher et al. (2019a) consider the complex problem of the distribution of dynamic berths (DBAP) in which vessels can dock anywhere on the berth without requiring a berth discretization and the problem of assigning quay cranes (QCAP). For a continuous problem with constant crane assignment, a mixed integer linear model is proposed. The model is extended by adding several families of admissible inequalities. The authors also propose an iterative branch-and-cut algorithm. Experiment results show that problems with up to 40 vessels can be optimally solved.

A variant of the DBAP and QCAP problem in which each berth can serve multiple vessels at the same time if their total length is equal to or less than the length of the berth is discussed by El Hammouti et al. (2019). A Modified Sailfish Optimizer (MSFO) metaheuristic based on the hunting behavior of a sailfish has been developed. Computational experiments on examples with 60 vessels and 13 berths show results comparable to other metaheuristic algorithms.

Cervellera et al. (2021) investigate application of complex Markov decision problems to the classical BAPs. The problem is considered as assigning berths to vessels according to a parameterized policy function that drives the temporal evolution of the environment, instead of the usual MILP formulation. Experiments show that this approach can be useful in some special cases of the problem.

A similar problem of train-to-train transshipment is considered by Cichenski et al. (2017), but the problem considered in our paper is different because of different types of transportation.

As for train formation problem (TFP), it is studied widely as a separate problem of railway companies.

Lan et al. (2019) optimize three subproblems of car flow routing, TFP, and train routing minimizing the total transportation cost, accumulation cost, and classification cost. A model for GUROBI solver is constructed for the small- and medium-size cases of the problem. For large instances, authors suggest a two-stage Benders-and-Price approach with an arc-based model with some variables fixed as the corresponding values fetched from the first phase.

Lin and Zhao (2019) study a train formation problem regard to recurring patterns in rail loading stations. A nonlinear 0-1 programming model for LINGO optimization package is constructed.

A combined algorithm based on a goal programming approach and an L-p norm method along with a Simulated Annealing algorithm are suggested by Alikhani-Kooshkak et al. (2019). Authors solve such problems as customers satisfaction level, optimizing yard activity and the problem of minimizing underutilized trains tonnage.

To obtain high-quality solution of the train formation problem in the seaport, it is necessary to take into account the features of the port. However, there are few publications that jointly consider the BAP and TFP, and since BAP is NP-hard then the joint problem is also NP-hard. For example, this joint problem is considered by Yan et al. (2020b). Operators are assumed to have low computing resources and tight time limits. The problem is formulated as an integer programming model. A tailored rolling horizon approach with the adaptive horizon and back-tracking strategy is proposed. Computation experiment is conducted using a GUROBI solver. Authors conclude that the performance of quay cranes is the most significant factor in this problem. In next paper, Yan et al. (2020a) the schedule plan of trains and the transshipment plan are determined simultaneously. A mixed integer programming (MIP) for the combined problem is proposed. Relaxation of some variables and adding some additional constraints makes the model more effective. Extended experiments still show that quay cranes are still the most important factor, and that increasing the storage cost of import containers leads to a more effective transshipment plan.

There are many companies, i.e. Russian Railways, which provide comprehensive logistics services. The income of these companies depends both on seaside and landside planning. Since combination of optimal solutions of BAP and TFP obtained independently won't guaranteed to be an optimal solution of combined BAP-TFP problem, we propose models for combined BAP-TFP problem. The objective of BAP-TFP problem is to unload marine cargo to trains and deliver to its destinations.

In this paper, we compare two models made by us for BAP-TFP problem: binary and integer. The purpose of integer model was to increase scheduling interval. The number of decision variables in the binary model depends on time period discretization, which is time interval over sampling rate. Since the corresponding decision variable in integer model was depending on vessels number, we expected significant increase in productivity i.e. computational time reducing.

The rest of paper is organized as follows. Section 2 describes the problem itself. Section 3 introduces binary and integer models. Section 4 gives a comparison between binary and integer models. Finally, section 5 presents our conclusions.

2. THE PROBLEM

The problem's objective is to deliver cargoes from harbors to the destination as soon and as cheap as possible. Cargoes arrive in harbor on vessels, which must be assigned for unloading at a specified berth at a certain time, then cargo trains must be formed which are railed to their destination. Each berth is equipped with a set of loaders, the type and number of which affect the unloading speed and the variety of cargo types that can be unloaded at that berth. Each cargo is of a specific type: coal, non-ferrous metals, etc.

The first part of the problem can be characterized as a dynamic BAP problem with discrete berth assignment (only one vessel may stay at a berth at any given time) and deterministic arrival and unloading times of vessels.

The second part, which is the problem of train formation, can be characterized as a cargo clustering problem based on such attributes as destination, cargo type, volume, arrival time, and weight coefficient.

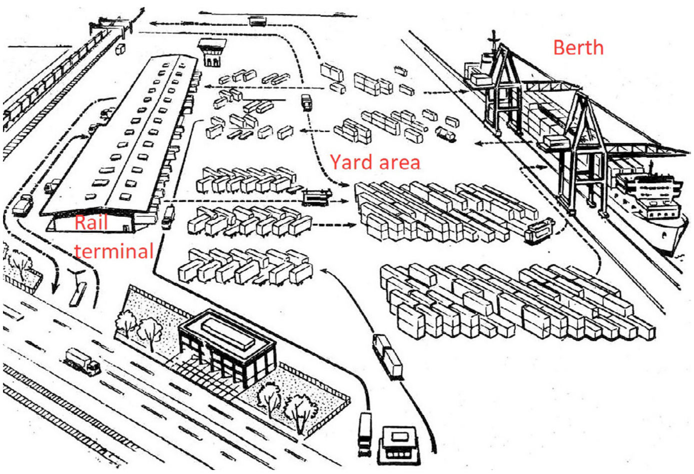


Fig. 1. Structure of a harbor.

The arrival times of vessels in the port during the planning period are pre-determined. Formed trains move directly to one of the destinations. All destinations are the nearest marshalling yards to the port.

For each cargo, a weight coefficient is assigned and there is a fixed cost for the formation of the trains. The objective function is to minimize the total weighted cost of delivering all cargoes to their destinations.

In Fig. 1 the scheme of a harbor is presented. At first, it is necessary to unload a vessel (BAP). After all goods are unloaded, it is possible to place them at trains (TFP) with further delivery by railway.

3. MODELS

Let's introduce the notation of the sets used in the problem:

- $\mathbf{B}$ – set of all vessels coming to the port;
- $\mathbf{M}$ – set of all berths;
- $\mathbf{J}$ – set of all goods;
- $\mathbf{K}$ – set of all product types;
- $\mathbf{D}$ – set of all destinations;
- $\mathbf{L}$ – set of all trains;
- $\mathbf{T} = \{t \in \mathbb{N} \mid t \leq T/\Delta T\}$ – set of time moments within the planning horizon.

We list the characteristics of their designation for vessels, cargo and berths necessary to draw up constrains.

For each train, we have:

- $D_L \in \mathbf{D}$ – destination;
- $c \in \mathbb{N}$ – maximum number of railcars;
- $\Delta \in \mathbb{N}$ – minimum time interval between train departures;
- $a \in \mathbb{N}$ – train formation cost.

For each product $J \in \mathbf{J}$ at vessel $B \in \mathbf{B}$ the following characteristics are known:

- $\mathbf{J}^B \subseteq \mathbf{J}$ – set of all goods at the vessel B;
- K_J – type of the good;
- r_B – release time of the vessel B;
- $m_J \leq c$ – amount of the product in railcars;
- w_J – weight of the good;
- $D_J \in \mathbf{D}$ – destination;
- d_J – time of cargo transportation from the port to the destination.

For each berth $M \in \mathbf{M}$ the following characteristics are known:

- $\mathbf{K}^M \subseteq \mathbf{K}$ – set of types of cargo that the berth is capable of unloading M;
- r_M – release time;
- v_M – the amount of cargo unloaded at the berth per time unit;
- $p_{B,M}$ – handling time of the vessel $B \in \mathbf{B}$ at the berth $M \in \mathbf{M}$,

$$p_{B,M} = \begin{cases} \sum\limits_{J \in \mathbf{J}^B} \dfrac{m_J}{v_M}, & K_J \in \mathbf{K}^M, \\ +\infty, & K_J \notin \mathbf{K}^M. \end{cases} \tag{1}$$

3.1 Decision variables

All parameters were denoted. Then two lists of decision variables are presented. For the binary model, the following decision variables were introduced:

- $x(B, M, t)$ – main binary decision variable, $x(B, M, t) = 1$, iff the vessel $B \in \mathbf{B}$ started unloading at the berth $M \in \mathbf{M}$ at the time $t \in \mathbf{T}$;
- $y(J, L)$ – binary decision variable, $y(J, L) = 1$, iff the product $J \in \mathbf{J}$ is assigned to the train $L \in \mathbf{L}$;
- $C_L \in \mathbb{N}$ – completion time of the train $L \in \mathbf{L}$.

In the integer model, another decision variables were created:

- $t(B, M, n)$ – main integer decision variable, $t(B, M, t) = T$, iff the vessel $B \in \mathbf{B}$ started unloading at the berth $M \in \mathbf{M}$ under number $n \in \mathbf{N}$ at the time T, $\mathbf{N} = \{x \in \mathbb{N} \mid x \leq N\}$, where

$$N = \max_{M \in \mathbf{M}} \left\{ \dim \bigcup_{K \in \mathbf{K}^M} \mathbf{B}^K \right\} \leq \dim \mathbf{B},$$

  where $\mathbf{B}^K$ – set of vessels with at least one cargo of the type K;
- $\delta(B, M, n)$ – binary decision variable, $\delta(B, M, n) = 1$ iff the vessel $B \in \mathbf{B}$ is unloaded at the berth B in schedule $M \in \mathbf{M}$ under number $n \in \mathbf{N}$;
- $y(J, L)$ – binary decision variable, $y(J, L) = 1$, iff the product $J \in \mathbf{J}$ is assigned to the train $L \in \mathbf{L}$;
- $C_L \in \mathbb{N}$ – completion time of the train $L \in \mathbf{L}$.

In the binary and in the integer models, the main decision variables characterize the same thing (start time of handling) but in different way. In the integer model additional variables $\delta(B, M, n) = 1$ are added in order to set simpler constraints. Both $y(J, L)$ and C_L are the same in the models.

The objective function is the same for both models and can be written as follows:

$$F = \underbrace{\sum_{J\in\mathbf{J}} w_J \cdot \left(\sum_{L\in\mathbf{L}} C_L \cdot y(J,L) + d_J\right)}_{\text{delivering time cost}} + \underbrace{a \cdot \sum_{L\in\mathbf{L}} \min\left\{1, \sum_{J\in\mathbf{J}} y(J,L)\right\}}_{\text{train formation cost}}, \quad (2)$$

where w_j – cargo weight coefficient, C_L – train release time, $y(J,L)$ – indicator if the cargo was assigned on the train, d_J – delivery time, a – train formation cost.

3.2 Time moments

In the integer model, the main variable defines the handling start time of the vessel by itself. In the binary model, handling start time of the vessel $B \in \mathbf{B}$ can by defined by the following way:

$$t^B_{start}(x) = \sum_{M\in\mathbf{M}} \sum_{t\in\mathbf{T}} t \cdot x(B,M,t). \quad (3)$$

Similarly, it is possible to set handling end time of the vessel $B \in \mathbf{B}$ at the binary model:

$$t^B_{end}(x) = t^B_{start}(x) + \sum_{M\in\mathbf{M}} \sum_{t\in\mathbf{T}} p_{B,M} \cdot x(B,M,t). \quad (4)$$

At the same time, it is possible to define release time of a product $J \in \mathbf{J}$ at the vessel $B \in \mathbf{B}$. All products are ready when the certain vessel is unloaded:

$$r_J(x) = t^B_{end}(x), \quad J \in \mathbf{J}^B. \quad (5)$$

3.3 BAP constraints

Several specific constraints appear in response to the joint problem of railway and sea port optimization. The differences between integer and binary model exist only in sea port constraints, which are presented at first.

Obviously, unloading of each vessel can be carried out only once at one berth. In binary model this means that only one decision variable $x(B,M,t)$ (indicator of handling start time) equals to one for each vessel:

$$\sum_{M\in\mathbf{M}} \sum_{t\in\mathbf{T}} x(B,M,t) = 1,\ \forall B \in \mathbf{B},\ \forall B \in \mathbf{B}. \quad (6)$$

In the integer model, this constraint can be written similarly. Instead of $x(B,M,t)$ we use $\delta(B,M,n)$ (index of unloading for each vessel):

$$\sum_{M\in\mathbf{M}} \sum_{n\in\mathbf{N}} \delta(B,M,n) = 1, \quad \forall B \in \mathbf{B}. \quad (7)$$

That is impossible to unload a vessel before its arrival or beyond a berth availability interval. In the binary model, it is possible to set all variables $x(B,M,t)$ equal zero for all forbidden time moments:

$$\forall t \in \mathbf{T}: \quad t \le max(r_B, r_M) \to x(B,M,t) = 0, \quad \forall B \in \mathbf{B},\ \forall M \in \mathbf{M}. \quad (8)$$

In the integer model, there are decision variables $t(B,M,n)$ which are linked with a time moments directly. This means we have to set the following constraint for them:

$$t(B,M,n) \ge max(r_B, r_M), \quad \forall B \in \mathbf{B}, \forall M \in \mathbf{M}, \forall n \in \mathbf{N}. \quad (9)$$

Because of the Far East region ports structure at any time moment at each berth, no more than one vessel is unloaded. In the binary model, tricky constraint was introduced. For each berth if any vessel is under unloading then during processing time of certain vessel no other vessel can be unloaded. For this issue, additional variable $\theta = \max\{0, t - p_b\}$ was introduced. Then we can sum up the following expression for each time moments $t \in \mathbf{T}$:

$$\sum_{B\in\mathbf{B}} \sum_{\theta=\max\{0,t-p_b\}}^{t} x(B,M,\theta) \le 1, \quad \forall M \in \mathbf{M},\ \forall t \in \mathbf{T}. \quad (10)$$

If $x(B,M,\theta)$ is equal to one, then at time period $[\theta;\ t]$ only one vessel is handled. In the integer model three constraints have been set for this issue:

$$t(B_i,M,n+1) \ge \big(t(B_j,M,n) + p_{B_j,M} \cdot \delta(B_j,M,n)\big) \cdot \delta(B_i,M,n+1), \quad \forall M \in \mathbf{M}, \forall B_i, B_j \in \mathbf{B},\ \forall n \in \mathbf{N} \setminus \{|\mathbf{N}|\}, \quad (11)$$

$$\sum_{B\in\mathbf{B}} \delta(B,M,n) \le 1,\ \forall M \in \mathbf{M},\ \forall n \in \mathbf{N}, \quad (12)$$

$$\delta(\overline{B},M,n+1) \le \sum_{B\in\mathbf{B}\setminus\{\overline{B}\}} \delta(B,M,n), \quad \forall \overline{B} \in \mathbf{B},\ \forall M \in \mathbf{M},\ 2 \le n < |\mathbf{N}|. \quad (13)$$

The first of them define that handling start time of a vessel which undergo unloading by the $(n+1)$-th at the berth B is greater than a handling end time of a vessel which is processed at the same berth by n-th. The next condition prohibits for each berth the possibility of unloading two vessels under the number n. The last constraint declares that after unloading the n-th vessel, the $(n+1)$-th vessel should be unloaded (omissions in numbering are prohibited).

Each berth can unload only an allowed product type set. This constraint is the same for the both models. If it is unfeasible to unload a certain vessel at a berth, then $x(B,M,t) = 0$ in the binary model:

$$K_J \notin \mathbf{K}^M \to x(B,M,t) = 0, \quad \forall M \in \mathbf{M},\ \forall B \in \mathbf{B},\ \forall J \in \mathbf{J}^B,\ \forall t \in \mathbf{T}. \quad (14)$$

The same condition was set for the integer model:

$$K_J \notin \mathbf{K}^M \to \delta(B,M,n) = 0, \quad \forall M \in \mathbf{M},\ \forall B \in \mathbf{B}, \forall J \in \mathbf{J}^B,\ \forall n \in \mathbf{N}. \quad (15)$$

3.4 TFP constraints

Constraints related to the assignment of cargo to trains are the same for both models. Each product unloaded from a vessel can be assigned to only one train:

$$\sum_{L\in\mathbf{L}} y(J,L) = 1,\ \forall J \in \mathbf{J}. \quad (16)$$

The number of railcars in each train does not exceed the specified number, then if a product is assigned to a train the remaining train capacity decreases by the volume of this product. Total volume of all goods assigned to a train cannot exceed a defined value:

$$\sum_{J\in\mathbf{J}} y(J,L)\cdot m_J \le c,\ \forall L\in\mathbf{L}. \quad (17)$$

We introduced virtual trains. The maximum number of trains required to deliver all cargo is equal to the number of all cargo arriving at all vessels during the planning period. Then the capacities of the sets of trains L and goods J coincide, i.e. $|L| = |J|$. We will assign a destination to each train as follows: $D_{(J_i)} = D_{(L_i)}$, where $i = 1, ..., |J|$. Initially, we will consider all trains "virtual". A destination is set for each virtual train, but no cargo is assigned that can be sent as part of it. If one or more products are assigned to a train in the final schedule, then the train ceases to be virtual. If no cargo is assigned to a certain train, then it remains virtual and is not included in the final schedule. In this case, the train can contain only goods that must be delivered to the destination to which the train is heading. The cost of forming a virtual train, unlike a real one, is not taken into account in the objective function, so the solution will contain a minimum number of real trains. The introduction of virtual trains makes it guaranteed to get a valid solution. For each product, the destination coincides with the destination of the train on which this product was assigned:

$$D_J \ne D_L \to y(J,L) = 0,\ \forall J\in\mathbf{J},\ \forall L\in\mathbf{L}. \quad (18)$$

The train departs not earlier than the goods specified on it are available. In this case, a train release time must be greater than the release time of all products assigned to it:

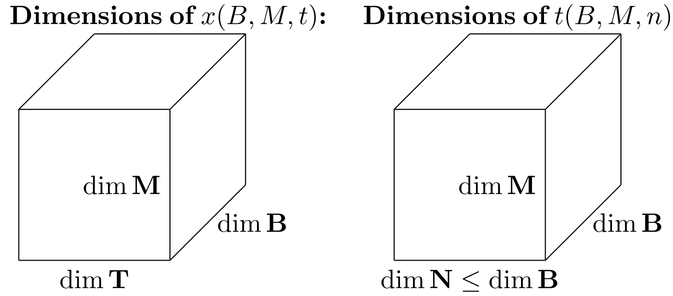
$$C_L \ge \max_{J\in\mathbf{J}}\{r_J(x)\cdot y(J,L)\},\ \forall L\in\mathbf{L}. \quad (19)$$

At least a specified interval Δ must pass between the departures of any two trains. Then if two trains are not virtual, the $\min\{1, \sum_{J\in\mathbf{J}} y(J,L_1), \sum_{J\in\mathbf{J}} y(J,L_2)\}$ will be equal to one. In opposite case (at least one train is virtual) this expression equals to zero and there is no necessity in such a constraint:

$$|C_{L_1} - C_{L_2}| \ge \Delta\cdot\min\{1, \sum_{J\in\mathbf{J}} y(J,L_1), \sum_{J\in\mathbf{J}} y(J,L_2)\},$$
$$\forall L_1, L_2\in\mathbf{L}. \quad (20)$$

4. COMPARISON

There are advantages as well as disadvantages in both models. The main difference is the number of main decision variables. Two of three parameters of decision variables $x(B,M,t)$ and $t(B,M,n)$ are the same, but the last one differs. In the binary model, the third parameter of the variable is a time moment, while in the integer model it is an index in loader planning. It is important to note that number of indexes $\mathbf{N}$ is less or equals to the number of all vessels $\mathbf{B}$ and significantly less than total time moments number $\mathbf{T}$. In the diagram below, there is a scheme of decision variables arrays.



The planning period was chosen as one week. That is the most popular period presented in scientific literature. The time unit is 20 min (discretization), so $\dim\mathbf{T} = 504$ (planning horizon – 1 week, time unit – 20 min) and $\dim\mathbf{B} \approx 10$.

Thus, changing a binary model to an integer one can significantly reduce the number of decision variables and constraints. At the same time, possible values of variables are greater in the integer models. In addition, the constraint (10) in the binary model was replaced by constraints (11)-(13) in the integer model. It is quite difficult to decide what is better. In the binary model this constraint is only but rather difficult for optimizer. The opposite issue is in the integer model: instead of one, there are three constraints, but all of them are easier.

5. RESULTS

Researchers use different solution methods for different sizes of input data and different problem statements. Both short planning horizons lasting from 6 hours and long ones – up to several months and years are considered. Since the problem is NP-hard, it is impossible to get an optimal solution of large-sized examples using an optimizer. In this paper, a planning horizon of 1 week was chosen, since at such an interval release times of vessels become known, and there is also enough time to obtain a solution using the optimizer. We considered the Far Eastern region, where 4 ports are located: Vladivostok, Vanino, Nakhodka, Vostochny, in which there are 15, 13, 5 and 4 berths, respectively. The number of incoming vessels depends on the port, but varies from 3 to 12. Vessels can contain several types of cargo at the same time: ferrous and non-ferrous metals, coal, timber, alumina and containers. At the same time, containers are equivalent to one railway car, and other types of cargo are measured in tons. To unify the volume of all types of cargo, everything was converted into a Twenty–foot Equivalent Unit (TEU) - a conventional unit of measurement for the capacity of cargo vehicles. Thus, one container, TEU and a standard railcar are equivalent and correspond to the same volume of cargo. There are from 1000 to 5000 TEU on each vessel. For small ports with a low unloading rate, vessels from 150 TEU are possible. The total volume of cargo arriving at the port per week was calculated based on the reporting data for 2020. The rate of unloading of vessels at berths varies for each type of cargo and for each berth and lies in the interval from 15 to 50 TEU per unit of time (20-minute interval). Seven nearest marshalling yards have been selected as destinations, the travel time to which from each port is determined using Dijkstra's algorithm. The

Berth x Vessel	Binary model			Integer model		
	Time, sec.	Gap, %	Decision variables	Time, sec.	Gap, %	Decision variables
4 x 3	4	0	7104	0.8	0	1092
4 x 4	4	0	8616	0.13	0	616
4 x 5	2	0	10460	0.14	0	480
5 x 3	1367	0	12672	3600	10.8	5157
5 x 5	3600	7	15356	3600	9.3	2881
5 x 7	21	0	21546	46	0	4151
13 x 6	3600	2.8	54812	3600	2.8	15968
13 x 8	3600	8.7	69708	3600	2.6	18124
13 x 10	3600	9.2	82080	3600	4.5	22180
15 x 8	3600	1	83432	3600	1	23912
15 x 10	3600	4.8	107106	3600	0.9	30912
15 x 12	3600	8.5	116802	3600	3.4	28242

technical speed of movement of freight trains is assumed to be 44 km/h. The length of each train cannot exceed 80 rail cars. If cargo arrives in a volume exceeding 80 TEU (80 railway cars), such cargo is divided into several with the same destination, but the volume of which does not exceed 80 TEU (for example, 123 TEU = 80 TEU + 43 TEU). This allows us to reduce the number of goods under consideration. Instead of considering each TEU separately, batches of TEUs with the same data are considered at once, which reduces the number of decision variables used. An interval of 20 minutes was selected per unit of time in the model. This is due to the fact that it allows you to significantly reduce the set of all possible time moments, and also correlates well with the time discretization adopted by rail and sea freight. Thus, a planning horizon of 1 week is equivalent to 504 intervals of 20 minutes. A weight coefficient is set for each cargo, taking a value from 1 to 10, while they differ for each type of cargo. For example, for coal it lies in the range from 1 to 3, and for containers from 7 to 10. The cost of forming one train (including assigning a locomotive to it) is assumed to be 200, which is on average equivalent to forming a new train of 40 railcars. The model is implemented in C++ and solved using the Gurobi 9.1.2 optimizer on an Intel Core i9-10980HK processor, 32 Gb RAM. The choice of the Gurobi solver is justified by the fact that it is well acclaimed in solving BAP problems having a scale similar to ours. Although Gurobi works slower on average on large input data compared to heuristic algorithms, it gets a solution that is closer to optimal. Priority is given to the quality of the solution, not the speed of calculations, because the solution time is much less than the planning horizon. At the same time, the duration of possible delays arising in the schedule as a result of inaccuracy of the solution significantly exceeds the solution time. Input data sets of various dimensions were generated, the maximum of which included 15 berths and 12 vessels. To find a solution, the optimizer's working time was limited to one hour. The results of the experiments and comparison are presented in Table 1.

REFERENCES

Al-Refaie, A. and Abedalqader, H. (2021). Optimal berth allocation under regular and emergent vessel arrivals. *Proceedings Of The Institution Of Mechanical Engineers Part M-Journal Of Engineering For The Maritime Environment*, 235(2), 642–656. doi: 10.1177/1475090220937888.

Alikhani-Kooshkak, R., Tavakkoli-Moghaddam, R., Jamili, A., and Ebrahimnejad, S. (2019). Multi-objective mathematical modeling of a n integrated train makeup and routing problem in an Iranian railway company. *Scientia Iranica*, 26(6), 3765–3779. doi:10.24200/sci.2018.20782.

Cervellera, C., Gagger, M., and Maccio, D. (2021). Policy Optimization for Berth Allocation Problems. In *2021 International Joint Conference On Neural Networks (IJCNN)*, IEEE International Joint Conference on Neural Networks (IJCNN). Int Neural Network Soc; IEEE Computat Intelligence Soc. doi: 10.1109/IJCNN52387.2021.9533891. International Joint Conference on Neural Networks (IJCNN), ELECTR NETWORK, JUL 18-22, 2021.

Cichenski, M., Jaehn, F., Pawlak, G., Pesch, E., Singh, G., and Blazewicz, J. (2017). An integrated model for the transshipment yard scheduling problem. *JOURNAL OF SCHEDULING*, 20(1), 57–65. doi:10.1007/s10951-016-0470-4.

Correcher, J.F., Alvarez-Valdes, R., and Tamarit, J.M. (2019a). New exact methods for the time-invariant berth allocation and quay crane assignment problem. *European Journal Of Operational Research*, 275(1), 80–92. doi:10.1016/j.ejor.2018.11.007.

Correcher, J.F., Van den Bossche, T., Alvarez-Valdes, R., and Vanden Berghe, G. (2019b). The berth allocation problem in terminals with irregular layouts. *European Journal Of Operational Research*, 272(3), 1096–1108. doi:10.1016/j.ejor.2018.07.019.

El Hammouti, I., Lajjam, A., El Merouani, M., and Tabaa, Y. (2019). A modified sailfish optimizer to solve dynamic berth allocation problem in conventional container terminal. *International Journal Of Industrial Engineering Computations*, 10(4), 491–504. doi: 10.5267/j.ijiec.2019.4.002.

Lan, Z., He, S., Xu, Y., and Wang, Y. (2019). Optimizing Train Formation Problem With Car Flow Routing and Train Routing by Benders-and-Price Approach. *IEEE Access*, 7, 178496–178510. doi: 10.1109/ACCESS.2019.2958601.

Lim, A. (1998). The berth planning problem. *Operations Research Letters*, 22(2), 105–110. doi: https://doi.org/10.1016/S0167-6377(98)00010-8.

Lin, B. and Zhao, Y. (2019). The Systematic Optimization of Train Formation in Loading Stations. *Symmetry-Basel*, 11(10). doi:10.3390/sym11101238.

Yan, B., Jin, J.G., Zhu, X., Lee, D.H., Wang, L., and Wang, H. (2020a). Integrated planning of train schedule template and container transshipment operation in seaport railway terminals. *Transportation Research Part E-Logistics And Transportation Review*, 142. doi: 10.1016/j.tre.2020.102061.

Yan, B., Zhu, X., Lee, D.H., Jin, J.G., and Wang, L. (2020b). Transshipment operations optimization of sea-rail intermodal container in seaport rail terminals. *Computers & Industrial Engineering*, 141. doi: 10.1016/j.cie.2020.106296.