Складання вантажного плану контейнеровоза, що враховує максимальну кількість факторів, вимагає врахування структурних обмежень контейнерів і судна, обмежень при розміщенні і правил сегрегації в разі небезпечних вантажів.

Облік обмежень при розміщенні контейнерів з небезпечними вантажами, так званих IMO контейнерів (IMO – International Maritime Organization), видається актуальним, тому що в даний час відзначається тенденція зростання обсягу перевезень небезпечних вантажів.

Пропонований підхід для вирішення завдання автоматизації складання вантажного плану на контейнерному судні полягає в поділі завдання на два етапи. На першому етапі проводиться розрахунок допустимого положення контейнерів з урахуванням конструктивних обмежень і сумісності небезпечних вантажів, на другому – розрахунок параметрів безпеки (остійності, міцності та інших).

Пропонується булева математична модель цілочисельного лінійного програмування, що враховує конструктивні особливості контейнерів, судна і правила розміщення небезпечних вантажів відповідно до Кодексу IMDG (International maritime dangerous goods code), а також модифікований адитивний алгоритм для розв'язання задачі складання вантажного плану контейнеровоза. Для перевірки математичної моделі був обраний класичний алгоритм, який спирається на ідеї загального методу гілок і меж. У зв'язку з тим, що отримана математична модель для задачі завантаження контейнерного судна небезпечними вантажами має специфічний вид, цей алгоритм був доповнений тестами, які дозволяють відкидати деякі рішення без безпосередньої перевірки.

Наводиться приклад рішення задачі з розміщення вантажів у трюмі з урахуванням структурних обмежень контейнерів і правил розміщення небезпечних вантажів відповідно до Кодексу IMDG, який був отриманий за допомогою модифікованого адитивного алгоритму

Ключові слова: контейнеровози, вантажний план, булева математична модель, небезпечні вантажі, адитивний алгоритм

UDC 629.123.03
DOI: 10.15587/1729-4061.2019.183385

CONSTRUCTION OF A MATHEMATICAL MODEL AND A METHOD FOR ARRANGING HAZARDOUS CARGOES ON A CONTAINERSHIP

K. Kamieniev
Postgraduate student*
E-mail: chp.od.ua@gmail.com
A. Kamienieva
PhD, Associate Professor**
E-mail: avk@te.net.ua
M. Tsymbal
Doctor of Technical Sciences, Professor*
E-mail: tsn53@ukr.net
*Department of Shipping***
**Department of Theory of
Automatic Control and Computing***
***National University
"Odessa Maritime Academy"
Didrikhsona str., 8, Odessa, Ukraine, 65029

Received date 09.10.2019
Accepted date 08.11.2019
Published date 13.12.2019

1. Introduction

The world community has been increasingly focused on the issue related to improving the safety of navigation, as solving it successfully would have a positive effect on reducing the number of accidents and, as a result, decreasing the number of casualties, harm to health, to the environment, to assets and industrial processes. Given an intense competition in the container fleet, the requirements for timely, reliable, and safe delivery of cargoes become even more important, so every opportunity to improve the safety of navigation should be used.

The safety of navigation is a multifaceted problem requiring that a large number of factors should be taken into consideration, many of which depend on the correct loading of a vessel.

The ship is loaded according to a pre-approved load plan, which includes all cargoes, their allocation for holds (for containerships), their attachment, the order of loading, as well as the distribution of ballast and ballast operations. Such a plan largely determines the voyage safety. Compiling a load plan for a containership in advance requires additional consideration of parameters of the containers themselves, such as their types, dimensions, features of contents, etc. When transporting hazardous cargoes in containers, attention should be paid to the requirements for the location and segregation of such cargoes.

The current trend demonstrates an increase in the volume of dangerous cargo transportation. However, such deliveries exert a negative impact on the environment and, in the event of an accident, could result in injury or death. This threat is another reason to focus on improving the safety and efficiency of transporting dangerous cargoes [1].

The task is complicated by the fact that the capacity of containerships in operation has been increasing. According to DynaLiner analysts, as of January 1, 2018, there were 451 ultra-large containerships (ULCS) in the world [2]. ULCS denote a container capacity of 10,000 TEU (twenty-foot equivalent unit). According to BIMCO (Baltic and International Maritime Council), the world's largest international non-governmental organization in the field of shipping, more than 80 % of ships launched in 2018 had a capacity of 15,000 TEU and above.

The increase in the capacity of containerships and, therefore, the number of factors to be considered, predetermines the relevance of automating the pre-planning of cargo placement on ships and compiling a load plan.

Given the complexity of the task, different mathematical models are used depending on specific statement and, therefore, different approaches are applied to solve it.

This problem has been termed the Master Bay Plan problem; solving it has been often addressed in modern scientific literature.

2. Literature review and problem statement

A detailed description of this problem is given in [3]. Formally, the main issue related to a bay plan is to determine the pattern of arranging a set of containers of different types within the finite cargo space of a containership, taking into consideration the structural and operational limitations for both the containers and the vessel itself.

Different approaches are used to solve a given problem.

Paper "Stowing a containership: the master bay plan problem" [3] gives a simplified model of linear programming that is solved by dividing the problem into three stages.

The first stage is termed "Pre-processing"; its main task is to prevent the arrangement of containers that does not meet the strictly specified conditions. Out of the set, which includes all possible positions of a container, those are excluded in which the container cannot be by default. Thus, for a refrigerated container, one excludes positions outside the access to power sources, for some types of dangerous cargoes – positions on the deck, etc. As a result of this sorting, the initial set is significantly reduced. The second stage is termed "Preloading Procedure," at which containers are divided into sets depending on the unloading port. In a given model, individual bays are assigned to certain ports, and the loading order is arranged so that the containers intended for the first unloading ports are closer to the center of the vessel. As a result, those that do not meet the specified conditions for beys are excluded from the existing set of positions.

The third stage implies solving a system of linear equations, taking into consideration the sets excluded at the first and second stages. It should be noted that this algorithm does not consider either the parameters of strength or stability. Instead, different assumptions are used. For example, lighter containers are loaded onto heavier ones. Or, the total weight of the containers located in the bow relative to the midsection of the vessel should be equal to the total weight of the containers located in the stern part of the vessel. Such assumptions may exclude solutions that are valid for significant parameters.

In addition, the cited work did not address the rules for the placement of dangerous cargoes.

Article [4] gives a model of integer linear programming for the task of loading containers. However, the given model does not take into consideration the rules of placing dangerous cargoes, which excludes a significant part of actual transportation and makes this model inapplicable for many voyages.

Work [5] gives a Boolean model of integer programming for the task on determining the position of containers aboard a ship along its route in order to minimize the number of unproductive movements. However, when drawing up a mathematical model, it was assumed that all containers were the same size while the conditions of stability and the rules for placing dangerous cargoes were not considered.

Authors of [6] propose an integer model for Master Bay Planning, taking into consideration segregation rules for containers with dangerous cargoes. However, the authors do not account for the vessel draft, which could cause problems regarding the requirements from the Load Line convention. In addition, the authors do not offer an algorithm of solution and do not give the qualitative results from calculating the solution to the master problem of a bay plan, which does not make it possible to evaluate the stated results.

Paper [7] uses a heuristic algorithm that does not take into consideration the rules for placing dangerous cargoes. The main purpose of a given algorithm is to improve the parameters of stability. The examples show its effectiveness, but this algorithm does not take into consideration the overall strength of a vessel while the roll and differentiation are changed by rolling a ballast.

Article [8] considers the loading algorithm taking into consideration the restrictions imposed on containers with dangerous cargoes, using a branch and bound method. However, the cited article addresses only a few limitations, as indicated by the authors themselves.

Paper [9] uses a particular case of the branch and bound method to determine the position of containers in the bey taking into consideration the restrictions imposed on containers with dangerous cargoes, but does not take into account the dimensions of containers.

Article [10] addresses the arrangement on containerships the type of river–sea. The study considers two optimization criteria: operations shift and a stability coefficient. The model combines limitations that reflect the actual operation of a terminal and meet the assigned structural and operational constraints associated with both the vessel and containers. The model is designed for river–sea containerships, but does not take into consideration the restrictions imposed on containers with dangerous cargoes.

Work [11] compares various heuristic methods for the task on determining the placement of containers, within the so-called bay plan problem. However, it is assumed that the vessel is initially empty while the rules for placing dangerous cargoes are not taken into consideration.

Authors of [5] tested the effectiveness of integer programming models to solve the bay plan problem.

Thus, given the complexity of the problem, different approaches are used to solve the problem of automating the loading of containerships.

Mathematical models adopt various simplistic assumptions that make the models in question applicable only in individual cases. In this case, the reported models do not take into consideration all the factors that are necessary for safe and reliable cargo transportation. Due to these reasons, construction of a mathematical model aimed at compiling a load plan for a containership is an unresolved task and requires a more detailed study.

3. The aim and objectives of the study

The aim of this study is to develop a mathematical model for compiling a load plan for a containership and to obtain the permissible position of containers taking into consideration the structural limitations and compatibility of dangerous cargoes. That would make it possible to automate the compilation of a preliminary load plan for containerships, which could reduce the impact of the human factor thereby improving the safety of navigation.

To accomplish the aim, the following tasks have been set:

– to consider the structural limitations of containers and a ship, as well as the rules for placing dangerous cargoes, and to take them into account in the mathematical model;

– to conduct a numerical experiment with “severe” restrictions: in this case, the structural restrictions for containers and a vessel, as well as the rules for placing dangerous cargoes in accordance with the IMDG Code, would make it possible to arrange different types of containers within an extremely limited number of positions.

4. Accounting for the structural limitations for containers, a ship, and rules for placing dangerous cargoes

The proposed mathematical model takes into consideration the structural limitations for containers, for a ship, as well as the rules for placing dangerous cargoes.

When transporting dangerous cargoes by sea, national and international legislation on packaging, labelling, arrangement, and clearance of cargoes must be met [12].

The main legal document regulating the transportation of dangerous cargoes by sea is the IMDG Code (International Maritime Dangerous Cargoes Code). The IMDG Code is compiled on the basis of official IMO (International Maritime Organization) documents.

The rules for the placement and segregation of hazardous cargoes are contained in section 7 of the International Code of Maritime Transport for Dangerous Cargoes [13]. Dangerous cargoes are divided into classes and subclasses. Table 1 gives 9 classes of dangerous cargoes.

Table 1

Classes of hazardous cargoes under the IMDG Code

Class number	Title
Class 1	Explosive materials
Class 2	Gases
Class 3	Flammable liquids
Class 4	Flammable solids; substances that can spontaneously ignite; substances emitting flammable gases in contact with water
Class 5	Oxidizing agents and organic peroxides
Class 6	Toxic and infectious substances
Class 7	Radioactive materials
Class 8	Corrosive substances
Class 9	Various dangerous substances and objects

In addition to the general requirements for the transportation of dangerous cargoes, containerships are subject to additional requirements. The requirements for segregation on containerships are given in the code in the form of segregation tables, such as Table 2, and in the form of drawings [13].

Table 2

Segregation table of dangerous cargoes

CLASS	1. 1, 1. 2, 1. 5	1. 3, 1. 6	1. 4	2. 2	2. 1	2. 3	3	4. 1	4. 2	4. 3	5. 1	5. 2	6. 1	6. 2	7	8	9
1. 1,1. 2, 1. 5	*	*	*	2	4	2	4	4	4	4	4	4	2	4	2	4	X
1. 3, 1. 6	*	*	*	2	4	2	4	3	3	4	4	4	2	4	2	2	X
1. 4	*	*	*	1	2	1	2	2	2	2	2	2	X	4	2	2	X
2. 1	4	4	2	X	X	X	2	1	2	X	2	2	X	4	2	1	X
2. 2	2	2	1	X	X	X	1	X	1	X	X	1	X	2	1	X	X
2. 3	2	2	1	X	X	X	2	X	2	X	X	2	X	2	1	X	X
3	4	4	2	1	2	2	X	X	2	1	2	2	X	3	2	X	X
4. 1	4	3	2	X	1	X	X	X	1	X	1	2	X	3	2	1	X
4. 2	4	3	2	1	2	2	2	1	X	1	2	2	1	3	2	1	X
4. 3	4	4	2	X	X	X	1	X	1	X	2	2	X	2	2	1	X
5. 1	4	4	2	X	2	X	2	1	2	2	X	2	1	3	1	2	X
5. 2	4	4	2	1	2	2	2	2	2	2	2	X	1	3	2	2	X
6. 1	2	2	X	X	X	X	X	X	1	X	1	1	X	1	X	X	X
6. 2	4	4	4	2	4	2	3	3	3	2	3	3	1	X	3	3	X
7	2	2	2	1	2	1	2	2	2	2	1	2	X	3	X	2	X
8	4	2	2	X	1	X	X	1	1	1	2	2	X	3	2	X	X
9	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Numbers and symbols in the table denote the following:

– 1 – "at a distance from";

– 2 – "separated from";

– 3 – "separated by a whole compartment or hold from";

– 4 – "separated by a longitudinal full compartment or hold from";

– X – check against the List of Dangerous Cargoes;

– * – Section 7 of the International Code of Maritime Transport of Dangerous Cargoes should be checked to establish rules for segregation between Class 1 cargoes.

Let us make a mathematical model of the problem on optimizing the placement of cargoes on a containership, which defines some limitations more accurately.

The bay of a containership is part of a freight hold; load plans are typically compiled in the form of bay plans (Fig. 1). Fig. 1 shows the bay (table) whose cells can either be filled with containers or remain empty.

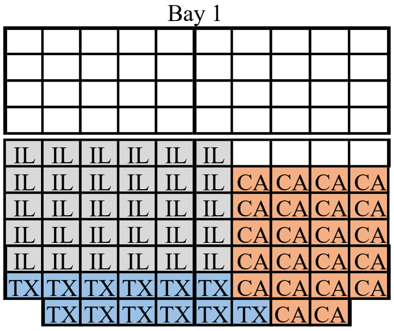

Fig. 1. Example of a single bay-plan

If a container takes position (i, j, k), the variable corresponding to the container is 1, otherwise it is 0.

Containers come in two main sizes:

– 20-foot equivalent unit (TEU) is 20 feet long, 8 feet wide and 8 feet 6 inches high;

– 40-foot equivalent unit (FEU) that is double the length at the same width and height.

Assume:

n^0_{FEU} – number of FEU containers with non-hazardous cargoes to be loaded onto a vessel;

n^1_{FEU} – number of FEU containers with Class 1 cargoes;

...

n^{17}_{FEU} – number of FEU containers with Class 17 cargoes.

By analogy:

n^0_{TEU} – number of TEU containers with non-hazardous cargoes;

n^1_{TEU} – number of TEU containers with Class 1 cargoes;

...

n^{17}_{TEU} – number of TEU containers with Class 17 cargoes.

We shall designate via $f(c, i, j, k)$ a FEU container with class c cargo in position (i, j, k), via $t(c, i, j, k)$ – a TEU container with class c cargo in position (i, j, k) (Fig. 2).

Variable c can take 18 different values: 1 – if a cargo class is 1. 1 (Table 2), 2 – if a cargo class is 1. 2, ..., 17 – if a cargo class is 9, 0 – if a cargo is non-dangerous.

Thus

$$f_{cijk} = \begin{cases} 1, & \text{if there is a class } c \text{ FEU} \\ & \text{in the position } (i, j, k), \\ 0, & \text{otherwise.} \end{cases}$$

$$t_{cijk} = \begin{cases} 1, & \text{if there is a class } c \\ & \text{TEU in the position } (i, j, k), \\ 0, & \text{otherwise.} \end{cases}$$

Consider the limitations:

– for the number of containers with class c^* cargoes –

$$\sum_i \sum_j \sum_k f_{c^* ijk} = 2n^{c^*}_{FEU},$$

where $c^* \in \{0, 1, ..., 17\}$;

– by analogy

$$\sum_i \sum_j \sum_k t_{c^* ijk} = n^{c^*}_{TEU},$$

where $c^* \in \{0, 1, ..., 17\}$. .

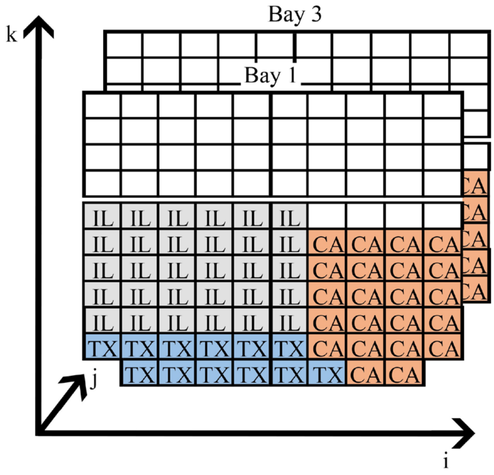

Fig. 2. Accepted coordinate system

Upon reducing to a standard form for the optimization problem, these limitations take the form:

$$-\left(\sum_i \sum_j \sum_k f_{c^* ijk}\right) \le -2n^{c^*}_{FEU}, \tag{1}$$

where $c^* \in \{0, 1, ..., 17\}$;

$$-\left(\sum_i \sum_j \sum_k t_{c^* ijk}\right) \le -n^{c^*}_{TEU}, \tag{2}$$

where $c^* \in \{0, 1, ..., 17\}$.

When the cargo hold contains only 1 bay, then only TEU containers can be loaded. When the hold contains 2 bays, FEU containers can also be loaded. In this case, FEU containers are marked as standard on one bay, and on the other bee they are marked with "X" (Fig. 3).

Thus, a FEU container occupies 2 places (along j):

$$| f_{c\, i\, j0\, k} - f_{c\, i\, (j0+1)\, k} | \le 0, \tag{3}$$

where $j0 \in J_{FEU}$.

Here, J_{FEU} is the set of indices (positions) where one can place FEU containers.

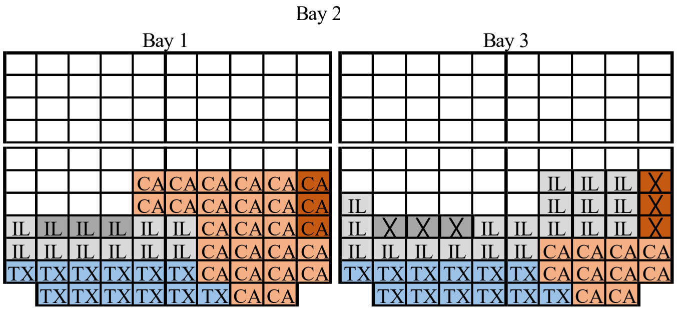
Bay 2
Bay 1
Bay 3
CA CA CA CA CA CA
CA CA CA CA CA CA
IL IL IL IL IL IL CA CA CA CA
IL IL IL IL IL IL CA CA CA CA
TX TX TX TX TX TX CA CA CA CA
TX TX TX TX TX TX CA CA
IL IL IL X
IL IL IL IL X
IL X X X IL IL IL IL IL X
IL IL IL IL IL IL CA CA CA CA
TX TX TX TX TX TX CA CA CA CA
TX TX TX TX TX TX CA CA

Fig. 3. Example of a bay plan with two bays in one hold

In position (i, j, k), the hold can either host a FEU container, or a TEU container, or it can be empty.

If $f_{c\,i\,j\,k}=1$, then $t_{c\,i\,jk}=0$, and vice versa, if $t_{c\,i\,j\,k}=1$, then $f_{c\,i\,jk}=0$, where $c\in\{0, ..., 17\}$. This limitation in the form of an inequality is as follows:

$$\sum_c \left(f_{c\,i\,j\,k}+t_{c\,i\,j\,k}\right)\le 1. \qquad (4)$$

Containers are designed in such a way that it is possible to put 1 FEU container on 2 TEU containers, but not the other way around. This adds an additional set of limitations that are often ignored.

Thus, if

$$f_{c\,i0\,j0\,k0}=1,$$

then

$$\sum_c \sum_{k\succ k0}\left(t_{c\,i0\,j0\,k}+t_{c\,i0\,(j0+1)\;k}\right)=0.$$

Hence, it follows

$$\sum_c \sum_{k\succ k0}\left(t_{c\,i0\;j0\;k}+t_{c\,i0(\,j0+1)\;k}\right)+M\,N_c\sum_c f_{c\,i0\;j0\;k0}\le M\,N_c, \qquad (5)$$

where $j0\in J_{FEU}$, M is the maximum number of containers in a stack, N_c is the number of classes of dangerous cargoes.

It is also impossible to load FEU containers on a TEU container if the stack heights do not match, that is a FEU container cannot be placed on adjacent TEU stacks of different heights. Denote:

$$A=\sum_c \sum_{k=1}^{k0} t_{c\,i0\;j0\;k}$$

and

$$B=\sum_c \sum_{k=1}^{k0} t_{c\,i0\,(\,j0+1)k}.$$

If $|A-B|\succ 0,$ that is the stacks are of different heights, then

$$\sum_c f_{c\,i0\,j0(k0+1)}+\sum_c f_{c\,i0\,(\,j0+1)(k0+1)}=0.$$

An additional logical variable $w_s\in\{0; 1\}$ is introduced for each s-th limitation [14]. Then each limitation is converted into a system of inequalities:

$$\begin{cases}\left|A_s-B_s\right|\le M\left(1-w_s\right),\\ \left[\sum_c f_{c\,i0\,j0(k0+1)}+\sum_c f_{c\,i0\,(\,j0+1)(k0+1)}\right]_s\le M\,w_s,\end{cases} \qquad (6)$$

where M is the maximum height of a stack.

Restrictions for the placement of containers with dangerous cargoes (Table 1) shall be considered using an example. Let position ($i0, j0, k0$) be taken by a FEU container of Class 4.1 (c=8). Then, $f_{8,\,i0,\,j0,\,k0}$=1. It is required to place a Class 5.2 container (c=12).

As it follows from Table 2, for containers with cargoes of Classes 4.1 and 5.2:

– $f_{12,\,i0,\,j0,\,k}=0,\;\; t_{12,\,i0,\,j0,\,k}=0,\;\; f_{12,\,i0,\,j0+1,\,k}=0,\;\; t_{12,\,i0,\,j0+1,\,k}=0$

for $\forall k$, that is

$$\sum_k \left(f_{12\,i0\,j0k}+t_{12\,i0\,j0k}+f_{12\,i0\,(\,j0+1)k}+t_{12\,i0\,(\,j0+1)k}\right)\le 0;$$

– $f_{12,\,i0-1,\,j0,\,k}=0,\; t_{12,\,i0-1,\,j0,\,k}=0,\; f_{12,\,i0-1,\,j0+1,\,k}=0,\; t_{12,\,i0,\,j0+1,\,k}=0$

for $\forall k$, that is

$$\sum_k \left(f_{12\,(i0-1)\,j0k}+t_{12\,(i0-1)\,j0k}+f_{12\,(i0-1)(\,j0+1)}+t_{12\,(i0-1)(\,j0+1)k}\right)\le 0;$$

– $f_{12,\,i0+1,\,j0,\,k}=0,\; t_{12,\,i0+1,\,j0,\,k}=0,\; f_{12,\,i0+1,\,j0+1,\,k}=0,\; t_{12,\,i0+1,\,j0+1,\,k}=0$

for $\forall k$, that is

$$\sum_k \left(f_{12\,(i0+1)\,j0k}+t_{12\,(i0+1)\,j0k}+f_{12\,(i0+1)\,(\,j0+1)k}+t_{12\,(i0+1)(\,j0+1)k}\right)\le 0.$$

Similar restrictions hold for the placement of containers with dangerous cargoes for all classes of cargoes.

All restrictions mostly take the form:

if

$$f_{c0,\,i0,\,j0,\,k0}=1$$

or

$$t_{c0,\,i0,\,j0,\,k0}=1,$$

then the sum of certain variables $f_{c,\,i,\,j,\,k}$ and $t_{c,\,i,\,j,\,k}$ is zero. In the form of inequalities, these restrictions take the form:

$$\sum_{cijk}\left(f_{cijk}+t_{cijk}\right)\le N\left(1-f_{c0i0\,j0k0}\right)$$

or

$$\sum_{cijk}\left(f_{cijk}+t_{cijk}\right)\le N\left(1-t_{c0i0\,j0k0}\right), \qquad (7)$$

where N is the total number of containers.

5. Method for solving the problem on compiling a load plan for a containership

The considered optimization problem can be attributed to a particular case of problems of integer linear programming – Boolean programming, under which variables can accept only two values – “0” and “1.” Using the Boolean variables simplifies computational procedures, as each variable takes only 2 values. This is taken into consideration when constructing an algorithm to solve problems involving Boolean variables. In the considered algorithm, computational operations are limited only to addition and subtraction. Therefore, an algorithm for solving problems with Boolean variables is sometimes termed an additive method. This algorithm relies on ideas from a general branch and bound method [15].

To implement the additive algorithm, the optimization problem must take the following form.

Minimize

$$z = \sum_{j=1}^{n} c_j x_j ,$$

where c_j>=0, under constraints

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i,$$

where x_j=0 or 1, i=1, 2, .., m, j=1, 2, .., n.

The simplex table for a given problem defines the acceptable solution when all c_j>=0 [15].

To solve the problem by an additive algorithm, the limitations are recorded in the form

$$\sum_{j=1}^{n} a_{ij} x_j + S_i = b_i, \tag{8}$$

where $S_i \geq 0$ is an additional variable corresponding to the i-th limitation, i=1, 2, .., m, j=1, 2, .., n.

The original problem on placing containers shall be stated as follows: to minimize function

$$z = \sum_{cijk} \left(f_{cijk} + t_{cijk} \right) \tag{9}$$

under the above constraints (1) to (7), which are written in form (8).

The basic idea of an additive algorithm is to sort out 2^n possible solutions to the original problem. The sorting procedure is carried out in a special way, which makes it possible to discard certain solutions without direct verification. Ultimately, implementing an additive algorithm requires a direct consideration of only part of the 2^n possible solutions.

At the first step, all the original variables are assumed to equal 0. This seems logical, because in the objective function all c_j>=0. The resulting solution (not a single container is loaded) is not acceptable, as some $S_i < 0$. Therefore, some original variables need to be assigned a value of 1. The purpose of this procedure is to meet the condition $S_i \geq 0$ for $\forall i$=1, 2, .., m, that is ensuring the admissibility of the solution. At each step of the algorithm, variables are determined that need to be assigned values 1 and 0. This selection is made using four classical tests [15], implemented in the object-oriented programming language C# (standardized as ECMA-334 and ISO/IEC 23270) [16]. When the algorithm was implemented, multidimensional arrays f_{cijk} and t_{cijk}, as well as a one-dimensional array w_s, were converted into a one-dimensional array (x).

```
public class Method
{
static public int test1(int[,] a, int[] s, int r)
{. . .}
static public int test2(int r, int[] c, int zmin, int zsol)
{. . .}
static public bool test3(int[,] a, int[] s, int[] nt)
{. . .}
static public int test4(int[,] a, int[] xnt, int[] s)
{. . .}
}
```

Given that the mathematical model of the problem on loading a containership takes a certain form, described above, the classic additive algorithm was supplemented with 4 more tests. These tests make it possible to assign zero values to unknown variables without additional checks.

This reduces the number of searches for possible solutions to the original problem.

Thus, it is impossible to put a TEU container on a FEU container, so if the solving process produced a variable value of $f(c0, i0, j0, k0)=1$, then $t(c, i0, j0, k)=0$ for $\forall k > k0$ and $\forall c \in \{0, 1,..., 17\}$:

```
static public void test5(int num, int[] x)
// x[num]=1
{. . .}
```

If the solution process produced a variable value of $f(c0, i0, j0, k0)=1$, then $t(c, i0, j0, k0)=0$ for $\forall c \in \{0, 1,..., 17\}$ and $f(c, i0, j0, k0)=0$ for $\forall c \in \{0, 1,..., 17\}$, $c \neq c0$.

Similarly, if $t(c0, i0, j0, k0)=1$, then $t(c, i0, j0, k0)=0$ for $\forall c \in \{0, 1,..., 17\}$, $c \neq c0$ and $f(c, i0, j0, k0)=0$ for $\forall c \in \{0, 1,..., 17\}$. This is a consequence of the statement that there may be only one container in the hold in position *(i, j, k),* or it may be empty:

```
static public void test6(int num, int[] x)
// x[num]=1
{. . .}
```

If a certain restriction has only non-negative coefficients, and the right-hand side of this restriction is 0:

$$\sum_{j=1}^{n} a_{i0\,j} = \sum_{j=1}^{n} \left| a_{i0\,j} \right| \succ 0$$

and $b_{i0} = 0$, then the variable corresponding to factor $a_{i0\,j0} \neq 0$ (an element from array f_{cijk} or t_{cijk}) is assigned 0. Such restrictions (7) in the problem on loading a vessel taking into consideration dangerous cargoes are necessarily present, as mentioned above. Test 7 makes it possible to exclude these variables from further consideration:

```
static public void test7(int[,] a, int[] b, int[] x)
// danger Cargoes
{. . .}
```

Any variable to which a value of 1 is assigned resulting in an objective function value greater than $\sum_{c=0}^{17} \left(n_{TEU}^{c} + n_{FEU}^{c} \right)$, should remain zero. Test 8 excludes such variables from consideration:

```
staticpublic int test8(int r, int[] c, int nc)
{. . .}
```

The algorithm under consideration is also supplemented with a module that makes it possible to set the initial location of containers.

The practical use of computer software intended to solve integer problems typically involves active interference of the operator in the computational process [15]. Human control over the course of computations makes it possible to improve the efficiency of program implementation. The specified module is also intended for this purpose.

6. Example of implementing the proposed method

As an example, consider 40 containers (5 *FEU* and 35 *TEU*), including 3 *FEU* and 1 *TEU* with a Class 4. 1 dangerous cargo. 1, located in the hold as shown in Fig. 4. It is required to place additional 6 containers (2 *FEU* and 4 *TEU*), including 1 *FEU* and 1 *TEU* with a dangerous cargo of Class 5. 2.

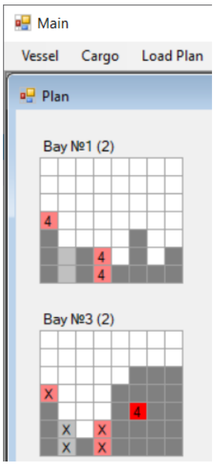

Fig. 4. Original placement of containers in the hold: ■ — *TEU* with a Class 0 cargo; ▨ — *FEU* container with a Class 0 cargo; 4 — *FEU* container with a Class 4.1 cargo; 4 — *TEU* container with a Class 4.1 cargo

According to restrictions (7), containers with dangerous cargoes of Class 5.2 can be placed only in positions (7, 0, 2), (7, 0, 3), (7, 0, 4), (7, 0, 5), (7, 0, 6), (7, 1, 5), (7, 1, 6), that is only in the last column of bays No. 1 and No. 3. As for *FEU* containers, a *FEU* container with a Class 5.2 cargo can only be placed in 2 places: (7, 0, 5), (7, 1, 5) and (7, 0, 6), (7, 1, 6) and only if there are no empty places under them. Such "severe" conditions are chosen to test and demonstrate the adequacy of the model and robustness of the method. In practice, a much larger number of positions are acceptable to accommodate containers with dangerous cargoes.

Fig. 5 shows a screenshot of one of the solutions obtained using the modified additive algorithm.

Thus, we have obtained the acceptable arrangement of containers, namely:

1. All original containers are placed in accordance with terms (1) and (2).
2. FEU containers occupy 2 places, and are not placed on stacks of TEU containers of different heights according to conditions (3) and (6).
3. Positions taken by TEU and FEU do not intersect according to condition (4).
4. TEU containers are not placed on FEU containers as required by condition (5).
5. The segregation of dangerous cargoes is met in accordance with condition (7).

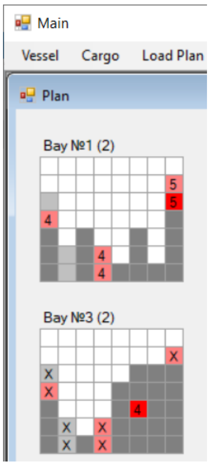

Fig. 5. Acceptable placement of containers in the hold: ■ — *TEU* container with a Class 0 cargo; ▨ — *FEU* container with a Class 0 cargo; 4 — *FEU* container with a Class 4.1 cargo; 4 — *TEU* container with a Class 4.1 cargo; 5 — *FEU* container with a Class 5.2 cargo; 5 — *TEU* container with a Class 5.2 cargo

7. Discussion of results from numerical experiment

The result of our numerical experiment is the derived acceptable positions of containers, one of which is shown in Fig.5. The initial data, as indicated above, were deliberately chosen to allow different types of containers to be placed within a very limited number of positions.

This may indicate the adequacy of the proposed mathematical model for a bay plan problem, as the result of the calculation using the model in question corresponds to the acceptable location of containers in the hold.

The mathematical model used is Boolean, which takes into consideration the structural features of containers, as well as a ship, and the rules for placing dangerous cargoes, in contrast to models considered by other authors. Restrictions include the impossibility of building a Boolean model, which would also test the parameters of stability, strength of the vessel, and other similar ones, which is why we decided to divide the process into two stages.

To verify the mathematical model, a classic non-heuristic algorithm was deliberately chosen, based on the ideas from a general branch and bound method, which was modified for a bay plan problem.

It should be noted that there is a relationship between the duration of calculations and the specific type of problem statement. The order under which limitations are considered has a direct impact on the effectiveness of the algorithm, as does the total number of variables. It is preferable to order the restrictions as their "rigidity" descends.

It should also be noted that there are limitations that are not considered in the proposed mathematical model, such as placing dangerous cargoes in open and closed holds, which need to be considered in the future.

In addition, the additive algorithm does not provide high efficiency. Therefore, it is advisable to investigate other algorithms, including heuristic ones, to solve the master problem of a bay plan, taking into consideration the requirements by the IMDG Code.

8. Conclusions

1. A mathematical model has been constructed for compiling a load plan for a containership, which reflects the structural limitations for containers, a ship, and the rules for placing dangerous cargoes. The classic algorithm, based on the ideas from the general branch and bound method, and consisting of 4 tests, was supplemented with 4 more tests to solve the master problem of a bay plan. That makes it possible to automate the pre-drafting of a containership's load plan, taking into consideration the requirements by the IMDG Code.

2. A numerical experiment has been carried out, resulting in the obtained acceptable solutions for a task on drawing up a containership's load plan. The calculation was performed using a modified classical algorithm under "severe" restrictions. The numerical experiment has shown that the constructed mathematical model is adequate to solve the set problem, as the result of the calculation using the model in question corresponds to the acceptable location of containers in the hold.

Reference

1. Galierikova, A., Sosedova, J. (2018). Intermodalni prijevoz opasnih tereta. Naše More, 65 (3), 8–11. doi: https://doi.org/10.17818/nm/2018/3.8
2. Krupneyshie vladel'tsy "megamaksov" v mire. Available at: https://ports.com.ua/articles/krupneyshie-vladeltsy-megamaksov-v-mire
3. Ambrosino, D., Sciomachen, A., Tanfani, E. (2004). Stowing a containership: the master bay plan problem. Transportation Research Part A: Policy and Practice, 38 (2), 81–99. doi: https://doi.org/10.1016/j.tra.2003.09.002
4. Wang, L., Ni, M., Gao, J., Shen, Q., Jia, Y., Yao, C. (2019). The Loading Optimization: A Novel Integer Linear Programming Model. Enterprise Information Systems, 13 (10), 1471–1482. doi: https://doi.org/10.1080/17517575.2019.1631964
5. Parreño-Torres, C., Alvarez-Valdes, R., Parreño, F. (2019). Solution Strategies for a Multiport Container Ship Stowage Problem. Mathematical Problems in Engineering, 2019, 1–12. doi: https://doi.org/10.1155/2019/9029267
6. Kebedow, K. G., Oppen, J. (2018). Including Containers with Dangerous Goods in the Multi-Port Master Bay Planning Problem. MENDEL, 24 (2). doi: https://doi.org/10.13164/mendel.2018.2.023
7. Zeng, M., Low, M. Y. H., Hsu, W. J., Huang, S. Y., Liu, F., Win, C. A. (2010). Automated stowage planning for large container-ships with improved safety and stability. Proceedings of the 2010 Winter Simulation Conference. doi: https://doi.org/10.1109/wsc.2010.5678873
8. Ambrosino, D., Sciomachen, A. (2015). Using a Bin Packing Approach for Stowing Hazardous Containers into Containerships. Springer Optimization and Its Applications, 1–18. doi: https://doi.org/10.1007/978-3-319-18899-7_1
9. Kamieniev, K. I., Kamienieva, A. V. (2018). Vykopystannia adytyvnoho alhopytmu dlia rozmishchennia nebezpechnykh vantazhiv na konteinernomu sudni. Sudovozhdenie: sbornik nauchnyh tpudov, 28, 70–77.
10. Yaagoubi, A. E., El Hilali Alaoui, A., Boukachour, J. (2018). Multi-objective river-sea-going container barge stowage planning prob-lem with container fragility and barge stability factors. 2018 4th International Conference on Logistics Operations Management (GOL). doi: https://doi.org/10.1109/gol.2018.8378102
11. Ambrosino, D., Anghinolfi, D., Paolucci, M., Sciomachen, A. (2010). An Experimental Comparison of Different Heuristics for the Master Bay Plan Problem. Lecture Notes in Computer Science, 314–325. doi: https://doi.org/10.1007/978-3-642-13193-6_27
12. On Transportation of Dangerous Cargos. Available at: https://zakon.rada.gov.ua/laws/show/1644-14?lang=en
13. IMDG Code (2012). Vol. 1. CPI Group (UK) Ltd, Croydon, 486.
14. Taha, H. (2018). Issledovanie operatsiy. Sankt-Peterburg: OOO «Dialektika», 1056.
15. Taha, H. (1985). Vvedenie v issledovanie operatsiy. Kn. 1. Moscow: Mir, 479.
16. Neygel, K., Iv'en, B., Glinn, D., Uotson, K., Skinner, M. (2011). C#4.0 i platforma.NET 4 dlya professionalov. Moscow: Dialektika, Vil'yams, 1440.