ARCHIVES OF TRANSPORT *ISSN (print):* *0866-9546*
Volume 44, Issue 4, 2017 ***e-ISSN (online):*** *2300-8830*
DOI: *10.5604/01.3001.0010.6162*

MULTI-OBJECTIVE EVOLUTIONARY METHOD FOR CARGO ARRANGEMENT IN A LOADING SPACE

Michał Taraska[1], Remigiusz Iwańkowicz[2]

[1,2] West Pomeranian University of Technology Szczecin, Faculty of Maritime Technology and Transport, Szczecin, Poland

[2]e-mail: iwankow@zut.edu.pl

Abstract: *The paper characterises a class of problems for packing boxes in the container. It presents the current state of knowledge in this area and distinguishes evolutionary algorithms, as the most promising in the search for quasi-optimal loading conditions. The method proposed in the paper focuses on certain criteria important from a practical point of view, which in a formalised manner have not been included so far in the solving-problems models. Apart from the traditional consideration of the problem of three-dimensional space loading maximisation, the proposed method considers the deviation of the loaded container weight centre from its symmetry planes and the availability of packages during unloading. New elements in the described method are: the applied criteria, penalty function, way of coding the container loading state in the evolutionary algorithm and fast crossover and mutations operators dedicated to the adopted coding. It was observed that the evolution in the developed algorithm occurs properly, that is seeking to minimise the criteria. The article also includes a calculation example showing the effect of the method with the discussion of the results indicating the advantages and disadvantages of the proposed solution. The performance of the algorithm has been considered in the context of time necessary to obtain the acceptable solution and quality of the obtained solution. It was found that the algorithm in its current form is a strong base for its further improvement.*

Key words: *cargo arrangement, three-dimensional loading, container packing, evolutionary optimization.*

1. Introduction

The containerization of cargo has an extremely important role in transporting cargos on the national, European and global arena. The 70s started with a growing trend unbroken to this day of transport operations using the loading units, such as a container. South-east Asia is an example of the dominance of this form of transport. According to forecasts of experts, in next 10 years the largest container ports in the world (Shanghai, Singapore, Hong Kong and Shenzhen) will need to increase their throughput several times to handle all the containers shipped there (Yap et al., 2013). This shows how important it is to deal with the problem of optimising the packing of three-dimensional space. In addition to the problem, popular among researchers, of maximising the use of the cargo space of the container, the issues of distributing the cargo within the container are also important. This problem can affect both the arrangement of weight of the packed boxes, and their availability during the unloading operations. The even weight arrangement of goods inside the container is extremely important from the point of view of safety. It allows the reduction of risk, resulting from the lack of stability and the possible tilting of containers, during cargo handling operations, and in the transport process. The problem of packages priority results from the fact that one container often contains goods to be unloaded at different points. Therefore, wrong deployment of goods in terms of availability leads to the performance of additional handling operations inside the container, what increases the costs and time of the whole transport process.

Searching for a way to fit the boxes in a container is called 3D bin packing problem (3D-BPP). Solving the problem of loading the three-dimensional space is based on the most effective use (filling) of the considered space, with a previously defined set of boxes. 3D-BPP is characterised by a large complexity increasing with the number of boxes and it is included in a group of NP-difficult problems (Bożejko et al. 2014; George & Robinson, 1980). A better use of the usable space of the container can lead to the reduction of the number of containers, necessary for the transport of the required volume of cargo, and thus limit the number of means of transport needed to transport the cargo. Such solution would have a positive effect on the environment, road infrastructure, safety and it would be a tool conducive to reducing congestion. An efficient method to quasi-optimise the process of packing the container would reduce the need to increase the dimensions of means of transport (mainly ships).

In the issues of packing the three-dimensional space we can distinguish two types of problems:

- problems of packing the cargo in the space,
- problems of distributing the cargo in the space.

In the packing problems, we take into consideration the spatial dimensions of the considered goods. These dimensions concern the basic sizes of lumps and figures, including: heights, length and width. The basic packing criterion is the maximisation of using the loading space (Gürbüz et al., 2009; Maarouf et al., 2008), however, some authors attempt to extend the problem with additional aspects. For example, the model presented in the paper (Kacprzak et al., 2015) does not allow packages floating freely, and strives for the greatest variety of loaded goods.

Problems from the group of cargo allocation, in turn, use a group of factors, which have a significant impact on the way of allocating the packaging in the container, at the same time without changing the space occupied by particular boxes. An important factor may include the order in which the boxes should be delivered to different points, because the way of packing the container can impact the performance of the unloading process.3D-BPPtend to be considered with multi-vehicle routing problems (MVRP). In paper (Suarez & Anticona, 2010) the authors assume that each vehicle delivers loads only to one place. It is an oversimplification of the problem of packing the set of boxes with diversified destination points and order of unpacking.

Among the quasi-optimisation methods used in the 3D-BPP solution, the greatest popularity was achieved by the evolutionary algorithms (Bożejko et al. 2014; Gonçalves & Resende, 2013) simulated annealing (Kacprzak et al., 2015) and taboo-search (Lodi et al., 2002). Quasi-optimisation is most often carried out in conjunction with heuristics adapted to the specific assumptions of the model (Wu et al., 2010). For example, authors of the paper (Gürbüz et al., 2009) proposed an LAFF algorithm, wherein the first arranged boxes have the largest wall surface, while the stacking of the boxes as low as possible is another criterion.

Problems of packing and arrangement can be considered as separate issues, or they can be combined for a comprehensive look at the problem of three-dimensional space utilisation. In the article, this problem was dealt with comprehensively. The focus was mainly on the problems of the arrangement of the cargo inside the container, but the indications of problems of space packing were also included.

The described method of the quasi-optimal arrangement of the boxes in the container is based on the evolutionary algorithm. An individual population is represented by a matrix, which contains information about the coordinates and the arrangement in the Euclidean space of all boxes from the given set (Fig. 1).

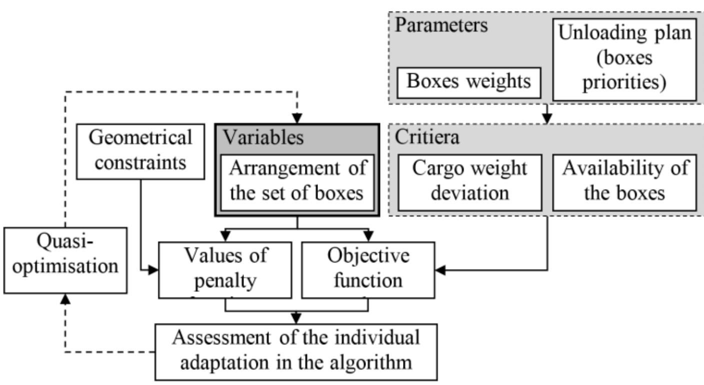

Fig. 1. Diagram of the proposed model for finding quasi-optimal solution.

Figure 1 illustrates the conceptual schema of the proposed algorithm. External data (in squares with dashed border) are the physical parameters of boxes and their unloading priorities. The weights of the criteria needed to determine the final value of the penalties function are also considered as external data. This allows for better control of the algorithm's performance (in specific situations where the criterion is undesirable, it can be disabled by setting the corresponding weight value to zero). The graph center represents the flow of data by an algorithm (continuous lines). Feedback loop where the proposed solution is transformed to obtain the best quasi-optimal solution is shown by a dashed line. New elements in the proposed method are the applied criteria, penalty function, way of coding the container loading state in the evolutionary algorithm and fast crossover and mutations operators dedicated to the adopted coding.

Apart from the traditional consideration of the problem of three-dimensional space loading maximisation, the method considers the availability of packages during unloading and the deviation of the loaded container weight centre from its symmetry planes.

2. Mathematical model

2.1. Decision variables

The rectangular space of the container is given with the fixed dimensions: l_k×b_k×h_k and the set n of rectangular boxes, which in this space are to be deployed. Each i-th box (i=1, 2, …, n) is described with parameters: length l_i, width b_i, height h_i, mass m_i and priority π_i. It is assumed that all parameters characterising the boxes are natural numbers. The priority is interpreted as follows: the box with a lower priority value will be removed from the container before the box with a higher value of priority.

The coordinate XYZ system is linked with the container, against which the geometric centre of each box is located – Fig. 2.

It is assumed that the box can only adopt such a position, in which each dimension of the main is directed parallel to one of the XYZ system axes. Thus, the location of the box in the container is defined by one of six permutations of dimensions: $\boldsymbol{\sigma}(i)^1 = (l_i, b_i, h_i)$, $\boldsymbol{\sigma}(i)^2 = (l_i, h_i, b_i)$, $\boldsymbol{\sigma}(i)^3 = (b_i, l_i, h_i)$, $\boldsymbol{\sigma}(i)^4 = (b_i, h_i, l_i)$, $\boldsymbol{\sigma}(i)^5 = (h_i, l_i, b_i)$ or $\boldsymbol{\sigma}(i)^6 = (h_i, b_i, l_i)$.

For permutation $\sigma(i)^p$ (where p=1, 2, …, 6) the dimension $\sigma(i)_1^p$ of i-th box is directed along the X axis, the dimension $\sigma(i)_2^p$ along the Y axis, while the dimension $\sigma(i)_3^p$ along the Z axis.

Decision variables are the coordinates of the centre of each box and the order of its dimensions. The arrangement of the n set of boxes can be written as a matrix:

$$\mathbf{S} = \begin{pmatrix} x_1 & y_1 & z_1 & p_1 \\ x_2 & y_2 & z_2 & p_2 \\ \dots & \dots & \dots & \dots \\ x_n & y_n & z_n & p_n \end{pmatrix} \qquad (1)$$

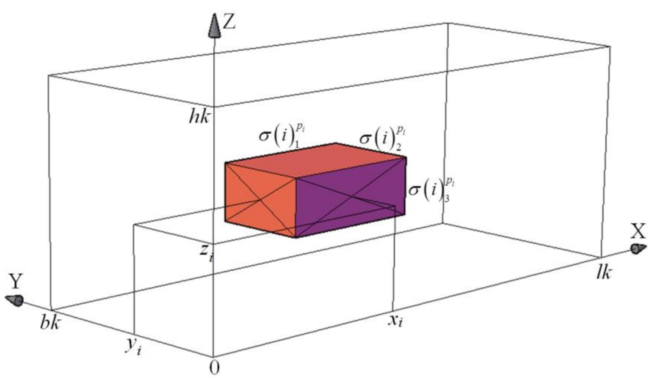

Fig. 2. The box placed in the container

where:
x_i, y_i, z_i – coordinates of the geometric centre of i-th box, $\forall i=1,...,n: x_i \in [1, l\_k-1]$, $y_i \in [1, b\_k-1]$, $z_i \in [1, h\_k-1]$,
p_i – number of permutation of the box dimensions, $\forall i=1,...,n: p_i \in \{1,2,...,6\}$.

Thus, the space of solutions includes $\left(6(l\_k-1)(b\_k-1)(h\_k-1)\right)^n$ of possible arrangement matrixes. We can observe a strong influence of the adopted length unit on the number of the search space. The problem of unit selection is a separate optimisation issue, where we should consider the minimisation of calculation time and maximisation of the utility of obtained solutions.

2.2. Constrains

It is assumed that an unacceptable solution is the one, wherein at least one of the boxes is not distributed inside the container or there is a couple of boxes, which penetrate each other. The conditions for the inclusion of boxes in the container is as follows:

$$\begin{aligned} &\forall i=1,2,...,n: \\ &0,5\cdot\sigma(i)_1^{p_i} \le x_i \le l\_k - 0,5\cdot\sigma(i)_1^{p_i} \wedge \\ &0,5\cdot\sigma(i)_2^{p_i} \le y_i \le b\_k - 0,5\cdot\sigma(i)_2^{p_i} \wedge \\ &0,5\cdot\sigma(i)_3^{p_i} \le z_i \le h\_k - 0,5\cdot\sigma(i)_3^{p_i} \end{aligned} \tag{2}$$

The conditions for the mutual non-penetration of packages is as follows:

$$\begin{aligned} &\forall i,j=1,2,...,n, \quad i\ne j: \\ &\begin{pmatrix} \left|x_i-x_j\right| \ge 0,5\cdot\sigma(i)_1^{p_i} + 0,5\cdot\sigma(j)_1^{p_j} \vee \\ \left|x_i-x_j\right| \ge \left|0,5\cdot\sigma(i)_1^{p_i} - 0,5\cdot\sigma(j)_1^{p_j}\right| \end{pmatrix} \wedge \\ &\begin{pmatrix} \left|y_i-y_j\right| \ge 0,5\cdot\sigma(i)_2^{p_i} + 0,5\cdot\sigma(j)_2^{p_j} \vee \\ \left|y_i-y_j\right| \ge \left|0,5\cdot\sigma(i)_2^{p_i} - 0,5\cdot\sigma(j)_2^{p_j}\right| \end{pmatrix} \wedge \\ &\begin{pmatrix} \left|z_i-z_j\right| \ge 0,5\cdot\sigma(i)_3^{p_i} + 0,5\cdot\sigma(j)_3^{p_j} \vee \\ \left|z_i-z_j\right| \ge \left|0,5\cdot\sigma(i)_3^{p_i} - 0,5\cdot\sigma(j)_3^{p_j}\right| \end{pmatrix} \end{aligned} \tag{3}$$

The acceptable solutions due to the conditions (2) and (3) can be very difficult to find in the search space. Therefore, restrictions in the calculation algorithm were included in a soft way, in the form of the so-called penalty function.

2.3. Objective function

In the assessment of the quality of the boxes arrangement plan in the container the information about their weight and priority is important.
The weight of all boxes must be expressed in the same unit. The described model takes into account the demand for such arrangement of the cargo weight in the container, in order to avoid dangerous tilting during transport operations or storage.
The problem of arranging the boxes in accordance with the priorities means that each box is available for unloading without the need to move the boxes unloaded later on. Of course, the availability of boxes can be considered in terms of defining the location of unloading door in the container. For example, in case of a courier car, this is most often one of the smallest sides of the container. So it is essential to properly distribute the boxes along the loading space and vertically (if the load is placed in layers).
Two assessment criteria are assumed for the quality of the box arrangement plan in the container. The first one is the minimisation of the total distance of the weight centre of the whole load from two vertical planes of the container symmetry:

$$\begin{aligned} f1(\mathbf{S}) = &\left|\sum_{i=1}^{n} m_i\left(x_i - \frac{1}{2}l\_k\right)\right| + \\ &\left|\sum_{i=1}^{n} m_i\left(y_i - \frac{1}{2}b\_k\right)\right| \to \min \end{aligned} \tag{4}$$

The second criterion concerns the non-compliance of the arrangement of boxes with the order of priorities:

$$\begin{aligned} &f2(\mathbf{S}) = \\ &\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[bool(\pi_i,\pi_j)\cdot\begin{pmatrix} bool(d_i,d_j)+ \\ bool(h_i,h_j) \end{pmatrix} + ... \right. \\ &\left. ... + bool(\pi_j,\pi_i)\cdot\begin{pmatrix} bool(d_j,d_i)+ \\ bool(h_j,h_i) \end{pmatrix}\right] \to \min \end{aligned} \tag{5}$$

where:

$$\forall i = 1,2,...n : d_i = l\_k - x_i - 0,5 \cdot \sigma(i)_1^{p_i} \quad (6)$$

$$\forall i = 1,2,...n : h_i = z_i - 0,5 \cdot \sigma(i)_3^{p_i} \quad (7)$$

$$bool(a,b) = \begin{cases} 0 & \text{if } a \geq b \\ 1 & \text{if } a < b \end{cases} \quad (8)$$

Boolean functions are used in this criterion, which assume a value of 0 or 1 according to equation (9). The value d_i is the distance of the box from the container door measured along the axis X (eq. 6), h_i is the distance of the box from the container floor measured along the axis Z (eq. 7).

In order to illustrate the criteria evaluation process, calculations for the hypothetic set of three boxes were performed. Dimensions, weights, arrangement in a container and priorities of boxes are shown in Table 1.

The value of the first criterion would be calculated as follows (according to equation 4):

$$f1(\mathbf{S}) = \left| \begin{matrix} 1(750 - 2942) + 0,8(400 - 2942) + \\ 0,5(4200 - 2942) \end{matrix} \right| + \left| \begin{matrix} 1(1000 - 1165) + 0,8(350 - 1165) + \\ 0,5(1600 - 1165) \end{matrix} \right| = 4196,1 \quad (9)$$

Distances between boxes and container are (according to equations 6 and 7):

$$\begin{aligned} d_1 &= 5884 - 750 - 0,5 \cdot 200 = 5034 \\ d_2 &= 5884 - 400 - 0,5 \cdot 400 = 5284 \\ d_3 &= 5884 - 4200 - 0,5 \cdot 300 = 1534 \end{aligned} \quad (10)$$

$$\begin{aligned} h_1 &= 1000 - 0,5 \cdot 600 = 700 \\ h_2 &= 500 - 0,5 \cdot 300 = 350 \\ h_3 &= 270 - 0,5 \cdot 100 = 220 \end{aligned} \quad (11)$$

Considering determined distances and assumed priorities of boxes we determine the second criterion:

$$\begin{aligned} f2(\mathbf{S}) = &\left(0 \cdot (1+0) + 1 \cdot (0+1)\right) + \\ &\left(0 \cdot (0+0) + 1 \cdot (1+1)\right) + \\ &\left(0 \cdot (0+0) + 1 \cdot (1+1)\right) = 4 \end{aligned} \quad (12)$$

Both criteria functions can be standardises so that their values are contained in the range [0,1]. The function $f1$ assumes the greatest value when the centres of all boxes are in the same corner of the container. The function $f2$ assumes the highest value, when all boxes are arranged in the container in reverse order to the set priorities of unloading. Finally, the following criteria functions are adopted in the standardised forms:

$$f1(\mathbf{S}) = \frac{\left| \sum_{i=1}^{n} m_i \left(x_i - \frac{1}{2} l\_k \right) \right| + \left| \sum_{i=1}^{n} m_i \left(y_i - \frac{1}{2} b\_k \right) \right|}{n \left(\frac{l\_k}{2} - 1 \right) \sum_{i=1}^{n} m_i + n \left(\frac{b\_k}{2} - 1 \right) \sum_{i=1}^{n} m_i} \rightarrow \min \quad (13)$$

$$f2(\mathbf{S}) = \frac{\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left[\begin{matrix} bool(\pi_i . \pi_j) \cdot \begin{pmatrix} bool(d_i, d_j) + \\ bool(h_i, h_j) \end{pmatrix} + \\ bool(\pi_j, \pi_i) \cdot \begin{pmatrix} bool(d_j, d_i) + \\ bool(h_j, h_i) \end{pmatrix} \end{matrix} \right]}{n^2 - n} \rightarrow \min \quad (14)$$

Table 1. Exemplary positions of 3 packages in a container

Box no. i	Centre coordinates			Box weight	Box dimensions			Priority
	x_i	y_i	z_i	m_i	$\sigma(i)_1^{p_i}$	$\sigma(i)_2^{p_i}$	$\sigma(i)_3^{p_i}$	π_i
1	750	1000	1000	1	200	500	600	3
2	400	350	500	0,8	400	100	300	2
3	4200	1600	270	0,5	300	500	100	1

Thus the normalized form of the exemplary calculations presented in equations (9) and (12) looks as follows:

$$f1(\mathbf{S}) = \frac{4196{,}1}{3(2942-1)(1+0{,}8+0{,}5)+3(1165-1)(1+0{,}8+0{,}5)} \approx 0{,}15 \qquad (15)$$

$$f2(\mathbf{S}) = \frac{4}{3^2-3} \approx 0{,}67 \qquad (16)$$

The proposed model is two-criteria hence the final objective function may take the form of the weighted average:

$$f(\mathbf{S}) = w \cdot f1(\mathbf{S}) + (1-w) \cdot f2(\mathbf{S}), \quad w \in [0{,}1] \quad (17)$$

Worth noting is the fact that using the parameter value w it is possible to adapt the model to the individual needs of the user. In extreme cases, the model is reduced to one-criterion.

3. Method of quazi-optimalization

3.1. Coding and fitness evaluation of individual

The state of container loading is represented in the evolutionary algorithm by an individual presented as the dimension matrix $n\times6$. Similarly, to matrix **S** (see (1)), the rows of the individual's matrix correspond the next boxes, and the first three columns are coordinates of the location of the box centre in the coordinate XYZ system (see Fig. 2). In order to accelerate the calculations, the coding of the individual uses the developed form of permutation of boxes sizes. Consequently, instead of the fourth column of matrix S, it was necessary to use three columns with the box dimensions in the order corresponding one of the six possible permutations.
The assessment of the individual's adaptation uses:

- two objective functions described in section 2.3 (equations (13) and (14)),
- penalty function, which counts the total amount of boxes which penetrating each other and protruding outside the container. In the case of very small values taken by the penalty function, it is assumed that the individual is a weak permissible solution,
- penalty functions for the height, on which the boxes are located. It allows to decrease the effect of boxes floating in the space without the support.

Evaluation of the overall adaptation of the individual is to calculate the weighted average values adopted by the objective and penalty functions.

3.2. Genetic operators

The population of individuals, after determining the adaptation, is subjected to the selection with by the roulette method. The number of pairs of random parents is equal to the population decreased by the number of elite individuals. It is allowed to choose the same individual as parents twice, for one child.
Each pair of parents is subject to crossing. The operator of crossing is to select a random natural number $m\in[1, n-1]$. The child formed in this process is the matrix $n\times6$, in which the rows $[1, m]$ are copied from the first parent, and the others come from the second parent. Such a process of crossing enables to obtain different descendants, even with the repeated selection of the same parents.
Each descendent is subjected to mutation. Two types of mutation are allowed: a shift and/or rotation of the box. Checking whether mutations took place occurs twice for each box, that's why the individual can mutate maximally $2n$ times. The process of mutation is controlled by two parameters (probabilities).
The shift of the box means the change of location of its geometric centre. The rotation of the box in the algorithm was performed by the permutation of the order of box dimensions. The occurrence of the mutation is controlled by the mutation probability index.

3.3. Parameters of algorithm

The input data for the algorithm are:

- parameters of boxes, which are the subject of deployment process – dimensions, weight and unloading priorities – presented in the form of a matrix with dimensions $n\times5$,
- dimensions of the container and the fixed location of unloading door,
- weight of criterion and penalty functions,
- evolution process parameters: population size, mutation probability, number of elite individuals, maximum number of generations.

The selection of criteria weights and evolution process parameters was conducted during the test calculations for the fixed sets of boxes and container dimensions. The basic goal was to obtain the permissible solutions, for which the penalty function assumes the zero value. It was observed that the evolution in the developed algorithm occurs properly, that is seeking to minimise the criteria.

4. Exemplary calculation

4.1. Problem definition

The given set of 28 boxes should be packed in the space of the twenty-feet container. Each box is described with five characteristics: length, width, height, weight and priority. The cargo must be provided to four unloading points, including the minimisation of the handling operations in each point. It is assumed that the total weight of boxes is contained in standards allowed for containers. It is assumed that the boxes are perfectly rigid and can be piled in stacks at any height, as long as the stack height does not exceed the container height. The details of each box are presented in Table 2.

Table 2. Dimensions of the analysed boxes

Box no.	Length	Width	Height	Weight	Priority
1,2,3,4	600	1000	1600	1	1
5,6	600	600	600	0,5	1
7,8,9,10	600	1000	1600	1	2
11,12,13,14	600	600	600	0,5	2
15,16,17,18	600	1000	1600	1	3
19,20,21,22	600	600	600	0,5	3
23,24	600	1000	1600	1	4
25,26,27,28	600	600	600	0,5	4

4.2. Quasi-optimization process and results

The algorithm parameters for calculations were selected based on the preliminary research. They adopted the following values:

- number of generations: 20000,
- number of individuals: 40,
- number of elite individuals: 2,
- selection parameter: 1,
- mutation probability: 0,05,
- penalty weight for penetration of the boxes: 1,2,
- penalty weight for weight deviation: 0,01,
- penalty weight for mixing the priorities: 0,1,
- penalty weight of gravity: 0,03.

Calculations were made on a computer with the following parameters:

- operating System: Windows 10 Home,
- system type: 64 bits,
- processor: Intel (R) Core (TM) i5-6300HQ CPU 2.30GHz,
- RAM: 8GB.

The evolution process was presented in Fig. 3.

In Fig. 3 the blue chart is the presentation of the average adaptation of the populations. The red colour was used for marking the course of change of the adaptation of the best individual in populations. The total calculation time lasted 780 seconds. At this time we managed to find a solution considered weak permissible solution. The values, which were adopted by individual penalties, and the function of adaptation were as follows:

- penalty for penetration of the boxes = 0,0006,
- penalty for the weight deviation = 0,
- penalty of mixing priorities = 0,0159,
- gravity penalty = 0,1072,
- adaptation of the individual = 0,0055.

The coordinates of location of centres of individual boxes for the best found solutions are presented in Table 3.

Visualisation of the obtained results is shown in Fig. 4 . On the graphics the blue colour was used to mark the packages with 1 priority, red for 2 priority, yellow for packages with 3 priority, while green – priority 4.

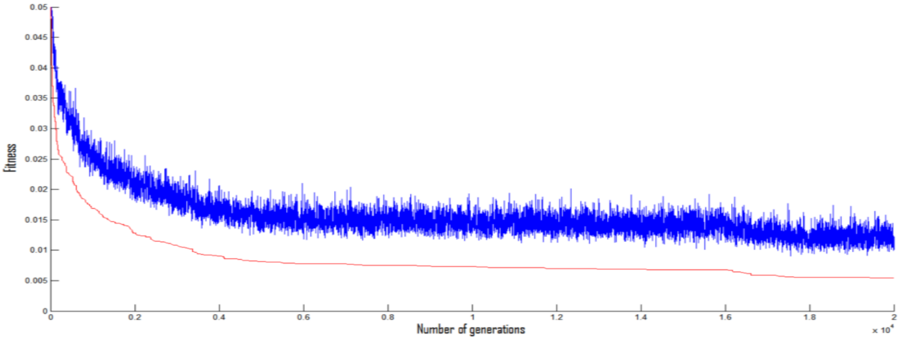

Fig. 3. The evolution process of the population and of the best adapted individual

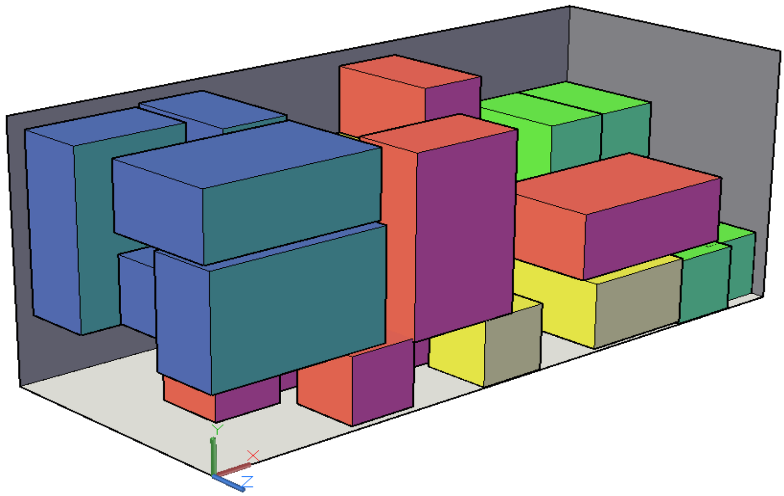

Fig. 4. Visualisation of the obtained results

Table 3. The best found solution

Box no.	Centre coordinates			Dimensions			Box no.	Centre coordinates			Dimensions		
	X	Y	Z	L	B	H		X	Y	Z	L	B	H
1	876	661	1956	1600	1000	600	15	2233	1738	808	600	1000	1600
2	600	1954	1440	1000	600	1600	16	2867	1787	810	600	1000	1600
3	1487	1823	1445	600	1000	1600	17	4317	802	328	1000	1600	600
4	857	380	1134	1600	600	1000	18	3536	500	806	600	1000	1600
5	747	1335	956	600	600	600	19	3585	2012	310	600	600	600
6	1553	985	950	600	600	600	20	2903	302	305	600	600	600
7	2403	934	829	1000	600	1600	21	3496	1389	314	600	600	600
8	3486	1807	1425	600	1000	1600	22	4950	917	315	600	600	600
9	2380	305	1405	1000	600	1600	23	5561	1826	801	600	1000	1600
10	4601	633	934	1600	1000	600	24	4950	1827	801	600	1000	1600
11	1569	1939	329	600	600	600	25	5117	300	305	600	600	600
12	1590	324	330	600	600	600	26	5578	420	301	600	600	600
13	804	908	398	600	600	600	27	5570	1022	300	600	600	600
14	1588	1202	335	600	600	600	28	4197	1921	302	600	600	600

4.3. Discussion

The obtained results allow to conclude that the described evolutionary algorithm copes well with the problems of placing the set of packages in the container. A small value of the penalty for the mutual penetration of packages in practice is negligible, especially when the boxes are not perfectly rigid and allow the squeeze of even a few millimetres.

Despite using the additional penalty, which simulates the effect of gravity, some boxes still have the tendency to float without support. This is the result of the competitive criterion of setting packages according to the priority. This problem can be eliminated by controlling the criteria weights and penalties in the assessment of the adaptation of individuals during the evolution process.

In addition to free floating boxes, there are situations in which the existing support is too small to allow the package to maintain equilibrium in the position proposed by the algorithm. In such cases it is recommended to use additional fixings and supports for particular boxes. In the future it is planned to develop a method with a criterion that minimizes the number of only partially supported boxes. Free floating boxes are less of a problem because they can be brought down to the nearest support surface. This solution is to be implemented with the development of the algorithm.

In the analyzed example all boxes were placed within the container. As the number of packages grows, the complexity of the computational complexity increases, and it is not guaranteed that the algorithm will find a solution where all the boxes fit in the container. It is assumed that if the algorithm at a fixed, acceptable time fails to find a solution with zero value of the penalty function for the penetration of packages, then it means that the set of boxes is too large. This does not mean that packages can not fit into the container at all, but that the time it takes to find such a solution exceeds the permissible practical framework. In this situation, the best solution is to divide the packages into smaller sets.

The algorithm has a strong mechanism for minimising the deviation of the loaded container weight, relative to the empty one, while maintaining the order of priority. As is typical for the evolutionary algorithms, the adaptations of the next populations are characterised by a loud noise. This is caused by the random nature of the operators of selection, crossing and mutation. The whole evolutionary process, however, goes in the right direction, gradually obtaining better values of adaptations of the whole populations. The adaptation values adopted by the best individuals in the given population achieved the best values about 17000 generations. The further calculations improved the adaptation of the whole generations, however, better individuals were not achieved in them. The computation time for a given case falls within the acceptable practice frames, however, for more complex examples it is recommended to use equipment with stronger computing parameters.

5. Summary

The problem of packing a rectangular three-dimensional space is a very important issue in the field of transport. It concerns most of the loading units, like: containers, boxes, parcels and trailers. In the proposed algorithm the problem was recognised in an innovative manner, taking into account the new criteria. In contrast to the traditional approach represented by the evolutionary algorithms, the proposed way for conducting calculations is not limited to only finding acceptable solutions. This provides him a greater freedom of action, and the cited computational example shows that this allows to obtain results satisfactory in the practical context. Targeting calculations towards the acceptable solutions was obtained thanks to the penalty function described in chapter 2.

The performance of the algorithm can be considered in the context of two indicators:

1) time necessary to obtain the acceptable solution,
2) quality of the obtained solution.

Shortening the computation time can be obtained by the code optimisation and usage of another programming language. Matlab is an environment for creating computational programs, but for that reason it lacks many of the options that exist in languages such as C or C++. For example, it is not possible to easily declare a variable type to control the size it occupies in memory or the lack of pointers that speed up operations performed on arrays.

While the improvement of the generated solutions can be achieved by extending the code with additional restrictions.

Another improvement of the algorithm which can be introduced would be the change of the way of scaling the criteria and penalties. Bringing them to the same order of magnitude would allow a better control of the penalties weighing index values, enabling the selection of optimal parameters for the operation of the algorithm. The program, in which the algorithm was implemented, does not have the interface appropriate for the user. Before implementing it to the practical uses, the appropriate layout and functional models should be developed, which would allow a simple operation.

The problem considered in the paper can be extended in many aspects. In order to adapt it to the individual cases, we should take care of the possibility to expand the set of criteria, which guide the algorithm. In practical issues also the possibility of conducting calculations for the boxes and containers other than rectangular would be useful. The possibility for defining more than one space which would be packed with a given set of parcels would also be helpful.

The feature of the algorithm, which is worth highlighting, is its versatility in transport issues. The program has a high potential for connecting it with different problems from the field of transport thanks to treating dimensions of packing space as a variable, and the ability to transform it into a single-criterion problem. It is well suited for analysing cases in the field of operational research. Taking into account the priorities of the parcels, already at the stage of loading it allows to minimise the number of handling operations during transport. For example, in the case of the extensive travelling salesman problem, the use of the proposed algorithm would allow to save a lot of time in individual unloading points. In conclusion, the algorithm in its current form is a strong base for its further improvement. It is designed for the use in practical problems, and thanks to its flexibility, it can be used in a wide range of transport issues.

References

[1] BOŻEJKO, W., KACPRZAK, Ł., WODECKI, M., 2014. Równoległe algorytmy populacyjne dla trójwymiarowego problemu pakowania. Konosala R. (eds.) *Innowacje w Zarządzaniu i Inżynierii Produkcji*. Opole: Oficyna Wydawnicza Polskiego Towarzystwa Zarządzania Produkcją, pp. 494-506.

[2] GEORGE, J. A., ROBINSON, D. F., 1980. A heuristic for packing boxes into a container. *Computer and Operations Research*, 7(3), pp. 147-156.

[3] GONÇALVES, J. F., RESENDE, M. G. C., 2013. A biased random-key genetic algorithm for a 2D and 3D bin packing problem. *International Journal of Production Economics*, 145, pp. 500–510.

[4] GÜRBÜZ, M. Z., AKYOKUŞ, S., EMIROĞLU, I., GÜRAN, A., 2009. An Efficient Algorithm for 3D Rectangular Box Packing. *Applied Automatic Systems: Proceedings of Selected AAS 2009 Papers*, Ohrid 26-29.09.2009, pp. 131-134.

[5] KACPRZAK, Ł., RUDY, J., ŻELAZNY, D., 2014. Wielokryterialny trójwymiarowy problem pakowania. *Gospodarka Materiałowa i Logistyka*, 11, pp. 40-44.

[6] LODI, A., MARTELLO, S., VIGO, D., 2002. Heuristic algorithms for the three-dimensional bin packing problem. *European Journal of Operational Research*, 141(2), pp. 410-420.

[7] SUAREZ, J. G., ANTICONA, M. T., 2010. Two GRASP Metaheuristic for the Capacitated Vehicle Routing Problem Considering Split Delivery and Solving the Three Dimensional Bin Packing Problem. *Advances in Information Sciences and Service Sciences*, 2(2), pp. 42–50.

[8] MAAROUF, W. F., BARBAR, A. M., OWAYJAN, M. J.,2008. A New Heuristic Algorithm for the 3D Bin Packing Problem. *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, pp. 342-345.

[9] WU, Y., LI, W., GOH, M., SOUZA, R., 2010. Three-dimensional bin packing problem with variable bin height. *European Journal of Operational Research*, 202(2), pp. 347-355.

[10] YAP, W. Y., LAM, J. S. L., 2013. 80 million-twenty-foot equivalent-unit container port? Sustainability issues in port and coastal development. *Ocean & Coastal Management*, 71, pp. 13-25.