The Swarm Hyperheuristic Approach for Solving Cargo Loading Problem

I. M. Latypov*

*Ufa State Aviation Technical University, latypov.ilshat@gmail.com

Abstract

Due to increasing goods turnover it is important to optimize the placement of the cargo into the containers of the vehicles for their further transportation. In this paper the problem of the placement of the paralellepipedal boxes into the containers taking into account technological restrictions is considered. The algorithm, based on hyperheuristic, is proposed.

1 Introduction

Three-dimensional Bin Packing problem is NP-hard combinatorial optimization problem. Exact solutions of such problems may be reached only by exhaustive search algorithms, which require unacceptably much time.

At this moment there are many heuristic algorithms for solving three-dimensional bin packing problems developed.

Paper [1] describes practical spheres, where the results of solving of packing problems may be used. Article [2] compares the results of different heuristics, using for solving bin packing problems.

Work [3] proposes the solution of parallelepiped packing problem by the algorithm, based on dynamic approach.

Works [4] and [5] make the survey of heuristics for 3D bin packing problems.

The authors of the article [6] propone the basics of the hyperheuristic approach for solving two-dimensional packing problems. This approach based on consistent usage of some simple heuristics. Hyperheuristics compared with other heuristics, shown its efficiency.

Work [7] investigates the schedule personnel problem, which solved by hyperheuristics.

Article [8] describes the method of using hyperheuristics with Genetic Algorithms for solving Class Timetabling problem. Similar approach (hyperheuristics + GA) used in work [9] for 2D Cutting Stock Problem.

2 Problem description

Three-dimensional packing problem is widespread problem in the literature. But in practical tasks some additional options must be added to the general statement of this problem. It is important to take into account such parameters as weight and load capacity of the objects. Also in practical tasks of the placement pallets are used. The placement must be rigid to avoid damage of the boxes during the transportation.

Thus there are addintional conditions of the problem:

1. The boxes must be placed on pallets and the pallets into the containers;
2. Total weight of the boxes, placed on one pallet must not exceed the capacity of that pallet;
3. Total weight of loaded pallets must not exceed the capacity of the container.

The goal is to minimize the number of containers used.

3 Solving method

For solving this problem the hyperheuristic-based algorithm is proposed. The idea of hyperheuristic algorithms is based on combining of the results of simple and widely used algorithms [6]. Hyperheuristic manages the sequence, which consists of many low-level heuristics, each of the applied to the problem in specific order. Trying to obtain the better result, high-level algorithm changes this sequence by adding new heuristic, removing or replacing it. By means of hyper-heuristics its possible to implicitly indicate the algorithm's operating time. During this time the mechanism of getting the result will be performed. Obviously, the greater this time, the more variants can be checked by the algorithm and the higher probability of getting the better result.

The Bees Algorithm is a population-based search algorithm. It mimics the food foraging behavior of swarm of honey bees. It can be used for both combinatorial optimization and functional optimization [10], [11]. Work [12] contains the description of this algorithm and comparison with Genetic Algorithm and Ant Colony Optimization algorithm.

As the union of these two algorithms, the Swarm Hyperheuristic algorithm was developed. It is the modification of Hyperheuristics, where the search engine is based on the Bees Algorithm. The Bees Algorithm on each step tries to find new solutions and to modify the best already found solutions.

The structure of the proposed algoirhtm is presented below:

1. To build the initial algoirthm X.
2. To assume $L = 0$, $L_{best} = 0$, $L_{good} = 0$.
3. Repeat the following steps until the operating time is over:
    (a) Algorithms generation (adding to L)
    (b) Adding the algorithms from L_{best} and L_{good} to L.
    (c) Lists cleaning: $L_{best} = 0$, $L_{good} = 0$.
    (d) Forming S_{best} set (adding items to L_{best}.
    (e) Modifiyng the algorithms from S_{best} (adding the obtained algorithms to L_{best}).
    (f) Forming S_{good} set (adding items to L_{good}.
    (g) Modifiyng the algorithms from S_{good} (adding the obtained algorithms to L_{good}).
    (h) If $F(X') < F(X)$ then $X = X'$, where X' is the best algorithm from S_{best}.
4. To output the best build algorithm X and the objective function result $F(X)$.

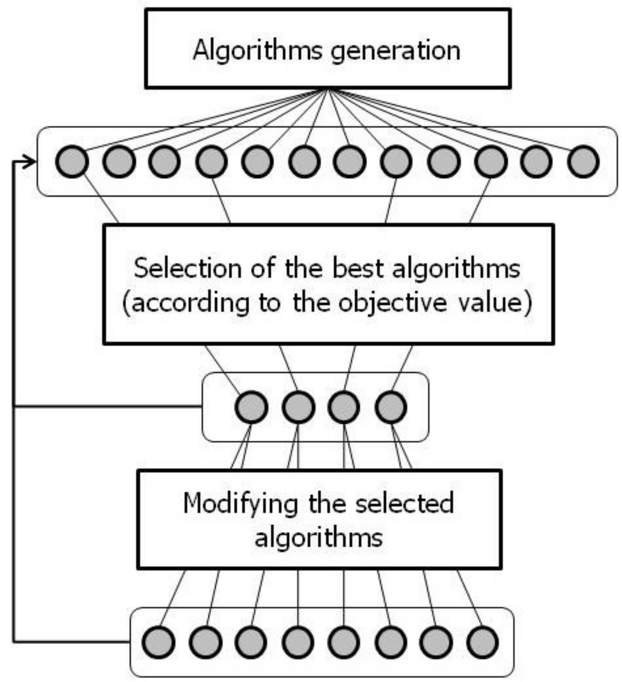

Figure 1: Scheme of the Swarm Heuristic algorithm

The general scheme of the algorithm SH is shown on the figure 1.

4 Tests

For experiments this algorithm was tested on simple three-dimensional strip packing problem without technological restrictions. The results of that experiment are shown on the table 1.

Table 1: Swarm Hyperheuristic results

N	Data sets	Packing coefficient	Occupied height	Gap
1	SP-L-08-1	89.78	724	11.38
2	SP-L-08-2	92.46	703	8.15
3	SP-L-10-1	92.59	702	8.00
4	SP-L-10-2	94.07	691	6.31
5	SP-L-12-1	91.04	714	9.85
6	SP-L-12-2	91.81	708	8.92
7	SP-L-16-1	90.03	722	11.08
8	SP-L-16-2	91.94	707	8.77
9	SP-L-20-1	91.42	711	9.38
10	SP-L-20-2	92.20	705	8.46
11	SP-L-30-1	93.53	695	6.92
12	SP-L-30-2	93.26	697	7.23
13	SP-L-40-1	91.16	713	9.69
14	SP-L-40-2	90.15	721	10.92
15	SP-L-50-1	94.61	687	5.69
16	SP-L-50-2	94.07	691	6.31

Table 2: Practical example - boxes list

Length	Width	Height	Weight	Amount
50	40	40	30	120
30	30	50	20	105
40	20	30	20	109

Test results indicate that algorithm is quite effective and it's results are comparable with other different known algorithms.

5 Application results

As the next step the developed algorithm was modified for solving the original problem. Based on this algorithm the application was developed.

As the example of application results input data sets are presented on tables 2, 3, 4. Output of the application is 3D model of the object's placement, which is shown on pictures 2 and 3.

According to this results to load all of the boxes we need 38 pallets and 2 containers.

Table 3: Practical example - pallets list

L	W	H	Weight	Max height	Capacity
50	40	40	30	120	300

Table 4: Practical example - containers list

Length	Width	Height	Capacity
580	230	180	4000

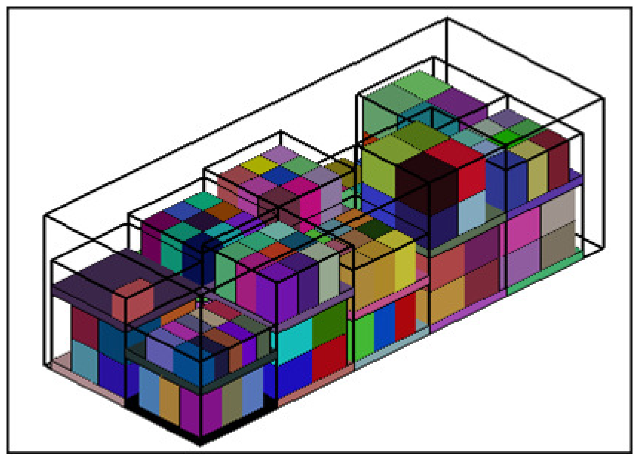

Figure 2: Application output (part 1)

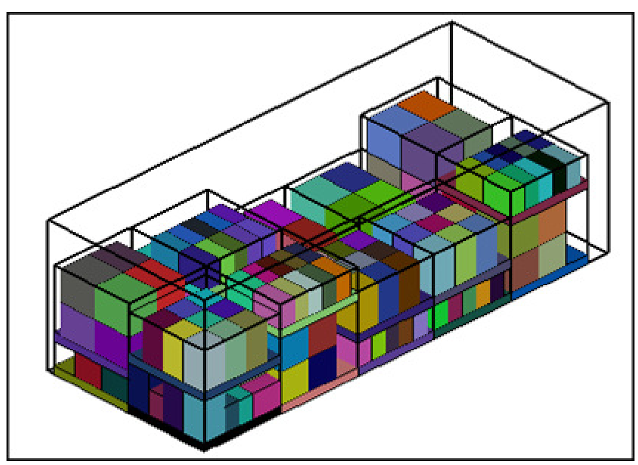

Figure 3: Application output (part 2)

Conclusion

The results achieved with the proposed algorithm are quite good, but there are a some ideas on ways to improve the results of the algorithm. One of them is parallelization of the algorithm, which will allow to check more solutions during the operating time.

The developed application allows to get the placement, which can be used in practical problems, due to taking into account technological restrictions.

Currently work on getting good lower bounds for three-dimensional packing problems is being conducted.

References

[1] G. Scheithauer, Y. Terno, A. Muller, R. Lindeman, *Zuschnitprobleme und ihre prakti-sche Losung.* Leipzig, 1987, p.207.

[2] E.E. Bischoff, M.D. Marriott, *A comparative evaluation of heuristics for container loading.* European Journal of Operation Research, 44(1990), p. 267-276.

[3] E.A. Mukhacheva, A.F. Valeeva, I.E. Totskov, *Methods for solving parallelepipedal packing problems based on dynamic approaches.* Informational technologies, 2001, 1, pp. 21-29.

[4] A. Lodi, S. Martello, D. Vigo, Heuristic algorithms for the three-dimensional bin packing problem. European Journal of Operational Research, 2002, N 141, pp. 410-420.

[5] D. Pisinger, Heuristics for the container loading problem. European Journal of Operational Research, 2002, N 141, pp. 382-392.

[6] I. Araya, B. Neveu, M.-C. Riff, *An Efficient Hyperheuristics for Strip-Packing Problems.*

[7] P. Cowling, K. Chakhlevitch, *Hyperheuristics for Managing a Large Collection of Low Level Heuristics to Schedule Personnel.*, Karlsruhe Institute of Technology, 2003, pp. 1214-1221.

[8] P. Ross, J.G. Marin-Blazques, Constructive Hyper-heuristics in Class Timetabling. Karlsruhe Institute of Technology, 2005, pp. 1493-1500.

[9] H. Terashima-Marin, A. Moran-Saavedra, P. Ross, *Forming Hyper-heuristics with Gas when Solving 2D-Reqular Cutting Stock Problems* Karlsruhe Institute of Technology, 2005, pp.1104-1110.

[10] D.T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, M. Zaidi, *The Bees Algorithm.* Technical note, Manufacturing Engeneering Centre, Cardiff University, UK, 2005.

[11] D. Karaboga, *An idea based on honey bee swarm for numerical optimization.* Technical report-TR06, Erciyes University, Engeneering Faculty, Computer Engineering Department, 2005.

[12] D.T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, M. Zaidi, The Bees Algorithm A Novel Tool for Complex Optimization Problems. Manufacturing Engeneering Centre, Cardiff University, UK.