

A Modified Artificial Bee Colony for Solving the Container Loading Problem

Ramadan A. Zeineldin
Associate Professor, Institute of
Statistical Studies and research
Cairo University

Alaa M. Morsy
Department of Operations Research
Cairo University

ABSTRACT

This paper presents a modified artificial bee colony for solving the container loading problem because the cost can be reduced by increasing the space utilization ratio. This problem is solved in a two phased a Modified Artificial Bee Colony Optimization (MABCO) and a Wall-building approach. In the first phase, MABCO with its probabilistic decision rule is used to construct a sequence of boxes. The boxes are then arranged into a container with the Wall-building heuristic in the second phase. The nectar information feedback of MABCO using neighborhood updating rule helped to improve the solutions. Computational experiments were conducted on benchmark data set and the results obtained from the proposed approach are shown to be comparable with other methods from the literatures.

General Terms

Artificial Bee Colony, Container Loading Problem

Keywords

Packing problem; Container loading; Bee colony; Wall-building; Meta-heuristic

1. INTRODUCTION

Container Loading Problem (CLP) is an ongoing field and has many uses in the world, especially in container distribution and transport companies [2,3]. In the area of production and distribution of products the efficient use of transportation mediums such as containers and palettes is of high economic importance. A high use of the applied transportation capacities causes considerable savings. Further effects are the reduction of the good traffic and the protection of natural resources. Computer supported packing methods can considerably contribute to the achievement of these goals. The modeling of practical problems concerning the optimal utilization of transportation devices leads to different kinds of packing problems [4]. Packing problems are a kind of combinatorial optimization problems that has many applications in the world such as steel cutting, glass, or paper container loading and pallet loading [1,8]. For organizations trying to minimize their operational costs, "inexpensive transportation" is a practical alternative. To take instead of that as an opportunity, products should be transported from one place to another, on time in an economic way. With the goal of transporting more products with low cost, less energy and time, especially in overseas logistics applications, products should be packed optimally or at least near optimally. This brings in the question of the effective use of containers [2]. CLP has been studied extensively by a lot of researchers with different objective functions and constraints. Since these problems are strong NP-hard [5], majority of the studies focuses on providing solutions from heuristics and meta-heuristic approaches. Methods to obtain a feasible solution include the use of different data structures such as trees and graphs, heuristics algorithms and Meta-heuristic

algorithms such as Genetic Algorithms (GA), Simulated Annealing (SA) and Tabu Search (TS). In this paper, a modified artificial bee colony (MABCO) with its probabilistic rule is developed to solve the CLP. The paper is organized as follows: section two is assigned to the problem definition and literature review. The artificial bee colony is described in section three. Section four is assigned to heuristic filling procedure. A modified artificial bee Colony Optimization with wall-building heuristic is described in section five

2. PROBLEM DEFINITION AND LITERATURE REVIEW

Container loading Problem (CLP) is an interesting and hard problem to solve. The three dimension Container loading problem is NP-hard [5,6,20]. The issue can be described as follows: Given a set of n items with width (w_i), depth (d_i) and height (h_i) and a single container with known dimensions (W, D, H) where $w_i \leq W, h_i \leq H$ and $d_i \leq D$, the problem is to pack items into the container without overlapping to maximize the utilization rate of the container [2,7,25]. The single container loading problem is a three dimensional packing problem in which a large rectangular box (the container) has to be filled with smaller rectangular boxes of different sizes. Fig.1 shows that CLP can be differentiated according to the mix of box types to be loaded. They vary from the completely homogeneous case, where boxes have exactly the same dimensions and orientations, to the strongly heterogeneous case [6].

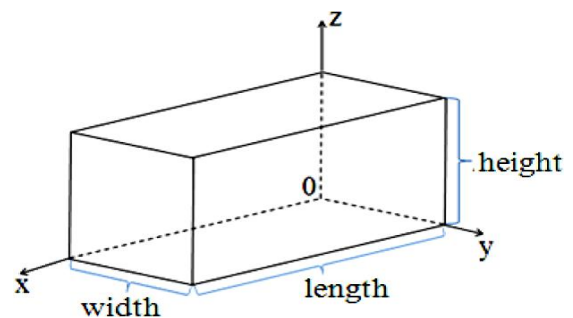


Figure 1: Container in three dimensional [1].

Seow et al. formulated problem as Container filling model (wall-building) [25]. Given a set of rectangular boxes $B = \{b_1, \dots, b_n\}$ and a rectangular container C . Let x_i, y_i and z_i represent the three dimensions of the box b_i , and X, Y and Z represent the three dimensions of the container. The objective of the problem is to select a subset $S \subseteq B$ and assign a position to each box, $b_j \in S$, in the 3D space of container C such that $\sum_{b_j \in S} x_j \times y_j \times z_j$ is maximized. The main aim of the CLP is to find the best Space Utilization Ratio (SUR) as formulated as eq.1:

$$\text{Max SUR} = \frac{\sum_{a=1}^b Va}{c} \quad (1)$$

subjected to the constraints that all boxes must be totally contained in C, no two boxes intersect in 3D space and every edge of the packed box must be parallel or orthogonal to the container walls [27]. Where: b = Quantity of packed box, V_n = Volume of the n th packed box, C = Volume of the container

The problem is solved under the following assumptions:

- Items are rectangular boxes defined with known dimensions (w_i, d_i, h_i).
- The boxes can be rotated (depending on the problem restrictions).
- Each box can be stacked on top of another.

Heuristics have been applicable alternative to find optimal or near optimal packing. Several heuristic procedures have been proposed for solving the CLP. Fanslau and Bortfeldt [14] classified approaches for the CLP according to packing heuristic and method type. Bortfeldt and Gehring [20] presented a hybrid genetic algorithm (GA) for the container loading problem with boxes of different sizes and a single container for loading. Pisinger [13] presented a heuristic algorithm for the Knapsack Container Loading Problem. Eley [17] showed a new algorithm based on the generation of homogeneous block arrangements was developed in order to solve single container problems. Morabito and Nales [19] presented a new approach to solve unconstrained two dimensional non-guillotine cutting problems, where a rectangular shaped plate must be cut into rectangular shaped pieces and all cuts are orthogonal to one side of the plate. Bischoff and Ratcliff [10] presented highlighted some important short comings in the existing theoretical literature on container loading. Bortfeldt and Gehring [4] presented a parallel tabu search algorithm for solving the container loading problem with the parallelization approach.

3. ARTIFICIAL BEE COLONY ALGORITHM

Artificial Bee Colony Algorithm (ABC), proposed by Karaboga in 2005 for real parameter optimization, is a recently introduced optimization algorithm which simulates the gathering behavior of a bee colony [11,12]. ABC classifies the foraging artificial bees into three groups, namely, employed bees, onlooker bees and scout bees. Fifty percent of the colony is made up of employed bees, and the other fifty percent includes onlooker bees. Employed bees search about the food around the food source in their memory, meanwhile they pass their food information to onlooker bees. Onlooker bees tend to select good food sources from those founded by the employed bees, and then further search the foods around the selected food source. Scout bees are translated from a few employed bees, which abandon their food sources and search new ones. ABC is an iterative process.

The units of the basic ABC can be explained as follows:

3.1 Initialization of the population

The initial population of solutions is filled with SN (population size) number of randomly generated n -dimensional real valued vectors (i.e., food sources). Let $X_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,n}\}$ represent the i th food source in the population, and then each food source is generated as follows:

$$x_{i,j} = x_{\min,j} + \text{rand}(0,1)(x_{\max,j} - x_{\min,j}), \quad (2)$$

where $i = 1, 2, \dots, SN$, $j = 1, 2, \dots, n$. $x_{\min,j}$ and $x_{\max,j}$ are the lower and upper bounds for the dimension j , respectively. These food sources are randomly assigned to SN number of employed bees and their fitnesses are evaluated [15,33].

Population in the beginning is a crucial task in evolutionary algorithms because it can affect the convergence speed and the quality of the final solution. If no information about the solution is available, then random initialization is the most commonly used method to generate candidate solutions (initial population) [12]. The chaotic maps have been used to initialize the population so that the search space information can be extracted to increase the population changes and diversity. At the same time, according to Rahnamayan [15], replace the random initialization with the opposition based population initialization can get better initial solutions and then precision convergence speed. So a novel initialization approach which employs opposition based learning method and chaotic systems to generate the beginning population. Here, sinusoidal iterator is selected and its equation is defined as follows:

$$Ch_{k+1} = \sin(\pi Ch_k), Ch_k \in (0,1), k=0,1,2,\dots \quad (3)$$

Where Ch_{k+1} chaotic system to generate the beginning population, k is the iteration counter and K is the preset maximum number of chaotic iterations. The mapped variables in eq.(3) can distribute in search space with periodicity, randomness and irregularity.

3.2 Initialization of the bee phase

At this stage, each employed bee X_i generates a new food source V_i in the neighborhood of its present position by using solution search equation as follows:

$$v_{i,j} = x_{i,j} + \emptyset_{i,j}(x_{i,j} - x_{k,j}), \quad (4)$$

where $k \in \{1, 2, \dots, SN\}$ and $j \in \{1, 2, \dots, n\}$ are randomly chosen indexes; k has to be different from i ; $\emptyset_{i,j}$ is a random number in the range $[-1, 1]$. Once V_i is obtained, it will be evaluated and compared to X_i . If the fitness of V_i is equal to or better than that of X_i , V_i will replace X_i and become a new member of the population; otherwise X_i is retained. In other words, a greedy selection mechanism is employed between the old and candidate solutions.

An onlooker bee evaluates the nectar information taken from all the employed bees and selects a food source X_i depending on its probability value p_i . Once the onlooker has selected her food source X_i , she produces a modification on X_i by using Eq.(4). As in the case of the employed bees, if the modified food source has a better or equal nectar amount than X_i , the modified food source will replace X_i and become a new member in the population [28,33].

When a food source X_i cannot be further improved through a predetermined number of trials limit, the food source is assumed to be abandoned, and the corresponding employed bee becomes a scout. The scout produces a food source randomly as follows:

$$x_{i,j} = x_{\min,j} + \text{rand}(0,1)(x_{\max,j} - x_{\min,j}), \quad (5)$$

Where $j = 1, 2, \dots, n$.

Differential Evolution (DE) algorithm a simple efficient evolutionary algorithm for many optimization problems. It

follows the general procedure of an evolutionary [16,34]. After initialization, DE enters a loop of evolutionary operations: mutation, crossover, and selection. There are several variant DE algorithms which are different in that their mutation strategies are adopted differently. The following is a mutation strategy frequently used in the literature:

$$DE/best/1: V_i = X_{best} + F (X_{r1} - X_{r2}) \quad (6)$$

Where $i = \{1, 2, \dots, SN\}$ and r_1 and r_2 are mutually different random integer indices selected from $\{1, 2, \dots, SN\}$, commonly known as scaling factor or amplification factor, is a positive real number, typically less than 1 that controls the rate at which the population evolves. The best solutions in the current population are very useful sources that can be used to improve the convergence performance. Based on the variant DE algorithm and the property of ABC, the solution search equation is devised as follows:

$$ABC/best/1: v_{i,j} = x_{best,j} + \phi_{i,j} (x_{r1,j} - x_{r2,j}) \quad (7)$$

Where the indices r_1 and r_2 are mutually exclusive integers randomly chosen from

$\{1, 2, \dots, SN\}$.

3.3 Calculating probability values involved in probabilistic selection

When all employed bees complete their searches, they start share their information related to the nectar amounts and the positions of their sources with the onlooker bees on the dance area. An onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source site with a probability related to its nectar amount. This probabilistic selection depends on the fitness values of the solutions in the population. A fitness based selection scheme might be a roulette wheel, ranking based, stochastic universal sampling, tournament selection or another selection scheme. In basic ABC, roulette wheel selection scheme in which each slice is proportional in size to the fitness value is employed as follows:

$$p_i = \frac{f_i}{\sum_{j=1}^{SN} f_j} \quad (8)$$

Where f_i is the fitness value of solution i . obviously, the higher the f_i is, the more probability that the i_{th} food source is selected.

Main steps of the artificial bee colony algorithm are summarized in the following flow chart:

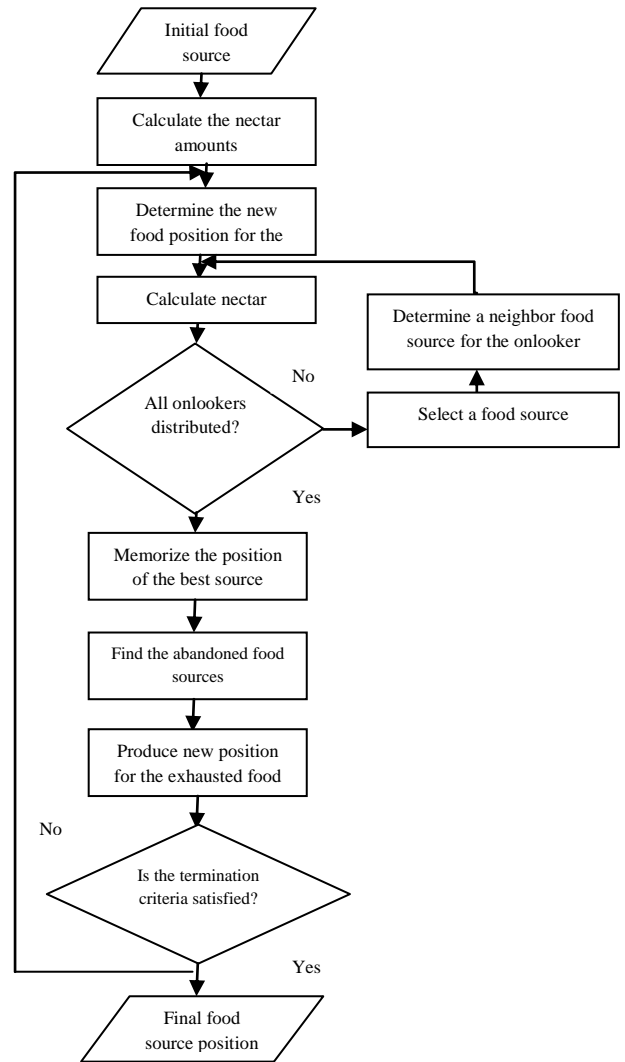


Figure 2: Flow chart of ABC algorithm [28].

4. HEURISTIC FILLING PROCEDURE

A heuristic filling procedure depended on approach called wall-building. The wall-building and layering approach, first introduced by Pham et al [21], is most commonly used and modified by later researchers for its high quality and high efficiency [22]. It should be noted that, for each meta-heuristic approach, a heuristic filling approach is needed to solve the problem. Without this approach, we cannot compute the objective function value of a solution which is volume utilization in the classical CLP.

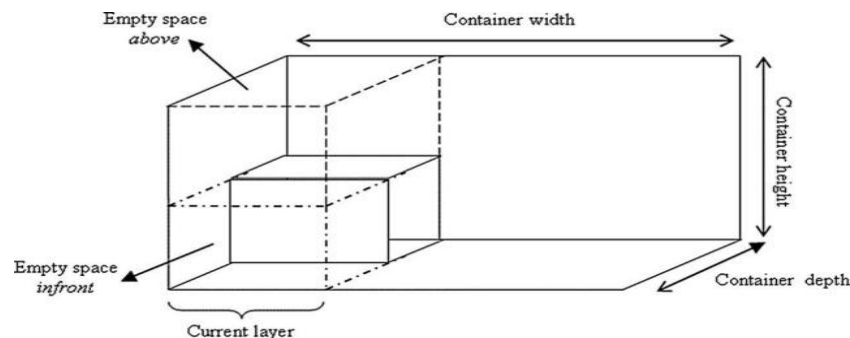


Figure 3: Empty space in front and above of the LDB [3,7].

The proposed heuristic filling procedure is a wall-building approach which loads the container layer by layer in a recursive manner. In order to fill a layer, its dimensions should be determined. This is done by choosing the width of the layer. Layers are filled one at a time. If it is not possible to fill a layer with the boxes in the set of available boxes, the current layer is closed and a new layer is started. The procedure is repeated until it is not possible to locate a new layer to the remaining container width or when the set of available boxes is empty. As a result of this packing process, the container is filled with the isolated vertical layers where spanning of the boxes between layers is avoided [7].

4.1 Determination of the layer dimensions

Before we starting filling process, we must to determine the dimensions of a layer since the width of a layer must be carefully selected to obtain a good performance [13]. The width of each layer w_L is set equal to the width of the layer determining box (LDB) [23]. In order to determine the LDB, first the boxes among the set of available boxes are sorted by width dimension in non increasing order. Thus, the box with the greatest width dimension is given the highest priority. If the boxes with the same width, the box with the smallest depth dimension d_i is given a higher priority. Finally, the highest priority box in the set of available boxes is chosen as the LDB.

Following the determination of the LDB, the layer having width w_L equal to the width w_i of the LDB, height h_L and depth d_L equal to those of the container is filled. As a result, the dimensions of the layer are determined as follows:

$$w_L = \max (w_i), \text{ where } i = 1, \dots, n$$

$$h_L = H$$

$$d_L = D$$

4.2 Filling the layer

When we are finished determination of the layer dimensions, it is possible to fill the layers. When the first box (that is LDB) is allocated into the layer, three empty spaces namely 'beside', 'in front' and 'above' of the packed box are produced. In the given situation (see Figs.3 and 4), only two of these spaces 'in front' and 'above' occurs as a result of the allocation of the first box into the layer (shown in Fig.3.). However, when a box having a width smaller than the LDB is packed into the layer, an empty space besides the packed box occurs as seen in Fig. 3. In the proposed filling procedure, the empty spaces are filled in the following order: first the empty space 'in front' of the packed box is filled, then empty spaces 'beside' and 'above' the packed box is filled respectively [3,7].

Suppose that there is a packed box in the layer as shown in Fig.3 and it is desired to pack the next highest priority box in the set of available boxes into the container. First, the space 'in front' of the packed box is checked. If it is possible to allocate this box into this space, then it is packed there and removed from the set of available boxes. Since, there is not an empty space 'beside' the packed box in the current layer the space 'above' the packed box is

checked. If it is possible to allocate this box into this space, the box is packed to this space and the packed box is removed from the set of available boxes. Otherwise, the suitability of the next highest priority box in the set of available boxes to the available empty spaces in the layer is investigated. The process is repeated in a recursive manner for each box in the set of available boxes and for all empty spaces available in current layer until it is not possible to fill empty spaces with a box in the set of available boxes.

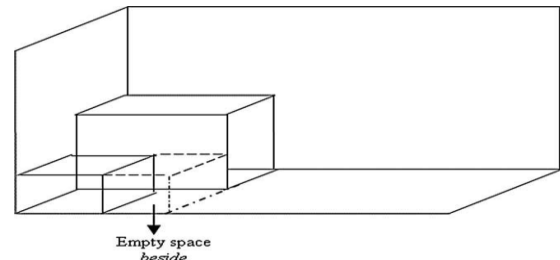


Figure 4: Empty space beside a packed box in the layer [3,7].

5. A MODIFIED ARTIFICIAL BEE COLONY OPTIMIZATION WITH WALL BUILDING (MABCO):

We can find the dimensions of a layer for a good container loading performance as discussed in section 4.1, the width of the layers is set equal to the width of the boxes, which has the highest priority in the set of available boxes. If the priority of the boxes in the set of available boxes can be altered, alternative widths for the layers can be considered to reach a good container loading performance. The priorities of the boxes (in the set of available boxes) are changed by enabling or disabling the rotation of the boxes. Each bee in the population represents a bit string of length equal to the number of the box types of the given CL problem. Each bit in this string shows an alternative orientation of a box type (there can be different rotation orientations for different problems). The MABC approach makes an analogy to a honey bee's colony that tries to find promising food sources in hive. The natural food for aging process of a honey bee colony starts with a number of scout bees from the colony searching for food sources. When scout bees find a rich food source, they begin the so called waggle dance in the hive [24]. The bee dance, which is a form of communication between the bees in the colony, includes information about the distance, direction and quality of the food source. Equipped with this important knowledge, the colony sends follower bees more follower bees are sent to more promising food sources to these food sources to collect food. While collecting it, bees evaluate the food level of the source and collect the needed information for the next waggle dance. The employed MABC mimics the food for bagging process of the honey bee colony [26]. There are several parameters should be determined. These key parameters are the number of scout bees n , the number of selected sites m , the number of elite sites e chosen from m sites, the number of bees recruited to search e elite sites nep , the number of bees recruited to search $m-e$ other sites nsp and termination criteria.

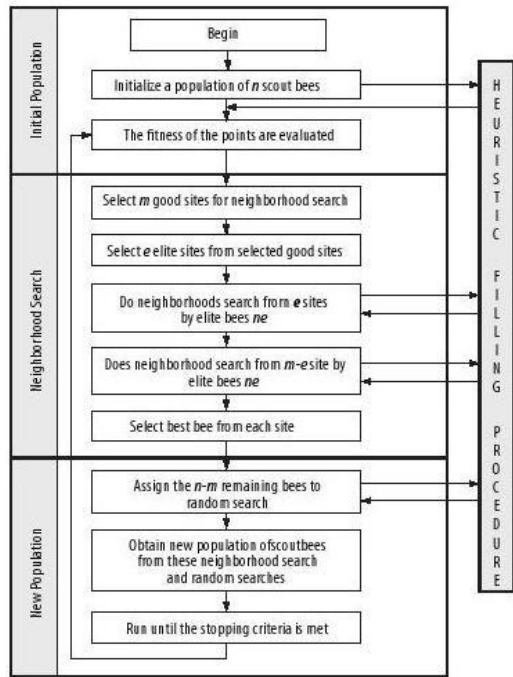


Figure 5: the approach of the MABCO

As can be seen in Fig. 5, the approach starts with n scout bees being placed randomly in the search space. These n scout bees represent the initial population. Following the acquisition of random initial solutions, the latter found by scout bees are evaluated by the proposed heuristic filling procedure. Bees that have good fitness among this initial population are selected so that m sites are chosen for neighborhood search which is primarily around the best sites among these m sites known as elite sites e and other selected sites $m-e$. Here, elite sites (e) represent more promising solutions searched with nep bees greater than nsp bees searching the other ($m-e$) sites. To make the next bee population, the bees (evaluated with the heuristic filling procedure) with the highest fitness value are selected from each m site. In order to complete population to n bees, the remaining $n-m$ bees are assigned randomly to the search space in order to find new solutions.

6. COMPUTATIONAL RESULTS

The computational experiments are conducted on benchmark data set from the literature. The data set consists of 15 problem instances from LN named LN1 to LN15 proposed by Loh and Nee [29], each with a different number of box type and container dimensions. The characteristic of the problems change from weakly to strongly heterogeneous. The proposed MABCO was coded using Microsoft Visual C#. The parameters of the Modified ABCO for test case is set to the following; number of bees (population) $n = 20$; number of selected sites $m = 4$; number of elite sites $e = 2$; number of bees send to elite points $nep = 4$ and number of bees send to other selected points $nsp = 2$ for 500 iterations.

Table 1: LN test problem results with different methods

Packing Space/volume utilization ratio (SUR) %						
Prob no	Loh and Nee [29]	Ngoi et al [30]	Bischof et al [31]	Bischof and Ratcliff [10]	Liang et al [32]	MABC proposed
LN1	78.1	62.5	62.5	62.5	62.5	62.5
LN2	76.8	80.7	89.7	90	89.7	94.9
LN3	69.5	53.4	53.4	53.4	53.4	53.4
LN4	59.2	55	55	55	55	55
LN5	85.8	77.2	77.2	77.2	77.2	77.2
LN6	88.6	88.7	89.5	83.1	91.4	90.7
LN7	78.2	81.8	83.9	78.7	84.6	84.7
LN8	67.6	59.4	59.4	59.4	59.4	59.4
LN9	84.2	61.9	61.9	61.9	61.9	61.9
LN10	70.1	67.3	67.3	67.3	67.3	67.3
LN11	63.8	62.2	62.2	62.2	62.2	62.2
LN12	76.3	78.5	76.5	78.5	78.5	78.5
LN13	77	84.1	82.3	78.1	85.6	85.6
LN14	69.1	62.8	62.8	62.8	62.8	62.8
LN15	65.6	59.6	59.5	59.5	59.5	59.5
AVG	74.2	69	69.5	68.6	70	70.37

Table 1 showed LN test problem results with different methods. The result of MABCO proposed from problem 1 (LN1) to problem15 (LN15) .

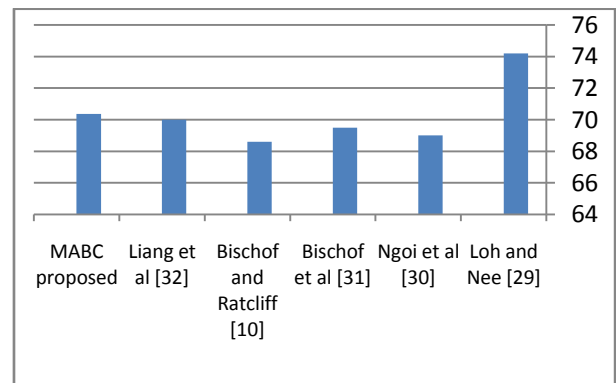


Figure 6: Results with different methods

As shown in fig.6 the average to solve 15 problems and the graphical method to describe the different between other.

7. CONCLUSION

In this, the container loading problem (CLP) solved by applying a modified artificial bee colony optimization (MABCO) with its probabilistic rule and promising food sources feedback to neighbor with the wall-building heuristic. From the computational experiments, the proposed MABCO generally yield comparable results. Therefore, the developed method has the capability to solve the CLP. Future work will focus on the exploration of the probability function and improve local neighborhood search which might further increase the space utilization ratio.

8. ACKNOWLEDGEMENTS

We would like to thank the anonymous referees for the valuable comments which improved the paper significantly. We would like to acknowledge the support provided by Institute of Statistical Studies and Research (ISSR) of University of Cairo.

9. REFERENCES

- [1] Huang, W, He, K. A caving degree approach for the single container loading problem, *European Journal of Operational Research* 196 (2009) 93–101.
- [2] Dereli, T, Das, S, G. A hybrid bee algorithm for solving container loading problems, *Applied Soft Computing* 11 (2011) 2854–2862.
- [3] Dereli, T, Das, S, G. A hybrid simulated annealing algorithm for solving multi-objective container loading problems, *Applied Artificial Intelligence* 24 (5) (2010) 463–48.
- [4] Bortfeldt, A, Gehring, H, Mack, D. A parallel tabu search algorithm for solving the container loading problem, *Parallel Computing* 29 (2003) 641–662.
- [5] Scheithauer, G. Algorithm for the container loading problem, *Operational Research Proceedings* 26 (1992) 445–52.
- [6] GonÇalves, F, J, Resende, C, G, M. A parallel multi-population biased random key genetic algorithm for a container loading problem, *Computers & Operations Research* 39 (2012) 179–190
- [7] Dereli, T, Das, S, G. Development of a decision support system for solving container loading problems, *Parallel Computing* 25(2) (2010) 138–147.
- [8] Chen, S, C, Lee, M, S, Shen, S, Q. An analytical model for the container loading problem, *European Journal of operational research* 80 (1995) 68-76.
- [9] Karaboga, D, Basturk, B. Artificial bee colony optimization algorithm for solving constrained optimization problems, *Advances in Soft Computing* (2007) 789-798.
- [10] Bischoff, E, E, Ratcliff, W, S, M. Issues in the development of approaches to Container loading, *Omega* 23 (1995) 377-39.
- [11] Karaboga, D. An idea based on honey bee swarm for numerical optimization. Technical Report-TR06, Kayseri, Turkey: Erciyes University; 2005.
- [12] Alatas, B. Chaotic bee colony algorithms for global numerical optimization, *Expert Systems with Applications* (2010); 37:5682–7.
- [13] Pisinger, D. Heuristics for the container loading problem, *European Journal of Operational Research* 141 (2002) 382–392.
- [14] Fanslau, T, Bortfeldt, A. A tree search algorithm for solving the container loading problem, *INFORMS Journal on Computing* 22(2) (2010) 222–35.
- [15] Rahnamayan, S. Opposition-based differential evolution. *IEEE Transaction on Evolutionary Computation* 2008; 12:64–79.
- [16] Storn, R, Price, K. Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* 2010; 23:689–94.
- [17] Eley, M. Solving container loading problems by block arrangement, *European Journal of Operational Research* 141(2) (2002) 393–409.
- [18] Bortfeldt, A, Gehring, H, Mack, D. A parallel tabu search algorithm for solving the container loading problem, *Parallel Computing* (2003);29(5):641–62.
- [19] Morabito, R, Arenales, M. An AND/OR-graph approach to the container loading problem, *International Transactions in Operational Research* (1994); 1(1): 59–73.
- [20] Bortfeldt, A, Gehring, H. A hybrid genetic algorithm for the container loading problem, *European Journal of Operational Research* 131(2001)143-161.
- [21] George, J, A, Robinson, D, F. A heuristic for packing boxes into a container, *Computers and Operations Research* 7 (1980) 147–156.
- [22] He, K, Huang, W. Solving the single container loading problem by a fast heuristic method, *Optimization Methods and Software* (2009) 1–15.
- [23] Gehring, H, Menschner, K, Meyer, M, A. Computer-based heuristic for packing pooled shipment containers, *European Journal of Operational Research* 44 (1990) 277–288.
- [24] Dyer, F, C. The biology of the dance language, *Annual Review of Entomology* 47 (2002): 917–949.
- [25] Seow, V, H, Majid, Z, A, Yap. Ant Colony Optimization for Container Loading Problem, *Journal of Mathematics and Statistics* 8 (2): 169-175, 2012.
- [26] Daş, G, S, Dereli, T. Container loading using hybrid bees algorithm, in *Proceedings of The EU/ME 2007 – Meta-heuristics in the Service Industry*, 8th Workshop of the EURO Working Group, EU/ME, The European Chapter on Meta-heuristics, University of Hohenheim, October 04–05 2007, Hohenheim, Stuttgart, Germany, 52–59.
- [27] Lim, A, Ying, W. A new method for the three dimensional container packing Problem, *AAAI American Association for Artificial Intelligence* 1(2001) 342-347.
- [28] Bahriye, A, Dervis, K. A modified Artificial Bee Colony algorithm for real-parameter optimization, *Information Sciences* 192 (2012) 120-142.
- [29] Loh, H, T, Nee, A, Y, C. A packing algorithm for hexahedral boxes, in: *Proceedings of the Industrial Automation Conference*, Singapore 2 (1992) 115-126.
- [30] Ngoi, B, K, Tay, A, M, L, Chua, E, S. Applying spatial representation techniques to the container packing problem. *Int. J. Prod* 32 (1994): 111-123.
- [31] Bischoff, E, E, Janetz, F, Ratcliff, M, S, W. Loading pallets with non-identical items. *Eur. J. Operat. Res* (1995): 681-692.
- [32] Liang, S, C, Lee C, Y, Huang S, W. A hybrid meta-heuristic for the container loading problem *Commun.IIMA7* (2007): 73-84.
- [33] Ali, K, Ahmad, A. Evaluating the effects of uncertainty in fuel price on transmission network expansion planning using DABC approach. *Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management Istanbul, Turkey*, July 3 – 6, (2012).
- [34] Jingqiao, Z, Arthur, C, S. A Adaptive differential evolution with optional external archive. *IEEE* (2009): 1051-8215.