KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2007
Cluj-Napoca (Romania), June 6–8, 2007, pp. 193–200

AN EVOLUTIONARY APPROACH FOR THE 3D PACKING PROBLEM

OANA MUNTEAN(1)

ABSTRACT. The three-dimensional packing is a real-world problem that arises in different industrial applications such as container ship loading, pallet loading, cargo management, warehouse management, etc. This work requires a system that efficiently place the boxes (objects of rectangular shape) so as to maximize the utilized space. This paper introduces a simple genetic algorithm, called the Genetic Packing Algorithm (GPA). The basic idea is to encode in each chromosome a permutation of the objects to be used. The algorithm is deeply described and several numerical experiments are presented in order to prove its effectiveness.

1. INTRODUCTION

The three-dimensional packing is a real-world nowadays problem that arises in different fields of industries dealing with container ship loading, pallet loading, cargo management, warehouse management. The problem may be simply stated as follows: Given a set of three-dimensional rectangular objects (parallelepipeds) it is required to arrange these in a three-dimensional package (container) in an efficient way: so as to maximize the filled space and without overlapping the objects.

The problem of packing boxes is a well-known NP-Complete [5] problem. Many algorithms have been developed on this field. However, no polynomial-time algorithm is known so far for it. Moreover, algorithmic research has provided strong evidence that it is unlikely to find a polynomial algorithm for this kind of problems. Such an algorithm is guaranteed to find an optimal solution in time that, even in the worst case, can be bounded by a polynomial in the size of the input. If no such bound can be guaranteed, the necessary time for solving instances tends to grow very fast as the instance size increases.

Evolutionary algorithms [6] are powerful optimization techniques inspired by Darwinian Theory of evolution and natural selection [4]. They have successfully been used for solving various difficult real-world problems.

2000 *Mathematics Subject Classification.* 68T20, 68W20.
Key words and phrases. 3D packing, Genetic Algorithm, Permutation encoding.

A new evolutionary approach (called Genetic Packing Algorithm (GPA)) for the three-dimensional packing problem is suggested in this paper. The individuals are encoded as permutations and are modified by using special genetic operators.

Several numerical experiments have been performed by using some randomly generated data. The relationship between the algorithm performance and various parameter settings has been analyzed. Results show that GPA is able to perform very well even when small populations are used.

The paper is organized as follows. Related work in the field of 3D packing is briefly reviewed in Section 2. Section 3 gives a short description of the 3D packing problem and the required parameters. The proposed algorithm is described in Section 4. Numerical experiments are provided in Section 5. Strengths and weaknesses of this approach are discussed in Section 6. Conclusions and future work directions are suggested in Section 7.

2. Related work

Several heuristic methods for solving this problem have been proposed. Among the most popular algorithms are wall building [8], guillotine cutting [2], cuboid arrangements [1], Tabu search [3], branch-and bound, etc. The wall building approach fills the container in a number of layers across the depth of the container. The stack building packs the boxes into suitable stacks which then are arranged at the floor of the container by solving a two-dimensional packing problem. The guillotine approach is based on a slicing tree representation of the packing. Each slicing tree corresponds to a guillotine partitioning of the container into smaller parts, where the leaf nodes correspond to boxes. The cuboid arrangement approach recursively fills the container with cuboid arrangements (arrangement of similar boxes). Cuboid arrangements will always provide a sufficient support of the boxes.

Only a few evolutionary algorithms were developed in this field. In [7] I. Ikonnen proposed a genetic algorithm for solving three-dimensional packing with convex objects having holes and cavities. This approach is strongly connected to the particular features of the objects to be placed, i.e. the dimensions and the orientation of the objects are very important for representation of the solution.

3. Problem statement

In the three-dimensional packing problem we have a set of n rectangular 3D objects (boxes) $i = 1, 2, \ldots, n$. Each box i has the dimensions denoted by w_i (the width), d_i (the depth) and h_i (the height).

Let us also denote by W, D and H the width, the depth and the height of the rectangular container that has to be filled with these boxes.

The task is to pack the boxes in such a way that the utilized volume (of the container) is maximized. The ideal case is when the whole volume is utilized.

There are some constrains:

- The boxes cannot overlap each other,
- The boxes are packed with each edge parallel to the corresponding container edge,
- The items may or not be rotated. For speed purposed, in this paper we have not allowed rotation. However, the algorithm can be easily adapted to include rotation.

4. Proposed algorithm

Genetic Packing Algorithm uses a specific representation and specific search operators. All elements of the algorithm are deeply discussed in this section.

4.1. **Individual Representation.** Each individual is a permutation specifying the order in which the objects are placed inside the box.

Example

Let us consider a container having the dimensions 30x20x30 units. It is requested to arrange the following boxes inside it:

(1) 9x10x16
(2) 5x9x12
(3) 14x11x4
(4) 22x8x16
(5) 17x7x7
(6) 3x8x9
(7) 9x5x5
(8) 7x7x8
(9) 14x8x9
(10) 4x5x17

An example of GPA chromosome is the following:

$$1, 5, 4, 8, 7, 10, 9, 3, 6, 2$$

The result of packing the boxes into the container as given by the previously described chromosome is given in Figure 1. There are 7 boxes that can be fitted there. The rest of the boxes are not taken into account in this case.

4.2. **Fitness Assignment Process.** The boxes are placed one by one in the container (according to the order given by the permutation encoded into the current chromosome). For each box we find the lowest position where it can be placed. If there are multiple possibilities we choose one of them randomly.

If a box cannot be placed it will skipped and the next box is considered.

The quality of an individual is equal to the uncovered volume (the waste) of the container. Since it is required to fill the entire space of the container we are

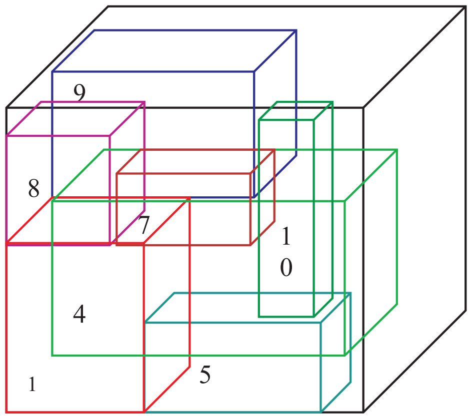

FIGURE 1. An example of packing 10 boxes into a container

dealing with a minimization type problem, which means that the best solution is given by the smallest fitness (we have to minimize it).

4.3. **Genetic Operators.** The main variation operators are recombination (crossover) and mutation.

The crossover operator proposed in this article uses one cutting point value and works as follows: a value p between 1 and n is randomly chosen and then the permutations are traversed from left to right and the values greater than p are exchanged between parents. It is necessary to mention that the interval for choosing this value is not $(1, n)$ as usual, but $(1, n/2)$. The reason for this choice is the fact that there might be some cases when not all the boxes can be used to fill the container. In this case, if the value of p is greater than the number of boxes that fitted into the container the resulting offspring from this recombination will provide the same fitness as their parents.

Mutation of a permutation was performed by swapping two randomly chosen positions.

4.4. **The Algorithm.** Genetic Packing Algorithm (GPA) uses steady state [9] as its underlying mechanism.

GPA starts by creating a random population of individuals. The following steps are repeated until a given number of generations is reached: Two parents are selected using a standard selection procedure. The parents are recombined in order to obtain two offspring. Each offspring is considered for mutation. The best offspring O replaces the worst individual W in the current population if O is better than W.

Genetic Packing Algorithm is outlined below:

Algorithm 1 Genetic Packing Algorithm

Randomly create the initial population $P(0)$
for t=1 to NumberOfGenerations **do**
  for k=1 to PopulationSize **do**
    p_1 = Select($P(t)$);{randomly select one individual from the current population}
    p_2 = Select($P(t)$); {select the second individual}
    Crossover (p_1, p_2, o_1, o_2); {crossover the parents p_1 and p_2 obtaining the offspring o_1 and o_2}
    Mutation(o_1); {mutate the offspring o_1}
    Mutation(o_2); {mutate the offspring o_2}
    if Fitness(o_1) $<$ Fitness(o_2) **then**
      if Fitness(o_1) $<$ the fitness of the worst individual in the current population **then**
        Replace the worst individual with o_1;
      else
        if Fitness(o_2) $<$ the fitness of the worst individual in the current population **then**
          Replace the worst individual with o_2;
        end if
      end if
    end if
  end for
end for

5. Numerical experiments

In this section we perform several numerical experiments in order to assess the performance of the Genetic Packing Algorithm. Two statistics are of high interest:

- The relationships between the filled volume and the population size.
- The relationships between the filled volume and the number of generations.

The general parameters of the GPA are given in Table 1.

Test data are taken from a real-world warehouse. There is a high diversity in the size of boxes. The container has 30x30x30. The width, depth and height of the boxes have the values between 5 and 15. Thirty boxes are involved.

TABLE 1. General parameters of the GP algorithm

Parameter	Value
Mutations	1 mutation / chromosome
Crossover probability	0.9
Selection	Binary Selection

5.1. **Experiment 1.** In this experiment the relationship between the filled volume and the population size is analyzed. The number of generations is set to 30. The other parameters are given in Table 1. Results are depicted in Figure 2.

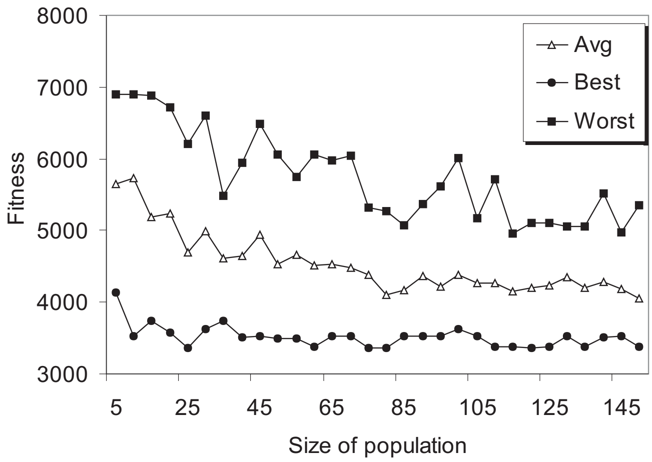

FIGURE 2. The relationship between the quality of the best chromosome and the population size. Results are averaged over 30 runs.

Figure 2 shows that the algorithm performs very well even if the population size is small (5 individuals). This means that we can obtain good results in a very short time.

5.2. **Experiment 2.** In this experiment the relationship between the filled volume and the number of generations is analyzed. The population size is set to 20 individuals. The other parameters are given in Table 1. Results are depicted in Figure 3.

Figure 3 shows that the performance of the GPA improves as the number of generations is increased.

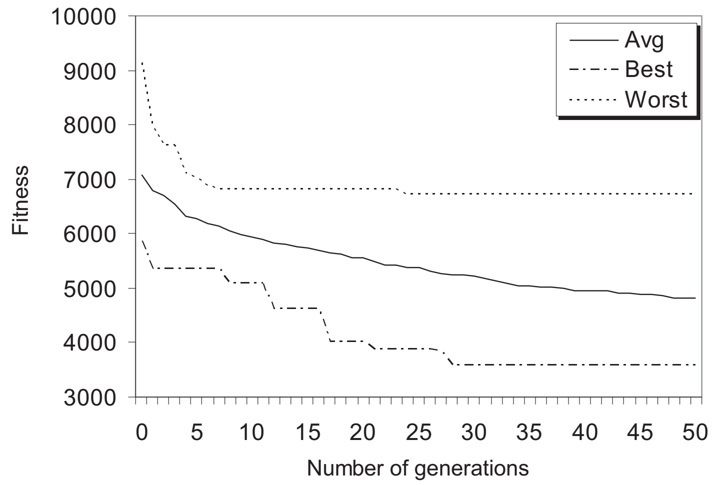

FIGURE 3. The relationship between the quality of the best chromosome and the number of generations. Results are averaged over 30 runs.

6. Strengths and weaknesses

The main reasons for using GA for 3D Packing problem are:

- Simple representation for the solution - each individual is represented as a permutation,
- Short running time - for a near optimal arrangement of 50 boxes, the solution is obtained in several seconds,
- Easy to compute fitness - each object is placed in the lowest position possible. This position is easy to be computed in GA algorithm,
- Good performance of operators - crossover and mutation used are well known and intensively studied, so the search space was efficiently explored.

However, this approach has some limitations:

- Usage of permutations for chromosome's representation can lead sometimes to some major changes in the solution. For example if a small perturbation takes place (e.g. mutation) the placement of the boxes following the changed box will radically change.
- It is needed parameters' tunning for obtaining good results. For instance, if the population is not large enough it is possible that the problem does not converge. More experiments have to be performed in order to adjust the value of GA's parameters.

7. Conclusions and further work

A new genetic approach for 3D packing problem has been proposed in this paper. Chromosomes are encoded as permutations which are varied by using special operators.

Numerical experiments have shown the ability of the algorithm to solve medium size instances of the problem by using very small populations is individuals. This means that we can obtain good results in a very short time.

Further efforts will be focused on analyzing the relationships between other parameters of the Genetic Packing Algorithm (such as mutation probability, crossover probability) and the GPA ability to find a very good solution of the problem.

Different heuristics for placement the boxes inside the container will be investigated in the near future. Their aim is to improve the quality of solutions and to speed up the algorithm.

References

[1] A. Bortfeldt, H. Gehring, and D. Mack. A parallel tabu search algorithm for solving the container loading problem. *Parallel Computing*, 29(5):641–662, 2003.
[2] L. Brunetta and P. Grégoire. A general purpose algorithm for three-dimensional packing. *INFORMS Journal on Computing*, 17(3):328–338, 2005.
[3] T. G. Crainic, G. Perboli, and R. Tadei. An interval graph-based tabu search framework for multi-dimensional packing, Nov. 04 2003.
[4] C. Darwin. *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*. John Murray 1^{st} Edition, London, 1859.
[5] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, New York, 1979.
[6] D. E. Goldberg. *Genetic algorithms in search, optimization and machine learning*. Addison Wesley, 1989.
[7] I. Ikonen, W. E. Biles, A. Kumar, J. C. Wissel, and R. K. Ragade. Genetic Algorithm for Packing Three-Dimensional Non-Convex Objects Having Cavities and Holes. In *Proceedings of the 7th International Conference on Genetic Algortithms*, pages 591–598, East Lansing, Michigan, July 1997. Morgan Kaufmann Publishers.
[8] D. Pisinger. Heuristics for the container loading problem, 2002.
[9] G. Syswerda. A study of reproduction in generational and steady state genetic algorithms. In G. J. E. Rawlins, editor, *Proceedings of Foundations of Genetic Algorithms Conference*, pages 94–101. Morgan Kaufmann, 1991.

(1) Faculty of Mathematics and Computer Science, Babeş-Bolyai University, Kogălniceanu 1, Cluj-Napoca, 400084, Romania

E-mail address: oana_muntean85@yahoo.com