Journal of Physics:
Conference Series

PAPER • OPEN ACCESS

Smart Packing Simulator for 3D Packing Problem Using Genetic Algorithm

To cite this article: U. Khairuddin *et al* 2020 *J. Phys.: Conf. Ser.* **1447** 012041

Smart Packing Simulator for 3D Packing Problem Using Genetic Algorithm

U. Khairuddin*[,1,3], N. A. Z. M. Razi[1], M. S. Z. Abidin[2] and R. Yusof[3]
[1]Department of Mechanical Precision Engineering, Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia
[2]Department of Information Systems, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 42300 Shah Alam, Selangor, Malaysia
[3]Centre for Artificial Intelligence & Robotics, Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia

E-mail: uswah.kl@utm.my

Abstract. Every year, at least 100 million tons of solid waste globally comes from packaging waste, in which partly created by inefficient packaging. Multiple box arrangement or bin packing solution directly addresses this problem which also affects storing space in production, manufacturing and logistics sector. Smart packing algorithm is designed for solving three-dimensional bin/container packing problem (3DBPP) which has numerous practical applications in various fields including container ship loading, pallet loading, plane cargo, warehouse management and parcel packing. This project investigates the implementation of genetic algorithm (GA) for a smart packing simulator in solving the 3DBPP applications. The smart packing system has an adaptable chromosome length GA for more robust implementation, where chromosome length will be changing with number of boxes. It can optimize multiple box arrangements and the boxes movements and positions are simulated through each GA generations, for realistic adaptation. The system is able to make optimum arrangement for the boxes so they can fit into a smallest container possible. The time taken for GA to converge varies with number of boxes.

1. Introduction

Box or bin packing is a big issue in logistics, transportation and manufacturing industries. In the year 2012, for the United States alone, 75.2 million tons of solid waste generated by packaging was reported by the Environment Protection Agency and most of the waste are from protective packaging, the extra materials added to fill in empty spaces inside a box in order to protect the package which valued at USD 22 billion [1]. It is imperative to optimize box packing arrangement in order to reduce packaging waste. Smart packing algorithm is designed for solving 3D bin/container packing problem such as, container ship loading, pallet loading, plane cargo and warehouse management [2], parcel packing, container and pallet loading [3]. Three-dimensional (3D) packing problem is a complex expansion of the two-dimensional (2D) and the one-dimensional (1D) bin packing problem [4] and can be found at the core of, many operational research (OR) problems [5]. A 3D packing problem seems simple to solve, however, multiple boxes and containers with various parameters such as dimensions, weight, orientation and arrangement somewhat made packing and arranging more difficult.

The 3D bin packing problem has widely been studied and many different algorithms were developed to solve the problem for example the “building growing” method [4], the “group packing approach” by [6], and the “peak filling slice push” which is practical in filling small volumes with different boxes size [7]. Generally, researchers have focused on different variants and constraints for 3DBPP. In [3], they are concerned with packing orthogonally a set of 3D rectangular-shaped items into a single bin, where the objective is to maximize the bin's utilization ratio, that is, the ratio of the sum of all packed items' volumes to the bin's volume. Kang et. Al. [8] considered three possible orientations for the boxes with three constraints; firstly, no penetration between the boxes lies in the container with the boundary surfaces of the container, secondly, no overlapping boxes, and lastly, each box is parallel to the container’s surfaces and is supported by either the floor of the container or other packed boxes. Many bin packing problems are designed using heuristics algorithm due to its NP-hard nature [9] for example, Wang et. Al. in [3] and [8] both used “hybrid genetic algorithm” for packing solution.

This paper presents a smart packing simulator for 3D packing problem using genetic algorithm (GA). The simulator will be able to simulate optimum arrangement of multiple boxes so they can fit into the smallest container possible, minimizing any empty spaces within the container. The graphical simulation will help user to adapt simulated box arrangement in reality.

2. Methodology
The smart packing simulator is designed using GA, an optimization algorithm used to find the optimal solution(s) to a given computational problem that maximizes or minimizes a particular function which was introduced by John Holland [10].

2.1. Smart packing strategy
For this system, GA with adaptable chromosome length were developed where the length of chromosome is dependent to number of boxes. User may input boxes dimensions (length, width, height) and the simulator will show the process of arranging the boxes throughout GA generations, and finally determine the optimum box arrangement. The workflow of the simulator is in figure 1.

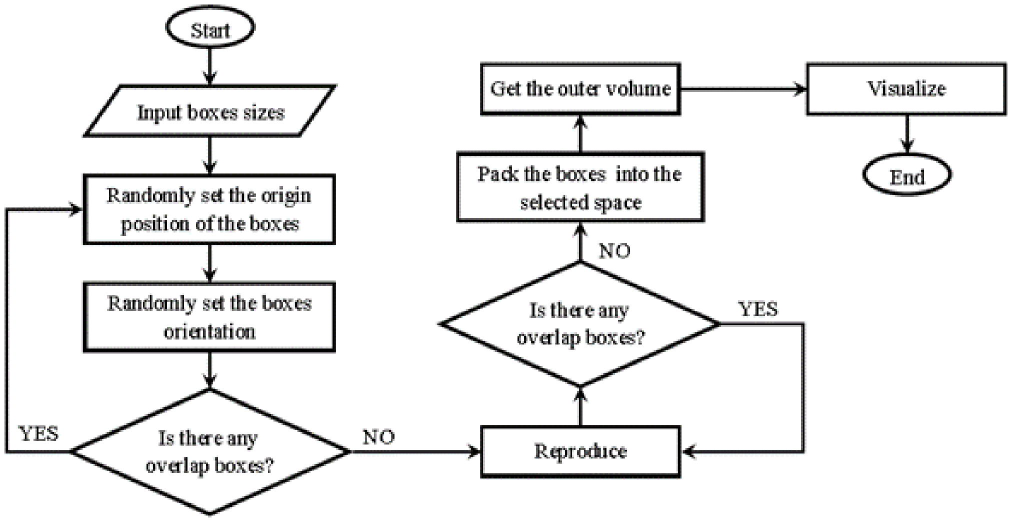

Figure 1. Flowchart of the smart packing system.

The packing strategy of this system is shown in figure 2 where initially the boxes are scattered in the volume space with random point of origin with random orientation and after every generation, the boxes will move closely to each other and finally arranged in a manner that the outer container has a minimum volume possible.

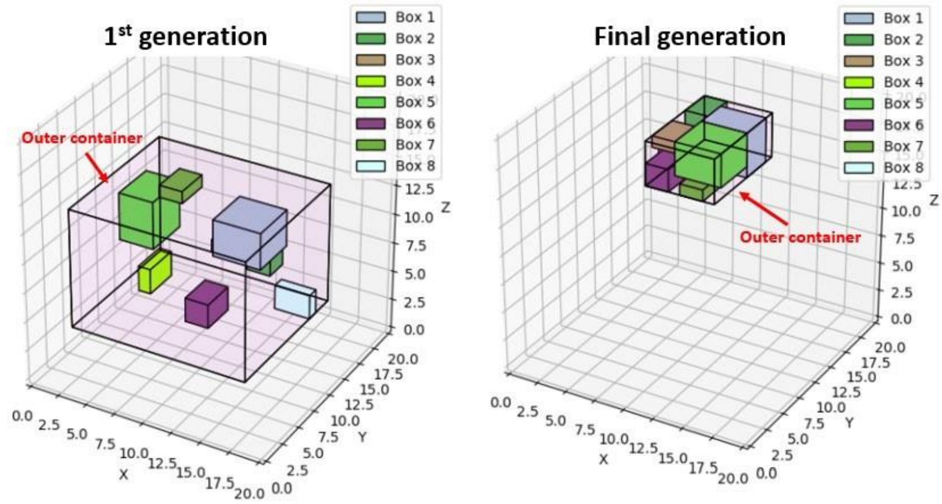

Figure 2. Packing strategy simulation

2.1.1. GA chromosomes and operations. Each box will occupy four spaces in the GA chromosome where the first space is for its initial position (x,y,z coordinates) in the volume space and the other three consecutive spaces are for its height, width and length. The GA chromosome length, $l_{chromosome} = 4n$ where n is the number of boxes. Common diploid approaches of GA such as crossover and mutation are included in the process and user may input maximum number of GA generations as stopping criteria.

2.1.2. GA crossover and mutation. The crossover operation was designed to fit the box packing problem where the crossover points must not separate any individual boxes' four-chromosome space. The crossover points are random in each generation and it will be in between any two boxes as shown in figure 3. New off-springs were built after crossover operations. The mutation operation will randomly change location and position of boxes in the chromosomes as shown in figure 4.

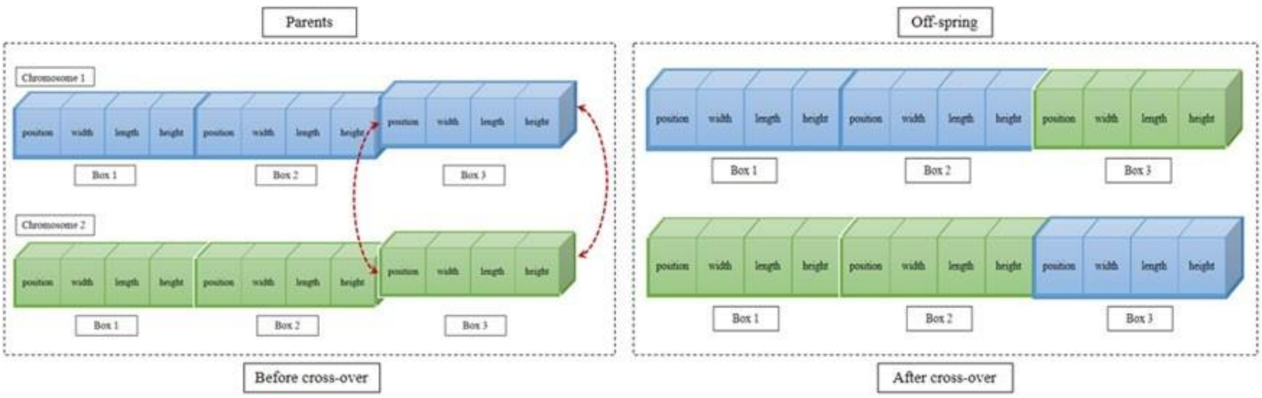

Figure 3. GA crossover for the smart packing system

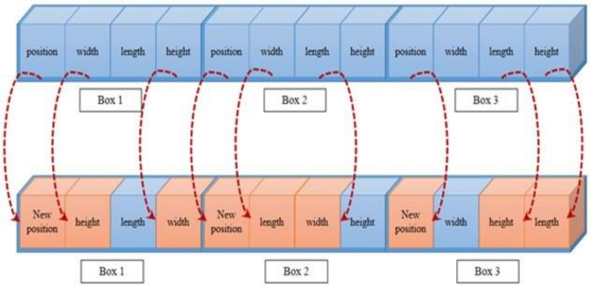

Figure 4. GA mutation for the smart packing system

2.1.3. Objective and fitness function. The objective of the smart packing system is to minimize the size of the outer container and the fitness function is set as below:

fitness = space left + distance to outer middle

where,

space left = outer container volume – total boxes volume

distance to outer middle = Σ (middle point of container – middle point of box

5% of Chromosomes with the minimum fitness value are chosen to be the elites for the next generation.

2.1.4. Constraints. The constraints for this algorithm are as follows; first, there should be no overlapping boxes. Second, the box must stick closely to the previous boxes. Third, the boxes can freely rotate (change position of height, width or length) to fit in the container. Whenever any randomly generated chromosome violated any of these criteria, a new chromosome will be generated as a replacement and will go through the same check to ensure non-violation to constraints.

3. Results and discussion

The proposed algorithm was tested using multiple number of boxes. The first case was for 4 boxes of different dimensions. The test ran for 400 generations and the minimum container size achieved was (7, 4, 3) with 12 m^3 empty space at generation number 201. The optimum arrangement is in figure 5. A plot of fitness versus GA generations is also included where the user can track fitness value as well as average fitness per generation.

The second case was for 8 boxes and the dimensions are the same as the previous case, but each box is duplicated. Three tests were carried out for this second case, the first two were for 500 GA generations, and the last one for 10,000 GA generations. The final optimum arrangements can be observed in figure 6. Information included in the figure is maximum number of generations, free space in outer container, outer container volume and dimension (size). Number of generations in which the optimum arrangement was found is also included at the bottom of the box arrangement plot.

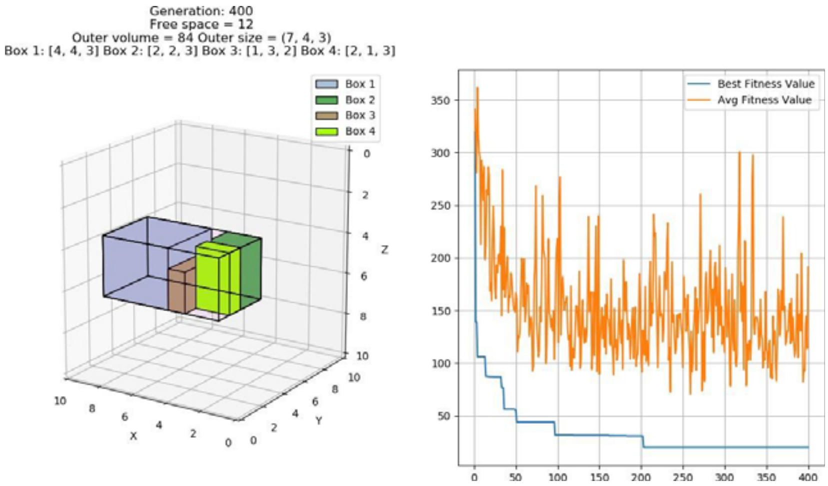

Figure 5. Optimum arrangement for 4 boxes of various dimensions (left) and a plot of fitness vs GA generations (right)

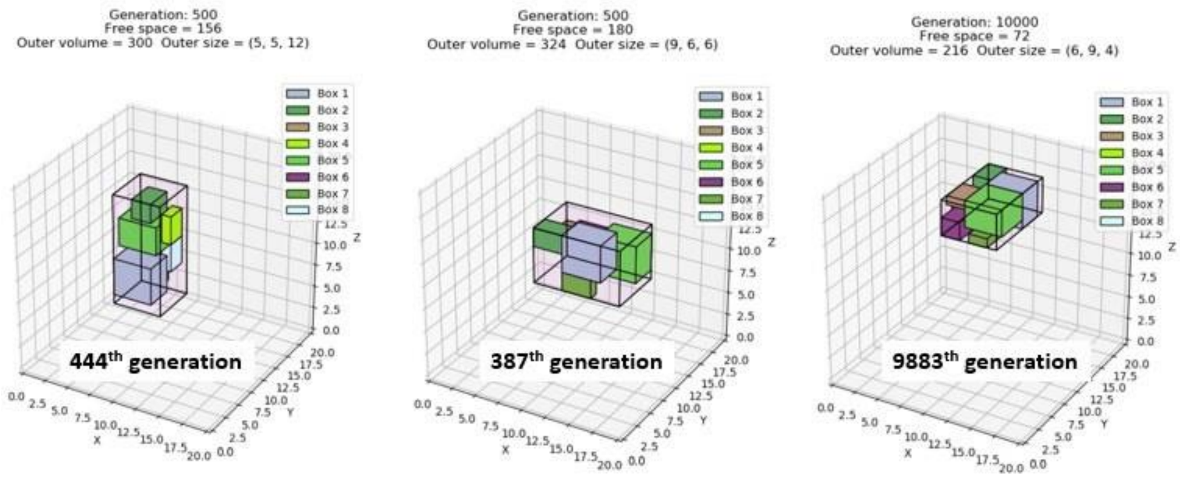

Figure 6. Optimum arrangements obtained for 8 boxes in three separate cases, 500 GA generations (left and middle) and 10,000 GA generations (right)

The first two tests are to observe whether 500 GA generations are enough to get an optimum arrangement and whether the result can be reproduced by the system. It turns out that the system cannot replicate the result because no random seed value was set before the test. When the test was run for 10,000 GA generations, the optimum arrangement is much smaller than the 500 generation tests. It shows that for big number of boxes, maximum number of generations must be high. However, more tests are needed to confirm the relationship between number of boxes and maximum number of GA generations.

The free space value depicts empty volume in the outer container. In real word application, these empty spaces will be filled with protective packaging. Smaller free space value means less protective packaging hence less solid waste will be generated.

4. Conclusion

This paper proposed a simple and reliable smart packing simulator for optimized boxes arrangements and minimization of outer container box size. The system, which is developed using GA, automate the arrangements based on natural selection process. Adaptable chromosome length made the system flexible to any number of boxes and is very robust, suitable for real life application. The final arrangement is simulated for easy referencing and adaptation.

In future work, more constraints will be added for example heavier box must be at the bottom, or fragile box must be on top. The system's model must be developed in order to know optimum GA parameters to obtain optimum box arrangements.

5. References

[1] Nuwer R and Kho J 2014 Thinking outside the box: unwrapping a massive packaging problem *The Guardian*

[2] Jaisree A D, Umagowri R and Kanna S K R 2016 Smart Evolutionary Algorithm for Constrained Container Loading Problem *International Journal of Latest Research in Engineering and Technology (IJLRET)* vol. **2**, no. 4, pp. 67–71

[3] Wang H 2010 A hybrid genetic algorithm for 3D bin packing problems *2010 IEEE Fifth Int. Conf. Bio-Inspired Comput. Theor. Appl.* pp. 703–707

[4] Lim A and Ying W 2001 A new method for the three-dimensional container packing problem *Proc. 17th Int. Jt. Conf. Artif. Intell*. Vol. **1**, pp. 342–347

[5] Nottingham T and User N E 2011 Algorithms and Data Structures for Three-Dimensional Packing

[6] Joung Y K and Noh S D 2014 Intelligent 3D packing using a grouping algorithm for automotive container engineering *J. Comput. Des. Eng.* vol. **1,** no. 2, pp. 140–151

[7] Maarouf W, Barbar A and Owayjan M 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

[8] Kang M I W H 2012 A hybrid genetic algorithm with a new packing strategy for the three-dimensional bin packing problem *Appl. Math. Comput.* vol. **219** pp. 1287–1299

[9] Loh K H, Golden B, Wasil E, and Smith R H 2008 Solving the one-dimensional bin packing problem with a weight annealing heuristic *Comput. Oper. Res.,* vol**. 35**, pp. 2283–2291

[10] Holland J H 1984 Genetic Algorithms and Adaptation. In: Selfridge O.G., Rissland E.L., Arbib M.A. (eds) Adaptive Control of Ill-Defined Systems. *NATO Conference Series (II Systems Science)*, vol **16**. Springer, Boston, MA

[11] Nuwer R and Kho J 2019 Thinking outside the box: unwrapping a massive packaging problem *The Guardian* pp. 1–7