

Hanoi University of Science

Biased random key genetic algorithm for 3D bin packing problem

Dinh Hoang Nhat Minh *

June 11, 2022



*Talented Program in Mathematics

Contents

1	Introduction	2
2	Packing Procedure	2
2.1	Chromosome representation and decoding	3
2.2	Empty maximal space selection	5
2.3	Box packing	6
2.4	State update information by Difference Process	6
2.5	Fitness function	8
2.6	Summary	8
3	Biased random key genetic algorithm	8
3.1	Architecture of the algorithm	8
3.2	Evolutionary process	8
4	Numerical experiments	10
5	Reference	11

1 Introduction

The three-dimensional bin packing problem (3D-BPP) consists in packing, with no overlapping, a set of three-dimensional rectangular shaped boxes (items) into the minimum number of three-dimensional rectangular shaped bins (containers).

All the bins have identical known dimensions (D, W, H) and each box i has dimensions (d_i, w_i, h_i) for $i = 1, \dots, n$. Without loss of generality one can assume that all input data are positive integers and that $d_i \leq D$, $w_i \leq W$, and $h_i \leq H$ for $i = 1, \dots, n$. It is assumed that the boxes can be rotated.

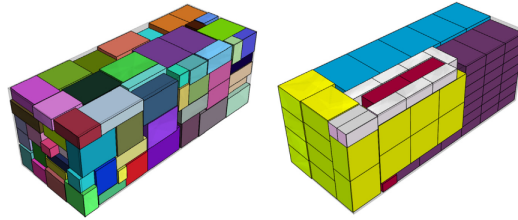


Figure 1. 3D Bin Packing Problem

Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves, Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

2 Packing Procedure

Example 2.1. We consider a 2D bin packing problem as an simple example to understand how the Biased random key genetic algorithm (BRKGA) algorithm works. Assume that we have to pack 3 boxes size $(2, 3)$, $(3, 5)$, $(1, 2)$ (blue box, red box, green box) into bins with size $(6, 6)$.

In example 2.1, we have 3 boxes to . We can decode a vector with 6 components (genes) are

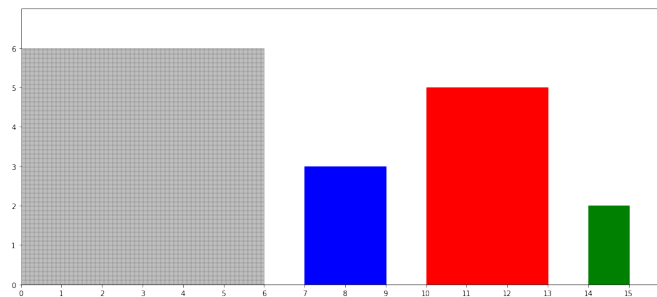


Figure 2. Example

random real numbers uniformly sampled from the interval $[0, 1]$ (which is called chromosome) to a solution to pack all the boxes into bins with size $(6, 5)$.

We consider a random chromosome $c = (0.8, 0.1, 0.6, 0.7, 0.9, 0.7)$.

The first 3 genes are used to obtain the Box Packing Sequence (*BPS*) and the genes 4 to 6 are used to obtain the Vector of Box Orientations (*VBO*).

For example, the packing order of blue box is presented by 0.8, the orientation of blue box is presented by 0.7.

Firstly, we decode the first 3 genes. Therefore, the *BPS* is (red box, green box, blue box).

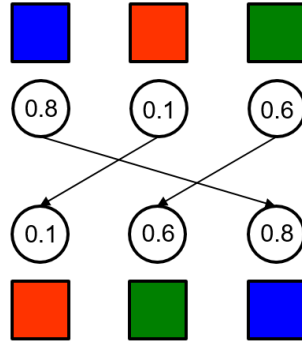


Figure 3. Decode to BPS

2.1 Chromosome representation and decoding

Definition 2.2. In general, the chromosome in the problem with n box to pack is made of $2n$ genes as depicted in the figure below. The first n genes are used to obtain the Box Packing Sequence and the genes $n + 1$ to $2n$ are used to obtain the Vector of Box Orientations.

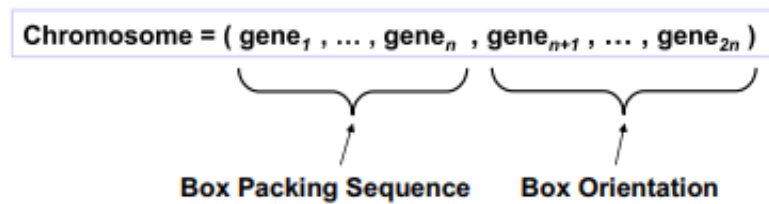


Figure 4. Chromosome

Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

Let BOs is the list of feasible orientations of box i , $nBOs$ is the number of feasible orientations. VBO_i is the gene that obtains the orientation of box i . Then we choose the orientation:

$$BO* = BOs(\lceil VBO_i \times nBOs \rceil).$$

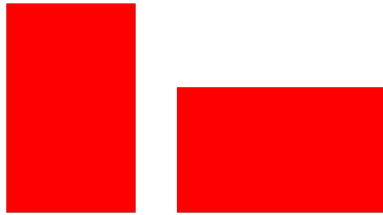


Figure 5. Orientaions of red box

In example 2.1, now we pack the red box into the bin, in this algorithm, we always pack the first box to the original point(the left - bottom corner). The red box have two feasible rotation as in the figure above, therefore $BOs = [vertical, horizontal]$. The red box orientation is determined by 0.9 from the VBO. Thus, we have

$$BO* = BOs(\lceil 0.9 \times 2 \rceil) = BOs(2).$$

So the box is packed into the bin with *horizontal* orientation.

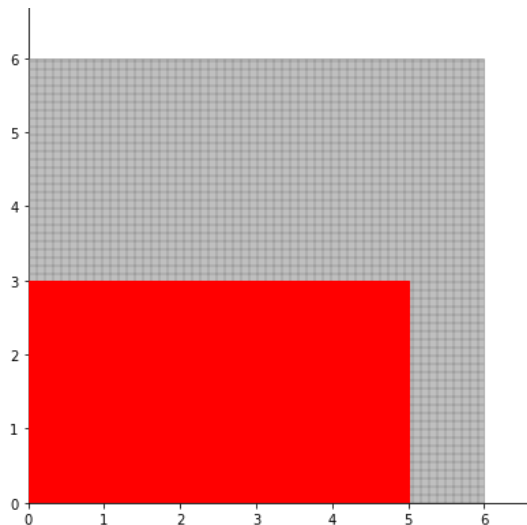


Figure 6. Bin after pack the red box

Then we pack the next box of the BPS is the green box. The value from the VBO which

is corresponding to the green box is 0.7, so we pack green box with horizontal orientation.

2.2 Empty maximal space selection

We have chosen the box to pack, now we have to decide the position to pack the box.

Definition 2.3. While trying to place a box in the bins we use a list S of empty maximal spaces (EMSs), i.e. largest empty rectangular spaces available for filling with boxes. EMSs are represented by their vertices with minimum and maximum coordinates $(x_i, y_i, z_i$ and X_i, Y_i, Z_i respectively).

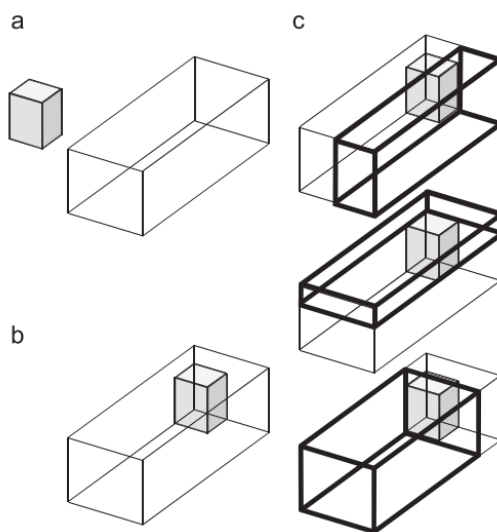


Figure 7. Empty maximal spaces in 3D packing problem

Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

To select EMS, we use the Distance to Front-Top-Right Corner (DFTRC) of the container as measure of compactness. We consider all the feasible orientation of the box to pack and all the EMSs. Each way to pack that box, we calculate the Euclidean distance between front-top-right corner of the box and the front-top-right corner of the the bin. We choose the EMS which maximizes the DFTRC. The first DFTRC is maximal DFTRC, so we choose the first EMS.

Note that if there is no EMS to select, we open a new bin to pack box to the original point.

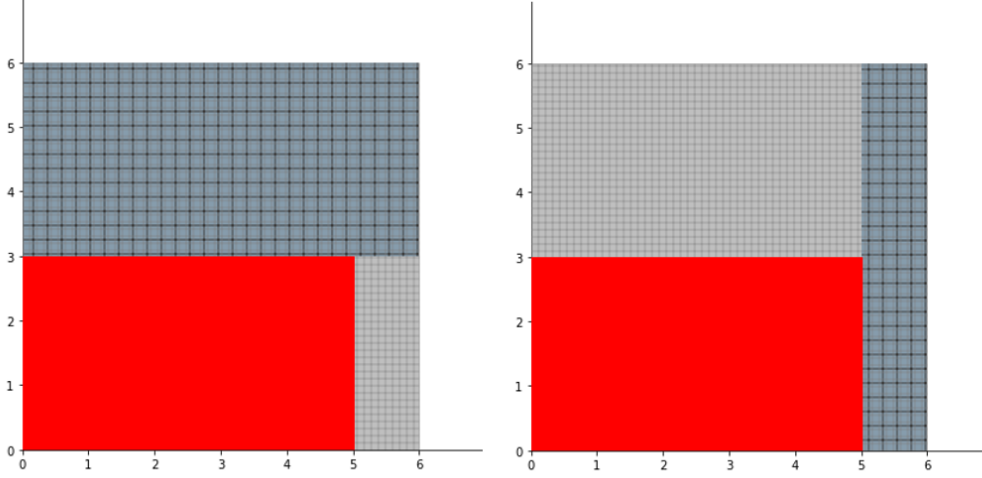


Figure 8. Empty maximal spaces in Example 2.1

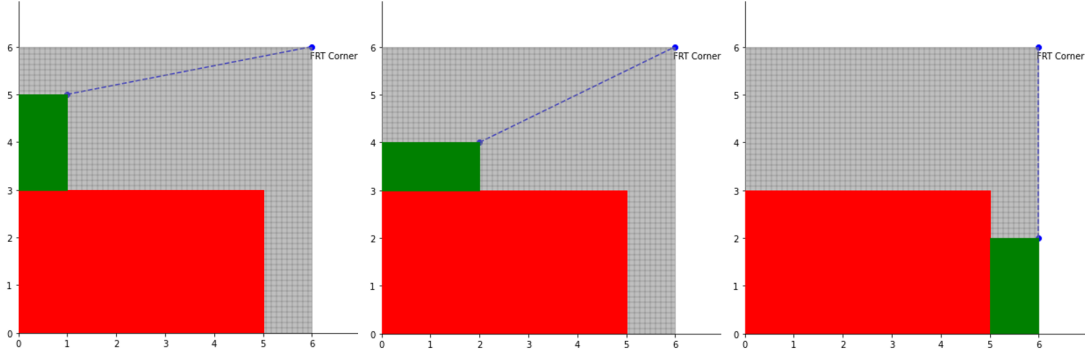


Figure 9. DFTRC in Example 2.1

2.3 Box packing

Now we have a box with orientation and an EMS to pack. We pack to box to the deep-bottom-left corner of the EMS.

In Example 2.1, we pack green box with horizontal orientation to the first EMS.

2.4 State update information by Difference Process

After packing, we delete all the EMSs that overlapped with the new box add new EMS that generate from old EMS.

Continue the process, we can finally have a complete solution.

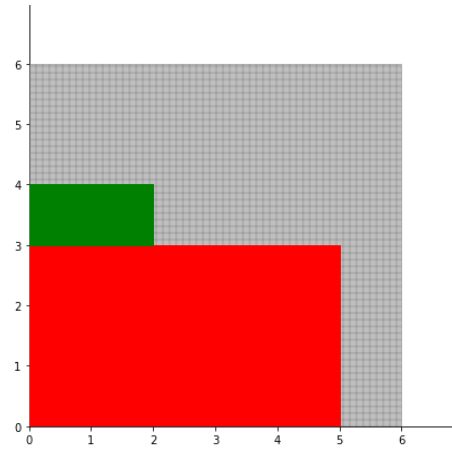


Figure 10. Packing box

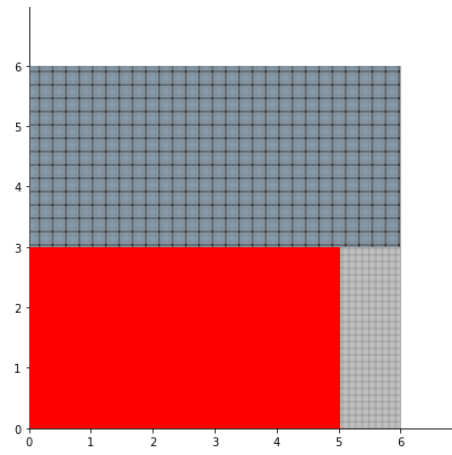


Figure 11. Old EMS

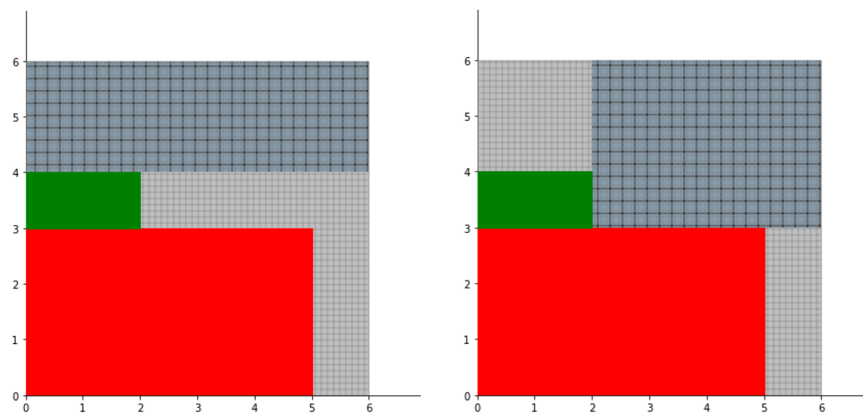


Figure 12. New EMSs after packing

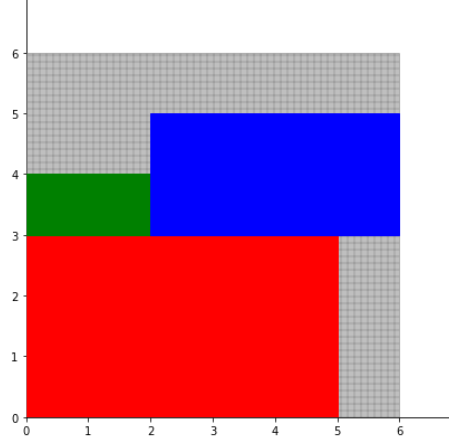


Figure 13. Complete solution corresponding to chromosome c in Example 2.1

2.5 Fitness function

To measure of fitness, we use a function which we call adjusted number of bins, aNB . The rationale for this new measure is that if we have two solutions that use the same number of bins, then the one having the least loaded bin will have more potential for improvement. Let *Least Load* be the load on the least loaded bin of a solution and let the capacity of the each bin be $BinCap = W \times H \times D$. The value of the adjusted number of bins is given by

$$aNB(chromosome) = Number\ of\ used\ bins + \frac{Least\ Load}{BinCap}.$$

2.6 Summary

The procedure combines the following elements the vectors BPS and VBO defined by the chromosome, the list S_b of empty maximal spaces for every open bin b , and the placement heuristics BFTRC. Each stage comprised of the following five main steps: Box selection, Bin and empty maximal space selection, Box orientation, Box packing, State information update.

3 Biased random key genetic algorithm

3.1 Architecture of the algorithm

3.2 Evolutionary process

In a random-key genetic algorithm, the population is initialized with random-key vectors whose components are random real numbers uniformly sampled from the interval $[0,1]$.

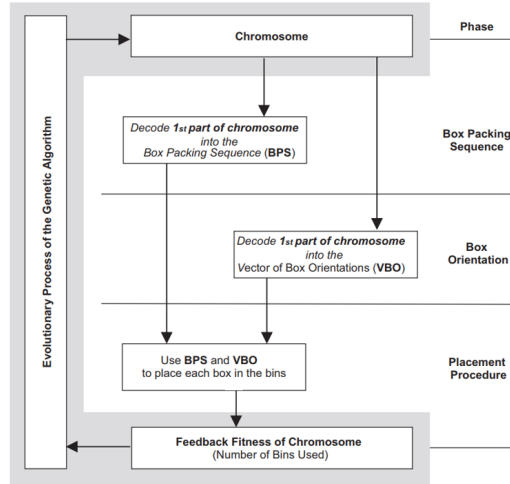


Figure 14. Architecture of the algorithm

Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

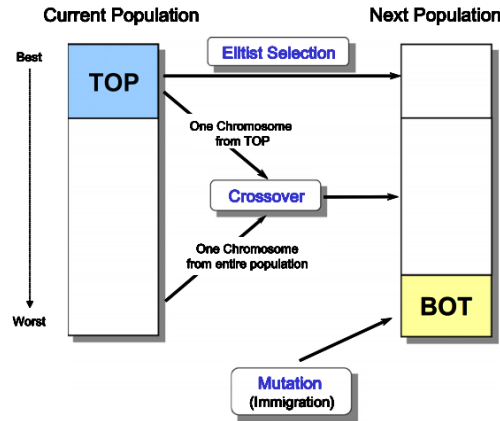


Figure 15. Evolutionary process

Source: A parallel multi-population biased random-key genetic algorithm for a container loading problem, Jose' Fernando Gonc-alves , Mauricio G.C. Resende, 2012, Computers Operations Research 39(2):179-190

Reproduction is accomplished by first copying some of the best individuals from one generation to the next,

Mutants are randomly generated from the same distribution as the initial population.

Assumes that a coin toss of heads selects the gene from the first parent, a tails chooses the gene from the second parent, and that the probability of tossing a heads is 0.7, i.e. the crossover probability $CProb = 0.7$.

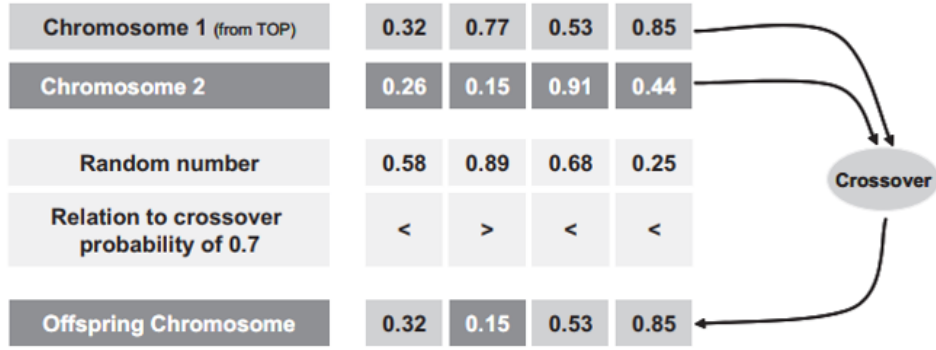


Figure 16. Crossover

Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

4 Numerical experiments

A standard benchmark set of 320 problems generated by Martello et al. (2000) was used for testing the 3D bin packing algorithm. The instance generator is available at <http://hjemmesider.diku.dk/~pisinger/codes.html>. We use the classes 6 and 7 which generate the boxes randomly from uniform distribution according to the intervals as follow:

- Class 6: bin size $W = H = D = 10$; $w_i, h_i, d_i \in [1, 10]$
- Class 7: bin size $W = H = D = 40$; $w_i, h_i, d_i \in [1, 35]$

The table below shows the result of our code with parameters: number of individuals is 100, number of generations is 10, number of elites is 10, number of mutants is 15, $CProb$ is 0.7. In the table, we compare our result with the result of paper [2] in reference with parameters: number of individuals is 30 times the number of boxes (p), number of generations is 200, number of elites is $p \times 0.1$, number of mutants is $p \times 0.15$, $CProb$ is 0.7.

Container size	10x10x10	10x10x10	40x40x40	40x40x40
Number of boxes	50	100	50	100
First time	7,75	17,22	7,29	13,29
Second time	11,56	13,11	4,25	9,39
Third time	9,25	15,58	7,38	11,5
Fourth time	7,22	16,25	6,31	12,1
Fifth time	7,8	14,51	6,23	11,5
Sixth time	7,25	14,1	5,54	10,34
Seventh time	5,85	13,1	6,19	10,05
Eighth time	8,15	15,78	5,68	11,43
Ninth time	6,62	16,13	5,42	11,31
Tenth time	7,38	13,44	5,41	10,12
Average	7,17	13,57	5,43	11,81
Result on paper	9,7	18,9	7,4	12,2

Figure 17. Result

5 Reference

- [1] A parallel multi-population biased random-key genetic algorithm for a container loading problem, Jose ´ Fernando Gonc-alves , Mauricio G.C. Resende, 2012, Computers Operations Research 39(2):179-190.
- [2] A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510.