Biased random key genetic algorithm for 3D bin packing problem

Dinh Hoang Nhat Minh

June 10, 2022

Outline

Problem

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 $i = 1, \ldots, n$. 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 $i = 1, \ldots, n$. It is assumed that the boxes can be rotated.

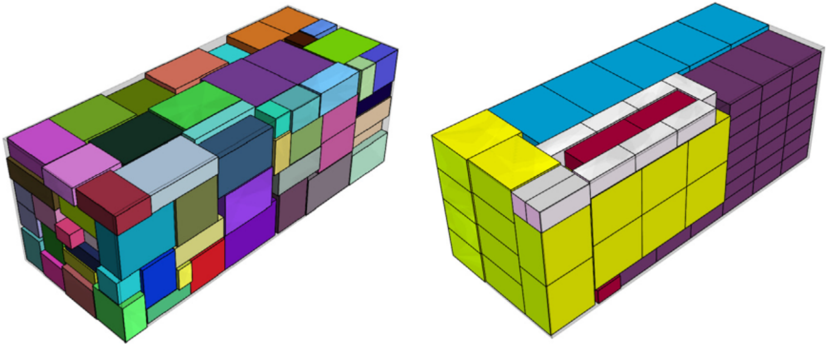


Figure 1: 3D Bin Packing Problem
Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

Example 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 $(6,6)$.

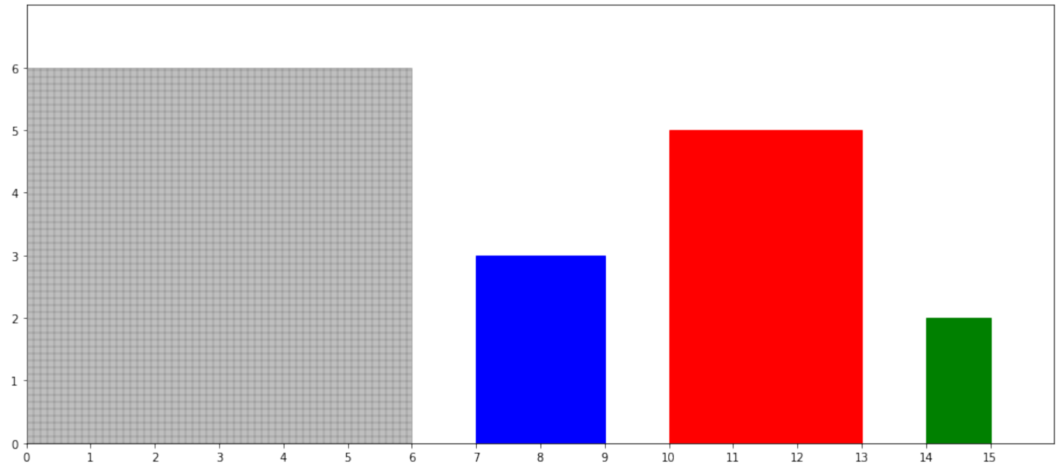


Figure 2: Example 1

Example 1

In example 1, we have 3 boxes to pack. We can decode a vector with 6 components (genes) are 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.

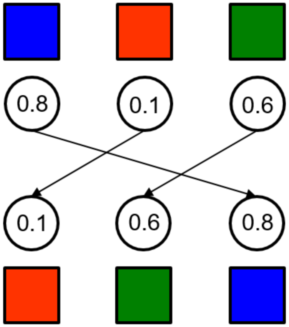


Figure 3: Decode to BPS

Therefore, the BPS is (red box, green box, blue box).

Chromosome

In general, the chromosome in the problem with n box to pack is 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.

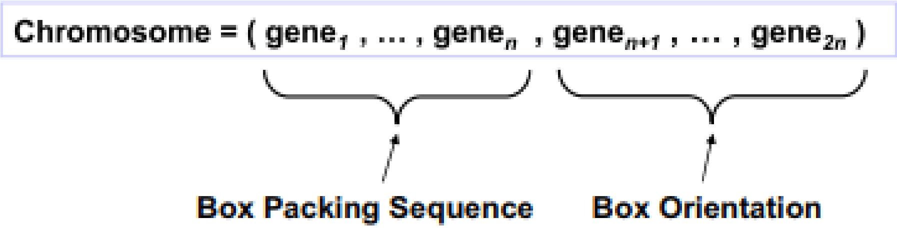


Figure 4: Chromosome
Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

Decode VBO

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:

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

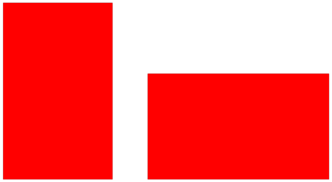


Figure 5: Orientaions of red box

In example 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.

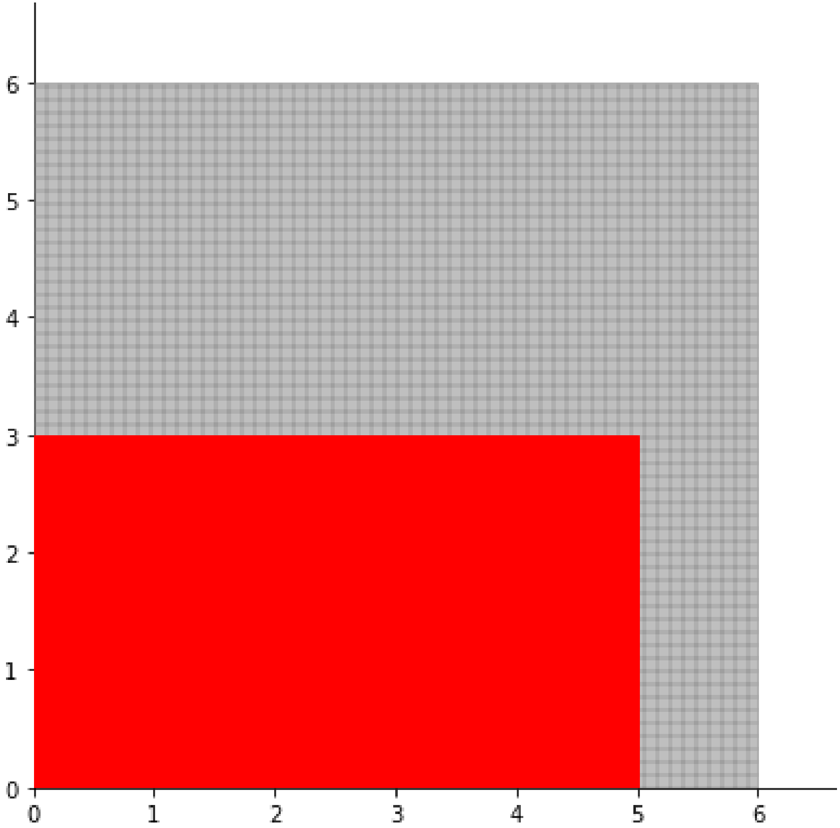


Figure 6: Bin after pack the red box

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.

The question is Where to pack?

Empty maximal space

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).

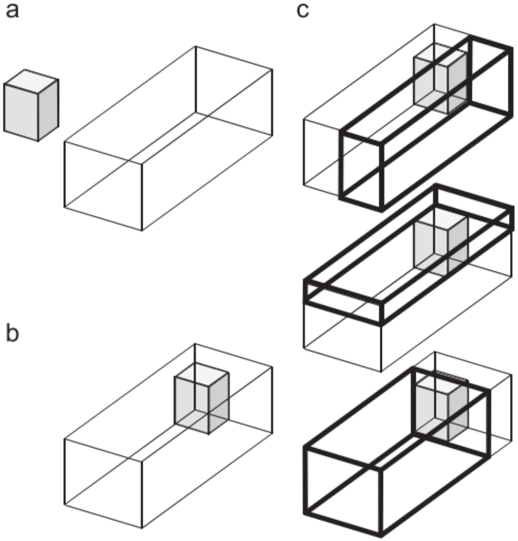


Figure 7: Empty maximal spaces in 3D packing problem
Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

Example 1

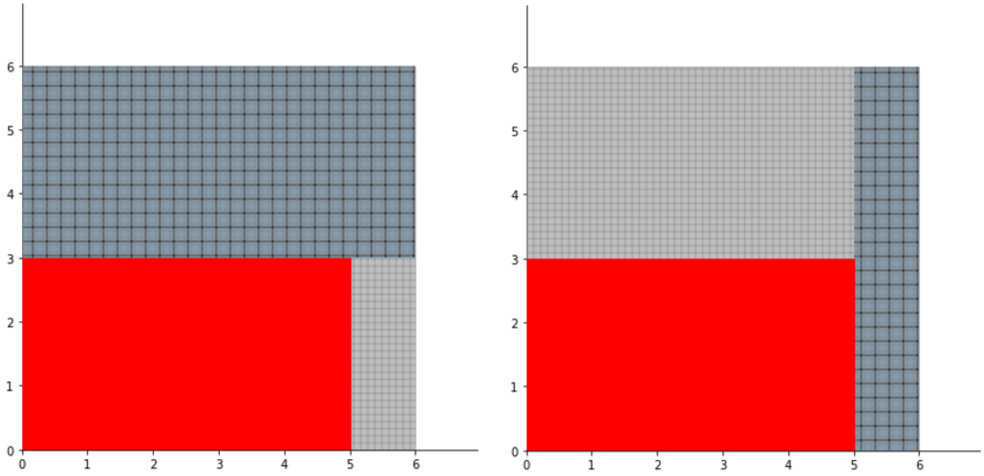


Figure 8: Empty maximal spaces in Example 1

EMS selection

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.

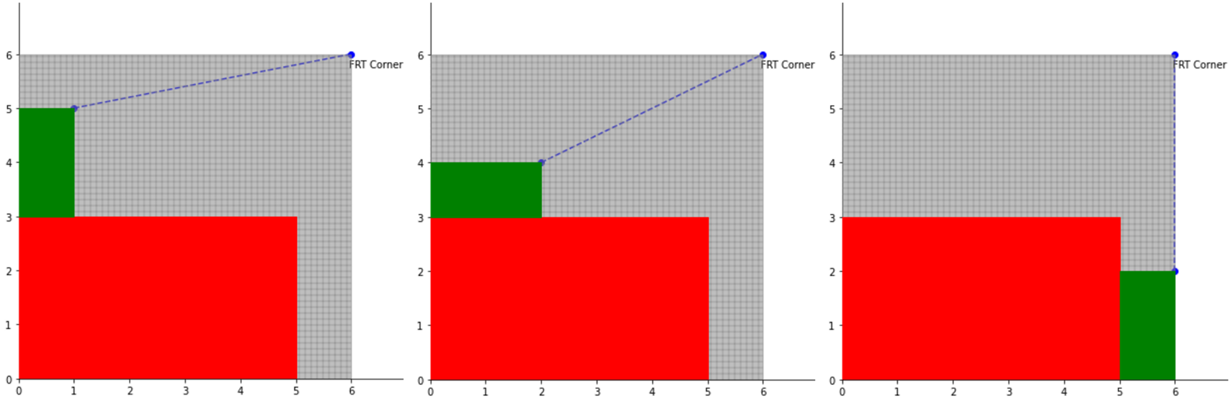


Figure 9: DFTRC in Example 1

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.

Packing box

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 1, we pack green box with horizontal orientation to the first EMS.

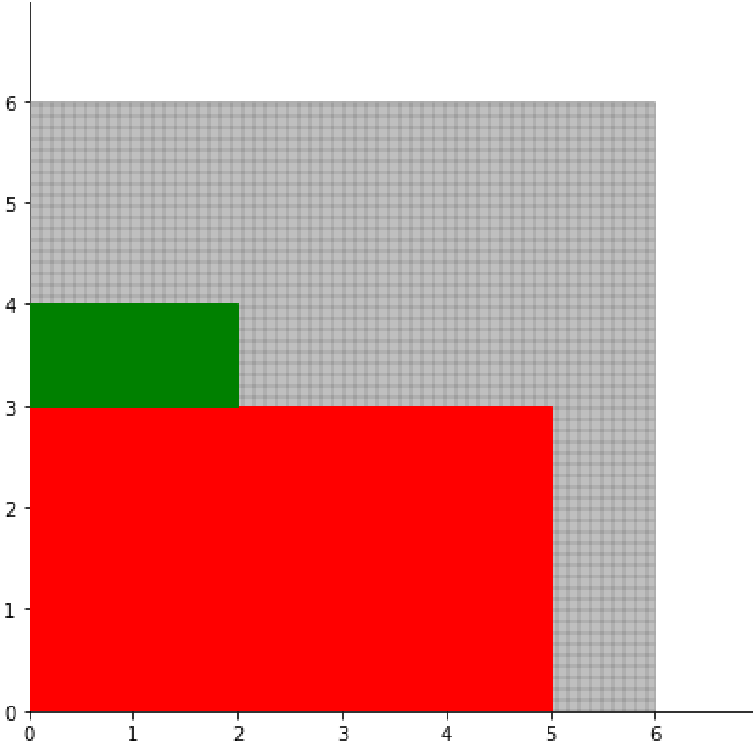


Figure 10: Packing box

Update information (list EMSs) by Difference Process

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

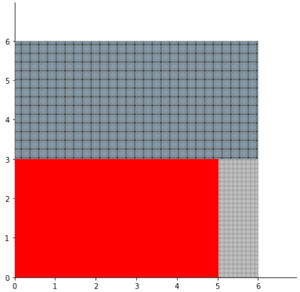


Figure 11: Old EMS

Update information (list EMSs) by Difference Process

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

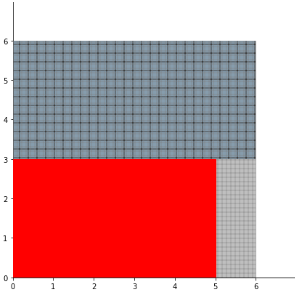


Figure 11: Old EMS

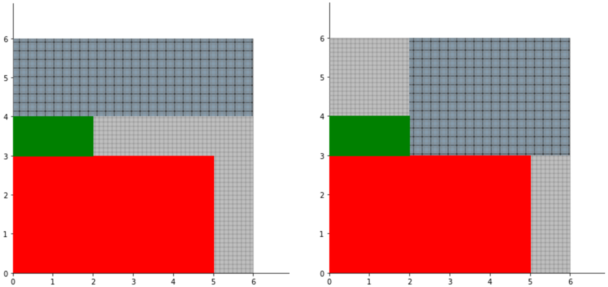


Figure 12: New EMSs after packing

Example 1

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

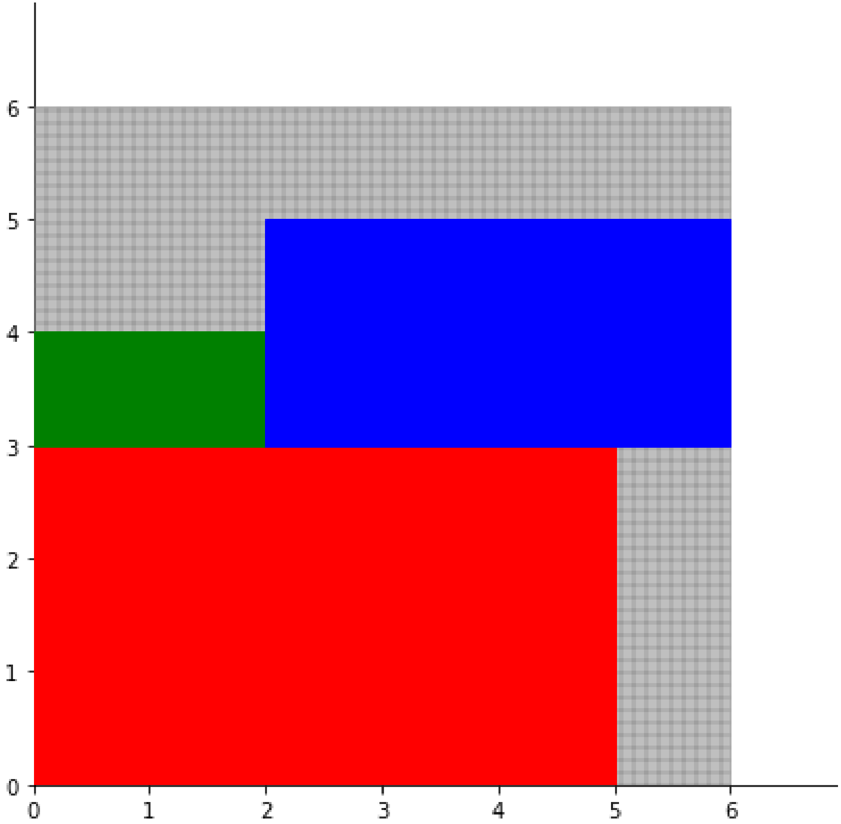


Figure 13: Complete solution corresponding to chromosome c in Example 1

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

$$aNB(chromosome) = Number\ of\ used\ bins + Least\ Load.$$

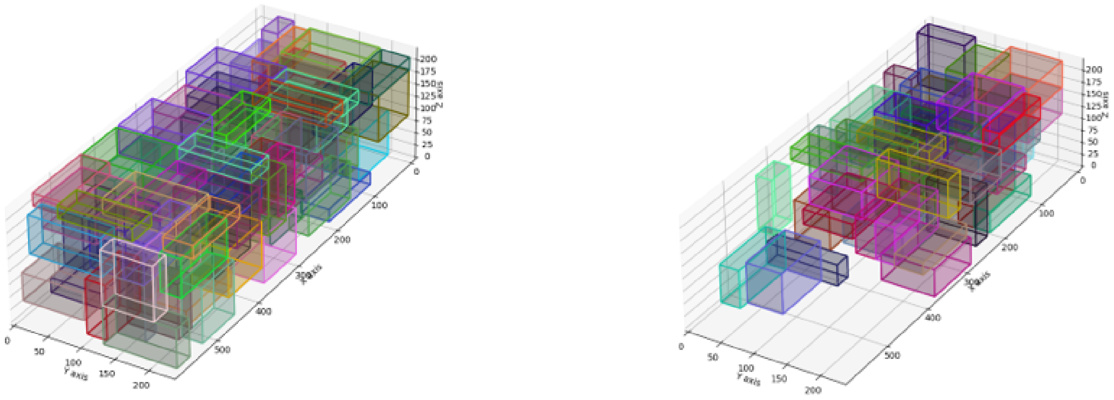


Figure 14: Packed bins

Architecture of the algorithm

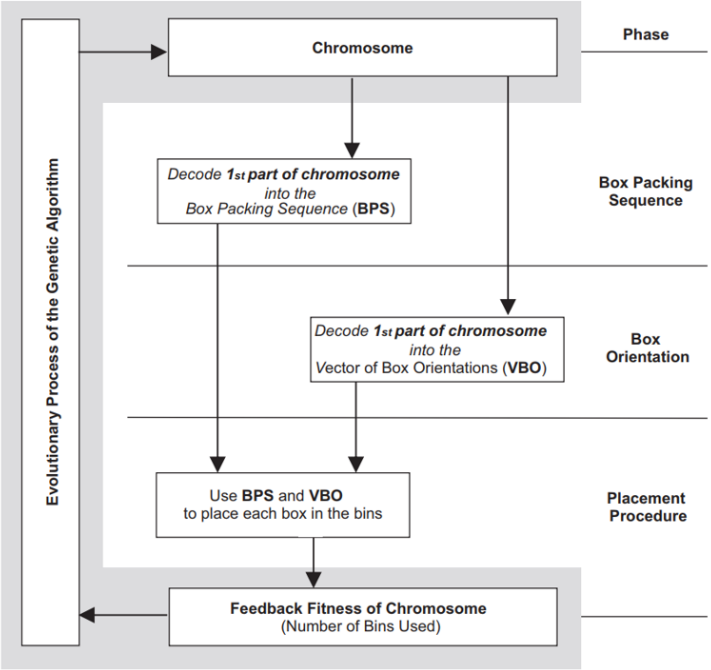


Figure 15: Architecture of the algorithm
Source: A biased random key genetic algorithm for 2D and 3D bin packing problems, José Fernando Gonçalves , Mauricio G.C. Resende, 2013, International Journal of Production Economics 145(2):500-510

Evolutionary process

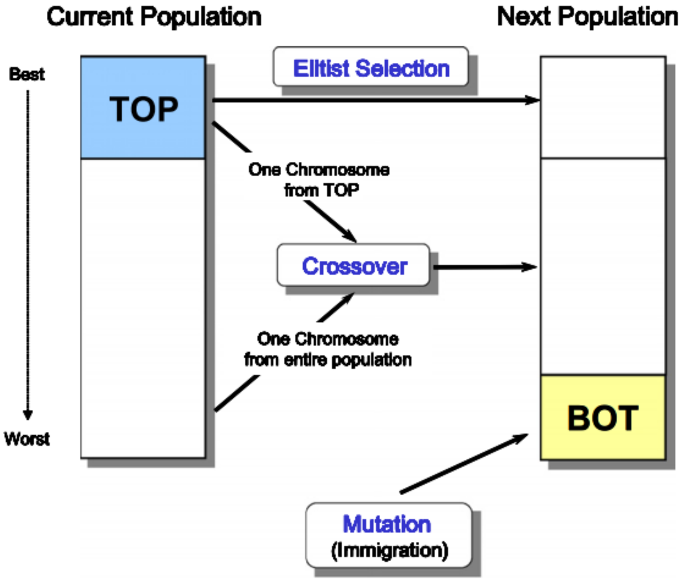


Figure 16: Evolutionary process
Source: A parallel multi-population biased random-key genetic algorithm for a container loading problem, Jose´ Fernando Gonc-alves , Mauricio G.C. Resende, 2012, Computers Operations Research 39(2):179-190

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].

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].

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

Evolutionary process

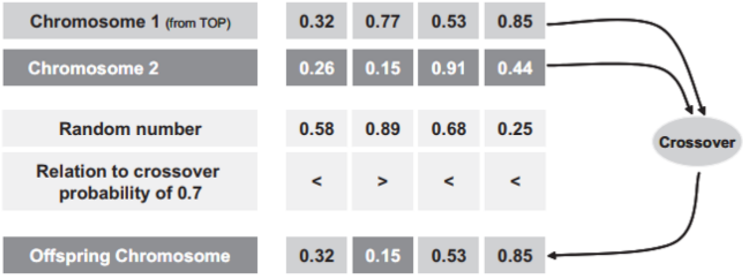
In a random-key genetic algorithm, the population is initialized with random-key vectors whose components are random real numbers uniformly sampled from the interval [0,1].

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.

Evolutionary process

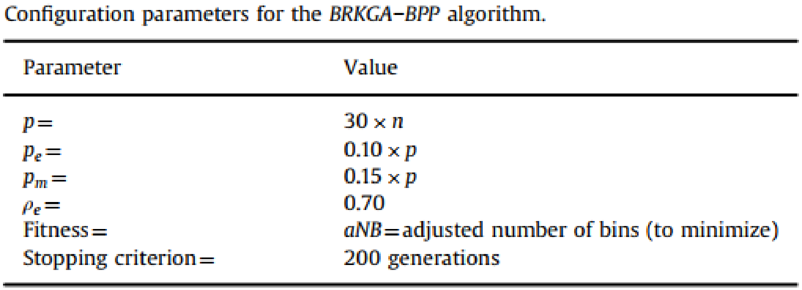
In a random-key genetic algorithm, the population is initialized with random-key vectors whose components are random real numbers uniformly sampled from the interval [0,1].
Reproduction is accomplished by first copying some of the best individuals from one generation to the next,
Mutants are randomly 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 $CProb = 0.7$.



Chromosome 1 (from TOP)	0.32	0.77	0.53	0.85
Chromosome 2	0.26	0.15	0.91	0.44
Random number	0.58	0.89	0.68	0.25
Relation to crossover probability of 0.7	<	>	<	<
Offspring Chromosome	0.32	0.15	0.53	0.85

Figure 17: 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

Configuration parameters

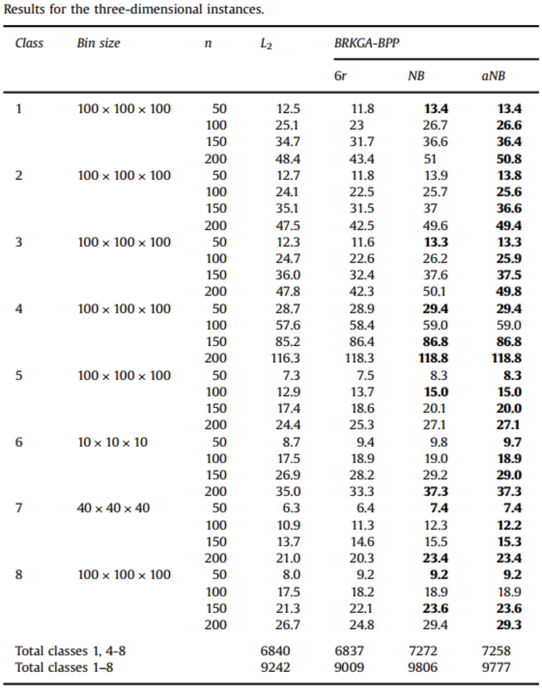


Configuration parameters for the *BRKGA–BPP* algorithm.

Parameter	Value
$p=$	$30 \times n$
$p_e=$	$0.10 \times p$
$p_m=$	$0.15 \times p$
$\rho_e=$	0.70
Fitness=	*aNB*=adjusted number of bins (to minimize)
Stopping criterion=	200 generations

Figure 18: Configuration parameters
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

Result



Results for the three-dimensional instances.

Class	*Bin size*	*n*	L_2	*BRKGA-BPP*		
				6r	*NB*	*aNB*
1	100 × 100 × 100	50	12.5	11.8	**13.4**	**13.4**
		100	25.1	23	26.7	**26.6**
		150	34.7	31.7	36.6	**36.4**
		200	48.4	43.4	51	**50.8**
2	100 × 100 × 100	50	12.7	11.8	13.9	**13.8**
		100	24.1	22.5	25.7	**25.6**
		150	35.1	31.5	37	**36.6**
		200	47.5	42.5	49.6	**49.4**
3	100 × 100 × 100	50	12.3	11.6	**13.3**	**13.3**
		100	24.7	22.6	26.2	**25.9**
		150	36.0	32.4	37.6	**37.5**
		200	47.8	42.3	50.1	**49.8**
4	100 × 100 × 100	50	28.7	28.9	**29.4**	**29.4**
		100	57.6	58.4	59.0	59.0
		150	85.2	86.4	**86.8**	**86.8**
		200	116.3	118.3	**118.8**	**118.8**
5	100 × 100 × 100	50	7.3	7.5	8.3	**8.3**
		100	12.9	13.7	**15.0**	**15.0**
		150	17.4	18.6	20.1	**20.0**
		200	24.4	25.3	27.1	**27.1**
6	10 × 10 × 10	50	8.7	9.4	9.8	**9.7**
		100	17.5	18.9	19.0	**18.9**
		150	26.9	28.2	29.2	**29.0**
		200	35.0	33.3	**37.3**	**37.3**
7	40 × 40 × 40	50	6.3	6.4	**7.4**	**7.4**
		100	10.9	11.3	12.3	**12.2**
		150	13.7	14.6	15.5	**15.3**
		200	21.0	20.3	**23.4**	**23.4**
8	100 × 100 × 100	50	8.0	9.2	**9.2**	**9.2**
		100	17.5	18.2	18.9	18.9
		150	21.3	22.1	**23.6**	**23.6**
		200	26.7	24.8	29.4	**29.3**
Total classes 1, 4-8			6840	6837	7272	7258
Total classes 1–8			9242	9009	9806	9777

Figure 19: Result
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

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.