

# An improved genetic algorithm for solving packing problem

LI Zhi-yan<sup>1, a</sup>

<sup>1</sup>Department of Mathematics and Physics, Hohai University, Changzhou Campus,  
Changzhou, P.R. of China

<sup>a</sup>email: Zhiyanli79@163.com

**Keywords:** Bin-Packing Problem; Genetic Algorithm; Best Fit Decrease; Combination Optimization

**Abstract.** Targeted at solving slow rate of convergence in the current genetic algorithm, an improved genetic algorithm is put forward in the article. The current genetic algorithm is improved by adding the best fit decrease algorithm to generate individuals into the initial population, conversing the preservation strategy and fitness of the optimum individual. To verify the validness of the algorithm, simulation experiments are designed. And the result of these experiments shows that the improved algorithm has the bigger probability to find the optimal solution and faster solution speed.

## Introduction

The bin-packing problem, widely existing in the transportation industry, construction industry, metal manufacturing industry and other industries, is the basis for many important actual optimization problems. The bin-packing problem belongs to NP-problems which can not be solved accurately. At present, there are two common solutions: heuristic algorithm and genetic algorithm [1-4]. Approximate solution for problems can be found by the heuristic algorithm, but in some extreme conditions, the result is found unsatisfactory. As a kind of random search algorithm, the genetic algorithm has strong global searching ability, but the main problems for it are relatively slow speed and poor timeliness for convergence to the globally optimal solution. Aimed at improving the current genetic algorithm, an improved genetic algorithm is brought forward in the article.

## Description and Solution to the Target Problem

The bin-packing problem requires putting  $n$  items into several boxes (up to  $n$  boxes). Each item has weight ( $w_j > 0$ ), while each box has a weight limit ( $c_i > 0$ ). The target is to find the optimal solution to put items into boxes, making the total weight of items in each box be less than the limit and using the minimum box quantity. Generally, all boxes have the same weight limit ( $c > 0$ ).

Mathematical expressions of the bin-packing problem are as follows:

$$\min \quad z(y) = \sum_{i=1}^n y_i \quad (1)$$

$$s.t. \quad \sum_{j=1}^n w_j x_{ij} \leq c y_i, \quad i \in N = \{1, 2, \dots, n\} \quad (2)$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j \in N \quad (3)$$

$$y_i = 0 \text{ or } 1, \quad i \in N \quad (4)$$

$$x_{ij} = 0 \text{ or } 1, \quad i, j \in N \quad (5)$$

Where,  $y_i = 1$  stands for items in the box  $i$ , otherwise the box  $i$  is empty.  $x_{ij} = 1$  stands for item  $j$  putting into the box  $i$ , otherwise the item  $j$  is not in the box  $i$ .

## Heuristic algorithm

Targeted at the problem, several heuristic algorithms are put forward, such as, NF algorithm, EF algorithm, BF algorithm, FFD algorithm, BFD algorithm, etc. Literature [4] compares and analyzes characters of these heuristic algorithms.

Tab.1. Characteristics of some heuristic algorithms

Heuristic algorithm	Whether online feature contained	Complexity Of time	Asymptotic worst-case performance ratio
Next fit algorithm (NF)	Online	$O(n)$	2
First fit algorithm (FF)	Online	$O(n \log n)$	1.7
Best fit algorithm (BF)	Online	$O(n \log n)$	1.7
First fit decrease algorithm (FFD)	Offline	$O(n \log n)$	11/9
Best fit decrease algorithm (BFD)	Offline	$O(n \log n)$	11/9

### Genetic algorithm

Literature [5] introduces genetic algorithms based on box coding scheme, item coding scheme and population coding scheme. Literature [6] demonstrates that the genetic algorithm based on population coding scheme is relatively better algorithm, which greatly reduces redundancy of algorithms and always converges to the globally optimal solution.

### Design of the improved algorithm

It is found that there are relatively many evolution algebras for converging to the globally optimal solution for the algorithm introduced in Literature [6], while its solution results are found unsatisfactory. By enlightenment of Literature [7] and Literature [8], strategies in following three aspects are proposed to adopt for improvement.

#### Add BFD algorithm to the initial population to generate individuals

Through comparison to several heuristic algorithms in table 1, it is seen that BFD algorithm and FFD algorithm are relatively better offline algorithms. Superior individuals generated by adding BFD algorithm and FFD algorithm in the initial population can strengthen genetic search ability, thus finding the most close optimal solution for the bin-packing problem.

BFD algorithm firstly conducts the descending sort to boxes in accordance with weight; then, puts boxes into the feasible boxes with the minimum residual capacity one by one. FFD algorithm firstly conducts the descending sort to boxes in accordance with weight; then, puts boxes into initialized boxes with minimum subscript. By a large amount of experiments, asymptotic worst-case performance ratio of the two algorithms is the same, but for the fitness function in formula (6), fitness of individuals generated by BFD algorithm is always no less than that of FFD algorithm, therefore, in the population-initialized, if individuals generated by BFD algorithm are chosen to be added, the ability to find the optimal solution can be improved.

#### Scale transform of the fitness function

If superior individuals and optimal individuals keeping strategies added into the population-initialized are adopted, it means that after evolution of certain algebra, due to relatively small individual fitness difference in the population, rate of convergence of the algorithm will become slow and continuous optimizing potential will decrease. To overcome the shortcoming, scale transform is conducted on the fitness function.

The fitness function introduced in Literature [6] is as follows:

$$f_{BPP} = \frac{\sum_{i=1}^N \left( \frac{F_i}{c} \right)^k}{N} \quad (6)$$

Where,  $N$  is the box quantity used in the solution,  $F_i$  is the sum of weight of all items in box  $i$  (filling degree of boxes),  $c$  is the weight limit of boxes, constant  $k$  which is 2, stands for the recognition degree for full loaded boxes.

Scale transform is done on the fitness function by adopting linear transformation:

$$f'_{BPP} = a \cdot f_{BPP} + b \quad (7)$$

Where,  $f'_{BPP}$  is the fitness function after the scaling,  $a$  and  $b$  are scale factors whose computing methods are as follows:

$$a = \frac{f_{avg}}{f_{avg} - f_{min}}, \quad b = \frac{-f_{min} \cdot f_{avg}}{f_{avg} - f_{min}} \quad (8)$$

Where,  $f_{avg}$  stands for the mean value of the original fitness,  $f_{min}$  stands for the minimum value of the original fitness,  $f_{max}$  stands for the maximum value of the original fitness.

### Description of the algorithm

Based on the existing algorithms [9] [10], steps of algorithms are improved. The improved algorithm steps are as follows:

Step 1: set the algorithm parameters, generate the initial population and add individuals generated by BFD algorithm in the initial population;

Step 2: calculate fitness of parent individual according to formula (6);

Step 3: copy optimal parent individual;

Step 4: conduct fitness scale transform according to formulas (7) and (8);

Step 5: choose, cross and conduct mutation operation to generate offspring;

Step 6: calculate fitness of offspring individuals according to formula (6);

Step 7: find the optimal offspring individual;

Step 8: compare fitness of optimal parent individual and optimal offspring individual. When fitness of the optimal parent individual is better than that of the optimal offspring individual, replace the optimal offspring individual with the optimal parent individual.

Step 9: verify termination conditions. If the maximum iteration algebra is reached or globally optimal solution is found, the algorithm is terminated and the optimal individual in step 7 is the solution to the genetic algorithm; otherwise turn to step 3.

### Simulation Verification

The experiment put 100 objects into many boxes that the weight is limited to 1, so as to seek for the optimal plan to distribute the objects into the boxes. In order to obtain the certain optimal solution, the weight of objects shall be designed specially. 100 objects needed to be put into the boxes shall be divided into two groups – A and B, and there are 50 objects in each group. The object weight of A group  $a_i, i=1, 2, \dots, 50$ , which complies the uniform distribution between 0.4~0.6. The object weight of B group  $b_i, i=1, 2, \dots, 50$ , which meets  $b_i=1-a_i$ . Obviously, the optimal solution for box number is 50 when putting the 100 objects into the boxes.

The experiment compares three algorithms – algorithm 1 is the algorithm in Literature [6]; algorithm 2 is the improved algorithm proposed in this paper; algorithm 3 is fundamentally the same as algorithm 2, and the only difference is that the individual generated by FFD algorithm is added to the initial population.

The operating parameter setting of algorithm: the value of crossover probability  $P_c$  is 0.8; the value of mutation probability  $P_m$  is 0.1; the value of maximum iterative times  $T$  is 1000; the value of population size is 100. The termination condition of algorithm is achieving the maximum iterative times or finding the optimal solution. The experiment conducts 30 simulation calculations to the three algorithms respectively. The iterative times of Gen that represents the termination of calculation. Figure 1 is the comparison figure of iterative times when the simulation calculation of three algorithms terminates.

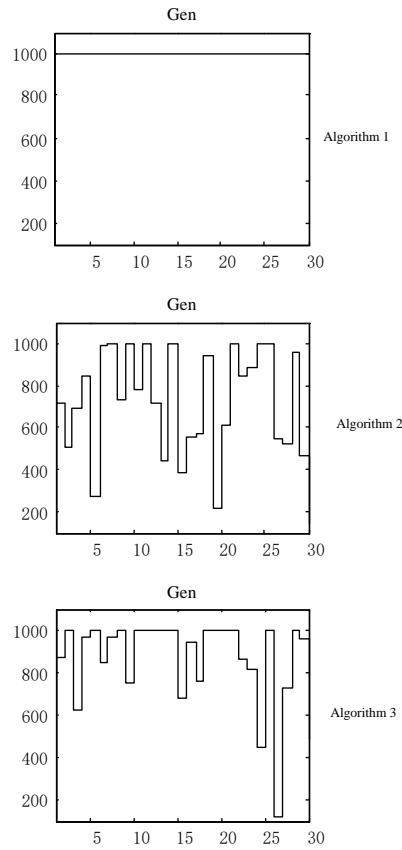


Fig.1. Comparison of iterative times when the simulation calculation terminates

Bin represents the calculation results when calculation terminates (number of boxes occupied by the optimal individual in population). Figure 2 is the comparison figure of calculation results when the simulation calculation of three algorithms terminates.

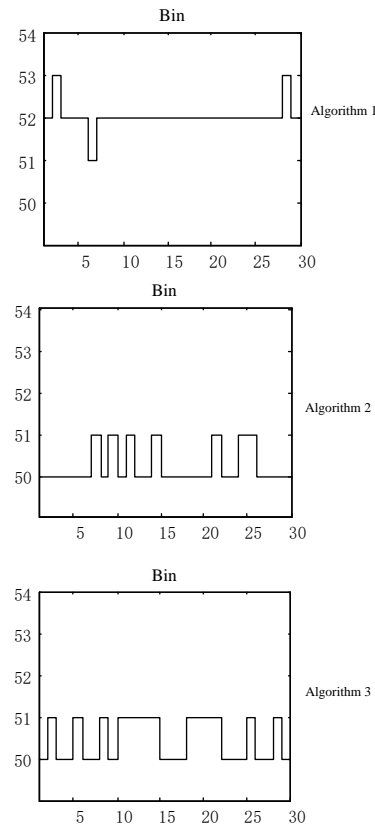


Fig.2. Comparison of calculation results when the simulation calculation terminates

The article conducts statistics of the experiment results by different classes, and the statistics indexes include: times for finding optimal solution during simulation calculation, success rate for finding optimal solution, minimum value of Bin, maximum value of Bin, minimum value of Gen, maximum value of Gen, average value of Gen and etc. Table 2 is the statistical table of simulation results of three algorithms.

Tab.2. Statistics of the simulations results

	Algorithm 1	Algorithm 2	Algorithm 3
Times for finding optimal solution	0	23	16
Success rate for finding optimal solution	0%	76.7%	53.3%
Minimum value of Bin	51	50	50
Maximum value of Bin	53	51	51
Average value of Bin	52.0	50.2	50.5
Minimum value of Gen	1000	142	116
Maximum value of Gen	1000	1000	1000
Average value of Gen	1000.0	711.4	861.3

The article also conducts mean difference test to the Bin value and Gen value of algorithm 2 and 3. The test result shows that, the average value of Bin and Gen of algorithm 2 is less than the index of same item of algorithm 3 at the remarkable level of 0.05. Hereby proves that, the method that adds the individual generated by BFD algorithm into initial population (algorithm 2) when improving the genetic algorithm is better than the method that adds the individual generated by FFD algorithm (algorithm 3) in deed.

We can know that, compared with the algorithm (algorithm 1) in Literature [6], the success rate for finding optimal solution of improved genetic algorithm (algorithm 2) within the limited evolutionary algebra increases significantly, and the calculation result is closer to the optimal solution. It clearly shows that the property is better when applying the improved genetic algorithm proposed in this article to solve the bin-packing problem.

## Conclusion

Aiming at the deficiency of the existing genetic algorithm, the article proposes an improved algorithm to solve the bin-packing problem. The algorithm improves the algorithm in Literature [6] through adding BFD individual into initial population, applying optimal individual storage strategy and conducting scale changes to fitness function. The simulation research result shows that the success rate for finding globally optimal solution of the improved algorithm proposed in this article is higher and the solution speed is faster, and it is a genetic algorithm that the property is better.

## Acknowledge

the National Science Foundation of China under Grant No.11471073

## References

- [1] Guanqun Bao, Liang Mi, Yishuang Geng, Mingda Zhou, Kaveh Pahlavan, A video-based speed estimation technique for localizing the wireless capsule endoscope inside gastrointestinal tract, 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Aug. 2014
- [2] Degui Zeng, Yishuang Geng, Content distribution mechanism in mobile P2P network, Journal of Networks, 9(5), 1229-1236, Jan. 2014
- [3] Gu W, Lv Z, Hao M. Change detection method for remote sensing images based on an improved Markov random field[J]. Multimedia Tools and Applications, 2015: 1-16.

[4] Chen Z, Huang W, Lv Z. Towards a face recognition method based on uncorrelated discriminant sparse preserving projection[J]. Multimedia Tools and Applications, 2015: 1-15.