

An Improved Genetic Algorithm for 0-1 Knapsack Problems

Wei Shen, Beibei Xu, Jiang-ping Huang

Faculty of Informatics & Electronics

Zhejiang Sci-Tech University

Hangzhou, China

Email: shenwei@zstu.edu.cn, xubeibei01@163.com, huangjiangping@yahoo.com

Abstract—The 0-1 knapsack problems is a problem in combinatorial optimization, which is NP-complete to solve exactly. A genetic algorithm is a kind of heuristic that mimics the process of natural evolution. We derived an improved solution for 0-1 knapsack problem based on the dual population genetic algorithm, which can overcome the defect of precocious and local convergence in iterative processes. The performance evaluation shows that the solution is better than the traditional genetic algorithm.

Keywords—0-1 knapsack problems; dual-population genetic algorithm; greedy criterion; sub-group competition; population collaboration

I. INTRODUCTION

0-1 Knapsack Problems is one of the typical NP problem in combinatorial optimization problems. The 0-1 Knapsack Problems have been used widely in the actual project, such as goods distribution, budget control and resource allocation. At present, there are a lot of solving methods of 0-1 Knapsack Problems [1] which can be divided into three categories: optimization algorithms, heuristic algorithms and hybrid algorithms. The optimization algorithms include dynamic programming and recursive algorithm; the heuristic algorithms consist of genetic algorithms, ant colony algorithms and greedy algorithms. the hybrid algorithms mainly refer to the use of integration between various algorithms.

Genetic Algorithm(GA) [2] is a kind of randomized search algorithms, which uses biological natural selection of biological, survival of the fittest and the genetic mechanism of species for reference. Genetic algorithm is characterized by the use of coding variable parameters for optimization and multiple points to searching. At present, there are many articles on using genetic algorithm to solve 0-1 knapsack problems. The studies have tended to be mature. The current work is mainly proceeded by improving the encoding mode, model structure, selection operator, adaptive and hybrid genetic algorithm. Good coding methods can optimize genetic manipulation, which include simple code of length, binary code, and complex orthogonal codes. As for model structure, single species has been developed into multi-population. Multi-population genetic algorithms can solve the problems of local convergence and local optimal solution in the genetic algorithm, because of each population's independent evolution and exchange between superior

individuals. However, because of the independent evolution of each population, genetic manipulation would waste a lot of time, which increases the time complexity. To improve that, a dual population genetic algorithm was suggested [3]. At present, Fang Bihe proposed dual-population genetic algorithm based on elimination mechanisms [3]. Yang Xiao-qin proposed entropy-based dual genetic algorithm [4]; Lu Xueyan raised bees bi-population genetic algorithm [5]. The performance of these structures are better than those used by traditional genetic algorithms, more stable than multi population genetic algorithms, and less time consuming. But those above two dual population structure did not give a good idea on reducing the convergence time and changing the population parameters in later iterations.

In this paper, we make several improvements to the three different dual populations structure mentioned above. We added populations cooperation ideas in the algorithm to solve the 0-1 knapsack problems, named as dual species collaboration competitive algorithm.

II. 0-1 KNAPSACK PROBLEMS

The common formulation of 0-1 Knapsack Problems can be described as follows : given m items, the volume of each item is $w_i (1 \leq i \leq m, i \in N)$, and the value of each item is $p_i, (1 \leq i \leq m), i \in N$, how to choose some items and put into the knapsack so that the sum of value of items in the knapsack is maximized and the sum of volume of them doesn't exceeding W . The mathematical model of 0-1 Knapsack Problems can be described as follows:

$$\max f(w_1, w_2, \dots, w_m) = \sum_{i=1}^m p_i x_i,$$

$$\sum_{i=1}^m w_i x_i \leq W,$$

where $x_i \in \{0, 1\}, i = 1, 2, \dots, m$, which means if $x_i = 1$, the item was put into the knapsack, otherwise, $x_i = 0$.

III. TECHNICAL APPROACHES

A. The basic idea of Dual population cooperative competition algorithm

The traditional two-population genetic algorithm is to utilize the two-species to evolve independently. Then the

best individual is distributed to the two groups. Hereafter a degree of individual whose cross-chromosome were exchanged to the other group, were selected from the two groups, respectively. By doing this we can ensure the diversity of the population, and overcome the problem of local convergence. Dual population is in this collaborative competition algorithm based on the increase of population thinking and collaborative thinking progeny double-population competitive co-evolution. The ideas we used are shown as below:

1) *Greedy Criterion*: Constructing the optimal solution gradually is used in the greedy algorithm (greedy method). At each stage, an optimal decision was made under certain criteria. Once the decision was made, it can not be changed. Greedy decisions [6] were based on a criteria called greedy criterion. As for the 0-1 knapsack problems, it means that the object relative weight values $\frac{p_i}{w_i}$ were first sorted in a descending order, then the object were put into the bag successively, until overall weight exceed the maximum capacity weight. However, in this way, we can only get the approximate optimal solution, can not guarantee the reliability.

In this article the greedy criterion is used to rebuild the individuals who are not meet the constraints in gene repair stage. We can determine the genes which need to be changed by comparing the density of goods' value, in order to make sure the individual meet the constraints and get the greatest fitness.

2) *Population Collaboration*: Two populations evolve through cooperate. The two populations proceed genetic operation through different crossover probability p_c and mutation probability p_m . One of the species has larger value of p_c and p_m , we call it ups populations, which is mainly used in the overall evolution procedure to provide new optimal individual to overcome the premature convergence and to prevent falling into local convergence. In this population, the best individual of each generation evolution is put into to the other species. The value of p_c and p_m in another population are small, we call it the main population, which is mainly used to find the outstanding individuals and maintain it. The structural model is shown as follows:

The two sub-populations evolve independently. After one generation's evolution, the optimal individual determined by Elite Strategy is saved to the next generation. Firstly, The population groups in population collaboration achieve their own individual evolution through competition which is based on independent evolutionary model. Secondly, based on information transmission between populations and survival of the fittest mechanism, we can compare and exchange to achieve mutual cooperation between individuals through the optimal solution set of each species. In this way we can not only realize the complementary strengths of different populations, but also improve the diversity of the population.

3) *Offspring Competition*: The collection of offspring refers to the individual parent with the chromosome's split and reorganization in the process of genetic cross, the offspring is likely to be fragments of the gene sequences generated through the restructuring of all individuals. Previous genetic algorithm, just simply collect a random of individuals to take one or two, which is not rigorous, easy to fall into the convergence stagnation, and resulted in the phenomenon that the offspring are worse than their parents. Besides it can damage the high fitness individual gene sequences, resulting in degradation of the population. By using sub-group competition, not only can we avoid the stagnation of convergence, but also can select outstanding individuals, obsolete fitness of the individual, and conducive to the evolution of population convergence.

B. idea of 0-1 Knapsack Problems

The exchange of information in populations avoid the occurrence of local convergence. The independent evolution make the evolutionary direction continuously approach the optimal solution. Both of the algorithms belong to the double population cooperative evolution algorithm. Using the algorithm to solve 0-1 knapsack problems can not only improve the efficiency of the genetic algorithm but also proved to be preciser and faster than traditional genetic algorithms. The design steps of double population genetic algorithm is listed below.

1) The Basic Idea:

- Step a Generate initial population, and the individual for the knapsack problem is a combination solution, but every gene's 0-1 in the individual chromosome represents the corresponding objects are selected or not. The scale of population is $POPSIZE = 100$, and set the length of the chromosome $Chrom = 50$.
- Step b If for the individual not meet constraint condition, we use the greedy regular to modify each individual of double population and make every individual meet constraint condition.
- Step c Calculate the fitness.
- Step d Select the highest fitness of individual, and retain elite.
- Step e Select the cross-individual and randomly generate the location of crossover point, in the meantime, switch the segment of genes for cross individual.
- Step f Perform subgroup competition, and execute inferiority elimination strategy for all individuals generated by the segment of genes exchanging.
- Step g Perform variation operation.
- Step h Calculate the fitness, and use greedy strategy to amend the individual in order to meet constraints.
- Step i Select competition groups and compare competition and implement co-evolution.
- Step j By genetic manipulation to produce a new generation of population, and return to step c, until meet the

convergence criteria.

2) The Introduction of The Method Used To Solve The Problem:

The Method Of Coding: In this article we use binary code [7] method to encode the individual. Every individual is a solution of 0-1 knapsack problems, and each individual's binary sequence expresses whether the corresponding items are selected or not.

Keep The Elite Individual: Elite individual refers to individual who has a high degree of individual adaptation. Retaining elite individuals can ensure high fitness individual information be saved, which ensures better and faster convergence of genetic algorithms to the optimal solution. And because of the randomness of operation, if the strategy of retaining elite individuals is not used [8], it may lead to the result that the new populations of individual have less fitness than the previous generation of population, thus causing the population's regress, and resulting in the loss of genetic information and unnecessary waste, affecting the performance of genetic algorithms. This algorithm used for each species have opened up a new memory units and used to store the best of contemporary individuals.

Operator Selection: This paper uses the proportional selection operator and the individual selection is determined by roulette wheel pattern. Then by parent offspring fitness rank ordering ,select individuals with higher adaption as a new generation.

Genetic Manipulation: This paper adopts two points' cross-methods. If it meets the condition of cross, it will produce two points at random, and then divide the gene segment [9] into two phases, exchanging one part of it. At the same time set the crossover probability value from 0.5 to 0.95. If the value is too big, the structure of excellence individual is likely to be damaged; if too small, it is easy to cause shrinkage stagnation. Adopting single point mutation. If meets mutation condition, the randomly generated point will be reversed, specifically, 0 becomes 1 and 1 becomes 0. The mutation rate is usually from 0.01 to 0.1.

The Collaborative evolutionary of population: The first thing is to select individuals participating in the exchange of information. In this paper, we first sort the individuals based on fitness, then randomly selected the first n excellent individuals for comparison. We reserve the individuals with higher adaptability and eliminate the low adaptability individual, and randomly generate new individuals to replace the obsoleted ones. At the same time, we chose the high degree of adaptability to replace the low degree of adaptability [3]. In this way the collaboration between the evolution of populations and the exchange of information can be achieved [2]. Then the next round of iteration begins.

Sub-group Competition: The first is to produce genetically split point, and then the sequence is divided into paragraphs. This paper uses two gene-secession law that divide the gene sequence into three sections, then the three

gene segments are exchanged with the gene segment in the parents. In this way these new individuals form a subgroup muster, then chose two individuals with high fitness as offspring from the subgroup muster to participate in the populations of other operations. But the individuals not selected in the sub-set are eliminated.

The Final Constraint Validation: It is necessary to test and verify the optimal individual ,in order to assure the satisfaction of kitbag algorithm's constraint conditions. If it doesn't meet the conditions, then re-iteration until satisfied. which ensures that the optimal solution is reasonable.

IV. PERFORMANCE EVALUATION

In evaluation, the length of the chromosome is set to $chrom = 50$; the population size is $POPSIZE = 100$, the maximum number of cycles is set as $MAX = 200$; the crossover probability is set as $p_c = 0.9$; the mutation probability is set as $p_m = 0.1$. The simulations were conducted on an Intel T2300 computer running Windows XP Professional with 896MB of RAM. We use the data comes from [10]. The values, volumes and the capacities of knapsack are as follows:

$$w_i = 220, 208, 198, 192, 180, 180, 165, 162, 160, 158, \\ 155, 130, 125, 122, 120, 118, 115, 110, 105, 101, \\ 100, 100, 98, 96, 95, 90, 88, 82, 80, 77, \\ 75, 73, 72, 70, 69, 66, 65, 63, 60, 58, \\ 56, 50, 30, 20, 15, 10, 8, 5, 3, 1,$$

$$p_i = 80, 82, 85, 70, 72, 70, 66, 50, 55, 25, \\ 50, 55, 40, 48, 50, 32, 22, 60, 30, 32, \\ 40, 38, 35, 32, 25, 28, 30, 22, 50, 30, \\ 45, 30, 60, 50, 20, 65, 20, 25, 30, 10, \\ 20, 25, 15, 10, 10, 10, 4, 4, 2, 1,$$

$$C = 1000,$$

The result is based on 100 experiments.

The results showed that: compared to elitist strategy genetic, the Dual population genetic algorithm is superior in search capabilities, speed, avoiding early maturity and other indicators. The convergence of dual population genetic algorithm will be stabilize around 60 era. The average error is between 0% and 2.7% among the optimal solutions. The maximum error is 2.7%. Thus, The average error of elitist strategy genetic algorithm is from 3.64% to 6.6%, and the maximum error is 6.6%. From Fig. 1(a), we can find that the dual population genetic algorithm's performance is much better and more stable than the elitist strategy genetic algorithm on the same Iterations. From Fig. 1(a) and Fig. 1(c), we can draw the conclusion that the performance of the improved dual population genetic algorithm is better than elitist strategy genetic algorithm's, however

algorithm	optimum / volume	worst value / volume	average value	Average of spending time s
elitist strategy genetic algorithm	2990/1000	2898/1000	2940	0.0516
dual population genetic algorithm	3103/1000	3019/1000	3072	1.19256
Improved dual population genetic algorithm	3093/1000	2967/1000	3054.367	0.756

Table I

THE EXPERIMENTAL RESULTS OF DUAL POPULATION GENETIC ALGORITHM, ELITIST STRATEGY GENETIC ALGORITHM AND THE IMPROVED DUAL POPULATION GENETIC ALGORITHM.

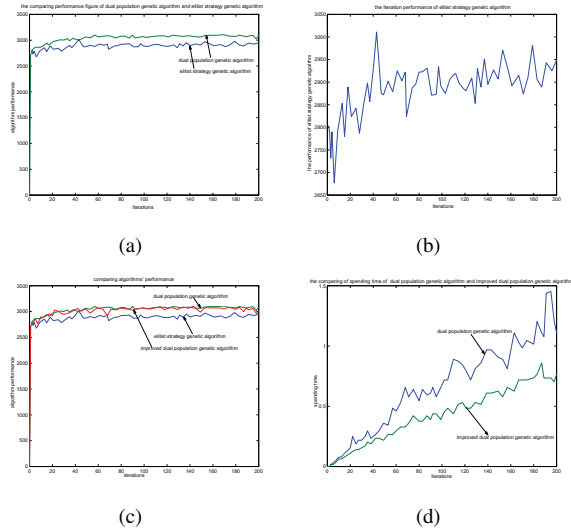


Figure 1. Result curves of genetic algorithms.

its performance and stable is worse than dual population genetic algorithm. Fig. 1(d) indicates that the spending time of improved dual population genetic algorithm is far shorter than elitist strategy genetic algorithm's. Fig. 1(b) shows that the convergence performance of elitist strategy genetic algorithm is not stable, whose reason is that a complete new individual can be generated with the change of iteration amount. Therefore, improved dual population genetic algorithm is better than elitist strategy genetic algorithm and dual population genetic algorithm.

V. CONCLUSION

The improved dual population genetic algorithm can enhance the evolved ability of the population, to some extent, overcome the premature convergence and the occurrence of premature, greatly improve the efficiency of the algorithm and get better solutions, can be a good solution to the 0-1 Knapsack Problems. Multi-population genetic algorithm will also be one of the main direction of genetic algorithm genetic algorithm in the future.

REFERENCES

- [1] C. Grosan and M. Oltean, "Improving the performance of evolutionary algorithms for the multiobjective 0/1 knapsack

problem using e-dominance," in *International Conference on Computational Science*, pp. 674–677, 2004.

- [2] J. Holland, *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence*. Cambridge Mass.: MIT Press, 1st MIT press ed. ed., 1992.
- [3] F. BiHe and Y. LeiLei, "Two-population genetic algorithm based on elimination mechanism," *Operations Research*, 2009.
- [4] L. M. Yang Xiaoqin and Z. Linxia, "Entropy-based dual genetic algorithm," *Pattern Recognition and Artificial Intelligence*, 2005.
- [5] L. Xueyan and Z. Yongquan, "Bee-based dual genetic algorithm," *Computer Engineering and Applications*, 2008.
- [6] L. Sachs, *Applied Statistics: A Handbook of Techniques*. Springer-Verlag New York Berlin Heidelberg Tokyo, 1982.
- [7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms, second edition," 2001.
- [8] X. S. H. AKCAY Y, LI Haijun, "Greedy algorithm for the general multidimensional knapsack problem," *Operations Research*, pp. 17–29, 2007.
- [9] L. Davis, ed., *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, 1991.
- [10] Y. T. S, "An improved genetic algorithm for knapsack problem," *Modern Computer*, pp. 44–47, 2009.