

A Novel Bat algorithm of solving 0-1 Knapsack Problem

Yanfeng Chen^{1,a}

¹Gansu Normal University for Nationalities, Hezuo, gansu, 747000, China

^aYueguangli7@163.com

Keywords: Bat Algorithm; 0-1 knapsack problem; active evolutionary operator; invalid bat

Abstract: 0-1 knapsack problem is a typical NP complete problem. In this paper, based on the bat algorithm, combined with the idea of genetic variation, the paper introduces the processing rules of the active evolutionary operator, invalid bat and the current optimal position. The simulation results show that the algorithm is superior to the basic bat algorithm in convergence speed and accuracy, and it can effectively solve the 0-1 knapsack problem.

1. Introduction

Knapsack problem is a typical combinatorial optimization problem in operations research ^[1]. It is a NP- complete problem, it is proposed by Markel and Hellman in the 50's of the last century ^[2]. The main idea of the knapsack problem is that a person has a lot of goods, the goods weights are not identical, the person must put some goods in the backpack. The weight of the goods is known, all possible goods are known, but the goods in the backpack are confidential, in addition additional backpack weight is limited. For the knapsack problem of large-scale to list all the possible items is not achieved in the calculation. In many types of knapsack problem, 0-1 knapsack problem is the most basic knapsack problems, other knapsack problem often can be converted into solving 0-1 knapsack problem.

Predecessors have done some in-depth research on the knapsack problem, get some classical methods, some methods for the knapsack problem can get very good results, but there are still many problems. First of all, most of the algorithms have big amount of computation, iteration time is long. For example, the exhaustive method and dynamic programming method ^[3], but the efficiency is too low, the robustness is not strong, only suitable for solving a smaller problem, but in practice, due to the corresponding search space can be very large, slowly solution, the efficiency will be very low. Second, the greedy algorithm ^[4], speed is fast, climbing ability is strong, but it is suitable for searching local optimal solution, may fall into local extreme value, not the global optimal solution. Third, the ant colony algorithm ^[5], it can get the approximate optimal solution, but when the data size is large, the convergence is too slow; genetic algorithm ^[6], it can get the approximate optimal solution, but may be premature. Fourth, knowledge evolution algorithm ^[7] and DNA calculation ^[8] can also effectively solve the knapsack problem, but these theories are not perfect.

2. 0-1 knapsack problem

0-1 knapsack problem: given n goods and a backpack. The weight of goods i is w_i , its value is v_i , the capacity of knapsack is c . Asked how to choose the goods of the backpack, so that the total value of goods of the backpack is maximum? The 0-1 knapsack problem can be formalized description: given c or $w_i > 0 \quad v_i > 0 \quad 1 \leq i \leq n$. Asked to find an n -dimensional vector $(x_1, x_2, \dots, x_n), x_i \in [0,1], 1 \leq i \leq n$, which makes $w_1x_1 + w_2x_2 + \dots + w_nx_n \leq c$ is less than or equal to c , and $v_1x_1 + v_2x_2 + \dots + v_nx_n$ reaches the maximum, 0-1 knapsack problem is a special integer planning problem:

$$\max \sum_{i=1}^n v_i x_i \quad s.t. \begin{cases} \sum_{i=1}^n w_i x_i \leq c \\ x_i \in [0,1], 1 \leq i \leq n \end{cases}$$

3. Bat algorithm

The bat algorithm (BA) is a new heuristic algorithm ^[9], it is proposed by XinShe Yang in 2010. Yang illustrates the basic idea of this algorithm and puts forward the basic assumption of the bat algorithm:

- 1) All bats use their echolocation perception and the distance of the target, they identify the target and the different of the background obstacles in a mysterious way.
- 2) The location of bats are x_i , flying with the speed of v_i , they search target at a fixed frequency f , variable wavelength λ and loudness A_i . They can determine the distance between himself and the prey and adjust the pulse wavelength (frequency) automatically, at the same time, when they close to the target, they adjust the pulse frequency $f_i \in [0,1]$.
- 3) There are a lot of changes in loudness, the assumption it is from the maximum value of A_{max} changes to the fixed minimum value of A_{min} .

Since the algorithm is proposed, some scholars have applied this algorithm to solve continuous function optimization problems, and have achieved good results. This paper presents a novel bat algorithm to solve the 0-1 knapsack problem, the simulation results show that the algorithm has better convergence speed and precision, and it can effectively solve the 0-1 knapsack problem.

4. The novel bat algorithm

In order to lead the current optimal bat to produce adaptive variation of the environment, the mutation factor is introduced to design the active evolutionary operator. For the 0-1 knapsack problem, the mutation factor is composed of $m_1 m_2 \dots m_d$, m_i expresses the probability of goods i is selected in a backpack in the basis of the prior knowledge. This probability is determined by the following two factors:

- (1) Per unit of weight value of goods i is avg_value_i, $avg_value_i = p_i / w_i$.
- (2) The ratio of weight of the goods i and backpack capacity is w_i / C .

According to the mutation factor, we can make the active evolution of the current optimal bats: the position of each bat is defined as a gene. When the evolution probability of this gene m_i is greater than random number, the bit reverse, thus generating an optimal solution of the bat, and compares it with the previous optimal bat, get better as the optimal bat.

If the total weight of a bat represents the selected item exceeds the maximum weight capacity of the backpack, it is called an invalid bat. For invalid bats, the gene of the bat can be judged according to the calculated mutation factor is 1 or 0. If the mutation probability m_i is less than that of the random number, the gene position is reversed. After the mutation, then determine whether the bat is valid, if it is still an invalid bat, continue to repeat the previous operation, until the bat becomes an effective bat.

In this paper, a new hybrid intelligent optimization algorithm----genetic variation bat algorithm is proposed. Assuming the bat population was n , the position of i $x(i) = (x_{i1}, x_{i2}, \dots, x_{id})$, where $i = 1, 2, \dots, n$. The basic steps of genetic variation bat algorithm can be summarized as follows:

The position of i bat in the initial population: $x(i) = round(rand(1, d))$, determine each bat of the population is effective, if it is invalid bat, until it becomes effective bat, initial speed $v(i)$, pulse rate $r(i)$, , loudness pulse $A(i)$, pulse frequency $F(i)$ and individual evaluation fitness $(i) = Z(x(i))$ ($i = 1, 2, \dots, n$).

Use the formula $q_i = avg_value^{-vc_i}$ calculation mutation factor

While (not meeting the end condition)

Use the formula $m_i = (q_i - \min\{q_i\}) / (\max\{q_i\} - \min\{q_i\})$ calculation mutation factor

for $i = 1:n$
Using formula $f_i = f_{\min} + (f_{\max} - f_{\min})\beta$, $v_i^{t+1} = v_i^t + (x_i^t - x_*)f_i$, $x_i^{t+1} = x_i^t + v_i^{t+1}$ to generate the new x_{new1}
if Rand > R(i)
Use $A_i^{t+1} = \alpha A_i^t$ to generate the new x_{new1}
Endif
Judge the bat whether is effective bat and deal with it
 $Z_{\text{new}} = Z(x_{\text{new}})$
If $\text{rand} < A(i) \& \& Z_{\text{new}} > \text{fitness}(I)$
 $x(i) = x_{\text{new}}$
 $\text{fitness}(i) = f_{\text{new}}$
Update $R(i)$ and $A(i)$ by $R_i^{t+1} = R_i^t * [1 - \exp(-\gamma t)]$.
Endif
Endfor
Update the current best solution x_* and its corresponding parameters
Active evolutionary strategy for the current optimal solution
Endwhile

5. The analysis of simulation

Run environment of algorithm is *MATLAB (R2010b)* under *Win7* operating system. Parameters: $f_{\min} = 0$, $f_{\max} = 1$, $\alpha = 0.95$, $\gamma = 0.9$, the lower bound of speed $v_{\min} = -0.5$, the upper bound of speed $v_{\max} = 0.5$, the initial velocity $v = 0$, the initial frequency $f = 0$, the initial loudness $A = 0.5$ and initial pulse frequency $R = 0.2$.

Goods number D=50, maximum weight limit C=11 258.

$v = (72, 490, 651, 833, 883, 489, 359, 337, 267, 441, 70, 934, 467, 661, 220, 329, 440, 774, 595, 98, 424, 37, 807, 320, 501, 309, 834, 851, 34, 459, 111, 253, 159, 858, 793, 145, 651, 856, 400, 285, 405, 95, 391, 19, 96, 273, 152, 473, 448, 231)$
 $w = (438, 754, 699, 587, 789, 912, 819, 347, 511, 287, 541, 784, 676, 198, 572, 914, 988, 4, 355, 569, 144, 272, 531, 556, 741, 489, 321, 84, 194, 483, 205, 607, 399, 747, 118, 651, 806, 9, 607, 121, 370, 999, 494, 743, 967, 718, 397, 589, 193, 369)$

In order to verify the accuracy of the algorithm, the above examples are run 50 times and the results are shown in Table 1. "Ratio" refers to the ratio of the best value of the algorithm, "-inf" indicates that the bat is not effective in the initialization of the bat is effective, all bats are invalid in the specified number of cycles. As can be seen from table 1, the genetic variation bat algorithm in this paper is better than the basic bat algorithm in precision and stability, it can effectively solve the 0-1 knapsack problem.

Table1 Optimization results of novel bat algorithm and the traditional bat algorithm

Algorithm	The number of population	Maximum number of iterations	Worst value	Average value	Optimal value	Ratio
The algorithm of the paper	100	600	13636	14667	16055	0.9971
Traditional bat algorithm	100	600	10269	12954	16005	0.9940

6. Conclusions

In this paper, the mutation strategy is introduced into the mutation strategy, and the active evolution operator is designed, which not only realizes the active evolution of bats, but also enriches the diversity of the population. The genetic variation bat algorithm is superior to the basic bat algorithm in the convergence speed and accuracy, and it has a certain feasibility and better search ability.

Acknowledgment

This work is supported by Gansu Provincial College graduate tutor of scientific research project (GSGXGO30) and Dean Fund of Gansu Normal College for Nationalities (15-07).

References

- [1] Syslo M M. Discrete Optimization Algorithms [M]. Englewood Cliffs, New Jersey: Prentice-Hall, 1983(07):118-165.
- [2] Ralph Merkle and Martin Hellman. Hiding Information and Signatures in Trapdoor Knapsacks[J]. IEEE Transactions on Information Theory, 1978, 24(5): 525–530.
- [3] Cao Zhoujin. An algorithm of exhaustive, search and dynamic programming about 0-1 knapsack problem [J]. artificial intelligence and recognition.2009,5 (12): 3193-3194.
- [4] You Wei. The algorithm research of 0/1 knapsack problem based on greedy strategy[J]. Computer and modernization . 2007 (4): 10-16.
- [5] Liu Huaying, Lin Yue, Liu Jinyue. An ant colony algorithm for solving 0/1knapsack problem[J]. Journal of Daqing Petroleum Institute .2005, 29 (3): 59-63.
- [6] Han Yu. Hybrid genetic algorithm for solving knapsack problem[D]. Beijing: Beijing University of Posts and Telecommunications, 2009
- [7] Ma Huimin, Ye Chunming, Zhang Shuang. Knowledge evolution algorithm for knapsack problem[J]. Computer engineering. 2009, 35 (6):208-212.
- [8] Shi Xiaolong, Xu Jin. DNA calculation and knapsack problem[J]. Computer engineering and application, 2003 (27): 44-52.
- [9] Xinshe Yang. A new meta heuristic bat-inspired Algorithm [J], Nature Inspired Cooperative Strategies for Optimization (NICSO2010)(Eds. J. R. Gonzalez et al.), SCI 284, 2010:65-74.