

# 求解 0-1 背包问题的改进混合遗传算法<sup>①</sup>

刘寒冰, 张亚娟

(黄河科技学院 信息工程学院, 郑州 450063)

**摘 要:** 针对一种混合遗传算法所采用的贪心变换法的不足, 给出了一种改进的贪心修正法; 并基于稳态复制的策略, 对遗传算法的选择操作进行改进, 给出了随机选择操作. 在此基础上, 提出了一种改进的混合遗传算法, 并将新算法用于解决大规模的 0-1 背包问题, 通过实例将新算法与 HGA 算法进行实验对比分析, 并研究了变异概率对新算法性能的影响. 实验结果表明新算法收敛速度快, 寻优能力强.

**关键词:** 混合遗传算法; 0-1 背包问题; 贪心变换; 随机选择; 贪心修正

## Improved Hybrid Genetic Algorithm for Solving 0-1 Knapsack Problem

LIU Han-Bing, ZHANG Ya-Juan

(School of Information Engineering, Huanghe College of Science and Technology, Zhengzhou 45006, China)

**Abstract:** An improved greedy correction method is advanced for overcome the flaw of greedy transform method adopted by hybrid genetic algorithm (HGA). And based on steady state reproduction strategy, the choice method of random selection is advanced. These new methods are combined with genetic algorithm to propose a high-efficient hybrid genetic algorithm (IHGA), and new algorithm was used to solve large-scale 0-1 knapsack problem. By many simulation experiments, IHGA algorithm is compared with HGA algorithm, and how the mutation probability affect the performance of the new algorithm has been studied. The experimental results show that the new algorithm has higher convergent speed and better optimization capability.

**Key words:** hybrid genetic algorithm; 0-1 knapsack problem; greedy transform; random selection; greedy correction

0-1 背包问题属于组合优化理论中最重要的问题之一, 是计算科学中的一个经典 NP 问题, 也是算法界最典型的一个实例. 求解该问题的算法很多, 主要分为精确算法和近似算法两大类. 精确算法主要有分支限界法、回溯法和动态规划算法等<sup>[1]</sup>. 精确算法求解 0-1 背包问题时, 是先找出问题的所有候选解, 然后按一定的方式在候选解中找出满足约束条件的最优解. 这类精确算法能找到问题的最优解, 具有较高的精确度, 但随着待选择物品数量和背包容量的增长, 算法的时间复杂度对于输入是指数级的, 计算效率较低. 因此, 对该问题进行研究的一些学者提出了一些近似算法来解决该问题, 如贪心算法<sup>[1]</sup>、遗传算法<sup>[2-5]</sup>、微粒群算法<sup>[6]</sup>、蚁群算法<sup>[7]</sup>等, 这些算法虽然不能保证求

出问题的最优解, 但能在可接受的时间内求出最优解的近似解.

遗传算法是模拟生物在自然环境中的遗传和进化过程而形成的一种自适应全局优化概率的近似算法. 该算法通过对生物遗传和进化过程中选择、交叉、变异机理的模仿, 来完成对问题最优解的自适应搜索过程<sup>[2]</sup>. 该算法具有全局优化性、通用性、并行性、稳健性和简单性等特点. 近年来, 将遗传算法用于解决 0-1 背包问题已取得良好进展, 但处理后期收敛速度、局部的寻优能力方面仍无法达到理想的效果.

贪心算法是一种用来解决较复杂的最优化问题的近似算法. 该算法解决问题时是从问题的某一个初始解出发逐步逼近给定的目标, 以尽可能快的速度求得

<sup>①</sup> 基金项目: 郑州市重点实验室资助项目(121PYFZX177)

收稿时间: 2014-10-17; 收到修改稿时间: 2014-11-28

更好的解. 因此, 该算法具有设计简单、复杂度低、运行速度快等特点. 但该算法并不从整体最优解考虑, 它所做出的选择只是在某种意义上的局部最优选择, 因此, 用贪心算法解决 0-1 背包问题, 求解的效率较高, 但不能保证得到最优解.

文献[8]中将贪心算法与遗传算法相结合形成混合遗传算法(Hybrid Genetic Algorithm, HGA). 在 HGA 中应用贪心算法, 是为了把经过交叉、变异后得到的不满足约束条件的染色体变成满足约束条件的染色体, 这样能提高种群的多样性和求解效率, 但后期的收敛速度和寻优能力并没有明显改进.

为此, 本文在 HGA 算法的基础上, 提出了一种求解 0-1 背包问题的改进混合遗传算法, 该算法采用新的贪心修正法对染色体进行修正, 同时对选择操作进行改进, 并通过实例进行仿真实验, 实验表明该算法具有较高的收敛速度和寻优能力.

## 1 相关研究

### 1.1 0-1 背包问题

给定  $n$  种物品和一个背包. 物品  $i$  的重量是  $w_i$ , 其价值为  $v_i$ , 背包的容量为  $C$ . 问应如何选择装入背包的物品, 使得装入背包中物品的总价值最大? 在选择物品装入背包时, 对每物品只有两种选择, 不装和全装. 此问题的数学描述是:

$$\begin{aligned} \max \quad & \sum_{i=1}^n x_i v_i \\ \text{s.t.} \quad & \begin{cases} \sum_{i=1}^n x_i w_i \leq C \\ x_i \in \{0,1\}, i=1,2,\dots,n \end{cases} \end{aligned} \quad (1)$$

其中,  $x_i$  代表第  $i$  种物品的选择情况,  $x_i=1$  代表第  $i$  种物品装入背包,  $x_i=0$  代表第  $i$  种物品不装入背包. 公式(1)是价值和表达式, 公式(2)是约束条件.

### 1.2 染色体的修正

在解决 0-1 背包问题的遗传算法中, 将随机产生的初始种群中的染色体和经过交叉、变异操作后得到的新染色体分为三类: 不满足约束条件的染色体、满足约束条件但背包资源利用不足的染色体、满足约束条件并且背包资源充分利用的染色体. 文献[8]中提出的混合遗传算法(HGA)中应用贪心算法, 是为了把经过交叉、变异操作后得到的不满足约束条件的染色体通过贪心变换操作修正为满足约束条件的染色体. 贪

心变换的基本思想: 对于不满足约束条件的染色体, 将该染色体中所表示的已选中的物品按单位重量价值比  $v_i/w_i$  从大到小的次序进行排序, 然后根据排列次序将物品装入背包, 直到背包无法装入为止, 将剩余的未装入背包的物品状态由“装”改变为“不装”. 分析 HGA 算法中贪心变换操作发现存在不足: 在贪心变换方法中只能对不满足约束条件染色体进行修正使其转化为满足约束条件的染色体, 但修正后的染色体不一定是充分利用背包资源的染色体, 因而就会影响算法的收敛速度和寻优能力. 因此, 本文中对不满足约束条件的染色体和满足约束条件但背包资源利用不足的染色体设计了贪心修正法进行修正, 使其转化为背包资源充分利用的染色体, 进而提高算法的收敛速度和寻优能力.

### 1.3 子代的选择

对文献[9]中提到的稳态繁殖的选择策略进行研究. 稳态繁殖的选择策略的基本思想是: 在每个繁殖代, 将通过交叉、变异和贪心变换操作后得到的子代染色体, 按比例  $t(0 < t < 1)$  选择最优的部分染色体替换父代中较弱的染色体. 这种选择方式能保留最优染色体也能保持种群的多样性, 但是也存在不足, 主要是替换比例  $t$  的选择问题,  $t$  的值越大, 算法收敛的速度就越快, 就会引起染色体早熟. 针对这一不足, 本文采用随机选择策略, 每代中替换率的选择是根据父代和子代中染色体的适应度而随机确定的, 每次选择时替换率都是变化的, 只在子代染色体的适应度值高于父代染色体的情况下, 子代染色体才可以替换掉父代染色体, 否则父代染色体仍保留在下一代群体中, 故它既能有效地维持种群多样性, 又能避免因替代率太高而引起染色体早熟现象.

### 1.4 迭代终止条件

由于该算法采用的是随机选择法, 新生种群中的染色体是按适应度的降序进行排列的, 所以当新生种群中第一个染色体的适应度与最后一个染色体的适应度相等时, 表示种群中所有染色体的适应度都相等. 繁殖过程就可以结束. 因此, 新算法迭代终止条件是: 新生种群中第一个染色体的适应度与最后一个染色体的适应度相等.

## 2 算法思想及描述

本文提出了一种改进的混合遗传算法 IHGA

(Improved Hybrid Genetic Algorithm). 该算法首先对初始种群中的染色体和通过交叉、变异操作后得到的不满足约束条件的染色体和满足约束条件但背包资源利用不足的染色体采用贪心修正法进行修正, 使其转化为为背包资源充分利用的染色体, 然后对父代染色体和子代染色体进行随机选择产生下一代种群. 算法的流程图如图 1 所示.

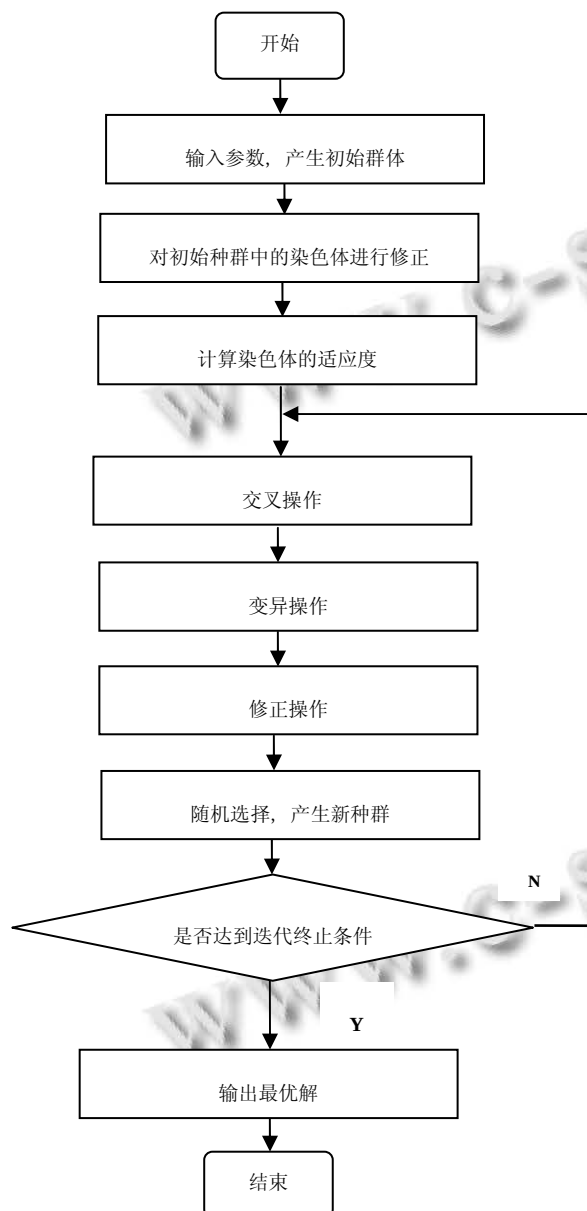


图 1 算法流程图

IHGA 算法解决 0-1 背包问题的主要步骤如下:

输入:  $n$  种物品的价值  $V=(v_1, v_2, \dots, v_n)$ , 重量  $W=(w_1, w_2, \dots, w_n)$ , 背包的容量  $C$ , 交叉概率  $PC$ , 变异

概率  $PM$ , 种群规模  $M$ .

输出: 根据 0-1 背包问题的数学表达式, 则输出的最优解是一个  $n$  元 0-1 向量  $(x_1, x_2, \dots, x_n)$ .

### 1) 种群初始化

采用二进制编码. 随机生成  $M$  个长度为  $n$  的“0”, “1”编码的串结构数据. 每个串结构数据代表一个染色体,  $M$  个染色体构成了一个群体, 称为初始种群, 从初始种群进行迭代.

### 2) 染色体修正

对初始种群中的染色体进行修正, 修正主要分为两步:

第 1 步: 修正不满足约束条件的染色体. 对于该部分染色体, 用 HGA 算法中的贪心变换法进行修正, 使其转换为满足约束条件的染色体, 如果转换后的染色体是背包资源充分利用的染色体, 则修正结束, 否则转为第 2 步.

第 2 步: 修正满足约束条件但背包资源利用不足的染色体. 对于该部分染色体, 用贪心修正法修正.

设  $X=(x_1, x_2, \dots, x_n)$  是一个背包资源利用不足的染色体, 该染色体代表的选择是有  $r$  个物品未装入背包, 背包剩余容量为  $C_0$ , 通过贪心修正法将其修正为充分利用背包资源的染色体  $Y=(y_1, y_2, \dots, y_n)$ .

贪心修正法的基本思想: 对染色体  $X$  中未装入背包中的物品按单位重量价值比的降序进行排序, 然后按排序的次序将相应的物品装入背包, 直到无法再装入为止.

实现的步骤如下:

排序. 对  $X$  中未装入背包中的物品按单位重量价值比  $v/w$  从大到小对物品的序号排序, 形成长度为  $r$  的序列  $\{b_1, b_2, \dots, b_r\}$  (即  $b_1$  为单位重量价值比最大物品的序号,  $b_r$  为单位重量价值比最小物品的序号).

优化.

① 令  $k=1$ ,  $W_0=0$ ,  $Y=X$ ;

② 如果  $W_0 + w_{b_k} \leq C_0$ , 则置  $y_{b_k}=1$ ,  $W_0=W_0+w_{b_k}$ ;

③  $k=k+1$ , 如果  $k \leq r$  转②, 否则, 结束优化.

### 3) 计算个体适应度

染色体对环境的适应程度叫做适应度. 为了体现染色体的适应能力, 引入了对问题中的每一个染色体都能进行度量的函数, 叫适应度函数. 本算法中适应度函数为:

$$f(x) = \sum_{i=1}^n v_i x_i \quad (3)$$

#### 4)交叉

用赌轮法随机取两个染色体,依照概率  $PC$  选择个体进行交叉操作,采用单点交叉,将被选择出的两个个体  $S1$  和  $S2$  作为父母个体,随机产生一个在 1 到  $n$  之间的数  $c$ ,将  $S1$  和  $S2$  的低  $c$  位交换,产生新个体  $P1$  和  $P2$ .

#### 5)变异

依照  $PM$  对繁殖个体进行变异操作,采用基位变异,将个体的基因链的各位按概率  $PM$  进行异向转化,“0”变异为“1”,“1”变异为“0”.

#### 6)选择

本文中采用随机选择法.在每一个繁殖代,将通过交叉、变异和贪心变换操作后得到的子代染色体加入到父代染色体中形成一个新的群体,对新群体中的个体按适应度的降序进行排序,按排好序的次序从中选择前  $M$  个染色体作为新生种群.这种选择策略中替换率  $t$  不是常量,而是根据父代和子代中染色体的适应度随机变化的变量,因而能解决稳态繁殖的选择策略中存在的不足.另外,新生代的产生是从子、父两代染色体群体中选择较优的,能保证新生种群中染色体的最优性和多样性,因而能提高算法收敛速度.

用数组 `oldpop` 存储当前要进行繁殖操作的种群,数组 `oldpop` 中的个体是有序的,排序标准是根据个体的适应度进行降序排列,用数组 `newpop` 存储由 `oldpop` 通过交叉、变异和贪心变换操作后产生的子代染色体.

随机选择操作实现步骤如下:

① 对数组 `newpop` 中的个体按适应度的降序进行排序;

② 将数组 `oldpop` 中的个体与数组 `newpop` 中的个体按适应度从大到小进行合并排序,将排好序的个体保存到数组 `tmp` 中;

③ 取数组 `tmp` 中的前  $M$  个个体,形成下一代种群.

### 3 仿真实验

为了测试 IHGA 算法的性能,并研究变异概率的取值对 IHGA 性能的影响.本文对 IHGA 算法和 HGA 算法进行仿真实验.实验中用到实例为 0-1 背包问题的经典实例,实例选自文献[3],两个实例的实验次数均为 50 次.实验参数:  $n$  代表物品个数,  $C$  代表背包容

量,  $J$  代表 50 次实验中得到的最满意值(背包中所装物品价值和的最大值),  $JC$  代表满意值在 50 次实验中出现的次数,  $JZ$  代表繁殖代数的均值.

实例 1 如下:

$n=50, C=1000$

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

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

实例 2 如下:

$n=100, C=6178$

$V=\{597,596,593,586,581,568,567,560,549,548,547,529,529,527,520,491,482,478,475,475,466,462,459,458,454,451,449,443,442,421,410,409,395,394,390,377,375,366,361,347,334,322,315,313,311,309,296,295,294,289,285,279,277,276,272,248,246,245,238,237,232,231,230,225,192,184,183,176,174,171,169,165,165,154,153,150,149,147,143,140,138,134,132,127,124,123,114,111,104,89,74,63,62,58,55,48,27,22,12,6\}$

$W=\{54,183,106,82,30,58,71,166,117,190,90,191,205,128,110,89,63,6,140,86,30,91,156,31,70,199,142,98,178,16,140,31,24,197,101,73,169,73,92,159,71,102,144,151,27,131,209,164,177,177,129,146,17,53,164,146,43,170,180,171,130,183,5,113,207,57,13,163,20,63,12,24,9,42,6,109,170,108,46,69,43,175,81,5,34,146,148,114,160,174,156,82,47,126,102,83,58,34,21,14\}$

#### 3.1 IHGA 算法与 HGA 算法的性能比较

实验中用到的交叉概率  $PC$  和变异概率  $PM$  分别为 0.5 和 0.01. 实验结果如表 1 所示.

表 1 实验结果

| 参数   | 实例 1( $n=50$ ) |      | 实例 2( $n=100$ ) |       |
|------|----------------|------|-----------------|-------|
|      | HGA            | IHGA | HGA             | IHGA  |
| $M$  | 200            | 200  | 200             | 200   |
| $J$  | 3119           | 3119 | 26499           | 26553 |
| $JC$ | 3              | 31   | 4               | 39    |
| $JZ$ | 300            | 31.8 | 500             | 63.5  |

从表 1 的实验数据可以看出,在实例 1 中算法 HGA 和 IHGA 都能得到问题的最优值,在实例 2 中,

IHGA 算法能求出问题的最优值, 而 HGA 算法仅能求出最优值的近似值, 这说明 IHGA 算法的寻优能力较强; 在 50 次实验中, IHGA 算法出现的最优值次数明显高于 HGA 算法, 这说明 IHGA 算法的求解效率较高; 实例 1 中 HGA 算法设置的繁殖代数 300, 而 IHGA 算法繁殖代数的均值仅为 31.8, 实例 2 HGA 算法设置的繁殖代数为 500, 而 IHGA 算法的平均繁殖代数仅为 63.5, 这说明采用贪心修正法和随机选择策略能提高 IHGA 算法的收敛速度和寻优效率. 从实例 1 和 2 的平均繁殖代数可以看出问题规模越大 IHGA 算法的寻优能力越强.

### 3.2 不同变异概率下 IHGA 算法的计算性能

为研究变异概率  $PM$  对 IHGA 算法的影响, 在交叉概率不变的情况下, 变异概率  $PM$  取 5 个不同值 (0.005, 0.01, 0.02, 0.05, 0.10), 每种取值对实例 1 分别进行 50 次仿真实验, 多次实验得到的最满意值都为最优值 3119, 满意值出现次数  $JC$  随变异概率变化曲线如图 2 所示, 繁殖代数均值  $JZ$  随变异概率变化曲线如图 3 所示, 图 2、3 中纵坐标为变异概率.

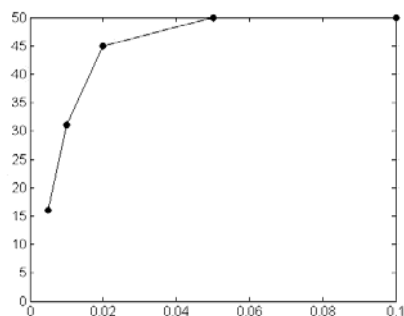


图 2 满意值出现次数变化曲线

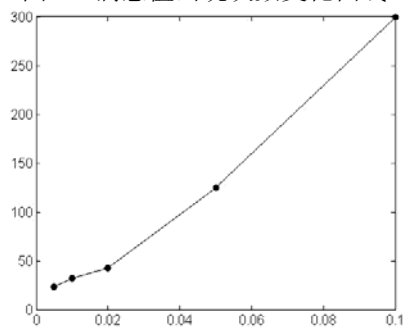


图 3 繁殖代数均值变化曲线

从图 2 可以看出变异概率越大, 满意值出现的次数也越多, 但当变异概率增加到 0.05 时, 满意值出现的次数达到最好值 50 次. 从图 3 可以看出, 变异概率

越大, 种群的平均繁殖代数越多, 当变异概率在 0.005 到 0.02 范围内变化时平均繁殖代数变化曲线增长缓慢, 变异概率在 0.05 到 0.1 范围内变化时平均繁殖代数变化曲线增长较快. 这是因为变异概率增大能够增加染色体的多样性, 提高算法的寻优能力, 但算法的收敛速度反而降低. 因此, 变异概率对算法的影响较大, 取值在 0.01 和 0.05 之间较好.

### 4 结语

本文提出了一种解决 0-1 背包问题的混合遗传算法 IHGA, 该算法通过用贪心修正法对非优染色体进行修正和优化, 提高种群中染色体的质量, 进而提高算法的寻优能力; 对经过交叉、变异和修正操作后得到的子代染色体和父代染色体进行随机选择操作得到新生种群, 该操作既能保留父代中的最优染色体也能保持新生种群的多样性, 进而提高算法收敛速度. 用实例对算法进行仿真实验, 实验结果表明 IHGA 算法具有高效性和稳定性, 能够胜任大规模 0-1 背包问题的求解.

### 参考文献

- 1 王晓东. 计算机算法设计与分析. 第 3 版. 北京: 电子工业出版社, 2008.
- 2 周明, 孙树栋. 遗传算法原理及应用. 北京: 国防工业出版社, 1999.
- 3 吕晓峰, 张勇亮, 马羚. 一种求解 0-1 背包问题的改进遗传算法. 计算机工程与应用, 2011, 47(34): 44-46.
- 4 王秋芬, 梁道雷. 一种求解 0-1 背包问题的启发式遗传算法. 计算机应用与软件, 2013, 30(2): 33-37.
- 5 Singh, RP. Solving 0-1 Knapsack problem using Genetic Algorithms. Proc. of the 3rd IEEE International Conference on Communication Software and Networks. Xi'an, China. 2011. 591-595.
- 6 Rana S, Jasola S, Kumar R. A review on particle swarm optimization algorithms and their applications to data clustering. Artificial Intelligence Review, 2011, 35: 211-222.
- 7 何小锋, 马良. 求解 0-1 背包问题的量子蚁群算法. 计算机工程与应用, 2011, 47(16): 29-31.
- 8 徐宗本. 计算智能—模拟进化计算. 北京: 高等教育出版社, 2005.
- 9 董鹏. 求解 0/1 背包问题的快速收敛的混合遗传算法. 计算机工程与应用, 2008, 44(30): 47-49.