

基于改进模拟退火的遗传算法求解 0—1 背包问题

张盛意, 蔡之华, 占志刚

(中国地质大学(武汉) 计算机学院, 武汉 湖北 430074)

摘 要: 引入改进的模拟退火思想来改进遗传算法. 本算法结合了遗传算法和模拟退火算法的优点, 并有效地克服了各自的弱点, 使其在优化性能、优化效率和可靠性方面具有明显的优越性. 运用本算法求解不同种群规模的 0—1 背包问题, 数值试验结果表明, 算法既具有较快的收敛速度, 又能够收敛到最优解, 优于遗传算法和模拟退火算法.

关键词: 0—1 背包; 遗传算法; 模拟退火

中图分类号: TP31

文献标识码: A

文章编号: 1000—7180(2011)02—0061—04

Solving 0—1 Knapsack Problem Based on Genetic Algorithm with Improved Simulated Annealing

ZHANG Sheng-yi, CAI Zhi-hua, ZHAN Zhi-gang

(School of Computer, China University of Geoscience, Wuhan 430074, China)

Abstract: The paper brings in simulated annealing to improve the performance of genetic algorithms. The algorithm combines the advantages and avoids the disadvantages of genetic algorithm and simulated annealing algorithm. It has superiority in performance, efficiency and reliability. This algorithm is used to solve 0-1 knapsack problem on different scaled datasets, and the results is better than either genetic algorithm or simulated annealing algorithm.

Key words: 0—1 knapsack problem; Genetic Algorithms; Simulated Annealing

1 引言

遗传算法与模拟退火算法是最近几年用于优化问题的两种智能算法. 遗传算法采用了生物进化论的思想, 通过自然选择和适者生存的竞争策略来求解优化问题. 虽然遗传算法有较强的全局搜索性能, 但它在实际应用中容易产生早熟收敛的问题, 而且在进化后期搜索效率较低^[1]. 模拟退火算法是模拟热力学中物理淬火过程的一种学习规则, 该算法既能向目标函数优化的方向迭代, 又能以一定的概率接受目标函数劣化的情况, 从而避免了陷入局部最优解, 保证获得全局最优解的可靠性, 但收敛速度较慢^[2]. 本文通过将模拟退火的思想引入遗传算法, 有效地缓解了遗传算法的选择压力, 并根据 0—1 背包问题的实际情况对交叉和变异算子进行了设计, 不但增强了遗传算法的全局收敛性, 并且使算法在进

化后期有较强的爬山性能, 加快了进化后期的收敛速度.

2 问题描述

背包问题(knapsack problem)的一般提法是: 已知 n 个物品的重量(weight)及其价值(或收益 profit)分别为 $W_i > 0$ 和 $P_i > 0$, 背包的容量(contain)为 $C > 0$, 选择哪些物品装入背包可以使得在背包的容量约束限制之内所装物品的价值最大呢?

该问题的模型可以表示为下述 0/1 整数规划模型.

目标函数:

$$\max f(x_1, x_2, \dots, x_n) = \sum_{i=1}^n p_i x_i$$

$$\text{约束条件:} \begin{cases} \sum_{i=1}^n w_i x_i \leq C \\ x_i \in \{0, 1\} \quad (i = 1, 2, \dots, n) \end{cases}$$

式中, X_i 为 0-1 决策变量, $X_i = 1$ 时表示将物品 i 装入背包中, $X_i = 0$ 时则表示不将其装入背包中^[3].

3 模拟退火算法简介

模拟退火^[4]算法是一种有效的全局优化算法,在模拟退火算法中,适当地控制温度的下降过程实现模拟退火,从而得到全局的最优解.

模拟退火算法来源于固体退火原理,将固体加温至充分高,再让其徐徐冷却,加温时,固体内部粒子随温度升高变为无序状,内能增大,而徐徐冷却时粒子渐趋有序,在每个温度都达到平衡态,最后在常温时达到基态,内能减为最小^[5]. 基本流程如下:

(1) 初始化:初始温度 T (充分大),初始解状态 S (是算法迭代的起点),每个 T 值的迭代次数 L ;

(2) 对 $k = 1, \dots, L$ 做(3) ~ (6) 步;

(3) 产生新解 S' ;

(4) 计算增量 $\Delta t' = C(S') - C(S)$, 其中 $C(S)$ 为评价函数;

(5) 若 $\Delta t' < 0$ 则接受 S' 作为新的当前解,否则以概率 $\exp(-\Delta t'/T)$ 接受 S' 作为新的当前解;

(6) 如果满足终止条件则输出当前解作为最优解,结束程序. 终止条件通常取为连续若干个新解都没有被接受时终止算法;

(7) T 逐渐减少,且 $T \rightarrow N$ (N 一般取足够小的常量),然后转(2).

4 遗传算法简介

遗传算法是将问题的每一个可能性解看作是群体中的一个个体(染色体),并将每一个染色体编码成串的形式,再根据预定的目标函数对每个个体进行评价,给出一个适应值. 算法将根据适应度值进行寻优过程,遗传算法的寻优过程是通过选择、杂交和变异三个遗传操作来具体实现的.

搜索能力由选择算子和杂交算子决定,变异算子则保证了算法能够搜索到问题空间的尽可能多的点,从而使其具有搜索全局最优的能力. 在遗传算法中,定义长度较短、低阶且适应值超过平均适应值的模式在群体中数目的期望值按指数递增,这个结论称为遗传算法的基本定理. 遗传算法是通过定义长度短、确定位数少、适应度值高的模式的反复抽样、组合来寻找最佳点,称这些使遗传算法有效工作的

模式为积木块,是遗传算法构造答案的基本材料. 但归根到底,要使遗传算法有效工作必须按照遗传算法的模式定理(或积木块假设)根据具体问题设计合理编码方案^[6].

在运行遗传算法程序时,需要对一些参数作事先选择,它们包括种群的大小、染色体长、杂交率、变异率、最大进化代数等,这些参数对 GA 的性能都有很重要的影响.

5 改进算法求解过程描述

基于背包问题的模型,我们设计了相对应的染色体编码方法:将待求解的各量 X 表示成长为 n (n 为待装入背包的物品数)的二进制字符串 $X[i]$ ($i = 1, 2, \dots, n$). $X[i] = 0$ 表示物品 i 不放入背包内, $X[i] = 1$ 表示物品 i 放入背包内. 例如:101010100...00101 代表一个解,它表示将第 1, 3, 5, 7, ..., $n-2$, n 号物体放入背包中,其他的物体则不放入. 本算法定义背包容量和物品个数分别为: $\text{contain} = 1000$, $\text{lchrom} = 50$.

5.1 设计

在这个过程中,本文设计了不同的编码方式(操作算子)来验证算法的性能.

(1) 初始化种群:从 knapin. txt 中读取背包问题的相关信息并将解空间的数据进行二进制编码,转换为遗传空间的基因型串(即染色体)结构数据,如本算法以 50 个物品为例: $\text{oldpop}[i]. \text{chrom}[j] = \text{rand}() \% 2$, 则解空间内一个基因串长度为 50 的二进制串;定义整数 lchrom 作为染色体的长度,并且随机产生 popsize 个染色体作为初始种群,在产生这 popsize 个染色体时,本文选取的是 weight 小于背包容量的个体作为种群的成员;

(2) 评估种群中个体适应度:评价函数对种群中的每个染色体(chrom)求得其个体适应度,以基因被选择与否,计算 $\text{pop}[i]. \text{fitness} = \text{cal_fit}(\text{pop}[i]. \text{chrom})$, 按照下列公式计算种群中个体权重与适应度(收益):

$$\text{weight} = \sum_{i=0}^{\text{lchrom}-1} \text{weight}[i] * \text{chrom}[i]$$

$$\text{fitness} = \sum_{i=0}^{\text{lchrom}-1} \text{profit}[i] * \text{chrom}[i]$$

设置适应度的惩罚函数: $\text{pen} * (\text{contain} - \text{weight})$, 降低适应度最小个体的适应度,以减少它被选出的概率^[7], 其中参数 $\text{pen} = 1.5$ (pen 的确定需经过实验测试得到), minpop 为种群中适应度最小个体的索引;

(3) 选择:精英选择,保留父代和子代的最优个体.选择把当前群体中适应度较高的个体按某种规则或者模型遗传到下一代种群中,这里所用的规则是:①(轮盘赌选择)染色体在种群中被选择的可能性与其个体的适应度的大小成正比;②(锦标赛选择)随机从原始种群中选择10个个体(染色体),比较其适应度,从中选择适应度最大个体;

(4) 交叉:采用双亲单子法,定义参数 p_c 作为杂交操作的概率(这里使用了两种:①固定杂交概率 p_c ;②自适应杂交概率 $p_c^{[8]}$),由(3)选择得到的两个个体以概率 p_c 交换各自的部分染色体(这里使用了单点杂交和多点杂交),得到新的两个个体.其中自适应杂交概率的计算以如下公式进行:

$$p_c = \begin{cases} k & f' < f_{avg} \\ k * \frac{f' - f_{avg}}{f_{best} - f_{avg}} & f' > f_{avg} \end{cases}$$

式中, f' 为所得两个父个体的平均适应度, f_{avg} 为种群平均适应度, f_{best} 为种群个体最大适应度, $k = 0.9$;

采用自适应杂交概率在算法前期作用不大,但是在后期,当种群适应度普遍较高时,有利于保持算法的稳定性(当两个父体适应度较高时,相应的杂交概率自适应的变小,从而有利于保留优势个体);

(5) 变异:定义参数 $P_m = 0.05$ 作为变异操作的概率,由(4)得到每个个体中的每个基因值都以概率 P_m 进行变异,这里未采用自适应的变异(由于问题规模不大,这样可能适得其反);

(6) 退火过程:在一个种群进化周期内(退火周期),设定初始退火温度 T 和每个 T 值的迭代次数(相当于种群演化次数) L ;在每一个退火周期内,令 $T = k * T$,进行退火,其中 k 为退火系数,控制着降温的速率,对算法结果的好坏有决定性作用.依据遗传操作得到新种群,然后比较新老种群中各自最大适应度个体,如果 $\maxfitness \geq \text{oldmax}$ (其中 $\maxfitness$ 为新种群中个体最大适应度, oldmax 为老种群中个体最大适应度),则接受新种群作为下一次退火迭代的初始种群,并令新种群的最好个体为当前最好解;如果 $\maxfitness < \text{oldmax}$,则以概率 $\exp \frac{\maxfitness - \text{oldmax}}{T}$ 接受新种群为下一次退火迭代的初始种群,若不能接受,则先以原始种群的最大适应度个体替换新种群的最小适应度个体,然后以新组成的种群作为下一次退火迭代的初始种群.这样做能够很好的保持种群多样性,以克服遗传算法容易陷入局部最优的缺陷.退火过程直到温度下降到一个

预定小的值或者达到既定的退火周期为止.

5.2 算法的具体过程

(1) 读入背包问题信息,以二进制编码初始化种群,评估种群中个体的适应度;

(2) 初始化退火(进化)周期 $C = 1$,进入一次退火过程中:

① 设置退火初始温度 $T = 3000$;

② 初始化退火迭代次数 $L = 100$,对初始化的种群进行遗传操作,按5.1节中的(3)和(4)对种群进行选择 and 杂交操作,然后对新产生种群的每个染色体按5.1节中的(5)进行变异操作;

③ 评估种群的适应度,然后按5.1节中的(6)进行模拟退火操作;

④ 每次退火操作后,都保存到目前为止适应度最佳个体的相关信息,避免退火过程中产生的全局最优个体被遗传退火操作“干掉”;同时令 $L = L - 1$,若 $L > 0$,开始下一次迭代,直到 $L = 0$ 时转到⑤;

⑤ 令 $T = k * T$, $T > 10$ 转到②,开始新一个 T 值的退火循环;否则向下执行;

(3) 保存一个退火周期里的最佳个体信息,令 $C = C + 1$,转①;

(4) 全部退火周期结束后,保留并输出最佳个体.

6 实验

为了能够验证混合算法的有效性和稳定性,本文通过数据实验对改进的混合算法与传统的GA和SA算法进行对比.其中实验数据一部分来源常见的遗传算法文献,另一部分则通过随机产生.

试验环境:CPU—Intel(R) Core(TM)2 Duo 2.0 GHz 内存—1GB;操作系统—Windows Xp;开发程序—VC++ 6.0.

参数设置:种群规模 popsize=200;染色体长度 lchrom=50;初始温度: $T = 3000$;降温系数: $d = 0.8$;退火终止温度:10;每个温度内的迭代次数: num=100.

计算分析过程:分别用遗传算法、模拟退火算法以及本文提出的改进算法对每个数据集都计算10次,保存计算结果.然后针对每个数据集,分析三种算法得到的最优值、最差值、平均值、偏差和改进比例.

① 偏差=(最优值-平均值)/最优值;

② 改进比例=(改进算法的最优值-GA/SA

的最优值)/(GA/SA 的最优值)。

实验结果见表 1。

表 1 实验结果

数据集	算法	最优值	最差值	平均值	偏差	改进比例
Data1	GA	15 909	15 502	15 720.4	1.19%	1.21%
	改进算法	16 102	16 102	16 102	0	NA
	SA	15 955	15 736	15 868.4	0.55%	0.92%
Data2	GA	3 076	3 000	3 034.7	1.34%	0.46%
	改进算法	3 090	3 081	3 086.8	0.10%	NA
	SA	3 082	3 024	3 056.3	1.88%	0.29%
Data3	GA	3 746	3 691	3 708.5	1.47%	5.88%
	改进算法	3 966	3 966	3 966	0	NA
	SA	3 908	3 845	3 892.6	1.61%	1.48%

注: Data1 数据来源于文献[9]; Data2 数据来源于文献[10]。

附: Data3 数据

Weight = { 80, 82, 85, 70, 72, 70, 82, 75, 78, 45, 49, 76, 45, 35, 94, 49, 76, 79, 84, 74, 76, 63, 35, 26, 52, 12, 56, 78, 16, 52, 16, 42, 18, 46, 39, 80, 41, 41, 16, 35, 70, 72, 70, 66, 50, 55, 25, 50, 55, 40 }

Profit = { 200, 208, 198, 192, 180, 180, 168, 176, 182, 168, 187, 138, 184, 154, 168, 175, 198, 184, 158, 148, 174, 135, 126, 156, 123, 145, 164, 145, 134, 164, 134, 174, 102, 149, 134, 156, 172, 164, 101, 154, 192, 180, 180, 165, 162, 160, 158, 155, 130, 125 }

Contain = 1 000

7 结束语

本文算法测试不同规模的数据集都取得了很好的结果。相比 GA 和 SA 而言, 本算法在这些数据集上都有较大的改进。

理论上, 当所谓的退火三原则(初始温度足够高, 降温速度足够慢, 终止温度足够低)满足时, 模拟退火算法以概率 1 达到全局最优解。但是初始温度及降温函数的选取会让算法的性能差异很大, 就像使用遗传算法时使用不同的选择算子和杂交算子对算法的性能影响很大一样。本文用了不同的遗传算子进行实验, 最终确定退火参数。

每一个退火周期相当于一个种群的进化周期, 在进化过程中同时进行遗传操作和退火操作, 控制参数 T 的下降幅度可能导致算法进程迭代次数的增加, 算法进程可以接受更多的变换, 访问更多的邻域, 搜索更大范围的解空间, 返回更高质量的最终

解, 当然也花费更多的 CPU 时间。在模拟退火算法中, 温度下降太快, 可能会丢失极值点; 温度下降过慢, 算法的收敛速度又大为降低, 本文通过多次试验选择了降温系数和初始温度。

由于遗传算法会以极快的速度收敛到局部极优, 而且往往很难跳出局部极优, 所以本文用模拟退火来克服这个问题。模拟退火算法的一个基本思想是跳出局部极优, 直观的结论是在较高的温度下没能跳出局部极优, 则在低的温度下跳出局部最优解的可能性也比较小。所以退火周期越多, 跳出局部极优的可能性越大。

参考文献:

- [1] 王小平, 曹立明. 遗传算法[M]. 西安: 西安交通大学出版社, 2005.
- [2] 邵罕, 孙艳歌, 邹长安. 基于遗传模拟退火算法的 Ad Hoc 网络 Qos 路由优化[J]. 微电子学与计算机, 2009, 26(9): 162-165.
- [3] 余祥宣, 崔国华, 邹海明. 计算机算法基础[M]. 2 版. 武汉: 华中科技大学出版社, 2004.
- [4] 韩小雷. 粒子群—模拟退火融合算法及其在函数优化中的应用[D]. 武汉: 武汉理工大学, 2008.
- [5] 吴志远, 邵震鹤, 吴新余. 遗传退火进化算法[J]. 上海交通大学学报, 1997, 31(12): 69-71.
- [6] 潘正君, 康立山. 演化计算[M]. 北京: 清华大学出版社, 1998.
- [7] 张晶, 翟鹏程, 张本源. 惩罚函数法在遗传算法处理约束问题中的应用[J]. 武汉理工大学学报, 2002, 24(2): 56-59.
- [8] 黄勇军, 武友新, 刘华斌. 应用改进的遗传算法求解 TSP 问题[J]. 计算机工程与设计, 2007, 28(24): 5909-5911.
- [9] 金慧敏, 马良. 遗传退火进化算法在背包问题中的应用[J]. 上海理工大学报, 2004, 26(6): 561-564.
- [10] 张铃, 张钊. 佳点集遗传算法[J]. 计算机学报, 2001, 24(9): 917-922.

作者简介:

张盛意 男, (1985-), 硕士研究生. 研究方向为演化计算、数据挖掘。

蔡子华 男, (1963-), 教授, 博士生导师. 研究方向为数据挖掘、机器学习、演化计算。

占志刚 男, (1985-), 硕士研究生. 研究方向为演化计算、多目标优化。