第 26 卷 第 2 期
2011 年 4 月

天津科技大学学报
Journal of Tianjin University of Science & Technology

Vol.26 No. 2
Apr. 2011

改进的 DNA 遗传算法在指派问题中的应用

李孝忠，任吉栋
（天津科技大学计算机科学与信息工程学院，天津 300222）

摘 要：提出了一种基于优秀基因片段思想的 DNA 遗传算法，将这段基因片段提取出来并将它遗传到后代中，可以加快收敛速度. 给出了 DNA 遗传算法的结构，讨论了选择、交叉和变异算子的具体操作，并将其运用到指派问题最优解的求解中，给出了具体的实现方法. 仿真实验验证了算法的有效性和实用性.

关键词：DNA 计算；遗传算法；指派问题
中图分类号：TP18 **文献标志码**：A **文章编号**：1672-6510(2011)02-0061-04

Improved DNA Genetic Algorithm and Its Application to Assignment Problem

LI Xiao-zhong, REN Ji-dong
(College of Computer Science and Information Engineering, Tianjin University of Science & Technology, Tianjin 300222, China)

Abstract: A DNA genetic algorithm based on the thought of excellent gene was proposed, extracting the excellent gene and inheriting it to future generations can speed up the convergence. The structure of DNA genetic algorithm was given and its specific operations including select, crossover and mutation operator were discussed. The method mentioned above was applied to the optimal solution of assignment problem and the concrete implementation was given. The effectiveness and practicality of the algorithm were verified by computer simulation at last.

Keywords: DNA calculation; genetic algorithm; assignment problem

遗传算法是近年来迅速发展起来的一种全新的随机搜索与优化算法，它具有不依赖于问题模型、易于并行处理、有较强的全局搜索功能、鲁棒性强等特点，适用于处理传统搜索方法难以解决的复杂和非线性问题[1]. 但是，遗传算法也存在早期收敛[2]和微调能力差[3]等不足.

DNA 计算是一种新的计算模式. 1994 年 Adelman 首次用实验显示了 DNA 计算的可能性[4]. 其最大优点是充分利用了 DNA 分子具有海量存储遗传密码以及生化反应的海量并行性. 近年来，一大批科学家投身于 DNA 计算这一新的研究领域，提出了许多组合优化问题的 DNA 计算模型[5].

为了进一步模拟生物的遗传机理和基因调控机理，一些学者提出了基于 DNA 编码的遗传算法[6]，并将其用到组合优化问题的求解当中[7]. DNA 遗传算法既可以提高 DNA 计算的效率，又可以拓宽遗传算法的应用范围. DNA 遗传算法还需要在常规的遗传算法上进行改进. 本文基于优秀基因片段思想提出了选择、交叉和变异操作的新方法来提高进化效率和加快收敛速度.

1 DNA 遗传算法

1.1 DNA 遗传算法的基本概念

1.1.1 DNA

DNA 是在 DNA 计算中起中心作用的分子，是重要的基因物质，携带着生物的遗传信息. DNA 的基本元素是核苷酸，由于化学结构的不同，核苷酸的碱基

收稿日期：2010–11–04；**修回日期**：2011–01–11
基金项目：国家自然科学基金资助项目(61070021)
作者简介：李孝忠（1962—），男，山东人，教授，博士，lixz@tust.edu.cn.

划分为腺嘌呤(A)、鸟嘌呤(G)、胞嘧啶(C)和胸嘧啶(T)四类碱基. 碱基之间的配对关系是:A 与 T 配对,C 与 G 配对,这种配对原则称为 Watson-Crick 互补性原则. DNA 通过磷酸二酯键可组成 DNA 单链,利用互补性原则,单链很容易形成双链分子.

DNA 分子具有变性和复性的性质. DNA 变性就是指 DNA 分子中维持双螺旋稳定性的氢键和疏水键的断裂,使稳定的双螺旋结构松解为无规则线性结构的现象. DNA 复性是指变形的 DNA 链在适当条件下,二条互补链全部或部分恢复到天然双螺旋结构的现象. 热变性 DNA 一般经缓慢冷却后即可复性,此过程称之为“退火”. 热循环就是利用这两性质使来源不同的 DNA 片段按碱基互补关系形成杂交双链分子的一个过程.

1.1.2　生物酶

限制内切酶能够识别特定的碱基序列,并在相应的位置切断作分离操作. 聚合酶能够在 DNA 序列的一端上添加核苷酸使序列加长,作复制操作.

1.2　DNA 遗传算法的结构

DNA 遗传算法的结构与传统的遗传算法相类似,如图 1 所示.

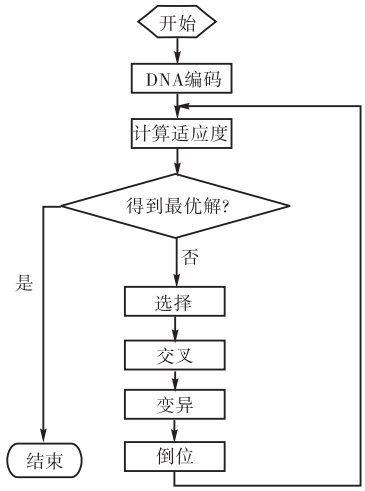

图 1　DNA 遗传算法流程图

Fig.1　DNA genetic algorithm flowchart

1.2.1　DNA 编码

使用 n 个具有任意长度的 DNA 链组成初始编码群体 $P(t)$. 一条 DNA 链由 4 种碱基 A,T,G,C 的序列构成. 初始化时,待解问题的设计参数是通过 4 个字母的符号集 Σ(A,T, G,C)编码以形成染色体,即 DNA 链. DNA 编码是一个关键的环节.DNA 链的长短将直接影响问题求解的精度和收敛度.

1.2.2　计算适应度

按照编码规则,根据具体问题构造具体的适应度函数. 通过这种评价函数决定哪个染色体是适合问题的最佳解,适应度函数值越大,解的质量越好.

1.2.3　选择

与常规遗传算法类似,从初始编码群体 $P(t)$ 中以一定的概率选出个体来繁殖后代. 经常使用的方法有期望法、精华保存法和适应度比例法等.

1.2.4　交叉

交叉操作是将两条 DNA 链进行互换重组,其效率和精度直接影响到计算结果.

1.2.5　变异

以一定的概率从 DNA 群体 $P(t+1)$ 中随机地选择若干个个体. 随机地选取某一基因位进行变异. 变异是为了使 DNA 遗传算法具有局部随机搜索能力,从而来维持群体的多样性,避免出现早熟现象.

1.2.6　倒位

在 DNA 链中两个随机选择位置之间的某些碱基的顺序进行倒位. 它可以使在父代中离得很远的位在后代中靠在一起. 相当于重新定义基因块. 倒位操作是可选的,根据问题的需要而定.

2　改进的 DNA 遗传算法

2.1　改进思想

一个适应度高的个体中包含着一段好的基因片段. 将这段基因片段提取出来并将它遗传到后代中,可以加快收敛速度.

2.2　改进技术

基于上述思想,本文在传统 DNA 遗传算法的基础上对 DNA 遗传算法做了以下几方面的改进:

选择操作改进. 采用“轮盘赌法”选择适应度高的个体,其优点是对适应度低的个体也给予选择的机会,这样保证了优秀基因片段来源的多样性. 然后在适应度高的个体中根据一定的规则选择出一段优秀的基因片段. 确定一个长度 L,从选出的每个适应度高的个体中随机选出 n 个长度为 L 的子序列,从中选出适应度最高的作为优秀基因片段.

交叉操作改进. 丁永生等[8]提出了采用基因转移操作来取代交叉算子. 将染色体中某一段好的基因串直接传递给其他染色体,进行信息之间的重组. 从而产生更好的个体,来加快搜索速度. 本文以此为基础,分两种情况产生新的群体. 适应度高的个体按照一定的概率进行优秀基因片段的互换,适应度低的个体分别以优秀基因片段为母体随机产生新的个体.

变异操作改进. 为保持种群的多样性和产生新

的基因信息，变异操作在所有新个体中进行. 文献[9]在研究 DNA 序列模型时指出在同一个 DNA 序列的不同位置，存在 hot spot 和 cold spot，位于 cold spot 的碱基其变异概率远远小于 hot spot 的碱基. 对于 DNA 遗传算法而言，在进化的不同阶段，不同位置的码位对问题答案的影响是不一样的. 基于上述思想，变异概率应该是一个动态变化的过程，变异的方法也与传统的单个基因位的变异不同. 从所有 DNA 链中选出适应度最高的个体，随机地选取该 DNA 链中的某一段 DNA 序列，将其传递给其他个体的对应部分.

2.3　算法编码及操作算子

2.3.1　编码

DNA 序列采用 A、G、C、T 四个字母来对长度为 L，由腺嘌呤、鸟嘌呤、胞嘧啶、胸腺嘧啶四种碱基组成的 DNA 序列进行编码.

2.3.2　操作算子

(1) 选择算子. 根据适应度函数值 f，采用“轮盘赌法”定义最好的 $N/2$ 个个体为中性个体，剩下的为有害个体. 为便于计算，设 N 为偶数. 在 $N/2$ 个中性个体中选出 $N/2$ 个长度为 l_0 的优秀基因片段，即当前序列的子序列 R.

(2) 交叉算子. 交叉分两种情况：中性个体两两互换优秀基因片段，两个序列与对方子序列 R 相对应的地方分别被对方的子序列 R 所替换；有害个体交叉的概率为 100%. 通过两种情况的交叉操作得到 N 个子代个体.

(3) 变异算子. 根据问题的特点选择是否使用动态的变异概率.

3　改进的 DNA 遗传算法求解指派问题

3.1　指派问题

指派问题的提法是：某单位需完成 n 项任务，恰好有 n 个人可承担这些任务. 各人完成任务不同，效率也不同.那么，如何进行指派，使总效率最高呢?

设决策变量为 $x_{ij}(i,j=1,2,\cdots,n)$，当指派第 i 个人去完成第 j 项任务时，$x_{ij}=1$；否则，$x_{ij}=0$. 设第 i 个人完成第 j 项任务的时间为 $c_{ij}(i,j=1,2,\cdots,n)$，$c_{ij}>0$. 若要求指派后的总时间最小，则最优指派问题的数学模型为

$$\min z=\sum_{i=1}^{n}\sum_{j=1}^{n}c_{ij}x_{ij}$$

$$s.t.\begin{cases}\sum_{i=1}^{n}x_{ij}=1(j=1,2,\cdots,n)\\ \sum_{j=1}^{n}x_{ij}=1(i=1,2,\cdots,n)\\ x_{ij}=1\text{或}0\end{cases}$$

从模型中的约束条件的结构不难看出：设决策变量所组成的矩阵为 $\boldsymbol{A}=(x_{ij})_{n\times n}$，则最优指派问题的解 $x_{ij}(i,j=1,2,\cdots,n)$ 为可行解的充分必要条件为 $\boldsymbol{A}$ 中恰有 n 个数为 1，第 j 列恰有 1 个数为 1($j=1,2,\cdots,n$)，第 $i(i=1,2,\cdots,n)$ 行恰有 1 个数为 1.

3.2　算法编码及算法步骤

3.2.1　DNA 编码

根据指派问题的特点，对每个人的每项任务进行编码，n 项任务 n 个人来完成总共需要 $n\times n$ 个编码. 编码包括 3 bp(base pairs)长度的数值序列和 10 bp 长度的标志序列，分三种情况. 第一列的编码先是数值序列，然后是标志序列；最后一列的编码与第一列的正好相反，先标志序列再数值序列；中间列为两个标志序列中间夹一个数值序列. 如果有后标志序列，则它唯一地表示了该列，如果有前标志序列，则它是前一列后标志序列的补序列. 例如第一列的编码为 XXXAACGTGATGC，第二列的编码为 TTGCACACGXXXGTAGCTAGTG. 其中 XXX 为数值序列，表示此人完成该任务所需的时间. 第一列编码中的后标志序列 AACGTGATGC 表明该编码为第一列编码. 至于编码属于哪一行则通过添加药剂来实现. 从编码得出每个序列的长度为 $13\times n-10$.

3.2.2　算法步骤

(1) 设置最大进化代数 G 和初始种群数 N. 将 $n\times n$ 个编码片段放在一起进行热循环，在热循环过程中前一列的后标志序列和与它互补的后一列的前标志序列退火，在聚合酶的作用下形成双链. 在聚合前，为每一行的 n 个片段设置相同的特性标记，为每一列设置不同的特性标记，使得处在相同行不同列的片段互相排斥不能连接，并且这些标记可以叠加，保证已经连接的片段不能再与该片段上已有的处在相同行上的片段相连. 在此基础上得到 N 个 DNA 序列作为初始种群.

(2) 初始种群中每个序列的单链部分就是每个人完成任务所需的时间. 根据适应度将 N 个 DNA 序列分成两部分. 本文的适应度函数设计如下：

$$f(x)=\begin{cases}C_{\max}-g(x) & g(x)<C_{\max}\\ 0 & \text{其他}\end{cases}$$

式中：$g(x)$ 是相应指派方案中所有人的完成时间和；

C_{max} 为进化过程中 $g(x)$ 的最大值或当前群体中 $g(x)$ 的最大值. 从公式可以看出,完成时间和小的指派方案相应的适应度值大,所以保留的可能性也较大. 采用“轮盘赌法”将适应度高的 $N/2$ 个个体保存下来,并从这些个体中提取出 $N/2$ 个优秀片段. 从 1 到[$n/2$]中随机生成一个数 a,在 $N/2$ 个体中截取 $N/2$ 个从 a 列到 $a+[n/2]-1$ 列长度为[$n/2$]的片段,为了满足这个要求,用限制性酶的特殊酶切位点来完成. 然后计算每个的时间值,选择时间最短的. 重复执行 $N/2$ 次,产生 $N/2$ 个优秀片段.

(3)分别以步骤(2)产生的优秀片段为母板,继续与 $n\times n$ 个片段进行热循环,产生 $N/2$ 个带有优秀片段的新个体,与保存下来的时间短的 $N/2$ 个个体组成新的 N 个个体.

(4)由于问题的编码不涉及到高低位,所以不采用动态变异概率. 设定一个变异概率 p,对每个个体都产生一个随机概率 p_i,如果 $p_i \leqslant p$ 则进行变异. 变异时从 1 到 n 中随机生成一个数 b,将序列中的第 b 个片段剪切下来,这样产生两段序列,并将第二段序列的最后一个片段也去掉. 然后将剩下的两段序列放在不包含剪切下来的两个片段的试管中进行热循环,将两段序列连接产生新的 DNA 序列.

将上述步骤产生的种群重复进行步骤(2)、(3)、(4)的操作 $G-1$ 次,得到最优解.

4　仿真实验

根据上述步骤将 DNA 编码改成实数编码,所有的生物操作采用编程实现,在 Visual C++环境下采用 C++语言编写程序进行仿真. 设有 A、B、C、D、E、F、G、H 八项任务由 0、1、2、3、4、5、6、7 八个人去完成,其完成时间如表 1 所示. 实验参数为:最大进化代数 G 为 20、初始种群 N 为 10、变异概率 p 为 0.1.

表 1　8×8 指派问题时间数据

Tab.1　Time data of 8×8 assignment problem　　d

任务编号	被指派人							
	A	B	C	D	E	F	G	H
0	14	5	2	1	4	8	7	5
1	5	5	2	4	3	2	13	5
2	2	9	8	7	11	5	12	3
3	10	4	2	4	5	8	10	6
4	14	11	2	5	4	1	3	9
5	9	5	3	7	6	1	7	8
6	4	8	2	5	10	4	5	12
7	2	4	2	14	3	5	9	3

实验表明,平均迭代 4.5 次就可以收敛到最优方案:7A—3B—6C—0D—1E—5F—4G—2H,而传统的 DNA 遗传算法平均迭代 9.5 次才能达到最优解. 改进前后的实验对比见表 2. 可以看出,改进后的方法收敛更快.

表 2　改进前后的实验对比

Tab.2　Results of comparison experiment

方法	平均最短时间/d	平均迭代次数
改进前	26	9.5
改进后	21	4.5

5　结　语

本文在传统 DNA 遗传算法的基础上,根据优秀基因片段思想,对选择、交叉和变异操作进行改进,给出了具体操作. 针对指派问题的最优解求解,提出具体的算法步骤,并进行仿真,初步验证了改进算法的有效性和实用性. 但是对于如何选择优秀基因片段本文并没有给出确定的规则,需要根据具体的问题进行具体的分析,还需要在这方面进行进一步的研究.

参考文献:

[1] 俞国燕,王筱珍. 改进遗传算法的应用研究[J]. 机械制造,2007,45(513):58–60.

[2] 蒋腾旭,谢枫. 遗传算法中防止早熟收敛的几种措施[J]. 计算机与现代化,2006,136(12):54–56.

[3] 徐丽娜. 神经网络控制[M]. 哈尔滨:哈尔滨工业大学出版社,1999:1–15.

[4] Adelman L M. Molecular computation of solutions to combinatorial problems [J]. Science,1994,266(5187):1021–1024.

[5] 丁永生. 计算智能:理论、技术与应用[M]. 北京:科学出版社,2004:294–303.

[6] 丁永生,任立红. DNA 遗传算法及其函数寻优应用[C]//中国控制与决策学术年会论文集. 沈阳:控制与决策编辑部,2000:235–239.

[7] 张雷,杨大地,冉戎. 基于 DNA 遗传算法的曲面最短路径问题[J]. 计算机工程,2007(16):181–185.

[8] Ren L H,Ding Y S,Ying H,et a1. Emergence of self-learning fuzzy systems by a new virus DNA-based evolutionary algorithm [J]. Intelligent Systems,2003,18(3):339–354.

[9] Neuhauser C,Krone S M. The genealogy of samples in models with selection[J]. Genetics,1997(2):519–534.