基于改进的粒子群遗传算法的 DNA 编码序列优化

崔光照[1),2)] 李小广[1)] 张勋才[1),2)] 王延峰[1),2)] 李翠玲[1)]

[1)](郑州轻工业学院电气信息工程学院 郑州 450002)

[2)](河南省信息化电气重点实验室 郑州 450002)

摘 要 在 DNA 计算中,DNA 编码序列的设计是影响 DNA 计算可靠性的重要手段.在不同的 DNA 序列设计中,应该选择适当的约束条件,并且根据相应的约束条件提出每个 DNA 应该相应满足的评估公式.文中从 DNA 编码设计应满足的多约束条件中选取适当的约束条件,提出评估公式,并采用改进的粒子群遗传算法来解决多目标优化问题.同时根据得到的序列与已有序列在综合适应度函数结果上进行对比,结果证明了该方法的有效性.

关键词 DNA 计算;DNA 编码;多目标优化;改进的粒子群遗传算法

中图法分类号 TP301 **DOI 号**: 10.3724/SP.J.1016.2009.00311

The Optimization of DNA Encodings Based on Modified PSO/GA Algorithm

CUI Guang-Zhao[1),2)] LI Xiao-Guang[1)] ZHANG Xun-Cai[1),2)] WANG Yan-Feng[1),2)] LI Cui-Ling[1)]

[1)](*School of Electrical and Electronic Engineering, Zhengzhou University of Light Industry, Zhengzhou* 450002)

[2)](*Henan Key Laboratory of Information-based Electrical Appliances, Zhengzhou* 450002)

Abstract The design of DNA sequence is important in improving the reliability of DNA computing. Some appropriate constrained terms that DNA sequence should satisfy are selected, and then the evaluation formulas of each DNA individual corresponding to the selected constrained terms are proposed. Modified Particle Swarm Optimization/Genetic Algorithm (MPSO/GA) is presented to solve the multi-objective optimization problem. At last the comparison of the results with the known DNA sequences in fitness function value is made to prove the feasibility and efficiency of the method.

Keywords DNA computing; DNA coding; multi-objective optimization; modified particle swarm optimization/genetic algorithm

1 引 言

美国加州大学的 Adleman 博士[1-2]利用现代分子生物技术提出 7 个顶点的哈密尔顿有向路问题的 DNA 分子生物算法,并且成功地在 DNA 溶液的试管中进行了实验,成为 DNA 计算发展的里程碑.总的来说,DNA 计算可以分为 3 个步骤:首先是编码,也就是把具体的问题映射到 DNA 分子链上;其次是通过各种生物酶的作用,在适当的条件下,使所代表问题的各个 DNA 分子链进行充分的混合杂交反应;最后是萃取阶段,即把杂交反应的结果还原为原问题的解.

通过 DNA 计算的过程可以知道,编码问题是

收稿日期:2009-07-06;最终修改稿收到日期:2009-11-17.本课题得到国家自然科学基金(60573190,60773122,60970084)、河南省基础与前沿技术研究项目(082300413203,092300410166)和河南省科技创新人才计划(科技创新杰出青年)(094100510022)资助.**崔光照**,男,1957 年生,博士,教授,主要研究领域为 DNA 计算、基因网络以及信息控制等. E-mail: cgzh@zzuli.edu.cn. **李小广**,男,1983 年生,硕士研究生,主要研究方向为 DNA 计算. **张勋才**,男,1981 年生,博士,副教授,主要研究方向为智能信息处理、算法分析与设计等. **王延峰**,男,1973 年生,博士,副教授,主要研究方向为 DNA 计算 、基因网络. **李翠玲**,女,1984 年生,硕士研究生,主要研究方向为信息安全.

首要问题.一个好的 DNA 编码序列就是能够确保随后进行的各种生化反应不出现任何错误,而且反应产物中包含有足够多的稳定可靠的能够被成功提取的原始问题的解.在实际的操作过程中,DNA 编码设计就要通过序列优化尽量减少 DNA 计算过程中的错误杂交.错误杂交的类型可以分为两种[3-4]:第一是假阳性,也就是不完全互补的 DNA 分子在适当的条件下能够杂交形成双链分子;第二是假阴性,也就是完全互补的 DNA 分子在反应过程中由于种种原因而没有杂交.假阳性主要是由于杂交的两个 DNA 分子间的序列有足够的"相似度"而造成的;而假阴性则主要是由反应条件及生化操作本身的失误引起的.为了确保 DNA 计算结果的可靠性,就必须最大限度地降低错误杂交的可能性.因此我们利用各种约束条件来筛选序列,使之满足可靠的 DNA 计算的需要.已经提出的组合约束以及热力学约束有汉明距离约束、二级结构约束、连续性约束、解链温度、GC 含量等.基于这些约束条件,Frutos 等提出了模板编码方法[4],Feldkamp 提出了一个设计 DNA 序列的 DNA 序列编译算法[5],Deaton 提出了遗传算法来设计 DNA 序列[6-7].

DNA 编码问题实质上是满足多约束条件的多目标组合优化问题.文献[8]用基本遗传算法针对多个约束条件产生了较好的序列,其中约束条件是针对每代群体的所有个体所提出的.本文将改进的粒子群算法(MPSO)[9-10]与基本遗传算法(SGA)[8]相结合,提出了改进的粒子群遗传算法(MPSO/GA)来对 DNA 序列进行优化,同时提出了针对单个个体的约束条件评估公式.在遗传算法的基础上,结合改进的粒子群算法,产生出了与之前文献相比更好的 DNA 序列.

2 基本编码问题

DNA 编码问题是 DNA 计算中的核心问题,它创造性地将现实问题映射为特殊的编码 DNA 分子序列.这些 DNA 分子序列应能够确保随后进行的生化反应不出现任何错误,而且反应产物中须包含有足够多的、稳定可靠的、能被成功提取的原始问题的解.一个好的编码序列就是要最大限度地促进期望的杂交,同时限制不期望杂交的发生,综合平衡各个反应条件,从而为最后解的提取提供有力支持.

概括来讲,DNA 编码问题可以表述为:在 DNA 分子的 4 个碱基 $\Sigma_{DNA}=\{A,G,C,T\}$ 上,存在一个长度为 n 的 DNA 分子的编码集合 S,显然 $|S|=4^n$.求 S 的一个子集 $C\subseteq S$ 使得 $\forall s_i,s_j\in C$ 满足 $\tau(s_i,s_j)\geq k$.其中 K 为正整数,τ 是评价编码性质的准则,如汉明距离、GC 含量等.

接下来的问题就是要在寡核苷酸空间中找到合适的评价编码性质的准则 τ 以及确定各个 DNA 编码约束条件的复杂性.如前所述,DNA 编码问题实质是多目标组合优化问题,也就是说编码要同时满足多个目标约束.这个问题将在第 4 节给出.

3 改进的粒子群遗传算法

3.1 基本遗传算法

遗传算法是模拟达尔文生物进化论的自然选择和遗传学机理的生物进化过程的计算模型,是一种通过模拟自然进化过程搜索最优解的方法,它最初由美国 Michigan 大学 Holland 教授于 1975 年首先提出来的.遗传算法以决策变量的编码作为运算对象,而不需要待解决问题的具体领域信息,同时它也不受搜索空间是否连续或者可微的限制.遗传算法主要可以分为 4 个部分:初始状态的确立、适应度函数的制定、遗传操作的进行以及控制参数的选取.

编码机制是遗传算法的基础.首先要创建一个随机的初始状态,即初始解,将这些解比喻为染色体或基因,该种群被称为第一代.接着对每一个解(染色体)指定一个合理的适应度值,根据问题求解的实际接近程度来指定.然后就是各种进化操作(选择、交叉和变异).在以上的各种操作过程中,控制参数的选取是不可忽略的因素,适当地选取可以使算法得到令人满意的结果.图 1 为基本遗传算法的流程图.

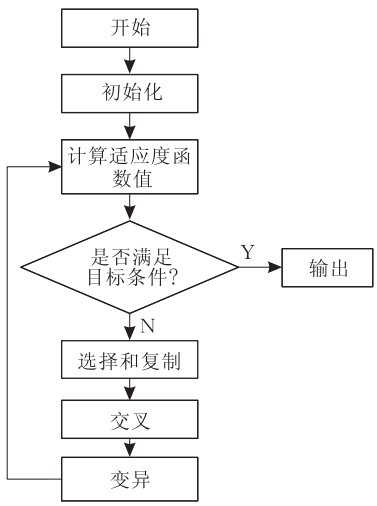

图 1 遗传算法流程图

3.2　改进的粒子群优化算法

粒子群算法自诞生以来就以其规则简单,易于编程实现成为处理多目标优化问题的重要工具.传统的粒子群算法是求解连续优化问题的有力工具,但是对于离散问题却无能为力.这里我们引入了四进制粒子群算法用于求解 DNA 编码这样的离散空间问题.

(1)传统粒子群算法:传统粒子群算法是受鸟群觅食行为的启发而提出的,其基本思想是通过群体中个体之间的协作和信息共享来寻找最优解.该算法将每个个体看作是搜索空间中的一个没有体积的微粒,并且在搜索空间中从一个随机初始位置(x_i)和随机初始速度(v_i)飞行.每个粒子代表解空间的一个候选解,它们的飞行速度根据它本身的飞行经验和同伴的飞行经验来进行动态调整.每个粒子在飞行过程中所经历过的最好位置就是粒子本身找到的最优解,整个群体所经历过的最好位置就是整个群体目前所找到的最优解.每个粒子都通过上述两个极值(个体最优解和整体最优解)不断地更新自己,从而产生新一代群体.对于第 t 次迭代,粒子 i 将根据下面的公式来更新自己的位置和速度:

$$\begin{cases} v_{id}^{t+1}=w\times v_{id}^{t}+c_1\times rand()\times(p_{id}-x_{id}^{t})+ \\ \qquad c_2\times rand()\times(p_{gd}-x_{id}^{t}) \\ x_{id}^{t+1}=x_{id}^{t}+v_{id}^{t+1} \end{cases},$$

其中 $rand()$ 是均匀分布于(0,1)区间的随机数,c_1,c_2 为学习因子,ω 为惯性权值,p_{id} 为个体极值,p_{gd} 为群体极值.粒子在解空间中不断跟踪个体极值与全局极值进行搜索,直到达到规定的迭代次数或者满足规定的误差标准为止.

(2)四进制粒子群算法:基于传统粒子群算法的不足,Kenndy 和 Eberhart 在文献[11]提出了一种二进制离散型 PSO 算法[12]用于解决组合优化问题,在一定程度上完善发展了传统粒子群算法.这里我们对文献[11]进行了改进,在离散四进制空间中,粒子 x_{id} 要趋向于判决选择为 0,1,2 和 3,由参数 v_{id} 决定一个概率选择阈值,如果偏高,粒子就会更可能选择为 0,1;如果偏低,就更倾向于选择 2,3.并且这个阈值位于[0,1]范围之内,这里引入 sigmoid 函数:$S(v_{id})=1/(1+\exp(-v_{id}))$,改进后的速度和位置更新公式如下:

$$\begin{cases} v_{id}^{t+1}=w\times v_{id}^{t}+c_1\times rand()\times(p_{id}-x_{id}^{t})+ \\ \qquad c_2\times rand()\times(p_{gd}-x_{id}^{t}) \\ x_{id}^{t+1}=Mod((x_{id}^{t}+f(v_{id}^{t+1})),4) \end{cases},$$

其中 $f(v)$ 被定义如下:

$$f(v)=\begin{cases} 0, & rand()>r\&rand()<S(v) \\ 1, & rand()<r\&rand()<S(v) \\ 2, & rand()\leq r\&rand()\geq S(v) \\ 3, & rand()\geq r\&rand()\geq S(v) \end{cases}.$$

(3)扰动策略的引入:经典的粒子群算法的粒子们在搜索过程中,总是追逐当前全局最优点和自己迄今搜索到的最优点,因此速度会很快降低到接近于 0,容易陷入局部最优值.我们课题组的秦利敏等人在文献[10]引入了扰动策略:即如果迄今搜索到的全局最优适应值连续 u 步迭代没有更新,那么依概率 r 随机选择一定数量的粒子,重置它们的速度.u 是自然数,称为设定的扰动因子,r 为[0,1]上的一个随机数.扰动策略表示为

$$\text{If } t-t_u>u,\ \text{Then reset } v,$$

其中 t_u 表示最近一次更新搜索到的全局最优点适应值的迭代步.图 2 为改进的粒子群算法流程图.

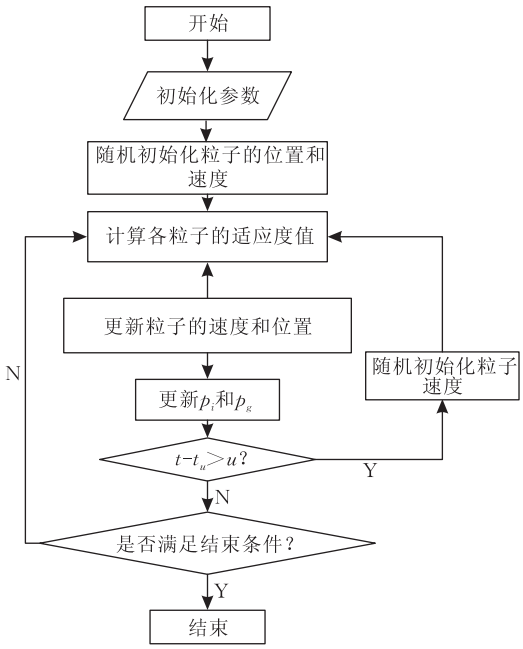

图 2　改进粒子群算法流程图

3.3　改进的粒子群遗传算法(MPSO/GA)

该算法是以基本遗传算法为基础,同时将改进的粒子群算法作为遗传算法的一个重要算子,具体算法步骤如下:

1. 设定参数,并随机产生初始种群;

2. 计算每个个体的适应度函数值,并且按照适应度函数值进行排序;

3. 判断是否满足目标条件(包括程序收敛以及达到指定的进化代数),如果满足,结束进程,输出结果;否则进行下一步;

4. 更新个体种群. 根据适应度函数值的大小确定一部分个体直接进入下一代种群，剩余个体通过 MPSO 算法优化过后进入下一代种群；

5. 对新一代种群执行遗传算法的复制、交叉和变异等操作，转步 2.

该算法的流程图如图 3 所示.

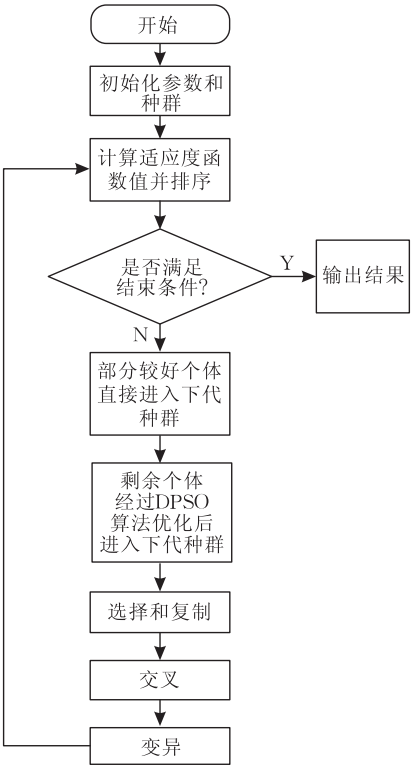

图 3 MPSO/GA 算法流程图

4 算法约束条件

通过很多的算法与文献，我们已经知道，越来越多的约束条件被用于评估 DNA 编码的好坏. 例如 GC 含量约束、T_m 值约束、连续性约束、二级结构约束、汉明距离约束、自由能约束等等. 但是通过文献[13]我们可以知道并不需要把每个约束条件考虑在内. 综合目前 DNA 计算中的主要组合约束，我们选取了以下几个方面作为评估项.

(1) GC 含量约束. GC 含量就是碱基 G 和 C 在一个 DNA 分子序列里所占总碱基的比重.

由于碱基 G 和 C 之间有 3 个氢键连接，而碱基 A 和 T 之间有两个氢键连接，所以 GC 含量对保持序列的化学性质的稳定性非常重要. 一般要求 $GC(x)\in[40,60]$.

$$f_{\mathrm{GC}}(x_i)=-|GC(x_i)-GC(x_i)_{\mathrm{defined}}| \tag{1}$$

其中 $GC(x_i)$ 表示序列 x_i 的 GC 含量，$GC(x_i)_{\mathrm{difined}}$ 表示所指定的 GC 含量，其计算公式如下：

$$GC(x)=\frac{\#G+\#C}{|x|},$$

其中 $\#G$ 和 $\#C$ 分别表示序列 x 里面的碱基 G 和 C 的个数，$|x|$ 表示序列 x 里碱基总个数.

(2) 连续性约束. 如果 DNA 序列里某一字母连续出现(比如出现“AAAA”或者“GGGG”，那么 DNA 结构就会变得不稳定.

$$f_{\mathrm{Con}}(x_i)=-\sum_{j=1}^{n}(j-1)N_j^{(i)} \tag{2}$$

其中 $N_j^{(i)}$ 表示同一字母在指定 i 序列中出现 j 次的次数.

(3) Hairpin 约束. 发卡结构可以引起 DNA 分子的自杂交，一般应该予以限制.

$$f_{\mathrm{Hairpin}}(x_i)=\sum_{r=5}^{n-2\cdot pinlen}\ \sum_{c=pinlen+[r/2]}^{n-pinlen-[r/2]}Hairpin(x_i,c) \tag{3}$$

其中，r 为形成发卡最小的环长度；$pinlen$ 为形成发卡茎所应有的最小长度. 并且 $Hairpin(x_i,c)$ 定义如下：

$$Hairpin(x_i,c)=\begin{cases}1, & d(x_i,c)>pinlen/2\\ 0, & d(x_i,c)\le pinlen/2\end{cases},$$

其中 $d(x_i,c)$ 表示序列 x_i 沿其 c 点折叠两段序列的逆补距离.

(4) Hamming 距离约束. 在 DNA 序列编码设计中，Hamming 距离描述了两个不同 DNA 序列之间的非相似程度. 为了减少不同 DNA 序列之间的相似性从而避免错误杂交的产生，一般要求两个 DNA 序列 x_i 和 x_j 之间的 Hamming 距离不小于某个阈值 d，即 $H(x_i,x_j)\ge d$.

$$f_{\mathrm{Hamming}}(x_i)=\min_{1\le j\le n,\,j\ne i}\{H(x_i,x_j)\} \tag{4}$$

(5) 逆距离约束. 与 Hamming 距离相似，要求序列 x_i 和序列 x_j 的逆序列 x_j^R 之间的 Hamming 距离不小于某个阈值 d.

$$f_{\mathrm{Inverse}}(x_i)=\min_{1\le j\le n}\{H(x_i,x_j^R)\} \tag{5}$$

(6) 适应度函数设计. 本文所定义的多目标优化问题属于最大化问题，我们采用加权平均法来处理每个 DNA 个体各约束项的评估函数：

$$Fitness(x_i)=\sum_{j=1}^{m}w_jf_j \tag{6}$$

其中 m 是评估项个数，w_j 为各个评估项 f_j 的权重，并且有

$$f_j\in\{f_{\mathrm{GC}}(x_i),f_{T_m}(x_i),f_{\mathrm{Con}}(x_i),f_{\mathrm{Haipin}}(x_i),f_{\mathrm{Hamming}}(x_i)\}.$$

5　算法仿真结果及分析

针对以上约束条件以及目标函数设计编码序列模型,在 Matlab 7.0 环境下,使用 MPSO/GA 算法进行仿真,运行环境是Pentium Dual E2104,1.6GHz,512MB,Microsoft XP. 参数设置如下:(1)基本遗传算法. 最大进化代数为 300,种群规模为 20,DNA 序列编码长度为 20,交叉率为 0.85,变异率为 0.005. (2)MPSO 算法. 最大进化代数为 200,学习因子分别为 $c_1=2$, $c_2=1.8$,惯性权重因子 w 从 2 降低到 0.8,扰动因子 $u=10$,最大速度为 4.

为了评价本算法所产生的 DNA 序列的性能,我们将所得到结果与文献[8]的结果进行了比较. 表 1 和表 2 分别列出了本算法以及文献[8]所产生的 DNA 序列编码的数据分析. 其中"*Hd*"表示 Hamming 距离,"*Id*"表示逆距离.

表 1　MPSO/GA 算法产生的 DNA 序列

DNA 序列(5′-3′)	*Hd*	*Id*	GC 成分/%	Continuity	Hairpin	Fitness value
CGTGTGCAGTACTGAGTATG	15	14	50	−1	−1	40
AGTAGTTCTCAGACGCTGCT	12	13	50	0	0	39
GCATGATCGATCTCGTCAGA	14	13	50	−2	0	39
ATCGGTAGTCGTAGACGTCT	12	15	50	−1	−1	38
TTGCAGTCAGTCGACTAGTG	13	14	50	−1	−1	37
TACGCTCACGATAGCCATGT	14	13	50	−3	0	37
AGTGCTACCTCGCTGTGATA	12	12	50	−3	−2	34

表 2　文献[8]所产生的 DNA 序列

DNA 序列(5′-3′)	*Hd*	*Id*	GC 成分/%	Continuity	Hairpin	Fitness value
GAGTTAGATGTCACGTCACG	15	14	50	−1	−4	37
AGGCGAGTAGGGGTATATCT	14	12	50	−4	−1	34
TTATGATTCCACTGGCGCTC	13	13	50	−4	0	33
CCTGTCAACATTGACGCTCA	11	11	50	−3	−2	31
CGCTCCATCCTTGATCGTTT	11	13	50	−5	−2	31
ATCGTACTCATGGTCCCTAC	11	10	50	−3	−2	28
CTTCGCTGCTGATAACCTCA	11	10	50	−3	−1	28

从表 1 与表 2 可以明显看到,MPSO/GA 算法明显比文献[8]算法的适应度函数有所改进,同时在图 4 指出了该算法的进化收敛图. 其中横轴表示进化代数,竖轴表示适应度函数值(每代群体经过排序后最差个体的适应度值). 由图 4 可以看出,算法具有较好的收敛性.

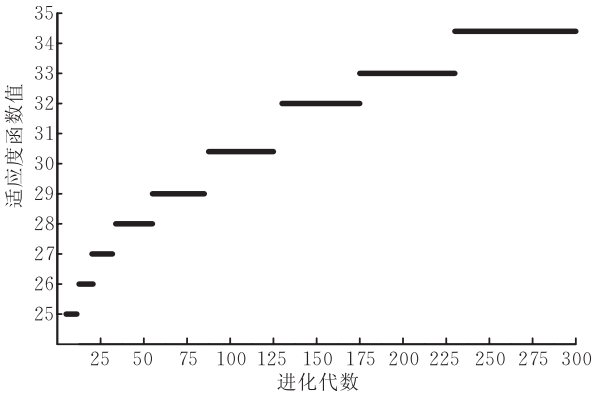

图 4　MPSO/GA 算法的进化收敛图

6　结论与展望

本文从 DNA 编码的多个约束条件选取合适的约束,通过将多约束转化为多目标优化问题,提出了 MPSO/GA 算法对 DNA 计算中的编码序列实现了优化,通过与前人对比,产生了较好的 DNA 序列,验证了该算法的可行性和有效性. 当然,进一步的工作可以结合其他的智能优化算法,进而增强算法的寻优能力并且尽量减小陷入局部最优的概率.

参　考　文　献

[1] Adleman L M. Molecular computation of solution to combinatorial problems. Science, 1994, 66(11): 1021-1024

[2] Adleman L M. On constructing a molecular computer. Computer Science Department, University of Southern California, USA: Technical Report TR 79-387, 1995

[3] Deaton R, Garzon M. Thermodynamic constraints on DNA-based computing//Gheorghe Paun eds. Computing with Bio-Molecules, Springer-Verlag, 1998: 138-152

[4] Frutos A G, Liu Q, Thiel A J et al. Demonstration of a word design strategy for DNA computing on surfaces. Nucleic Acids Res, 1997, 25(23): 4748-4757

[5] Feldkamp U, Banzhaf W, Rauhe H et al. A DNA sequence compiler//Proceedings of the 6th DIMACS Workshop on DNA Based Computers, Leiden, NE, 2000

[6] Deaton R. A DNA based implementation of an evolutionary search for good encodings for DNA computation//Proceedings of the 1997 IEEE International Conference on Evolutionary Computation. Indianapolis, IN, USA, 1997: 267-271

[7] Deaton R J, Murphy R C, Garzon M H et al. Good encodings for DNA-based solutions to combinatorial problems//Proceedings of the DNA-Based Computers II. AMS DIMACS Series 44, Princeton, 1998: 247-258

[8] Shin S Y, Lee I H, Kim D et al. Multiobjective evolutionary optimization of DNA sequences for reliable DNA computing. IEEE Transactions on Evolutionary Computation, 2005, 9(2): 143-158

[9] Cui G Z, Niu Y Y, Wang Y F et al. A new approach based on PSO algorithm to find good computational encoding sequences. Progress in Natural Science, 2007, 17(6): 712-716

[10] Cui G Z, Qin L M et al. Modified PSO algorithm for solving planar graph coloring problem. Progress in Natural Science, 2008, 18(3): 353-357

[11] Kennedy J, Eberhard R. A discrete binary version of the particle swarm optimization//Proceedings of the IEEE International Conference on System, Man, and Cybernetics. Orlando, Florida, 1997: 4104-4108

[12] Fatih M and Liang Y. A binary particle swarm optimization algorithm for lot sizing problem. Journal of Economic and Social Research, 2003, 5(2): 1-20

[13] Tanaka F, Nakatsugawa M, Yamamoto M et al. Developing support system for sequence design in DNA computing//Proceedings of the 7th International Workshop DNA-Based Computers. Tampa, FL, USA, 2001: 340-349

CUI Guang-Zhao, born in 1957, Ph. D., professor. His current research interests include DNA computing, gene networks and information control.

LI Xiao-Guang, born in 1983, M. S. candidate. His research interests focus on DNA computing.

ZHANG Xun-Cai, born in 1981, Ph. D., associate professor. His research interests include intelligent computation and algorithm analysis and design.

WANG Yan-Feng, born in 1973, Ph. D., associate professor. His research interests include intelligent computation and gene networks.

LI Cui-Ling, born in 1984, M. S. candidate. Her research interests focus on information security.

Background

This paper is supported by the National Natural Science Foundation of China (grant No. 60573190, 60773122, 60970084), Basic and Frontier Technology Research Program of Henna Province (grant No. 082300413203, 092300410166), and Innovation Scientist and Technicians Troop Construction Projects of Henna Province (grant No. 094100510022).

There are three parts in DNA computing: encoding that maps the problems onto DNA strands, hybridization that performs the basic core processing and extraction that makes the results visible to the naked eye. DNA encoding problem, which has been proved to be an NP hard problem, is a key problem for DNA computing, and usually solved by optimization algorithms. In this paper a new efficient genetic algorithm for the design of DNA codeword is presented. Some proper code design criterias are selected to improve the coding of the DNA sequences. Simulation results show this method is feasible and efficient.