

文章编号:1000-582X(2012)08-034-08

等价划分策略的混沌克隆优化算法

舒万能^{1a,1b,2a}, 丁立新^{1a,1b}, 徐兴东^{2b}

(1. 武汉大学 a. 计算机学院; b. 软件工程国家重点实验室, 湖北 武汉 430072;

2. 中南民族大学 a. 计算机科学学院; b. 计算与实验中心, 湖北 武汉 430074)

摘要:克隆选择算法随机产生种群的方式, 将容易导致数字的取值非均匀的分布在解的空间, 从而增加数据冗余的现象。为了克服克隆选择算法的缺点, 将克隆选择算法和混沌优化相结合, 提出一种用于函数优化的混沌克隆优化算法。该算法利用混沌的随机性、遍历性和规律性来避免陷入局部极小值, 同时引入等价划分的策略, 减少了可能出现的数据冗余现象。仿真实验显示了所设计的算法能以较快的速度完成给定范围的搜索和全局优化任务。

关键词:混沌优化; 克隆选择算法; 等价划分; 数据冗余

中图分类号: TP 344. 1

文献标志码: A

Chaotic clone optimization algorithm based on equivalent division strategy

SHU Wanneng^{1a,1b,2a}, DING Lixin^{1a,1b}, XU Xingdong^{2b}

(1a. Computer School; 1b. State Key Laboratory of Software Engineering, Wuhan University,

Wuhan 430072, Hubei, P. R. China; 2a. College of Computer Science; 2b. Center of Computing

& Experimenting, South-Central University for Nationalities, Wuhan 430074, Hubei, P. R. China)

Abstract: The way of clonal selection algorithm randomly generated population that will easily lead to numbers of non-uniform distribution of values in the solution space, thus increasing the data redundancy phenomenon. To overcome the shortcomings of clonal selection algorithm, a chaotic clonal optimization algorithm for function optimizing is proposed by combining clonal selection algorithm, chaos optimization. This algorithm uses chaotic characteristics—randomness, ergodicity and regularity to avoid trapping around local optimal. Equivalent division strategy is introduced by reducing the possible data redundancy phenomenon. The simulation results show that the proposed algorithm can converge to the global optimum at quicker rate in a given range.

Key words: chaotic optimization; clonal selection algorithm; equivalence division; data redundancy

免疫系统作为一种分布式自学习系统, 能自适应地维持群体多样性且具有自我调节功能。克隆选择算法(clonal selection algorithm, CSA)是模拟免疫系统对病菌的多样性识别能力而设计出来的多峰值搜索算法^[1-2]。它的搜索目标具有一定的分散性,

独立性。它能在维持种群个体多样性的同时, 加速算法的收敛过程。克隆选择算法在函数优化、组合优化、模式识别、数据挖掘等方面取得了良好的效果, 但是该算法随机产生种群的方式, 将容易导致数字的取值非均匀的分布在解的空间, 从而增加数据

收稿日期: 2012-02-05

基金项目: 国家自然科学基金资助项目(60803095, 60975050; 60902053); 高等学校博士学科点专项科研基金资助项目(20070486081); 中央高校基本科研业务费专项资金(ZY11005)

作者简介: 舒万能(1981-), 男, 武汉大学博士生, 主要从事智能计算方向研究, (Tel) 13971565983; (E-mail) shuwanneng@yahoo.com.cn。

冗余的现象。文献[3]针对该缺点,提出一种改进的免疫克隆多样性算法,该算法采用实数编码,其变异算子产生的新抗体服从均匀分布,从而有效解决了高斯分布易陷入局部极值的问题;文献[4]结合小生境技术,提出一种新的免疫算法,但该算法没有克隆操作,虽然提高了收敛速度,但却限制了搜索空间。

混沌优化算法是一种寻优效率较高的随机搜索方法。它具有遍历性、随机性、规律性等特点,能在一定范围内按其自身规律不重复地遍历所有状态^[5-6]。混沌优化算法能将混沌变量线性映射到优化变量的取值区间,然后利用混沌变量进行搜索,它具有对初值敏感、易于跳出局部最优解、搜索速度快等特点,被广泛应用于非线性优化领域^[7-8]。混沌优化算法已经广泛地和遗传算法进行结合,并取得了许多成果。文献[9]把混沌变量加载于遗传算法(genetic algorithm, GA)的变量群体中,利用混沌变量对子代群体进行微小扰动,并随着搜索过程的进行逐渐调整扰动幅度来实现优化。文献[10]采用帐篷映射生成的序列经简单映射后利用高斯函数来决定交叉位置,然后用 logistic 产生的序列对种群进行变异。文献[11]先用 GA 得到适应值较大的部分个体,然后在这些个体的领域内进行混沌优化或变尺度混沌优化。

结合混沌优化和克隆选择算法各自的优点,并采用等价划分的策略来引导算法的克隆,变异和选择操作,提出了一种基于等价划分策略的混沌克隆优化算法(chaotic clonal optimization algorithm, CCOA)。实验表明该算法执行效率高,能有效地收敛到最优解或近似最优解。

1 等价划分

1.1 等价区间划分

首先对抗体的亲和度进行处理,使得所有抗体的亲和度值分布在 $[0, 1]$ 之间。对亲和度的处理如下

$$F(x) = \frac{f_{\max} - f(x)}{f_{\max} - f_{\min}}, \quad (1)$$

其中: $f(x)$ 为抗体 x 的亲和度值; f_{\max} 和 f_{\min} 分别为当前群体的抗体最大和最小亲和度值。将 $F(x)$ 称为抗体 x 的样本值。令 $a = \min\{F_1, \dots, F_n\}$, $b = \max\{F_1, \dots, F_n\}$,并记区间 $\Omega = [a, b]$ 。将 Ω 划分为 N 个不相交的等价区间。假设 $\sum = \{A_i | i \in n\}$ 是区间 Ω 的一个划分,即

- 1) $A_i \cap A_j = \phi, \forall i \neq j$;
- 2) $\cup A_i = \Omega$ 。

具体的等价划分策略如下

①第一次划分:移动指针 $N=1$,将区间 Ω 划分成 $[a, 2^{-1}], (2^{-1}, b]$ 2部分。将抗体集 X 中亲和度小于0.5的放在左区间 $[0, 2^{-1}]$,其他的放在右区间 $(2^{-1}, 1]$;

②第二次划分:将区间 $[0, 2^{-1}]$ 划分成 $[a, 2^{-2}], (2^{-2}, 2^{-1}]$ 2部分。将抗体集 X 中亲和度小于0.25的放在左区间 $[a, 2^{-2}]$,其他的放在右区间 $(2^{-2}, 2^{-1}]$, $N=2$;

③如此下去, $N=N+1$ 。当 $2^{-N} < a$ 时,等价区间划分结束;否则,继续将区间 $[a, 2^{N-1}]$ 划分为 $[0, 2^{-N}], (2^{-N}, 2^{-(N-1)}]$ 。并将抗体集中亲和度小于 2^{-N} 的放在左区间 $[0, 2^{-N}]$,其他的放在右区间 $(2^{-N}, 2^{-(N-1)}]$ 。

经过等价划分后,将区间 Ω 划分为 N 互不相交的区间,并记为 A_1, A_2, \dots, A_N ,则 $\bigcup_{i=1}^N A_i = \Omega = [a, b]$ 。落在每一个区间的抗体集,称为块,并分别记为 $X_1, X_2, \dots, X_i, \dots, X_N$,则 $\bigcup_{i=1}^N X_i = X$ 。其中, $X_i = \{X_i^1, X_i^2, \dots, X_i^{\{X_i\}}\}$, $\# \{X_i\}$ 为统计块 X_i 中抗体的个数, $X_i^{\{X_i\}}$ 为每个块内的抗体,并且按亲和度降序排序。

1.2 抗体的信息熵

将抗体落在每个区间的频率记为 $P(A_i)$, $P(A_i) = \frac{\# \{X_i\}}{n}$ 。定义抗体的信息熵函数 $H(B) = - \sum_{i=1}^N P(A_i) \log(P(A_i))$ 。

抗体的平均信息熵为 $\overline{H(B)} = |H(B)|/N$ 。抗体群平均信息熵越大,就表明种群的分布性越好。

2 混沌克隆优化算法

2.1 编码

常见的遗传算法编码方式有二进制、gray码和实数编码,其中二进制编码以其简单直观而最常用^[12-13]采用二进制编码。假设种群中抗体数目为 n , x_i^t 表示第 t 代的第 i 个个体, $i \in \{1, 2, \dots, n\}$ 。每个抗体的基因位数 $L=m$ 。采用二进制编码,这样每个抗体 $x_i^t \in \{\varphi\}^m, \varphi \in \{0, 1\}$ 。抗体 x_i^t 可以表示为 m 维的行向量,即 $x_i^t = [x_i^{t(1)} x_i^{t(2)} \dots x_i^{t(j)} \dots x_i^{t(m)}]$, $j = 1, 2, \dots, m$ 。记 $a_i^t = \min\{x_i^{t(1)}, x_i^{t(2)}, \dots, x_i^{t(j)}, \dots, x_i^{t(m)}\}$, $b_i^t = \max\{x_i^{t(1)}, x_i^{t(2)}, \dots, x_i^{t(j)}, \dots, x_i^{t(m)}\}$ 。第 t 代种群 $\mathbf{X}(t)$ 可以表示为一个 $n \times m$ 的矩阵 $\mathbf{A}(t) = [x_1^t x_2^t \dots x_n^t]^T$ 。

2.2 区间克隆

为了扩大解空间的搜索范围,对优秀的抗体采取区间克隆操作。将每个块内亲和度最大的抗体进

行克隆。对种群 $\mathbf{X}(t)$ 执行的克隆操作 $\Psi_c(\mathbf{X}(t))$ 如下定义

$$\begin{aligned} \Psi_c(\mathbf{X}(t)) &= \Psi_c(X_1, X_2, \dots, X_N) = \\ &(X_1) \cup \Psi_c(X_2) \cup \dots \cup (X_N) = \\ &(X_1) \cup (X_2) \cup \dots \cup (X_N) = \\ &\{X_1^{1,1}, \dots, X_1^{1,\#(X_1)}\} \cup \{X_{1,1,2}, \dots, \\ &X_2^{1,\#(X_2)}\} \cup \dots \cup \{X_N^{1,1}, \dots, X_N^{1,\#(X_N)}\} \end{aligned}$$

式中, $\Psi_c(X_i) = \{X_i^{1,1}, X_i^{1,2}, \dots, X_i^{1,\#(X_i)}\}$, $X_{i,j} = X_i$ ($i=1, 2, \dots, N, j=1, 2, \dots, \{X_i\}$)

2.3 Logistic 混沌变异

免疫学认为,亲和度成熟和抗体多样性的产生主要是依靠抗体的高频变异。在抗体进化过程中,对亲和度样本值小于平均信息熵的抗体,采取 Logistic 混沌变异操作。由于混沌运动具有遍历性、随机性等特点,混沌运动能在一定范围内按照其自身的规律不重复的遍历所有的状态^[14]。因此,如果利用混沌原理来产生初始种群,而不是采用通常的免疫算法随机产生种群的方式,数字的取值会均匀的分布在解的空间中,减少了可能出现的数据冗余^[15]。

考虑到 Logistic 混沌迭代映射

$$x_{n+1} = \mu x_n (1 - x_n), \quad (2)$$

其中 μ 为控制系统混沌行为的参数, $0 \leq x_n \leq 1, n=0, 1, 2, \dots$ 。当 $\mu=4$ 时,对任意 n 个具有微小差异的初值。将生成的混沌序列从 $[0, 1]$ 混沌空间映射为系统状态空间,混沌系列能够在系统状态空间中均匀分布。

Logistic 混沌变异如表 1 所示

表 1 Logistic 混沌变异

算法 1: Logistic 混沌变异
Step 1: 初始化进化代数 $t=0$, 并随机产生初始种群 $P(t)$ 。
Step 2: 计算抗体群中每个抗体的亲和度值和样本值。
Step 3: 将抗体群进行等价区间划分, 并计算抗体群的平均信息熵。
Step 4: 对亲和度样本值小于平均信息熵的抗体 x_i 进行解码, 将抗体的每个基因 $x_i^{(j)}$ 映射到区间 $[0, 1]$ 内, 即:
$y_i^{(j)} = \frac{x_i^{(j)} - a_i}{b_i - a_i} \quad i = 1, 2, \dots, n, j = 1, 2, \dots, m$
Step 5: 利用 Logistic 方程进行混沌迭代, 得到混沌基因系列:
$y_{i+1}^{(j)} = 4 \times y_i^{(j)} \times (1 - y_i^{(j)})$
经过混沌迭代后, 将基因系列 $x_i^{(j)}$ 产生新的抗体记为 x_i' ;
Step 6: 计算抗体 x_i' 的亲和度 $f(x_i')$ 。如果 $f(x_i') \geq f(x_i)$, 则新抗体 x_i' 替代 x_i
Step 7: $t=t+1$, 转到 Step 2。

2.4 选择

为了将优秀的抗体直接保留到下一代, 对亲和度较高(亲和度样本值较小)的抗体进行选择操作。对种群 $X(t)$ 执行的选择操作 $\Psi_s(X(t))$ 如下定义

$$\begin{aligned} \Psi_s(X(t)) &= \Psi_s(X_1, X_2, \dots, X_N) = \\ \Psi_s(X_1) \cup \Psi_s(X_2) \cup \dots \cup \Psi_s(X_N) &= \\ \Psi_s(X_N) &= \\ \{X_N^{1,1}, X_N^{1,2}, \dots, X_N^{1,\#(X_N)}\} \end{aligned}$$

根据等价划分策略, 不难分析出亲和度最大和最小的抗体分别落在块 X_N 和 X_1 。直接将亲和度较大的块 X_N 中所有抗体选择到下一代。

2.5 混沌克隆优化算法 CCOA

混沌克隆优化算法的步骤详细描述如表 2 所示。

表 2 混沌克隆优化算法

算法 2: 混沌克隆优化算法
Step 1: 初始化进化代数 $t=0$, 最大进化代数 MaxGen, 并随机产生初始种群 $P(t)$ 。
Step 2: 评价种群 $P(t)$ 中每个抗体的亲和度值和样本值的亲和度。
Step 3: 将抗体群进行等价区间划分, 并计算抗体群的平均信息熵。
Step 4: 对种群 $P(t)$ 执行区间克隆操作;
Step 5: 对种群 $P(t)$ 执行 Logistic 混沌变异操作;
Step 6: 对种群 $P(t)$ 执行选择操作;
Step 7: $t=t+1$; 如果 $t \leq \text{MaxGen}$, 则返回到步骤 2; 否则, 算法结束, 输出最优解。

3 CCOA 的收敛性分析

1990 年 Eiben 提出 GA 的一种抽象表示, 将进化过程定义为马尔可夫链, 利用转移概率矩阵相乘来进行状态的变换, 分析得出 GA 收敛到最优解的条件。Markov 链模型是研究 GA 算法、CSA 以及相关改进算法的概率行为的最普遍, 最自然的方法。在文献 [16] 中, 作者提出了一种变异协同进化免疫算法 MCCLONALG, 该算法的主要构成部分和经典的克隆选择算法基本类似, 在变异时将抗体集和变异集协同进化, 最后用 Markov 链证明了 MCCLONALG 算法是以概率 1 收敛的。文献 [17] 将遗传算法、混沌优化和变尺度方法相结合, 提出了一种改进的混沌优化算法 ICOA (improved chaotic optimization algorithm, ICOA), 该算法将混沌映射函数应用到进化过程中, 最后证明了 ICOA 是全局渐进收敛。在文献 [18] 中, 作者提出了一种新的遗

传混沌优化组合方法 GCOCM (genetic chaos optimization combination method, GCOCM), 该算法在传统遗传算法的基础上, 采用混沌变异的方式, 最后证明了该算法是以概率 1 收敛到全局最优解。

在 CCOA 算法中, 初始种群的产生过程是一个随机过程, 每一次新的种群产生只与它的父代种群相关, 并且种群所处的状态为离散状态, CCOA 算法的这些特点基本上符合 Markov 链的特征。因此, 下面将用 Markov 链对 CCOA 算法的收敛性进行分析。

定理 1: CCOA 的种群序列 $\{X_k, k \geq 0\}$ 是有限的齐次 Markov 链。

证明 用 m 表示为每个抗体的基因位数, l 为每位基因用二进制表示的位数, N 为种群的规模。由于 CCOA 采用二进制编码, 抗体的取值是离散的 0 和 1, 用 $\Omega = \{\varphi\}^{ml}$ 表示抗体的种群空间, $\varphi \in \{0, 1\}$, 每个抗体 $x_i^t \in \{\varphi\}^{ml}$, 种群所在的状态空间大小为 $N \times 2^{ml}$ 。因此, 种群是有限的。

CCOA 的种群 X_{k+1} 产生只与其父种群 X_k 有关, 即转移概率只与种群状态有关, 并且种群所处的状态为离散状态, 算法中采用的区间克隆、Logistic 混沌变异和选择操作都与时刻 k 无关, 因此

$$P_{ij} = P\{X_{n+1} = j \mid X_n = i\}, \quad (3)$$

所以 CCOA 的种群序列 $\{X_k, k \geq 0\}$ 是有限的齐次 Markov 链。

设 $X_k = \{x_1, x_2, \dots, x_N\}$, 下标 k 表示进化代数, X_k 表示进化到第 k 代的一个种群, x_i 表示第 i 个个体^[19]。设 f 是 X_k 上的亲和度函数, 令

$$S^* = \{x_i \mid \max_{x_i \in X_k} f(x_i) = f^*\}, \quad (4)$$

称 S^* 为全局最优解集, 其中 f^* 为全局最佳亲和度。

定义 1 设算法对应的种群序列 $\{X_k, k \geq 0\}$ 。如果当且仅当

$$\lim_{k \rightarrow \infty} P\{X(k) \subset S^*\} = 1,$$

则称算法是收敛的。

在文献[20]中, 可以得到以下定理:

定理 2 克隆选择算法 CSA 是以概率 1 收敛的。

从定理 1 中, 可以得出: CCOA 算法和其他 CSA、MCCLONALG、GCOCM、ICOA 算法一样, 它的种群序列都是符合有限的齐次 Markov 链。在定理 2 的基础上, 得出下面的结论

推论 1 CCOA 是以概率 1 收敛的。

证明: 提出的混沌克隆优化算法, 是在克隆选择算法的基础上, 引入了混沌变异的思想。由于设计的 CCOA 算法, 只是在经典的克隆选择算法的基础

上, 采用的混沌变异改良种群, 提出的等价区间划分思想, 是用来引导种群的克隆和变异操作, 这样可以提高算法的执行效率, 并没有改变克隆选择算法的本质特征, 因此定理 2 对 CCOA 算法同样适用。设 $\{X_k, k \geq 0\}$ 是 CCOA 的一个随机变量序列, 那么当算法迭代到足够多的次数后, 下面公式一定成立,

$$\exists x_i \in X(k) : \lim_{k \rightarrow \infty} P\{X(k) \subset S^*\} = 1.$$

通过上述的推导证明, 群体中包含全局最优解的概率接近 1, CCOA 是以概率 1 收敛的。

4 CCOA 的计算复杂度分析

计算复杂度是评价一种算法搜索性能的关键指标, 将在本节里分析 CCOA 算法的计算复杂度, 主要通过算法指令被执行最大次数来估计算法运行所需的计算量。

CCOA 算法的种群规模和抗体的基因位数分别为 N 和 l , 算法主要包括以下 5 个操作: 1) 计算抗体的亲合度和样本值; 2) 进行等价区间划分, 并计算平均信息熵; 3) 区间克隆操作; 4) Logistic 混沌变异; 5) 选择操作。在抗体群中, 由于要对每个抗体计算亲合度和样本值, 所以操作 (1) 的执行最大次数是 $2N$, 计算量至多为 $O(2N)$ 。操作 2) 对抗体群进行等价区间划分所需的计算量至多为 $O(\log_2 N)$ 。操作 3) 采取的区间克隆操作所需的计算量至多为 $O(Nl)$ 。操作 4) 采取的 Logistic 混沌变异所需的计算量至多为 $O(N^2)$ 。根据对选择操作的定义, 操作 5) 的计算量至多也为 $O(1)$ 。因此, 算法总的计算复杂度可以表示为 $O(2N + \log_2 N + Nl + N^2 + 1)$ 。由此推出 CCOA 算法的计算最高级为 $O(N^2)$, 这表明种群规模直接影响到算法的搜索速度。在 CSA 算法中, 抗体要依次经过克隆, 变异和选择操作, 计算复杂度的级别也为 $O(N^2)$ 。但是, CCOA 算法通过引入等价划分的策略, 能有效引导算子进行局部最优搜索, 减少了可能出现的数据冗余现象, 因而同一进化代数中能更快更好地找到满意解, 使得满足同样收敛条件的时间要快于 CSA 算法。

5 仿真实验与结果分析

为了更有效地评估设计的算法性能, 选择以下 6 个目标函数, 将 CCOA 算法与文献[17]的 ICOA 算法和文献[18]的 CSA 算法的性能进行对比实验。设计的测试函数如下

$$F_1 = \sum_{i=1}^n x_i^2;$$

$$F_2 = 100(x^2 - y)^2 + (1 - x)^2;$$

$$F_3 = (x^2 + y^2)^{0.25} \times \{\sin^2[50(x^2 + y^2)^{0.1}] + 1\};$$

$$F_4 = x^2 + 2y^2 - 0.4\cos(4\pi x) - 0.6\cos(6\pi y) + 1;$$

$$F_5 = \sum_{i=1}^n (x_i^2 - 10\cos(2\pi x_i) + 10);$$

$$F_6 = 0.5 - \frac{\sin^2 \sqrt{x^2 + y^2} - 0.5}{[1 + 0.001(x^2 + y^2)]^2}.$$

表 3 列出了每个测试函数的基本特征。

图 1-3 分别是函数 F_2, F_3, F_4 的部分三维曲面。

表 3 测试函数的基本特征

函数	变量取值范围	维数	全局最优解	特点
F_1	$x \in [-5, 5]$	$n=3$	最小值: $F_1(0, 0) = 0$	三维, 单峰的二次函数
F_2	$x, y \in [-2, 2]$	—	最大值: $F_2(-2, -2) = 3\ 609$	单峰, 非凸病态函数, 难以最小化
F_3	$x, y \in [-5, 5]$	—	最小值: $F_3(0, 0) = 0$	多峰(谷)、多变量、多极值点
F_4	$x, y \in [-5, 5]$	—	最小值: $F_4(0, 0) = 0$	多峰, 大量局部极小值
F_5	$x \in [-5, 5]$	$n=10$	最小值: $F_5(0, 0) = 0$	多峰, 大量局部极小值
F_6	$x, y \in [-100, 100]$	—	最大值: $F_6(0, 0) = 1$	多峰, 大量局部极大值

在实验中, 每种算法对 6 个测试函数各运行 50 次, 记录算法每次运行找到最优解的最大进化代数、最小进化代数、平均进化代数、搜索到的最优解和最差解、平均最优值、收敛到全局最优解的次数, 这些实验结果记录在表 4 中。实验参数设置如下: 交叉概率均为 0.75, 变异概率均为

0.15, 函数 F_1 和 F_2 的种群规模为 100, 函数 F_3, F_4 和 F_5 的种群规模为 50, 函数 F_6 的种群规模为 60。函数 F_1 和 F_2 的最大进化代数为 250, 函数 F_3 和 F_4 的最大进化代数为 1 000, 函数 F_5 的最大进化代数为 500, 函数 F_6 的最大进化代数为 800。

表 4 测试函数的仿真实验结果

函数	算法	最优解	最差解	平均最优值	找到最优解的进代次数			收敛到全局最优解次数
					最大	最小	平均	
F_1	CSA	0.932 7	34.122 4	2.347 5	234	150	186	0
	ICOA	0	32.012 4	1.975 6	245	132	187	5
	CCOA	0	10.425 8	1.056 7	225	85	140	18
F_2	CSA	2 741.45	1 328.23	2 685.87	196	80	124	0
	ICOA	3 021.23	1 435.72	3 980.53	234	185	201	0
	CCOA	3 608.80	1 505.24	3 418.87	248	118	151	10
F_3	CSA	4.351 7	0.442 3	1.024 5	871	650	679	0
	ICOA	0.011 8	4.452 6	0.683 4	970	657	814	11
	CCOA	0	3.381 6	0.127 3	812	581	672	14
F_4	CSA	0.010 2	51.872 3	1.012 4	1 000	659	763	7
	ICOA	0.011 2	44.724 5	0.654 5	960	725	749	15
	CCOA	0.001 1	39.821 3	0.487 4	821	650	678	21
F_5	CSA	0.241 2	7.875 4	0.735 3	460	254	278	0
	ICOA	0	8.769 1	0.347 2	480	274	308	13
	CCOA	0	6.381 2	0.186 6	298	176	213	19
F_6	CSA	1	0.208 3	0.952 6	754	419	517	17
	ICOA	0.997 6	0.311 4	0.976 2	715	476	512	29
	CCOA	1	0.182 5	0.987 2	534	400	447	31

从表 4 中,CCOA 算法的寻优能力最强,CSA 算法的寻优能力最差。特别是收敛到全局最优解次数和最优解这两项,表现得最为突出。三种算法在函数 F_3, F_4, F_5 和 F_6 中的进化代数明显大于在函数 F_1 和 F_2 ,这是因为函数 F_4 和 F_5 存在大量局部极小值,函数 F_6 存在大量局部极大值,函数 F_3 拥有多个极值点。在表 4 中,CCOA 算法能在较多的情况下跳出局部最优,这表明 CCOA 算法能有效克服局部最优,避免早熟收敛现象。从最差解和最优解这两项的波动区间来看,CCOA 算法表现的最为稳定。实验中还发现,当函数的维数 n 增加时,发现 CCOA 算法所增加的时间不是特别明显,这表明 CCOA 算法具有良好的可扩展性。

图 4-9 分别是三种算法在函数 $F_1 \sim F_6$ 中搜寻优化解的性能比较。在图 4 中,CCOA 算法在前期优势表现得非常明显,后期三种算法的性能趋向一致,这表明 CCOA 算法的收敛速度明显快于 CSA 和 ICOA 算法。在图 5 和图 6 中,虽然前期三种算法对函数优化的进化趋势大致相同,但随着算法更深层次的迭代,CCOA 算法表现出更好的性能。在图 7~9 中,三种算法的寻优能力差别不大。通过应用设计的算法对以上几个性能各异的经典函数进行测试,实验结果显示了算法具有较快的全局收敛速度和较强的搜索能力,能有效地克服和避免遗传算法的早熟收敛现象,同时具有良好的可扩展性和稳定性。

$$(x^2-y^2)^{0.25}((\sin(50(x^2+y^2)^{0.1}))^2+1)$$

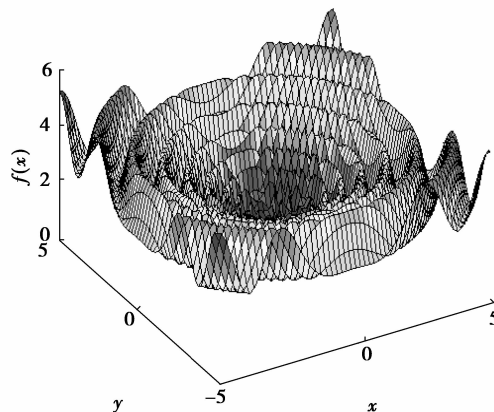


图 2 函数 F_3 的部分三维曲面

$$x^2+2y^2-0.4 \cos(4 \pi x)-0.6 \cos(4 \pi y)+1$$

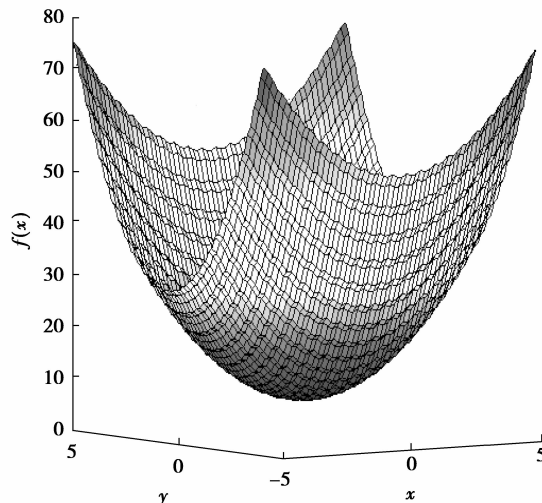


图 3 函数 F_4 的部分三维曲面

$$100(x^2-y)^2+(1-x)^2$$

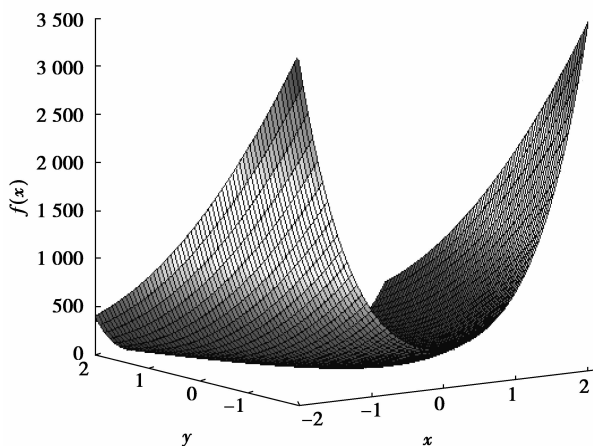


图 1 函数 F_2 的部分三维曲面

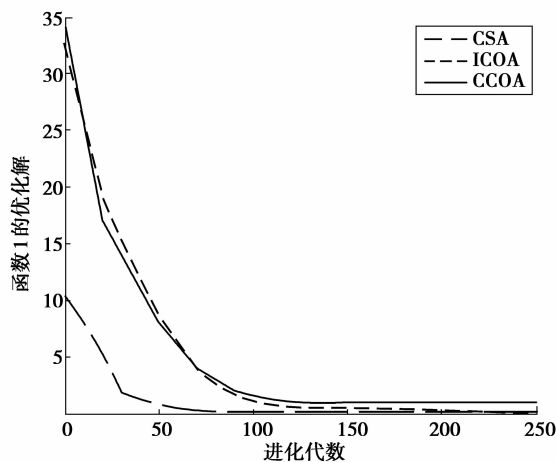


图 4 三种算法在函数 F_1 中的性能比较

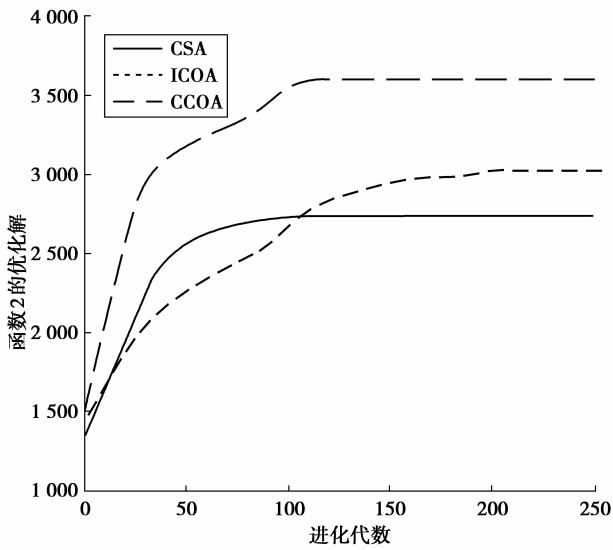


图 5 三种算法在函数 F_2 中的性能比较

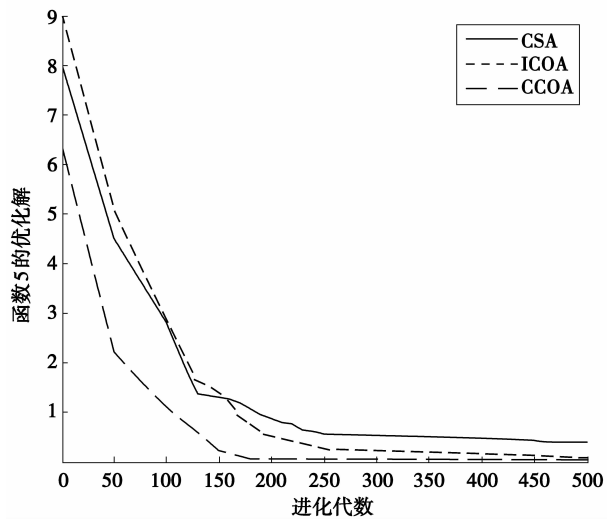


图 8 三种算法在函数 F_5 中的性能比较

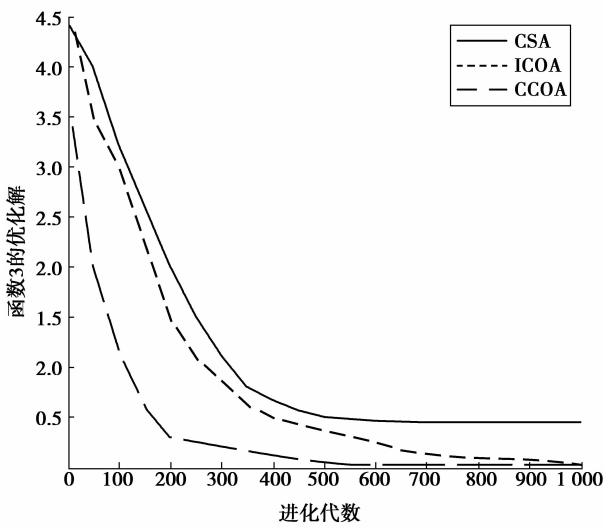


图 6 三种算法在函数 F_3 中的性能比较

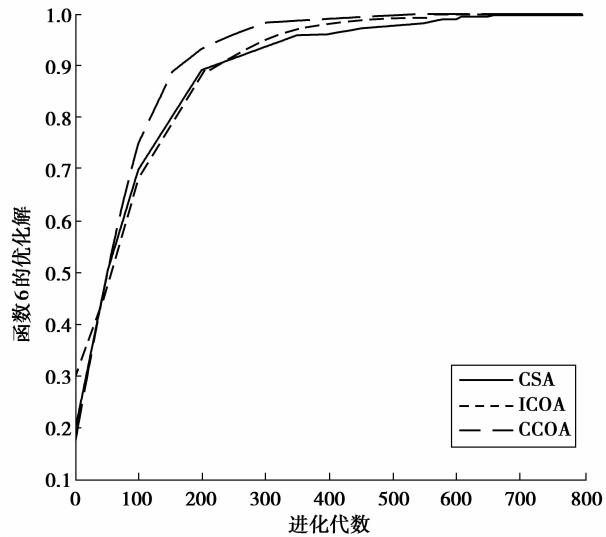


图 9 三种算法在函数 F_6 中的性能比较

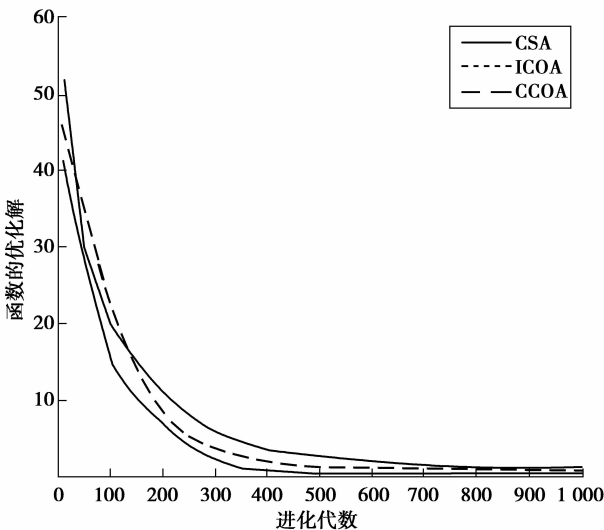


图 7 三种算法在函数 F_4 中的性能比较

6 结 论

为了克服克隆选择算法随机产生种群的方式, 从而造成数字的取值非均匀分布在解的空间, 导致增加数据冗余的问题。将克隆选择算法和混沌优化相结合, 提出了一种 CCOA 优化算法。将克隆选择算法的分散性, 独立性和混沌优化的遍历性、随机性、规律性等优点有效的进行结合, 充分的发挥了它们的优越性, 仿真实验表明设计的算法优于 CSA 算法和 ICOA 算法, 同时具有良好的扩展性和稳定性, 能有效地应用到函数的优化问题中来。

参考文献:

[1] De castro L N, Von zuben F J. Learning and optimization using the clonal selection principle [J].

- IEEE Transactions on Evolutionary Computation, 2002, 6(3): 239-251.
- [2] 刘丽珏, 蔡自兴. 变异协同进化的免疫克隆算法[J]. 控制与决策, 2008, 23(6): 613-618.
Liu Lijue, Cai Zixing. Immunity clone algorithm with mutation co-evolution [J]. Control and Decision, 2008, 23(6): 613-618.
- [3] 莫宏伟, 金鸿章. 用于函数优化的改进免疫克隆多样性算法[J]. 哈尔滨工程大学学报, 2004, 25(1): 76-79.
Mo Hongwei, JIN Hongzhang. The modified immune diversity algorithm used in function optimization [J]. Journal of Harbin Engineering University, 2004, 25(1): 76-79.
- [4] 张著洪, 黄席樾. 一种新的免疫算法及其在多模态函数优化中的应用[J]. 控制理论与应用, 2004, 21(1): 17-21.
Zhang Zhuhong, HUANG Xiyue. Novel immune algorithm and its application to multi-modal function optimization [J]. Control Theory & Application, 2004, 21(1): 17-21.
- [5] 郑皎凌, 唐常杰, 徐开阔, 等. 基于协同进化的异构种群挖掘混沌迭代函数[J]. 计算机学报, 2010, 33(4): 672-685.
Zheng Jiaoling, Tang Changjie, Xu Kaikuo. Mining chaotic iterative functions by co-evolution over heterogeneous populations [J]. Chinese Journal of Computers, 2010, 33(4): 672-685.
- [6] Chen H H, Shen G J, Lin Y L, et al. Chaos synchronization between two different chaotic systems via nonlinear feedback control [J]. Nonlinear Analysis: Theory, Methods & Application, 2009, 70(12): 4393-4401.
- [7] Li B, Jiang W S. Optimizing complex functions by chaos search [J]. An International Journal of Cybernetics and Systems, 1998, 29(4): 409-419.
- [8] 袁晓辉, 袁艳斌, 王乘, 等. 一种新型的自适应混沌遗传算法[J]. 电子学报, 2006, 34(4): 708-712.
Yuan Xiaohui, Yuan Yanbin, WANG Cheng, et al. A novel self-adaptive chaotic genetic algorithm [J]. Acta Electronic Sinica, 2006, 34(4): 708-712.
- [9] 姚俊峰, 梅焜, 彭小奇. 混沌遗传算法 CGA 的应用研究及其优化效率评价[J]. 自动化学报, 2002, 28(6): 935-943.
Yao Junfeng, Mei Chi, Peng Xiaoqi. The application research of the chaos genetic algorithm and its evaluation of optimization efficiency [J]. Acta Automatic Sinica, 2002, 28(6): 935-943.
- [10] 杨宇明, 李传东. 混沌在遗传算法中的应用[J]. 计算机工程与应用, 2004, 140(5): 76-77, 118.
Yang Yuming, Li Chuandong. Chaos in genetic algorithms [J]. Computer Engineering and Applications, 2004, 140(5): 76-77, 118.
- [11] 雷德明. 利用混沌搜索全局最优解的一种混合遗传算法[J]. 系统工程与电子技术, 1999, 21(12): 81-82.
Lei Deming. A hybrid genetic algorithm using chaos for globally optimal solution [J]. Systems Engineering and Electronic, 1999, 21(12): 81-82.
- [12] Chen H, Zhang J S, Zhang C. Chaos updating rotated gates quantum-inspired genetic algorithm [C] // Proceedings of the 2004 International Conference on Communication, Circuits and System, June 27-29, 2004, Chengdu, China; IEEE, 2004, 2: 1108-1112.
- [13] 肖人彬, 王磊. 人工免疫系统: 原理、模型、分析及展望[J]. 计算机学报, 2002, 25(12): 1281-1293.
Xiao Renbin, Wang Lei. Artificial immune system: principle, models, analysis and perspectives [J]. Chinese Journal of Computers, 2002, 25(12): 1281-1293.
- [14] Upadhyay R K. Dynamics of an ecological model living on the edge of chaos [J]. Applied Mathematics and Computation, 2009, 210(2): 455-464.
- [15] 李亚东, 李少运. 一种新的遗传混沌优化组合方法[J]. 控制理论与应用, 2002, 19(1): 143-145.
Li Yadong, Li Shaoyun. A new genetic chaos optimization combination method [J]. Control Theory & Applications, 2002, 19(1): 143-145.
- [16] 吴建辉, 章兢, 张小刚, 等. 一种求解 TSP 问题的分层免疫算法[J]. 计算机科学, 2010, 37(6): 256-260.
Wu Jianhui, Zhang Jing, Zhang Xiaogang, et al. Novel hierarchical immune algorithm for TSP solution [J]. Computer Science, 2010, 37(6): 256-260.
- [17] 费春国, 韩正之. 一种改进的混沌优化算法[J]. 控制理论与应用, 2006, 23(3): 471-474.
Fei Chunguo, Han Zhengzhi. An improved chaotic optimization algorithm [J]. Control Theory & Applications, 2006, 23(3): 471-474.
- [18] 焦李成, 尚荣华, 马文萍, 等. 多目标优化免疫算法、理论和应用[M]. 第1版. 北京: 科学出版社, 2010: 287-291.
- [19] 黄友锐. 智能优化算法及其应用[M]. 北京: 国防工业出版社, 2008: 62-68.