第 36 卷第 1 期
2019 年 1 月

控 制 理 论 与 应 用
Control Theory & Applications

Vol. 36 No. 1
Jan. 2019

弱连接多子群分子动理论优化算法

范朝冬[1,2], 刘颖南[1], 章　兢[1,3], 易灵芝[1], 肖乐意[3†]

(1. 湘潭大学 信息工程学院, 湖南 湘潭 411105; 2. 广西高校复杂系统与智能计算重点实验室, 广西 南宁 530006;
3. 湖南大学 电气与信息工程学院, 湖南 长沙 410082)

摘要: 针对分子动理论优化算法拓扑结构与"群集"现象的不足, 提出了一种弱连接多子群分子动理论优化算法. 该算法分为上下两层, 下层由一系列分子子群执行启发式快速搜索, 以提高算法的收敛速度; 上层中的混沌扰动子群基于混沌扰动机制, 以便停滞状态的分子子群能跳出局部极值; 上层中的免疫局部学习子群选取下层中的部分优秀个体进行局部学习, 以实现精细化搜索而提高算法的收敛精度. 仿真结果表明, 该算法在寻优精度、收敛速度以及求解偏移函数等方面均有良好的性能.

关键词: 分子动理论优化算法; 多子群; 弱连接; 群集现象; 混沌扰动

引用格式: 范朝冬, 刘颖南, 章兢, 等. 弱连接多子群分子动理论优化算法. 控制理论与应用, 2019, 36(1): 108 – 119

DOI: 10.7641/CTA.2018.70714

A weak linked multi-subpopulation kinetic-molecular theory optimization algorithm

FAN Chao-dong[1,2], LIU Ying-nan[1], ZHANG Jing[1,3], YI Ling-zhi[1], XIAO Le-yi[3†]

(1. College of Information Engineering, Xiangtan University, Xiangtan Hunan 411105, China;
2. Key Laboratory of Guangxi High Schools Complex System and Computational Intelligence, Nanning Guangxi 530006, China;
3. College of Electrical and Information Engineering, Hunan University, Changsha Hunan 410082, China)

Abstract: For overcoming the shortcomings of the topology and the 'cluster' phenomenon in the kinetic-molecular theory optimization algorithm (KMTOA), based on chaotic mapping and elite learning strategy, a weak linked multi-subpopulation kinetic-molecular theory optimization algorithm (WLMS–KMTOA) is proposed in this paper. WLMS–KMTOA includes two layers. In the lower layer, some subgroups perform heuristic search to improve the convergence rate of WLMS–KMTOA. In the upper layer, WLMS–KMTOA uses the chaotic sequence subpopulation to avoid falling into local optimum, and uses immune local learning subgroup to perform a refined search to improve the convergence accuracy. The simulation results show that WLMS–KMTOA has good performance in solution precision and convergence speed, and can be well applied to the functions with different shift values.

Key words: kinetic-molecular theory optimization algorithm; multiple subpopulations; weak link; clustering phenomenon; chaotic perturbation

Citation: FAN Chaodong, LIU Yingnan, ZHANG Jing, et al. A weak linked multi-subpopulation kinetic-molecular theory optimization algorithm. *Control Theory & Applications*, 2019, 36(1): 108 – 119

1　引言

分子动理论优化算法(kinetic-molecular theory optimization, KMTOA), 是由范朝冬博士于2013年提出的一种基于分子动理论的新型智能优化算法[1]. 该算法依据分子热运动中的吸引–排斥–波动机制设计了3种算子, 分子在不同算子的作用下迭代更新, 从而使种群在最优个体的引导下向最优解逼近. 与蚁群算法(ant colony optimization, ACO)[2]、人工蜂群算法(artificial bee colony algorithm, ABC)[3]、遗传算法(genetic algorithm, GA)[4]等智能优化算法相比, KMTOA原理简单、参数少、收敛快、效果好, 在实际应用中已表现出良好的优化性能[5]. 然而, 与绝大多数部分智

收稿日期: 2017–09–30; 录用日期: 2018–03–26.

†通信作者. E-mail: xiaolyttkx@163.com; Tel.: +86 15116143740.

本文责任编委: 陈增强.

国家自然科学基金项目(61573299, 51677063), 湖南省自然科学基金项目(2016JJ3125), 湖南省教育厅科学研究项目(15C1327), 湘潭大学校级课题(15XZX31), 广西高校复杂系统与智能计算重点实验室基金(2017CSCI04)资助.

Supported by the National Natural Science Foundation of China (61573299, 51677063), the Natural ScienceFoundation of Hunan Province (2016JJ3125), the Foundation of Hunan Educational Committee (15C1327), the Project of Xiangtan University (15XZX31) and the Foundation of Key Laboratory of Guangxi High Schools Complex System and Computational Intelligence (2017CSCI04).

能优化算法一样, KMTOA仍然存在缺乏局部搜索机制和错误性引导等不足[6].

针对群智能优化算法存在易陷入局部极值和收敛精度差等问题, 国内外学者引入了多子(种)群思想通过平衡算法的"局部开发"和"全局探索"能力而提高算法的性能[7]. 如: Rahnamayan等所提出的乌贼算法(cuttlefish algorithm, CFA)模拟乌贼细胞的变色机制, 将细胞种群分为4 个子群, 使用反射度和可见度作为全局最优的搜索策略, 大大提高了算法收敛速度[8]; Chang利用多子群的思想对微粒群算法(particle swarm optimization, PSO)进行改进, 成功求解了多模函数的多个极大值极小值问题[7]; 吴建辉等借鉴多子群与自适应思想, 使粒子群算法与免疫算法优势互补, 在高维与多模态函数优化问题上取得较好的效果[9]; 金敏等提出一种遗传算法与微粒群算法的多子群分层混合算法, 从子群组织结构角度较好地平衡了全局搜索与局部搜索, 在单模态和多模态问题上都取得了显著效果[10]. 此外, 在工程实际问题中, 多子群融合算法也取得了广泛应用, 如: 文献[11]和文献[12]分别将多种群遗传算法应用于多电平逆变器选择性谐波消除开关角计算和电力系统优化问题; 文献[13]中将多子群分层差分微粒群算法用于求解机器人逆运动学模型, 取得了良好的优化效果.

鉴于多种群思想在算法改进方面的重要作用, 综合KMTOA的拓扑结构与社会学中的"弱连接"理论, 本文提出了一种弱连接多子群分子动理论优化算法(weak linked multi-subpopulation kinetic-molecular theory optimization algorithm, WLMS–KMTOA). 该算法在检测到下层某个强连接分子子群的最优个体陷入停滞时, 跳出下层多子群的范围而将该子群中的个体送入上层的混沌扰动子群, 以增强算法的波动性而逃离局部极值; 上层的免疫局部学习子群, 对较优个体进行克隆与局部学习操作, 提高算法的收敛精度和精英保持能力. 仿真结果表明, 与蚁狮(ant lion optimizer, ALO)算法[14]、乌贼(cuttlefish algorithm, CFA)算法[8]和最优觅食(optimal foraging algorithm, OFA)[15]等算法相比, WLMS–KMTOA具有更高的寻优精度, 且在求解偏移函数的精度和收敛速度方面表现出明显优势.

2　KMTOA及其群集现象

2.1　KMTOA及其群集优势

KMTOA算法中的每个分子代表搜索空间中的一个解. 在搜索区域内, 各分子以一定的速度做加速度运动, 根据当前最优个体的位置动态调整自身的加速度及位置. 在KMTOA算法中, 分子i的速度和位置更新公式如下:

$$V_i(t+1) = (\omega_{\rm H} - \omega_{\rm L} \cdot \frac{t}{T}) \cdot V_i(t) + a_i, \quad (1)$$

$$X_i(t+1) = X_i(t) + V_i(t+1), \quad (2)$$

其中: V_i表示第i个分子的速度, X_i表示第i个分子的位置, a_i表示第i个分子的加速度, $\omega_{\rm H}$和$\omega_{\rm L}$为速度惯性调节因子. 式(1)中a_i分为3种情况: 引力、斥力和波动, 分别在不同的概率下决定分子的运动, 具体算法步骤和计算公式见文献[1].

由文献[1]可知: 与微粒群算法和拟态物理学优化算法等群体智能优化算法相比, KMTOA在求解F_1, F_2, F_3, F_4和F_6等大部分测试函数时都找到了全局最优解, 表现出明显优势. 为对其优势现象的本质进行深入分析, 本文首先进行了以下测试.

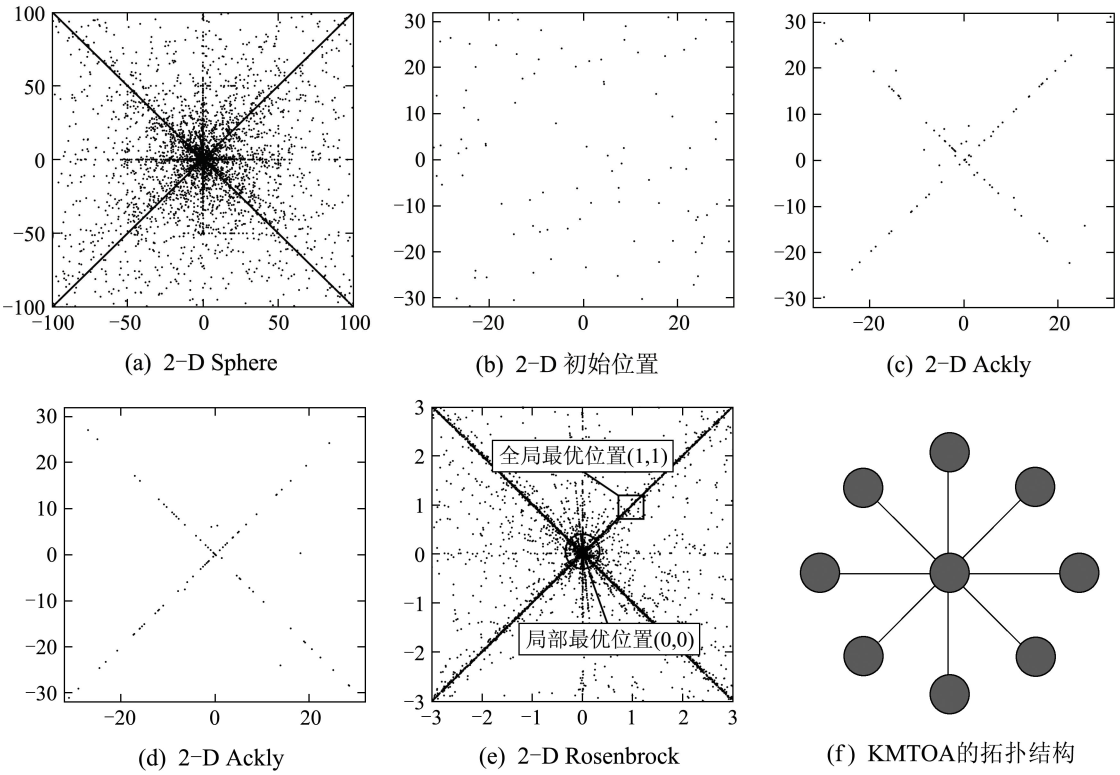
图1(a)为KMTOA对Sphere函数迭代500次的二维分子位置示意图, 共有25000 $(500*50)$个分子, 最优位置在(0, 0)处, 可以看到分子出现了惊人的群智能现象, 在搜索区域的两条对角线位置上产生了大量的分子. 为了进一步分析产生这种"群集"现象的原因, 图1(b)–1(d)记录了KMTOA在求解Ackly函数时的分子运动轨迹. 图1(b)为算法初始化100个分子的位置, 可以看出分子随机地分布在搜索空间中; 图1(c)已能清晰地看到大部分分子沿着搜索区域对角线位置的运动轨迹, 由于算法中存在一定概率的波动操作, 小部分分子随机出现在其它搜索区域; 图1(d)为算法迭代到100代时的分子分布图, 可以很清晰地看到与图1(c)出现了相同的现象, 说明这种"群集"现象是稳定存在的.

经过大量函数测试发现, 有两类函数会出现图1中从无序到有序的相变过程: 第1类函数为最优位置位于$(0, 0, \cdots, 0)$处且函数搜索区间对称的单峰测试函数; 第2类函数为部分全局最优值为0且搜索区间对称的多模函数. 可见, KMTOA算法这种内在的搜索机制是普遍存在的, 这从另一个角度解释了KMTOA 对于文献[1]中的大部分测试函数表现出优异性能的本质原因, 这是KMTOA算法相比其它群智能优化算法最大的优势.

2.2　KMTOA存在的不足

在分析KMTOA算法过程中, 发现以下几点不足:

1) KMTOA算法的上述"群集"现象虽具备一定的优势, 但也存在不足. 如: Rosenbrock函数的全局最优位置为$(1, 1, \cdots, 1)$, 而在寻优过程中发现算法快速收敛到$(0, 0, \cdots, 0)$后很难跳出此局部极值点, 图1(e)是截取[−3, 3]范围的分子二维位置示意图, 因此如何在不影响算法性能的情况下, 避免算法"先验"地陷入局部极值而影响算法后期的性能是算法改进的关键.



(a) 2–D Sphere　(b) 2–D 初始位置　(c) 2–D Ackly

(d) 2–D Ackly　(e) 2–D Rosenbrock　(f) KMTOA的拓扑结构

图 1 KMTOA算法群集现象与拓扑结构图

Fig. 1 The clustering phenomenon and topology of KMTOA algorithm

2) KMTOA中每个分子的位置与适应值包含了目标函数的大量信息, 在每次迭代过程中, 分子(包括最优个体)与最优个体进行吸引–排斥–波动操作, 最优个体类似于核心分子(引导分子), 引导其他分子趋向或远离当前最优个体. 算法的拓扑结构如图1(f)所示, 是一种星型拓扑结构, 每个分子仅与当前最优个体通信, 分子间大量有用的信息没有得到充分挖掘和利用.

这种星型拓扑以及算法本身的“群集”运动轨迹意味着KMTOA算法邻域搜索能力较弱, 对个体的响应存在不确定性. 如图2所示, 在应用KMTOA求解多峰Rastrigin函数时, 由每一代某单个分子的某一维与最优个体间距离变化图, 可以看出即使算法找到最优解, 整个分子系统依旧处于不稳定状态, 算法的寻优精度会受到影响, 而KMTOA对最优个体的保留策略又过于依赖精英保留策略, 说明其精英保留能力较弱.

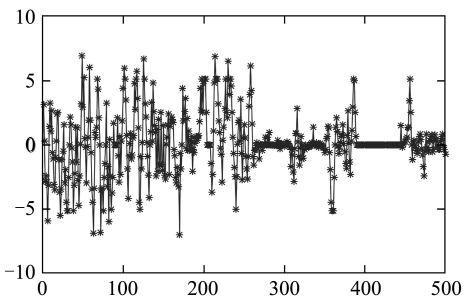


图 2 单个分子与最优个体间的距离

Fig. 2 Distance between a single molecule and an optimal individual

3) KMTOA算法的波动率和变异率较小且随机, 这无益于算法性能的提升. 算法的波动操作与吸引、排斥操作耦合紧密, 占有0.06的概率, 在实际求解问题过程中这对算法逃离局部最优、增加算法随机性有一定贡献, 所以需要加大算法的波动能力.

3 多子群弱连接模型

针对分子动理论优化算法的上述拓扑结构和优缺点分析, 设计了一种多子群弱连接模型对KMTOA算法进行改进. Granovetter于1973年指出: “弱连接”关系起到能决定社会最优结构的重要作用[16]. 所谓“弱连接”是指来自遥远的熟人带来的信息, 它能够提供一个圈子或集体不能得到的信息; “强连接”是指在一个关系密切的圈子中个人与个人之间有着频繁密切地联系, 这是一种能很快建立和接受的一种信息交流方式. 弱连接是在强连接的群体中架起一座桥梁, 以网络学习或者交流为主, 为那些没有直接联系的个体创造一条联系途径, 使新的观念能够在群体中传播. 显然, 由图1(f)可知, KMTOA中分子与最优个体之间是一种强连接关系. 为充分利用弱连接关系, 本文进行了以下改进:

定义 1　多子群弱连接拓扑结构.

本文所设计的多子群弱连接模型由5个子群所组成, 分为上下两层; 下层为分子动理论优化算法子群, 上层的左边是免疫局部学习子群(immune local learning subgroup, ILLG), 右边是混沌扰动子群 (chaotic perturbation subgroup, CPG), 多子群弱连接方式的拓扑结构如图3所示.

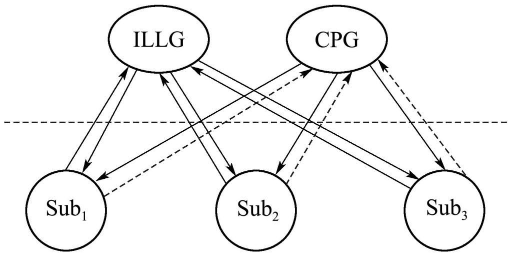


图 3 多子群弱连接拓扑图

Fig. 3 Multi-subpopulation weak connection topology

定义 2 下层分子动理论优化算法子群Sub_i.

Sub_i各个子群中有NP个体, Sub_i子群的初始个体按照文献[1]的方式生成, 其表达式为

$$x_{ij}^1 = L_i + \mathrm{rand}(0,1)(U_i - L_i). \tag{3}$$

受到反向学习原理[17]的启发, 将初始Sub_1搜索空间映射到另一个空间, 有助于改善算法性能, 拓宽分子搜索区域, Sub_2子群的初始个体生成表达式为

$$x_{ij}^2 = L_i + U_i - x_{ij}^1. \tag{4}$$

图4为Sub_2中个体映射到另一个空间示意图, 令$a = L_i,\ b = U_i$, a, b分别为搜索空间的下限和上限.

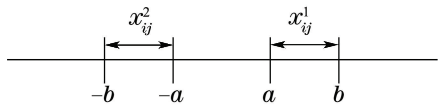


图 4 Sub_2中个体搜索空间示意图

Fig. 4 The diagram of the Sub_2's individual search space

同理, 将Sub_1搜索空间进一步映射到另一个空间, 避免3个子群趋同, Sub_3子群的初始个体生成表达式为

$$x_{ij}^3 = k \cdot (a_i + b_i) - x_{ij}^1, \tag{5}$$

其中: k为区间(0, 1)内的随机数, $a_i = \min(x_{ij}^1),\ b_i = \max(x_{ij}^1)$, 其个体映射空间如图5所示.

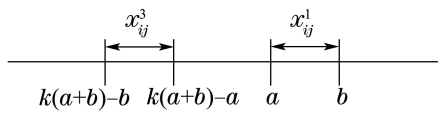


图 5 Sub_3中个体搜索空间示意图

Fig. 5 The diagram of the Sub_3's individual search space

此外, 受到文献[18]的启发, 对速度更新公式进行改进, Sub_1的速度公式中, 每个分子仅与最优个体通信, 而Sub_2, Sub_3引入了个体与个体之间的通信, 可以有效改善算法的优化性能:

$$\begin{cases} V_i(t+1) = (\omega_{\mathrm{H}} - \omega_{\mathrm{L}} \cdot \dfrac{t}{T}) \cdot V_i(t) + a_i, \\ V_i(t+1) = (\omega_{\mathrm{H}} - \omega_{\mathrm{L}} \cdot \dfrac{t}{T}) \cdot V_i(t) + a_i + \\ \qquad \varphi_i \cdot (x_i - x_k), \\ V_i(t+1) = x_{\mathrm{Best}} + a_i + \mu_i \cdot (x_i - x_k), \end{cases} \tag{6}$$

式中φ_i和μ_i分别是[0, 1.5]与[−1, 1]之间的随机数.

定义 3 混沌扰动子群.

Kent混沌序列产生的混沌变量具有随机性、遍历性等特点, 且在二维空间里分布比较均匀[19], 有利于算法跳出局部最优.

$$Z_{n+1} = \begin{cases} Z_n/\alpha, & 0 < Z_n \leqslant \alpha, \\ (1 - Z_n)/(1 - \alpha), & \alpha \leqslant Z_n \leqslant 1, \end{cases} \tag{7}$$

其中α为[0, 1]内的控制参数. 为提高算法的波动性能, 当前局部最小点连续g代更新陷入停滞时, 及时增大算法的扰动能力, 阻止种群陷入停滞, 设置算法停滞状态判断条件:

$$\gamma == g/T. \tag{8}$$

"$\gamma == g/T$"为判断"γ", "g/T"是否相等的布尔表达式. 当算法处于停滞状态时, 按方程(7)重新产生NP个混沌向量Y=$(Y_1, \cdots, Y_{NP})$, 以当前局部最小点X_{Best}为中心, $\dfrac{1}{2}R$为半径的邻域内混沌搜索, 混沌向量按公式(9)变换到解空间X_i, 利用生成的混沌个体X_i替换处于停滞状态的种群:

$$X_i = X_{\mathrm{Best}} + \frac{1}{2}R \cdot Y_i, \tag{9}$$

其中: $R = \rho \cdot (U_i - L_i)$, ρ为混沌搜索的收缩因子, 随迭代次数线性下降, 表示为

$$\rho_{t+1} = (1 - \beta \cdot t/T)\rho_t, \tag{10}$$

t为当前迭代次数, β为收缩因子控制参数.

定义 4 免疫局部学习子群.

下层分子动理论优化算法子群中的优秀个体送入免疫局部学习子群, 经过免疫克隆进化和优秀粒子间的局部学习操作, 增强算法的局部搜索能力, 提升算法的收敛精度.

1) 精英个体免疫克隆: 免疫系统中的抗原即为待解决的问题, 每个抗体即为问题的一个解, 从下层子群中选出精英个体, 将分子动理论优化算法中的适应度函数定义为亲和度函数, 即:

$$\mathrm{Affinity}(x_{\mathrm{m}}^{\mathrm{Sub}_i}) = \mathrm{fit}(x_{\mathrm{m}}^{\mathrm{Sub}_i}), \tag{11}$$

其中: $\mathrm{fit}(x_{\mathrm{m}}^{\mathrm{Sub}_i})$是每个子群$\mathrm{Sub}_i$选择的精英个体$x_{\mathrm{m}}$的适应度值, $\mathrm{Affinity}(x_{\mathrm{m}}^{\mathrm{Sub}_i})$为精英个体的适应度值, 按照以下公式进行克隆操作:

$$\beta(El_i) = \mathrm{round}(\mathrm{C} \cdot \alpha_{\mathrm{c}}(El_i)), \tag{12}$$

$$\alpha_{\rm c}(El_i)=(\sum_{j=1}^{N}\text{Affinity}(El_j))/\text{Affinity}(El_i),\quad(13)$$

其中: C为克隆常数; El_i为底层各子群选出的精英个体; N为下层子群的个数, $\text{Affinity}(El_j)$为精英个体El_j的亲和度值. 这样可以保证较优的精英个体能够被克隆多次, 提高算法的精英保留能力和收敛速度.

2) 免疫局部学习: 对精英个体进行免疫克隆, 提高算法的精英保留能力之后, 还应充分地利用精英个体. 因此, 设计精英局部学习机制对克隆后的精英个体进行局部微调, 以提高算法的精细搜索能力:

$$P'_{\rm id}=\begin{cases}P_{\rm id}+\varDelta(t,\dfrac{X_{\max}-P_{\rm id}}{\rm m}),\ \text{rand}\ \geqslant\sigma,\\ P_{\rm id}-\varDelta(t,\dfrac{P_{\rm id}-X_{\max}}{\rm m}),\ \text{其他},\end{cases}\quad(14)$$

式中: σ为方向阈值, 控制局部搜索的方向; $\varDelta(t,y)$ (y表示$\dfrac{X_{\max}-P_{\rm id}}{\rm m}$和$\dfrac{P_{\rm id}-P_{\max}}{\rm m}$)表示$[0,y]$范围内符合非均匀变异的一个随机数, m 为限幅常数. $\varDelta(t,y)$按照下式定义:

$$\varDelta(t,y)=y\eta_2(t),\ \eta_2(t)=1-r^{(1-t/T)^b},\quad(15)$$

式中: r为$[0,1]$之间均匀分布的随机数, b为系统参数.

4 WLMS–KMTOA算法流程

多子群弱连接分子动理论优化算法的具体过程如下:

Step 1 对下层N个子群初始化. 初始化算法的参数, 分别根据式(3)–(5)生成初始种群.

Step 2 计算子群中个体的适应值, 求出各子群的当前最优个体$X_{\rm Best}$.

Step 3 While $t<500$

For 每个子群中的个体$\rm Sub_i$

判断子群中个体吸引、排斥、波动条件, 并计算个体的加速度;

分别按照式(1)和式(2)更新分子的速度和位置;

更新各子群中的最优个体;

If $t<400$

判断各子群的最优个体是否停滞, 若停滞则执行Step 4, 否则$t=t+1$, 返回While;

Else If $400<t<500$

从$\{\rm Sub_1,Sub_2,Sub_3\}$中选择前10个精英个体, 执行Step 5;

End If

End For

End While

Step 4 按照式(7)生成混沌向量, 在规定的解空间范围内进行线性映射, 生成混沌个体及速度序列, 取代原种群的种群个体, 并计算适应值, 更新最优个体, 转回Step 3;

Step 5 对精英个体进行免疫克隆, 基于式(14)设计精英个体局部学习机制, 并更新最优个体, 转回Step 3;

Step 6 算法结束, 输出最优个体及其适应度值.

5 实验设计与结果分析

5.1 实验设计

实验采用MATLABR2014a进行仿真, 运行环境为WIN7平台下的Intel(R)Core(TM)i3处理器. 为测试改进算法的性能, 分别从文献[1]和文献[8]中选择20个基准测试函数作为测试对象, 算法参数设置:

$$\omega_{\rm H}=\beta=0.9,\ \omega_{\rm L}=\sigma=0.5,\ \alpha=0.4,$$
$$N=3,\ NP=50,\ T=500,\ C=b=2,$$
$$\rho_0=1,\ \gamma=0.01.$$

表1中F_1–F_5为单模态函数, 其中: F_2为区间不对称的Sphere函数, F_3为间断阶梯函数, F_5为噪声函数, 且都设置为100维; 表2中F_6–F_{14} 为多模态函数, 高维多模态函数局部极值点的数量随维度的增加呈指数增长, 求解难度较大, 能有效检验改进算法的性能; 表3中F_{15}–F_{20}为低维度的多模态函数, 虽无大量的局部极值点, 但函数的全局最优解往往不是0或搜索区间不对称, 具备良好的测试价值. 实验分别选取标准分子动理论优化算法、蚁狮算法、乌贼算法、最优觅食算法、增强差分进化算法(enhancing differential evolution utilizing eigenvector-based crossover operator, DE–EIG)[20]和竞争粒子群优化算法(Competitive swarm optimizer, CSO)[21]作为比较对象, 以验证改进算法优化性能.

表 1 单模态函数

Table 1 Unimodal functions

测试函数	维度	搜索范围	最优值
$F_1(x)=\sum_{i=1}^{D}x_i^2$	100	[−100, 100]	0
$F_2(x)=\sum_{i=1}^{D}x_i^2$	100	[−10, 190]	0
$F_3(x)=\sum_{i=1}^{D}\lvert x_i\rvert+\prod_{i=1}^{D}\lvert x_i\rvert$	100	[−10, 10]	0
$F_4(x)=\sum_{i=1}^{D}(\sum_{j=1}^{i}x_j)^2$	100	[−100, 100]	0
$F_5(x)=\sum_{i=1}^{D}ix_i^4+\text{rand}$	100	[−1.28, 1.28]	0

表 2 多模态函数

Table 2 Multimodal functions

测试函数	维度	搜索范围	最优值
$F_6=\sum\limits_{i=1}^{D-1}(100(x_i^2-x_{i+1})^2+(x_i-1)^2)$	100	[−50, 50]	0
$F_7=\sum\limits_{i=1}^{D}(\lfloor x_i+0.5\rfloor)^2$	100	[−10, 10]	0
$F_8=0.1\{10\sin^2(3\pi x_1)+\sum\limits_{i=1}^{D-1}(x_i-1)^2[1+\sin^2(3\pi x_{i+1})]+$ $(x_D-1)^2+[1+\sin^2(2\pi x_D)]\sum\limits_{i-1}^{D-1}(x_i-1)^2\}+\sum\limits_{i=1}^{D}u(x_i,5,100,4)$	100	[−10, 10]	0
$F_9=-\sum\limits_{i=1}^{D}(x_i\sin(\sqrt{\lvert x_i\rvert}))$	100	[−500, 500]	0
$F_{10}=\sum\limits_{i=1}^{D}(x_i^2-10\cos(2\pi x_i)+10)$	100	[−5.12, 5.12]	−41898.29
$F_{11}=20+e-20\exp(-0.2\sqrt{\frac{1}{n}\sum\limits_{i=1}^{D}x_i^2})-\exp(\frac{1}{n}\sum\limits_{i=1}^{D}\cos(2\pi x_i))$	100	[−32, 32]	0
$F_{12}=\begin{cases}\sin^2(\pi y_i)+\sum\limits_{i=1}^{D-1}[(y_i-1)^2(1+10\sin^2(\pi y_i+1))]+(y_i-1)^2(1+\sin^2(2\pi y_i))\\ y_i=1+\dfrac{x_i-1}{4}\end{cases}$	100	[−10, 10]	0
$F_{13}=\sum\limits_{i=1}^{D}\frac{x_i^2}{4000}-\prod\limits_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})+1$	100	[−600, 600]	0
$F_{14}=\sum\limits_{i=1}^{D}(i\times x_i^2)$	100	[−5.12, 5.12]	0

表 3 二维多模态函数

Table 3 2–D multimodal functions

测试函数	维度	搜索范围	最优值
$F_{15}=(x_1-x_2)^2+[(x_1+x_2-10)/3]^2$	2	[0, 10]	0
$F_{16}=-\cos x_1\cos x_2\exp(-(x_1-\pi)^2-(x_2-\pi)^2)$	2	[−100, 100]	0
$F_{17}=\sum\limits_{i=1}^{5}i\cos((i+1)x_1+i)\sum\limits_{i=1}^{5}i\cos((i+1)x_2+i)$	2	[−10, 10]	−186.7309
$F_{18}=[1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2)]\times$ $[30+(2x_1-3x_2)^2(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_1^2)]$	2	[−2, 2]	3
$F_{19}=(\frac{1}{500}+\sum\limits_{j=1}^{25}\frac{1}{i+\sum\limits_{i=1}^{2}(x_i-a_{ij})^6})^{-1}$	2	[−50, 50]	1
$F_{20}=(x_2-\frac{5.1^2}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6)^2+10(1-\frac{1}{8\pi})\cos x_1+10$	2	$-5\leqslant x_1\leqslant 10$ $0\leqslant x_2\leqslant 10$	0.3979

5.2 标准函数测试与分析

在测试过程中, 各算法的种群规模均设置为150, 最大迭代次数均为500, 独立运行50次所得到的最优值、平均值和标准差如表4–6所示. 其中, 最优值和平均值反映了解的质量及算法所能达到的精度, 标准差反映了算法的稳定性和鲁棒性. 综合3个表可知:

1) KMTOA与WLMS–KMTOA对F_1, F_{10}, F_{11}, F_{13}等非偏约束函数均收敛到理论最优值0. F_2为偏约束的Sphere函数, 与非偏约束Sphere函数F_1相比, KMTOA算法优化性能大为下降, 而WLMS–KMTOA算法有明显的优势. 2) CFA算法的数据选自文献[6] (—表示未对该函数进行测试), 因对不同的函数需要设置不同的参数[6], 故虽CFA算法的收敛速度快、精度高, 但算法参数的设置缺乏通用性, 调参困难, 在F_6, F_9等函数的优化性能上逊于WLMS–KMTOA算法. 3) CSO算法在部分函数上性能优于ALO, OFA和DE–EIG等算法, 但在除F_{12}的所有函数上求解精度均比WLMS–KMTOA算法低, DE–EIG算法在F_5, F_8, 和F_{12}三个函数上表现优于WLMS–KMTOA算法, 但WLMS–KMTOA算法的整体性能比DE–EIG算法更好. 4) WLMS–KMTOA算法在F_2, F_6, F_8等函数的收敛精度较高, 分别比KMTOA算法和ALO算法等高6个数量级以上. 这是因为WLMS–KMTOA算法采用了混沌扰动和免疫局部学习机制, 具有跳出局部极值和局部寻优能力.

表 4 单模态测试函数实验结果

Table 4 Experimental results on unimodal functions

测试函数	评价标准	DE–EIG	CSO	ALO	CFA	OFA	KMTOA	WLMS–KMTOA
	最优值	4.448e+04	4.165e−03	1.618	1.183e−93	7.419e+03	0	0
F_1	平均值	4.910e+04	6.492e−03	7.986	4.908e−82	1.064e+04	0	0
	标准差	8.691e+03	1.710e−03	2.866	1.675e−81	1.531e+03	0	0
	最优值	1.072e+05	9.862e+01	6.320		5.928e+04	1.138e+03	1.862e−97
F_2	平均值	1.245e+05	1.153e+02	17.751	—	7.219e+04	1.555e+03	4.661e−05
	标准差	8.945e+03	39.499	5.7224		5.592e+03	2.257e+03	1.064e–04
	最优值	9.262e−36	8.153e−03	35.471		66.001	0	0
F_3	平均值	2.927e−34	1.213e−02	288.532	—	76.659	0	0
	标准差	2.348e−34	1.625e−03	130.942		6.088	0	0
	最优值	3.697e−69	1.794e−73	15044.8678		1.767e−06	0	0
F_4	平均值	1.394e−65	1.251e−71	22044.0259	—	1.291e−03	0	0
	标准差	5.600e−65	1.805e−71	4140.0698		1.810e−03	0	0
	最优值	1.513e−06	3.961e−03	0.2841		1.199e+01	1.274e−05	8.405e−07
F_5	平均值	2.126e−05	6.741e−03	0.5519	—	20.736	6.309e−04	3.519e−04
	标准差	1.842e−05	1.095e−03	0.1464		4.636	6.138e−04	3.537e−04

表 5 多模态函数测试结果

Table 5 Experimental results on multimodal functions

测试函数	评价标准	DE–EIG	CSO	ALO	CFA	OFA	KMTOA	WLMS–KMTOA
	最优值	1.034e+08	7.799e−05	2325.809	94.8486	8.510e+07	9.862e+01	1.317e−10
F_6	平均值	2.226e+08	7.897e−04	11900.115	96.5564	1.391e+08	9.873e+01	2.250e−04
	标准差	6.024e+07	6.841e−04	10704.591	1.046	3.185e+07	6.387e−02	4.900e−04
	最优值	7.420e+02	0	2.607e−02		106	0	0
F_7	平均值	9.822e+02	0	7.583e−02	—	144.9	0	0
	标准差	9.622e+01	0	3.913e−02		17.623	0	0
	最优值	0	3.783e+02	196.835		1.982e+07	4.761	4.452e−05
F_8	平均值	0	4.882e+02	231.863	—	6.249e+07	7.065	4.145e−04
	标准差	0	1.553e+02	19.841		1.895e+07	9.604e−01	3.068e−03
	最优值	−8.730e+03	−4.189e+02	−18058.916	−2.091e+04	−1.170e+04	−3.115e+04	−4.091e+04
F_9	平均值	−7.061e+02	−4.189e+02	−18058.916	−1.795e+04	−1.033e+04	−2.789e+04	−3.623e+04
	标准差	1.764e+02	0	0	1.369e+03	4.88e+02	1.476e+03	2.276e+03
	最优值	1.056e+03	0	152.469	0	9.321e+02	0	0
F_{10}	平均值	1.108e+03	0	209.202	0	1.006e+03	0	0
	标准差	3.018e+01	0	46.286	0	2.551e+01	0	0
	最优值	1.721e+01	1.708	4.263	3.553e−15	1.510e+01	0	0
F_{11}	平均值	1.782e+01	1.708	6.198	3.553e−15	1.683e+01	0	0
	标准差	3.005e−01	0	1.131	0	4.904e−01	0	0
	最优值	3.420e−23	0	13.027		1.576e+02	2.220	4.617e−11
F_{12}	平均值	1.245e−17	1.838e−32	22.730	—	1.876e+02	3.135	2.589e−06
	标准差	4.065e−17	1.963e−32	5.315		1.474e+01	3.406e−01	6.023e−06
	最优值	3.744e+02	2.123e−09	0.691	0	7.148e+01	0	0
F_{13}	平均值	4.440e+02	3.489e−07	1.010	0	9.930e+01	0	0
	标准差	3.606e+01	4.116e−07	0.099	0	1.530e+01	0	0
	最优值	0	1.728e−146	21.101	4.924e−90	7.441e+02	0	0
F_{14}	平均值	0	9.305e−141	42.509	2.866e−83	9.294e+02	0	0
	标准差	0	2.944e−140	11.293	7.609e−83	1.164e+02	0	0

表 6 二维多模态函数测试结果

Table 6 Experimental results on 2–D multimodal functions

测试函数	评价标准	DE–EIG	CSO	ALO	CFA	OFA	KMTOA	WLMS–KMTOA
F_{15}	最优值	0	0	1.566e−17	0	0	0	0
	平均值	0	0	3.133e−15	0	1.358e−29	0	0
	标准差	0	0	2.848e−15	0	2.532e−29	0	0
F_{16}	最优值	−1	−1	−1	−1	−1	−1	−1
	平均值	−1	−1	−1	−1	−1	−1	−1
	标准差	0	0	0	0	0	0	0
F_{17}	最优值	−186.7309	−4	−186.7309	−186.7309	−186.7306	−186.7309	−186.7309
	平均值	−186.7309	−4	−186.7309	−186.7309	−186.7189	−186.7309	−186.7309
	标准差	0	0	0	0	1.088e−02	0	0
F_{18}	最优值	3.600e+01	3	3	3	3	3	3
	平均值	3.600e+01	3	3	3	3	3	3
	标准差	0	0	0	0	0	0	0
F_{19}	最优值	1.999	500	1.9921	1	1.99999	1	1
	平均值	3.748	500	1.9921	1	2.09989	1	1
	标准差	3.450	0	0	0	0.29992	0	0
F_{20}	最优值	2.605	8.789	0.3979		0.3979	0.3979	0.3979
	平均值	8.171	8.789	0.3979	—	0.3979	0.3979	0.3979
	标准差	1.887	0	0		0	0	0

5.3 Wilcoxon符号秩检验

表7和表8给出了Wilcoxon符号秩检验的结果, 其中: “R^+”表示WLMS−KMTOA算法样本数据优于其他6种算法的符号数; “R^-”表示逊于其他6种算法样本数据的符号数. 当两种算法存在显著差异且WLMS−KMTOA算法样本数据更优时, 用“+”号表示(反之, 则用“−”号表示); “=”号表示两种算法不存在较大差异[22]. 由表7–8可知: DE–EIG算法虽对F_5, F_8和F_{12}具备更好的优化性能, 但WLMS–KMTOA算法在13个函数上取得了更好的优化结果; CSO 算法仅对 F_{12} 取得相对更好的结果, 而 WLMS–KMTOA算法却对15个函数取得了更好的结果; ALO, CFA, OFA和KMTOA算法均没有在任何函数取得比WLMS−KMTOA算法更好的结果. 可见, 与其他算法相比, WLMS−KMTOA算法具备更强的寻优性能, 检验结果从另一个角度验证了算法改进的有效性.

5.4 偏移函数测试与分析

KMTOA算法在大部分二维测试函数中都表现出良好的性能, 因此需要进一步测试改进算法在最优值不为0、最优位置点不处于$(0, 0, \cdots, 0)$的多维(30维)偏移函数的优化性能. 为了检验WLMS–KMTOA算法对最优位置点不位于$(0, 0, \cdots, 0)$的偏移函数的优化效果, 引用文献[23]中的偏移指数与偏移量测试改进后的算法在不同偏移量下的优化性能. 如表9所示, U_k, L_k分别表示偏移函数搜索范围的最大最小值, SV为偏移量, SI为偏移指数, 以Sphere函数为例说明SV, SI: 当偏移指数为5时, 粒子x_i每一维上的偏移量为$0.5 \times ((100-(-100))/2) = 50$, 函数最优点由$(0, 0, \cdots, 0)$偏移到$(50, 50, \cdots, 50)$处, 搜索范围不变, 函数的最优适应度值仍为0. 偏移函数仿真实验参数设置为: 迭代次数500次, 种群个数150个, 独立运行次数50次.

表4–6是在函数无位置偏移情况下5种算法的实验结果, 而表10为不同偏移量情况下, WLMS–KMTOA算法和KMTOA算法的实验测试结果. 由表10可知, 从整体上看6种函数在不同位置偏移情况下, WLMS–KMTOA算法比KMTOA算法在寻优精度上有很大的提升, 即使对Rosenbrock这类复杂函数, 仍能取得良好的优化效果; 在$\mathrm{SI} = 5$时, 虽然WLMS–KMTOA与KMTOA在F_1, F_3, F_{13}都找到了理论最优值, 但WLMS–KMTOA算法不仅在F_4找到了理论最优值, 且在F_6, F_{11}两个函数上寻优精度也比KMTOA算法高. 这是因为WLMS–KMTOA算法基于多子群弱连接模型, 个体间的信息交流更为充分, 且采用混沌扰动机制增强了种群多样性, 避免了算法先验性地陷入$(0, 0, \cdots, 0)$局部位置; 免疫局部学习子群及精英保留策略提高了算法的精英保持能力, 使算法的收敛精度大为改善.

表 7 基于50次独立运行的Wilcoxon符号秩检验结果

Table 7 Wilcoxon signed ranks test based on 50 independent runs

函数	WLMS–KMTOA vs. DE–EIG				WLMS–KMTOA vs. CSO				WLMS–KMTOA vs. ALO			
	R^+	R^-	winer	p-value	R^+	R^-	winer	p-value	R^+	R^-	winer	p-value
F_1	1275	0	+	1.21e−12	1275	0	+	1.21e−12	1275	0	+	1.21e−12
F_2	1275	0	+	3.02e−11	1275	0	+	4.49e−13	1275	0	+	3.02e−11
F_3	1275	0	+	1.21e−12	1275	0	+	1.21e−12	1275	0	+	1.21e−12
F_4	1275	0	+	1.21e−12	1275	0	+	8.94e−14	1275	0	+	1.21e−12
F_5	20	1255	−	1.77e−10	1275	0	+	1.18e−10	1275	0	+	3.02e−11
F_6	1275	0	+	3.02e−11	1271	4	+	3.83e−06	1275	0	+	3.02e−11
F_7	1275	0	+	1.25e−12	0	0	=	1.00e+00	1275	0	+	1.21e−12
F_8	0	1275	−	1.21e−12	1275	0	+	6.73e−11	1275	0	+	3.02e−11
F_9	1275	0	+	1.68e−12	1275	0	+	1.21e−12	1275	0	+	1.24e−09
F_{10}	1275	0	+	1.21e−12	0	0	=	1.00e+00	1275	0	+	1.21e−12
F_{11}	1275	0	+	1.21e−12	1275	0	+	1.69e−14	1275	0	+	1.21e−11
F_{12}	0	1275	−	3.02e−11	0	1275	−	7.52e−12	1275	0	+	3.02e−11
F_{13}	1275	0	+	1.21e−12	1275	0	+	7.37e−13	1275	0	+	1.21e−12
F_{14}	0	0	=	1.00e+00	1275	0	+	7.37e−13	1275	0	+	1.21e−12
F_{15}	0	0	=	1.00e+00	0	0	=	1.00e+00	1275	0	+	1.21e−12
F_{16}	0	0	=	1.00e+00	0	0	=	1.00e+00	0	0	=	1.00e+00
F_{17}	0	0	=	1.00e+00	1275	0	+	1.69e−14	0	0	=	1.00e+00
F_{18}	1275	0	+	1.69e−14	0	0	=	1.00e+00	0	0	=	1.00e+00
F_{19}	1275	0	+	8.72e−07	1275	0	+	1.69e−14	1275	0	+	1.69e−14
F_{20}	1275	0	+	6.12e−14	1275	0	+	1.69e−14	0	0	=	1.00e+00
+/−/=	13/3/4				15/1/4				16/0/4			

表 8 基于50次独立运行的Wilcoxon符号秩检验结果

Table 8 Wilcoxon signed ranks test based on 50 independent runs

函数	WLMS–KMTOA vs. CFA				WLMS–KMTOA vs. OFA				WLMS–KMTOA vs. KMTOA			
	R^+	R^-	winer	p-value	R^+	R^-	winer	p-value	R^+	R^-	winer	p-value
F_1	1275	0	+	2.49e−12	1275	0	+	1.86e−14	0	0	=	1.00e+00
F_2			无		1275	0	+	3.43e−11	1275	0	+	3.01e−11
F_3			无		1275	0	+	4.69e−13	0	0	=	1.00e+00
F_4			无		1275	0	+	8.69e−13	0	0	=	1.00e+00
F_5			无		1275	0	+	6.39e−12	1266	9	+	2.65e−11
F_6	1275	0	+	3.02e−11	1275	0	+	3.02e−11	1275	0	+	3.02e−11
F_7			无		1275	0	+	1.21e−12	0	0	=	1.00e+00
F_8			无		1275	0	+	1.28e−12	1275	0	+	2.98e−11
F_9	1275	0	+	7.21e−12	1275	0	+	7.14e−12	1270	5	+	3.18e−11
F_{10}	0	0	=	1.00e+00	1275	0	+	1.21e−12	0	0	=	1.00e+00
F_{11}	1275	0	+	1.69e−14	1275	0	+	1.21e−12	0	0	=	1.00e+00
F_{12}			无		1275	0	0	3.02e−11	1275	0	+	3.02e−11
F_{13}	0	0	=	1.00e+00	1275	0	+	1.21e−12	0	0	=	1.00e+00
F_{14}	1275	0	+	1.69e−14	1275	0	+	1.21e−12	0	0	=	1.00e+00
F_{15}	0	0	=	1.00e+00	1275	0	+	6.18e−10	1275	0	+	1.21e−12
F_{16}	0	0	=	1.00e+00	0	0	=	1.00e+00	0	0	=	1.00e+00
F_{17}	0	0	=	1.00e+00	0	0	=	1.00e+00	0	0	=	1.00e+00
F_{18}	0	0	=	1.00e+00	0	0	=	1.00e+00	0	0	=	1.00e+00
F_{19}	0	0	=	1.00e+00	1275	0	+	1.69e−14	0	0	=	1.00e+00
F_{20}			无		0	0	+	1.00e+00	0	0	=	1.00e+00
+/−/=	5/0/7				16/0/5				6/0/14			

表 9 偏移指数与偏移量

Table 9 Shift index and shift value

SI	1	2	3	4	5	6
SV	$0.05\times\dfrac{U_k-L_k}{2}$	$0.1\times\dfrac{U_k-L_k}{2}$	$0.2\times\dfrac{U_k-L_k}{2}$	$0.3\times\dfrac{U_k-L_k}{2}$	$0.5\times\dfrac{U_k-L_k}{2}$	$0.7\times\dfrac{U_k-L_k}{2}$

表 10 不同偏移指数下的函数测试结果

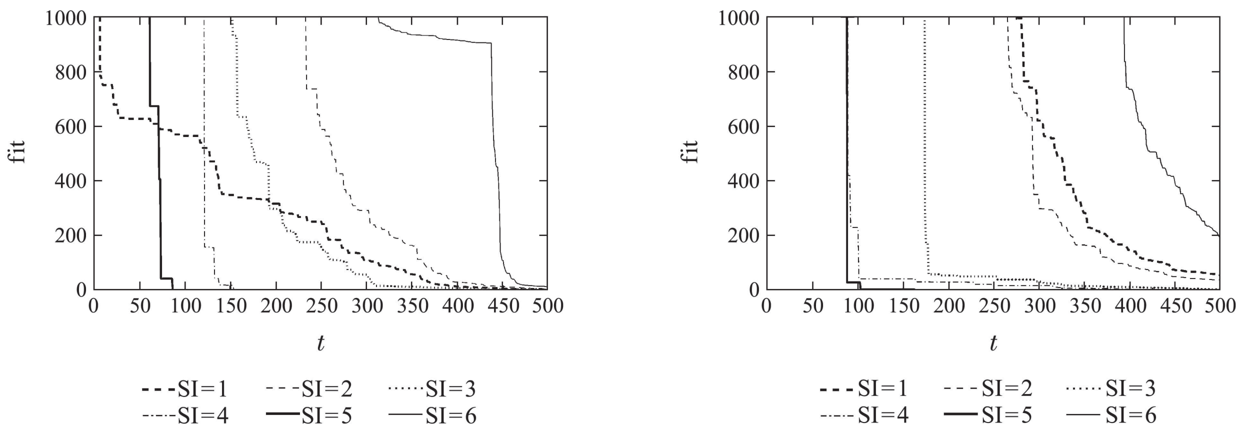
Table 10 Functions test results on different shift index

		F_1		F_3		F_4		F_6		F_{11}		F_{13}	
		平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差
SI=1	A	1.16e−1	7.44e−2	2.08e−1	8.74e−2	150.21	5.75e+2	31.11	7.97	9.05e−1	8.97e−1	1.74e−1	8.36e−2
	B	3.43e−6	8.47e−6	6.47e−4	6.77e−4	1.33e−3	1.80e−3	1.19e−4	2.02e−4	4.08e−5	3.51e−5	2.26e−3	5.70e−3
SI=2	A	5.86e−1	4.00e−2	1.61e−1	6.83e−2	764.02	1.71e+3	2.68e+2	9.34e+2	1.45e−1	2.34e−1	1.38e−1	6.34e−2
	B	7.55e−5	1.06e−4	1.85e−4	2.37e−4	2.17e−2	3.36e−2	9.52e−4	1.29e−3	1.53e−4	1.18e−4	7.22e−3	1.50e−2
SI=3	A	2.40e−2	2.75e−2	1.40e−	6.50e−2	2785.6	2.54e+3	2.45e+2	9.59e+2	3.76e−2	2.31e−2	6.35e−2	6.04e−2
	B	3.86e−4	9.16e−4	6.35e−4	7.00e−4	1.33	1.91	5.22e−3	9.42e−3	4.99e−4	4.28e−4	1.67e−3	3.48e−3
SI=4	A	9.02e−3	8.54e−3	1.39e−1	1.26e−1	3.76e+3	2.43e+3	4.19e+1	1.55e+2	2.23e−2	1.34e−2	5.04e−2	4.95e−2
	B	1.80e−4	2.93e−4	9.79e−4	9.36e−4	5.02e−2	9.29e−2	2.42e−3	3.94e−3	5.63e−4	5.65e−4	2.27e−3	3.45e−3
SI=5	A	0	0	0	0	5.56e+3	2.15e+3	1.07e+2	5.03e+2	8.88e−16	0	0	0
	B	0	0	0	0	0	0	9.38e−4	5.38e−3	2.14e−18	1.29e−18	0	0
SI=6	A	9.75e−2	9.34e−2	3.43e−1	2.14e−1	3.95e+3	1.02e+3	2.65e+2	4.81e+2	4.83e−1	6.06e−1	1.87e−1	1.28e−1
	B	1.01e−6	2.43e−6	8.73e−4	8.83e−4	2.96	1.99	1.19e−3	2.36e−3	4.69e−4	3.57e−4	1.55e−4	2.76e−4

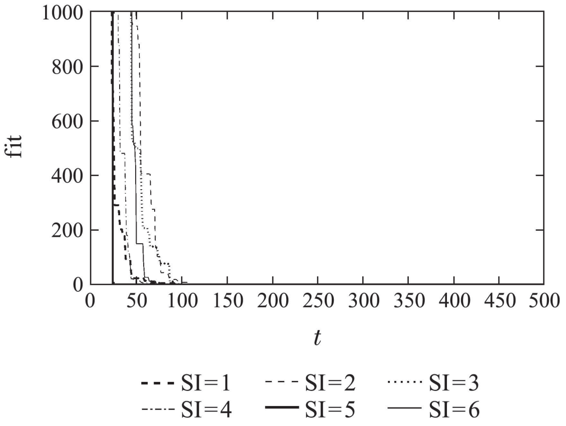
注: A表示KMTOA, B表示WLMS–KMTOA.

由图6所示的偏移函数收敛曲线可知: 在不同函数的位置偏移情况下, KMTOA算法收敛速度的下降幅度尤为明显, 尤其在SI = 6时, KMTOA算法收敛速度十分缓慢, 其原因在于, 对于偏移函数来说, KMTOA算法寻优过程中基于图1的高效搜索方式不再起到关键作用, 各个分子与最优个体星型拓扑通信方式的不足也凸显出来, 使算法的收敛精度和速度都受到影响. 而WLMS–KMTOA算法基于弱连接多子群结构, 有效增强了算法中个体与个体的信息交互, 优秀个体也得到了较好的利用, 因此SI = 6时, 算法收敛速度仍然较快, 且在设定的迭代次数内, KMTOA算法难以收敛到最优值, 而WLMS–KMTOA算法明显优于KMTOA算法, 这表明WLMS–KMTOA算法跳出局部最优的能力比KMTOA有所增强, 有利于解决复杂的优化问题.

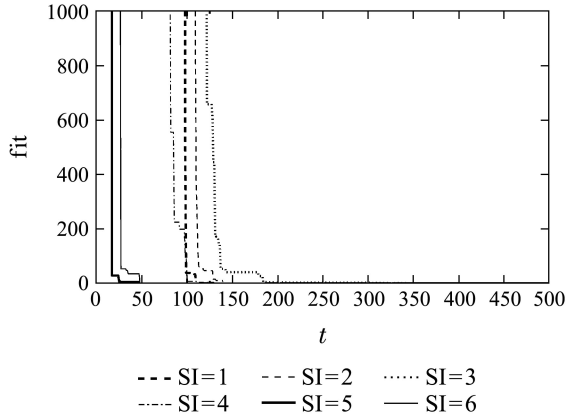
为说明KMTOA算法在SI = 5时收敛速度和精度大大提高的原因, 以偏移指数为5的Sphere函数进行测试, 由图7和图8可知, 此时KMTOA算法分子二维运动轨迹以右上对角线为主, 提高了寻优效率, 因此算法表现出很好的性能.



(a) KMTOA测试Sphere函数收敛曲线图　　(b) KMTOA测试Rosenbrock函数收敛曲线图



(c) WLMS–KMTOA测试Sphere函数收敛曲线图



(d) WLMS–KMTOA测试Rosenbrock函数收敛曲线图

图 6 偏移函数寻优情况对比图

Fig. 6 Comparison of performance of optimization on shift functions

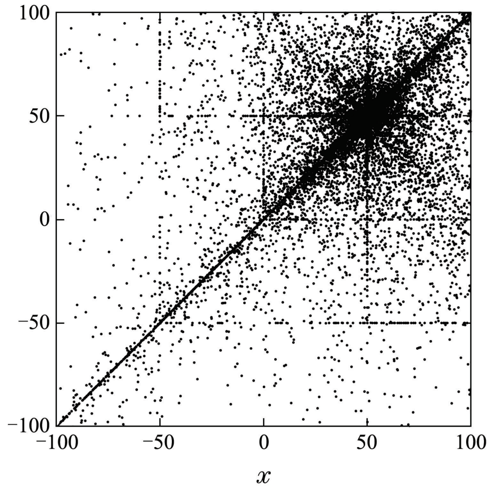


图 7 SI = 5时迭代500代二维分子位置图

Fig. 7 The locations of 2–D Sphere when SI = 5

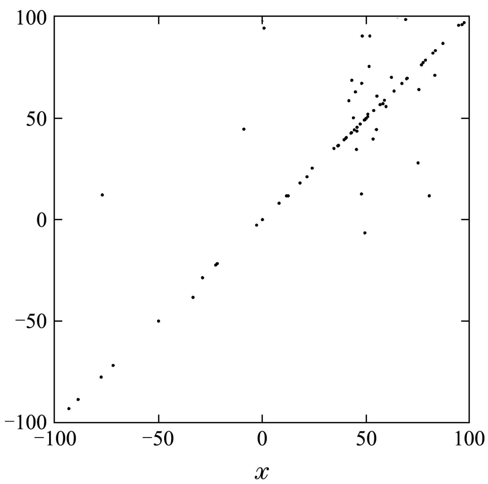


图 8 SI = 5时第60代二维分子位置图

Fig. 8 The locations of 2–D Sphere on the 60th generation when SI = 5

6 结论

本文提出了一种弱连接多子群分子动理论优化算法. 该算法基于多子群弱连接模型, 采用分子动理论优化算法进化下层子群, 以提高算法收敛速度; 上层中的混沌扰动子群通过混沌扰动机制提高了种群多样性, 有效地避免算法陷入局部最优; 上层中的免疫局部学习子群充分利用优秀个体进行局部搜索, 提高了算法精度. 算法融合了多子群、精英保留策略、弱连接网络、免疫克隆以及混沌扰动等优化思想, 充分利用分子间的信息与算法“群集”优势, 扬长避短, 有效地平衡了算法的全局搜索与局部开发能力. 与乌贼算法、蚁狮算法和最优觅食算法等当前表现较好的优化算法相比, WLMS−KMTOA算法不仅寻优精度高, 能有效克服“群集”现象的不足, 且在偏移函数求解方面表现出良好性能, 为求解工程应用中的偏移函数问题提供了一种新方法.

参考文献:

[1] FAN C D, OUYANG H L, ZHANG Y J. Optimization algorithm based on kinetic-molecular theory. *Journal of Central South University*, 2013, 20(12): 3504 – 3512.

[2] BO J, WANG Y, XU N. Study of Wireless sensor network route based on improved ant colony algorithm. *International Journal of Online Engineering*, 2016, 12(10): 86 – 90.

[3] SHARMA H, BANSAL J C, ARYA K V, et al. Lévy flight artificial bee colony algorithm. *International Journal of Systems Science*, 2016, 47(11): 2652 – 2670.

[4] CRUZ-VEGA I, GARCIA C A R, GIL P G, et al. Genetic algorithms based on a granular surrogate model and fuzzy aptitude functions. *Proceedings of the 23th IEEE World Congress on Computational Intelligence, Evolutionary computation*. Vancouver: IEEE, 2016: 2122 – 2128.

[5] FAN C D, REN K, ZHANG Y J, et al. Optimal multilevel thresholding based on molecular kinetic theory optimization algorithm and line intercept histogram. *Journal of Central South University*, 2016, 23(4): 880 – 890.

[6] FAN Chaodong, ZHANG Jing, YI Lingzhi. M-elite coevolutionary kinetic-molecular theory optimization algorithm. *Journal on Communications*, 2015, 36(7): 144 – 152.

(范朝冬, 章兢, 易灵芝. M-精英协同进化分子动理论优化算法. 通信学报, 2015, 36(7): 144 – 152.)

[7] CHANG W D. Multimodal function optimizations with multiple maximums and multiple minimums using an improved PSO algorithm. *Applied Soft Computing*, 2017, 60: 60 – 72.

[8] EESA A S, BRIFCANI A M A, ORMAN Z. A new tool for global optimization problems-cuttlefish algorithm. *International Journal of Mathematical, Computational, Natural and Physical Engineering*, 2014, 8(9): 1203 – 1207.

[9] WU Jianhui, ZHANG Jing, LI Renfa, et al. A multi-subpopulation pso immune algorithm and its application on function optimization. *Journal of Computer Research and Development*, 2012, 49(9): 1883 – 1898.
(吴建辉, 章兢, 李仁发, 等. 多子种群微粒群免疫算法及其在函数优化中应用. 计算机研究与发展, 2012, 49(9): 1883 – 1898.)

[10] JIN Min, LU Huaxiang. A multi-subgroup hierarchinal hybrid of genetic algorithm and particle swarm optimization. *Control Theory & Applications*, 2013, 30(10): 1231 – 1238.
(金敏, 鲁华祥. 一种遗传算法与粒子群优化的多子群分层混合算法. 控制理论与应用, 2013, 30(10): 1231 – 1238.)

[11] YE Manyuan, ZHOU Qiqi, CAI Hong, et al. Multiple population genetic algorithm based on multi-band SHEPWM control technology for multi-level inverter. *Transactions of China Electrotechnical Society*, 2015, 30(16): 111 – 119.
(叶满园, 周琪琦, 蔡鸿, 等. 基于多种群遗传算法的多电平逆变器多波段SHEPWM技术. 电工技术学报, 2015, 30(16): 111 – 119.)

[12] CHEN Biyun, WEI Xingqiu, CHEN Shaonan, et al. Power system multi-objective optimization based on multi-population genetic algorithm. *Proceedings of the CSU–EPSA*, 2015, 27(7): 24 – 29.
(陈碧云, 韦杏秋, 陈绍南, 等. 基于多种群遗传算法的电力系统多目标优化. 电力系统及其自动化学报, 2015, 27(7): 24 – 29.)

[13] XIE H, YU Wenke, YANG Peng, et al, Inverse kinematics solution method based on multi-subgroup hierachial hybrid of differential algorithm and chaotic particle swarm optimization. *Journal of Electronic and Instrumentation*, 2015, 29(10): 1456 – 1463.
(谢宏, 禹文科, 杨鹏, 等. 多子群分层差分粒子群算法的逆运动求解方法. 电子测量与仪器学报, 2015, 29(10): 1456 – 1463.)

[14] MIRJALILI S. The ant lion optimizer. *Advances in Engineering Software*, 2015, 83(C): 80 – 98.

[15] ZHU G Y, ZHANG W B. Optimal foraging algorithm for global optimization. *Applied Soft Computing*, 2017, 51: 294 – 313.

[16] GRANOVETTER S. The strength of weak ties. *American Journal of Sociology*, 1973, 78(6): 1360 – 1380.

[17] RAHNAMAYAN S, TIZHOOSH H R, SALAMA M M A. Opposition-based differential evolution. *IEEE Transactions on Evolutionary Computation*, 2008, 12(1): 64 – 79.

[18] LIANG J J, QIN A K, SUGANTHAN P N, et al. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Transactions on Evolutionary Computation*, 2006, 10(3): 281 – 295.

[19] MACKEY M C, TYRAN-KAMIŃSKA M. Central limit theorem behavior in the skew tent map. *Chaos Solitons & Fractals*, 2008, 38(3): 789 – 805.

[20] GUO S M, YANG C C. Enhancing differential evolution utilizing eigenvector-based crossover operator. *IEEE Transactions on Evolutionary Computation*, 2015, 19(1): 31 – 49.

[21] CHENG R, JIN Y. A competitive swarm optimizer for large scale optimization. *IEEE Transactions on Cybernetics*, 2015, 45(2): 191 – 204.

[22] DERRAC J, GARCÍA S, MOLINA D, et al. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm & Evolutionary Computation*, 2011, 1(1): 3 – 18.

[23] ZHU Qibing, WANG Zhenyu, HUANG Min. Fireworks algorithm with gravitational search operator. *Control and Decision*, 2016, 31(10): 1853 – 1859.
(朱启兵, 王震宇, 黄敏. 带有引力搜索算子的烟花算法. 控制与决策, 2016, 31(10): 1853 – 1859.)

作者简介:

范朝冬 博士, 讲师, 目前研究方向为智能信息处理、智能电网、模式识别与人工智能等, E-mail: fanchd@126.com;

刘颖南 硕士, 目前研究方向为智能信息处理、智能电网, E-mail: 201610171852@smail.xtu.edu.cn;

章 兢 博士, 教授, 博士生导师, 目前研究方向为智能计算、复杂系统控制等, E-mail: zhangj@ hnu.edu.cn;

易灵芝 博士, 教授, 硕士生导师, 目前研究方向为交流调速与电力电子装置、智能电网、新能源发电等, E-mail: ylzwyh@xtu.edu.cn;

肖乐意 博士研究生, 目前研究方向为图像处理、智能信息处理, E-mail: xiaolyttkx@163.com.