

文章编号:1671-6833(2018)06-0043-07

一种量子行为磷虾算法及其仿真分析

刘 振, 刘文彪, 鲁华杰

(海军航空大学 岸防兵学院, 山东 烟台 264001)

摘 要: 针对基本磷虾算法收敛效率低下, 容易收敛到局部极值的缺点, 基于协同进化和量子计算基本理论, 提出一种量子行为磷虾算法, 称为协同进化量子磷虾算法 (cooperative evolution quantum krill herd algorithm, CEQKHA). 该算法将磷虾种群划分为主种群和辅种群, 各种群能够独立进化, 并实现优良个体的交换. 利用量子进化行为方式更新磷虾个体位置, 引进 delta 势阱, 将粒子势阱中心设置为最优个体位置, 获取磷虾进化后的位置, 并分别将主种群和辅种群个体采用不同的位置更新方式, 提高种群勘探和开采的能力. 对所提出的算法进行了收敛性分析, 证明了所提出算法的收敛性能. 最后利用基准函数进行了仿真对比分析, 经仿真验证, 所提出的 CEQKHA 能得到更优解, 具备良好的优化性能.

关键词: 磷虾算法; 协同进化; 量子; 多种群; 势阱

中图分类号: TP15 文献标志码: A doi:10.13705/j.issn.1671-6833.2018.06.017

0 引言

磷虾算法作为一种低阶群智能进化算法^[1], 具有进化机制简单、收敛速度快等特性, 但同时也容易陷入局部极值的情况. 因此磷虾算法的改进方式不断涌现, 当前对其改进主要可以概括为以下几类:

(1) 与局部进化方式进行有效融合. 如文献[2]提出个体进化方式依据列维飞行, 并利用精英个体更新磷虾个体位置; 文献[3]引入反向学习方法初始化种群, 用于提高进化效率; 文献[4]将混沌算法的思想引入到磷虾算法中; 文献[5]对当前种群进化获得的新位置, 再次进行局部调整和二次开发, 并采用贪婪选择方式.

(2) 对磷虾算法中的进化参数进行调整, 文献[6]将静态常数 Δt 进行了自适应推广; 文献[7]不仅使 Δt 自适应变化, 同时对惯性权重依据正弦方式进行调整.

(3) 在磷虾个体的进化过程中引入传统进化规划的操作算子, 如个体迁移、交叉、变异等操作. 文献[8]引入了侧重于开采的磷虾迁移操作, 能让磷虾个体在进化后期处于全局最优值附近搜索; 文献[9]将交叉、变异引入到基本磷虾进化算

法中, 保证算法进化过程中的全局勘探和局部开采的有机统一.

(4) 与其他进化算法的有机融合, 文献[10]提出一种混合磷虾进化算法, 将磷虾个体依据粒子群算法中的粒子进化操作方式执行.

当前各种群智能进化算法大量涌现, 如头脑风暴优化算法^[11], 通过聚类 and 变异等操作方式驱动进化过程, 克服传统群智能进化算法存在的随机性和盲目性, 但相比磷虾算法缺少生物学进化基础, 种群进化中也没有考虑协同机制. 量子计算以经典的量子力学为理论基础, 充分利用量子的叠加性、相干性、纠缠性, 因而使量子计算具有极好的并行计算能力. 文献[12-14]将量子计算与人工智能演化算法相结合, 使量子计算技术可以结合人工智能进化优势, 从而使进化算法具备全局收敛能力强、鲁棒性好等特性. 文献[12]较早将量子编码引入到免疫克隆算法中; 文献[13]和[14]将量子力学思想分别与引力搜索算法和粒子群算法相融合, 构成量子引力搜索算法和量子粒子群算法, 有效提升了原有算法的收敛性能. 根据量子理论, 并利用协同进化思想, 笔者提出一种协同进化量子磷虾算法 (cooperative evolution quantum krill herd algorithm, CEQKHA). 将磷虾

进化种群分为主种群、辅种群,将磷虾个体的运动状态以量子态中的波函数表示,以一维 delta 势阱作为势场,在得到变量分离形式的波函数以后,适当选择势阱参数,使量子磷虾个体在更新位置过程中趋向于当前势阱中心最优个体的位置,从而提高个体局部开采能力.

1 基本磷虾算法及其特性分析

基本磷虾算法用于模拟磷虾群体受到食物源及群体影响的觅食行为,其觅食行为主要受食物吸引度、种群趋同性以及外部不确定因素影响.个体位置的进化过程可以利用拉格朗日模型表示为

$$\frac{d\mathbf{X}_i(t)}{dt} = \mathbf{N}_i(t) + \mathbf{F}_i(t) + \mathbf{D}_i(t), \quad (1)$$

式中: $\mathbf{X}_i(t) = \{x_{i,1}, x_{i,2}, \dots, x_{i,n}\}$ 表示第 i 个磷虾个体在 t 时刻的位置, n 为问题维数; $\mathbf{N}_i(t)$ 表示 t 时刻第 i 个磷虾个体受群体影响的向量; $\mathbf{F}_i(t)$ 表示第 i 个磷虾个体受到食物吸引而产生的运动向量; $\mathbf{D}_i(t)$ 表示外界不确定性因素对其运动位置影响的向量,则个体在下一时刻的位置更新可按照式(2)进行:

$$\mathbf{X}_i(t + \Delta t) = \mathbf{X}_i(t) + \Delta t \frac{d\mathbf{X}_i(t)}{dt}. \quad (2)$$

磷虾算法进化机制简便,所需控制参数少,由于编码机制设计过于简单,多样性不高,无法在进化过程中保持良好的全局勘探和局部开采性能.为提高基本磷虾算法的全局收敛性能,笔者提出一种协同进化量子磷虾算法.

2 协同进化量子磷虾算法

2.1 多种群协同进化框架结构

传统磷虾进化算法通常都维持单一种群进化模式,无法实现种群之间个体交流和信息的交互.多种群的进化模式能够有效地保证个体进化过程中的多样性^[15],使个体能够以较大的概率接近最优值.因此笔者将磷虾群体划分为主种群和辅种群,经分析验证设置辅种群数目为4,其进化结构如图1所示.

在协同进化量子磷虾算法中,设置主种群为 M ,辅种群为 $Sub_i (i = 1, 2, 3, 4)$. 每个辅种群在完成一次进化后,都会把当前寻找到的最优个体 $gb_i (i = 1, 2, 3, 4)$ 传递给主种群 M ,主种群 M 进行贪婪选择判断后,更新当前全局最优解集.

2.2 量子进化行为

一个微观粒子的量子态一般用波函数表示,

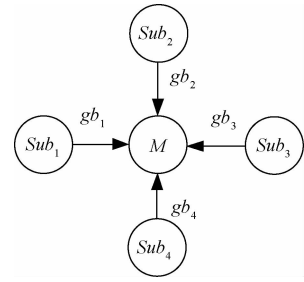


图1 种群协同进化框架

Fig.1 Framework for the cooperative evolution

因此波函数的演化状态确定以后,就足以描述整个系统的状态^[16].薛定谔方程具有和经典牛顿力学等同的作用,在任何时刻粒子的状态都可以用波函数 $\psi(\mathbf{X}, t)$ 表示为:

$$\hbar \psi \frac{\partial}{\partial t} \psi(\mathbf{X}, t) = H(\mathbf{X}) \psi(\mathbf{X}, t), \quad (3)$$

式中: \hbar 为普朗克常数; H 为与时间 t 无关的哈密尔顿操作符.

波函数幅值的平方为概率密度函数,故积分满足: $\int_{-\infty}^{+\infty} |\psi|^2 d\mathbf{X} = 1$. 由于 H 与时间 t 无关,波函数可以表示为时间片与空间片段的乘积:

$$\frac{j\hbar}{f(t)} \cdot \frac{d}{dt} f(t) = \frac{1}{\psi(\mathbf{X})} \cdot \left(-\frac{\hbar^2}{2m} \nabla^2 + V(\mathbf{X}) \right) \psi(\mathbf{X}), \quad (4)$$

式(4)中,等式左边是关于 t 的函数,而等式右边是关于 \mathbf{X} 的函数,两边相等并且都是能量 E . 因此,可以获得微分方程为:

$$\frac{1}{f(t)} \cdot \frac{d}{dt} f(t) = -\frac{jE}{\hbar}. \quad (5)$$

$$\text{由于 } E = \frac{1}{\psi(\mathbf{X})} \cdot \left(-\frac{\hbar^2}{2m} \nabla^2 + V(\mathbf{X}) \right) \psi(\mathbf{X}),$$

则可以得到 $H\psi(\mathbf{X}) = E\psi(\mathbf{X})$.

假定在磷虾算法中,所有个体都具备量子行为,选择其中最优的 G_{best} 个体构成吸引势场.不失一般性,假定用量子表示的个体都在一维空间中,因此所有向量 \mathbf{X} 将变成标量 x ,设定 $y = x - g_{\text{best}}$, 其中 g_{best} 表示最优个体的位置.应用 δ 势阱,令其中心在零点处,假定一维势阱的数目和 G_{best} 数目一致,则 $V(y) = -\gamma\delta(y) (\gamma > 0)$, 其中 γ 为势阱深度.势阱深度在原点处为无穷,而在其他处为零,此时薛定谔方程为:

$$\frac{\hbar^2}{2m} \frac{d^2}{dy^2} \psi(y) + \gamma\delta(y)\psi(y) + E\psi(y) = 0. \quad (6)$$

由于 $Q(y) = |\psi(y)|^2 = \frac{1}{L} e^{(-2|y|/L)}$, 令

$u = e^{(-2|y|^L)}$, 则得到 $y = \pm \frac{L}{2} \ln(1/u)$. 由于 $y = x - g_{\text{best}}$, 则得到 $x = g_{\text{best}} \pm \frac{L}{2} \ln(1/u)$. 令 $\frac{L}{2} = k|x - g_{\text{best}}|$, 其中 k 为常数, $|x - g_{\text{best}}|$ 可以度量新产生的个体位置与当前最优解集的个体偏差. 则对第 i 个磷虾个体 $X_i(t)$, 其第 d 维的位置更新可以表示为:

$$x_{i,d}(t+1) = g_{\text{best}}^d \pm k|x_{i,d}(t) - g_{\text{best}}^d| \cdot \ln(1/u). \quad (7)$$

2.3 不同种群个体多样化位置更新方式

为了充分发挥优良个体引导作用, 文献[17]以等概率方式使磷虾个体在最优值附近进行搜索; 文献[18]在其基础上, 引入了群体平均值, 增加了个体位置选择的随机性. 笔者借鉴上述思想, 提出在主种群中进行全局勘探, 采用式(8)更新当前磷虾个体的位置:

$$x_{i,d}(t+1) = g_1^d(t) \pm k|x_{i,d}(t) - x_{\text{mean}}(t)| \cdot \ln(1/u), \quad (8)$$

式中: $g_1^d(t)$ 表示当前代的最优解; $x_{\text{mean}}(t)$ 表示当前代的平均解.

在辅种群中, 为了进行局部精细开采操作, 采用式(9)更新当前磷虾个体的位置:

$$x_{i,d}(t+1) = k \left| \alpha \left(\frac{x_{i,d}(t) - g_1^d(t)}{2} \right) + (1 - \alpha) \cdot \left(\frac{x_{i,d}(t) - g_2^d(t)}{2} \right) \right| \cdot \ln(1/u) \pm g_2^d(t), \quad (9)$$

式中: $g_2^d(t)$ 表示截止到第 t 代为止的全局最优解; α 为调节系数, $0 \leq \alpha \leq 1$ 且为常数.

通过以上的描述, 笔者提出的协同进化量子磷虾算法(CEQKHA)整体流程可以表述如下:

步骤1 设定算法进化参数信息, 初始化种群 $P(t)$, 循环迭代次数 $t = 1$;

步骤2 将种群 $P(t)$ 划分为一个主种群和4个辅种群分别独立进化;

步骤3 在获得 $N_i(t)$ 、 $F_i(t)$ 和 $D_i(t)$ 后, 根据式(1)获得运动增量 $\frac{dX_i(t)}{dt}$, 完成一次种群进化和更新, 根据式(2)获得 $t+1$ 时刻第 i 个磷虾个体的运动位置向量 $X_i(t+1)$;

步骤4 将辅种群 $Sub_i (i = 1, 2, 3, 4)$ 中最优个体 $gb_i (i = 1, 2, 3, 4)$ 传递给主种群 M , 辅种群向主种群进行精英个体交流;

步骤5 按照2.2节的方式对磷虾个体进行量子进化, 并依据式(8)和式(9)进行更新;

步骤6 判断是否迭代结束, 没有结束, 则 $t = t+1$, 返回步骤3, 否则输出最优解, 算法结束.

3 收敛性分析

由式(1)和式(2), 并结合式(8)和式(9)可以得到

$$X(t+1) = X(t) + G(t). \quad (10)$$

其中, $G(t) = \frac{dX(t)}{dt} = N(t) + F(t) + D(t)$. 可知 $G(t+1)$ 不仅受到上一时刻磷虾位置 $X(t)$ 的影响, 同时也受到 $G(t)$ 的影响, 即

$$G(t+1) = \mu G(t) + \alpha X(t) + \beta. \quad (11)$$

因此, 根据式(10)和式(11), 可以得到

$$X(t+2) - (1+\mu)X(t+1) + (\mu-\alpha)X(t) = \beta. \quad (12)$$

由式(12)方程两边的构造可以看出, 其为一个差分方程, 因此给出定理1.

定理1 协同进化量子磷虾算法(CEQKHA)稳定收敛的充分条件: $\alpha < 0, \frac{\alpha}{2} - 1 < \mu < \alpha + 1$.

证明: 式(12)的特征方程为 $\lambda^2 - (1+\mu)\lambda + (\mu-\alpha) = 0$, 特征根为 λ_1 和 λ_2 , 则通解为 $X(t) = k_1\lambda_1^t + k_2\lambda_2^t + c$. 若要满足 $\lim_{t \rightarrow \infty} X(t) = X^*$, 即当迭代次数无穷时, 磷虾个体的位置趋向于稳定的最优值, 此时要求 $|\lambda_i| < 1, (i = 1, 2)$, 因此根据其判别式:

$$\Delta = (1+\mu)^2 - 4(\mu-\alpha) \geq 0,$$

则可得到

$$\alpha < 0, \frac{\alpha}{2} - 1 < \mu < \alpha + 1.$$

定理1得证.

4 仿真分析

利用基准函数 $f_1 \sim f_{18}$ 进行仿真分析, 其中惯性权重 $\omega^n = \omega^f = 0.9, C' = 2$, 最大引导速度为0.01, 最大扩散速度为0.005, 觅食速度为0.02, 种群规模 $NP = 100$, 循环迭代次数为100. 所有仿真分析均在 Intel(R) Core(TM) i3-3220 处理器, 内存2.00 GB, Windows 7 系统的计算机上进行, 仿真软件为 Matlab R2009a. 将笔者提出的协同进化量子磷虾算法(CEQKHA), 与基本磷虾算法(KHA)^[1]、对立搜索磷虾算法^[3](OKHA)以及改进磷虾算法^[5](IKHA)进行仿真对比分析, 变量取值范围均设定在 $[-100, 100]$ 内, 分别利用单模函数、多模函数进行仿真分析, 并分析进化框架结构和位置更新方式对算法影响.

$$f_1(x) = \sum_{i=1}^n |x_i| + \prod_{i=1}^n |x_i|;$$

$$f_2(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2;$$

$$f_3(x) = - \sum_i^n i x_i^2;$$

$$f_4(x) = \sum_{i=1}^{n-1} (100 (x_{i+1} - x_i^2)^2 + (x_i - 1)^2);$$

$$f_5(x) = \sum_{i=1}^n x_i^2;$$

$$f_6(x) = \sum_{i=1}^n (x_i + 0.5)^2;$$

$$f_7(x) = 418.982\,9 \cdot n - \sum_{i=1}^n x_i \sin(\sqrt{|x_i|});$$

$$f_8(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10);$$

$$f_9(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e;$$

$$f_{10}(x) = \frac{\pi}{n} \left[10 \sin(\pi y_1) + \sum_{i=1}^n (y_i - 1)^2 \cdot (1 + 10 \sin^2(\pi y_i + 1)) + (y_n - 1)^2 \right] + \sum_{i=1}^n u(x_i, 10, 100, 4);$$

$$y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a; \\ 0, & -a < x_i < a; \\ k(-x_i - a)^m, & x_i < -a, \end{cases} m = 5;$$

$$f_{11}(x) = 0.1 \left[\sin^2(3\pi x_1) + \sum_{i=1}^{n-1} (1 + \sin^2(3\pi x_i + 1)) (y_i - 1)^2 + (x_n - 1)^2 (1 + \sin^2(2\pi x_n)) \right] + \sum_{i=1}^n u(x_i, 5, 100, 4);$$

$$f_{12}(x) = 4.687 - \sum_{i=1}^n \sin x_i \cdot \left(\sin\left(\frac{i x_i^2}{\pi}\right) \right)^{2m}, m = 10;$$

$$f_{13}(x) = (x_1 - 1)^2 + \sum_{i=2}^n i (2x_i^2 - x_{i-1})^2;$$

$$f_{14}(x) = \sum_{i=1}^n i x_i^4 + \text{rand}(0, 1);$$

$$f_{15}(x) = (e^{-\sum_{i=1}^n (x_i/\beta)^{2m}} - 2e^{-\sum_{i=1}^n x_i^2}) \cdot \prod_{i=1}^n \cos^2 x_i, m = 5;$$

$$f_{16}(x) = \left(\left(\sum_{i=1}^n \sin^2(x_i) \right) - \exp\left(-\sum_{i=1}^n x_i^2\right) \right) \cdot \exp\left(-\sum_{i=1}^n \sin^2 \sqrt{|x_i|}\right);$$

$$f_{17}(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2)^2;$$

$$f_{18}(x) = 0.26(x_1^2 + x_1^2) - 0.48x_1x_2.$$

4.1 单模函数对比分析

将笔者提出的 CEQKHA, 利用单模基准函数 $f_1 \sim f_6$ 进行仿真分析, 其中 $f_1 \sim f_3$ 维数为 5, $f_4 \sim f_6$ 维数为 10. 并与基本磷虾算法 (KHA)、对立搜索磷虾算法 (OKHA) 以及改进磷虾算法 (IKHA) 进行对比, 其中基准函数的全局最优值均为 0.4 种算法分别独立运行 30 次, 统计获得的最优值、平均值和标准差, 对比分析结果如表 1 所示.

表 1 单模函数仿真对比分析

Tab. 1 Comparison results for unimodal functions				
函数	算法	最优值	均值	标准差
f_1	KHA	7.36×10^{-4}	8.97×10^{-3}	6.52×10^{-3}
	OKHA	6.82×10^{-4}	6.15×10^{-3}	2.38×10^{-4}
	IKHA	5.17×10^{-4}	4.05×10^{-3}	6.34×10^{-4}
	CEQKHA	4.36×10^{-4}	3.89×10^{-3}	4.28×10^{-4}
f_2	KHA	4.93×10^{-3}	2.71×10^{-2}	3.25×10^{-3}
	OKHA	2.40×10^{-3}	4.24×10^{-3}	5.13×10^{-4}
	IKHA	3.19×10^{-3}	8.56×10^{-3}	7.25×10^{-4}
	CEQKHA	5.65×10^{-4}	1.79×10^{-3}	1.06×10^{-4}
f_3	KHA	1.78×10^{-5}	4.27×10^{-4}	6.54×10^{-4}
	OKHA	3.23×10^{-6}	5.18×10^{-5}	5.94×10^{-6}
	IKHA	4.93×10^{-6}	1.87×10^{-5}	3.24×10^{-6}
	CEQKHA	2.32×10^{-7}	6.18×10^{-6}	1.56×10^{-8}
f_4	KHA	2.67×10^{-3}	4.91×10^{-2}	7.76×10^{-5}
	OKHA	5.90×10^{-4}	1.52×10^{-3}	5.65×10^{-5}
	IKHA	4.43×10^{-4}	6.56×10^{-3}	5.72×10^{-5}
	CEQKHA	5.17×10^{-4}	2.83×10^{-4}	9.10×10^{-6}
f_5	KHA	4.25×10^{-3}	5.97×10^{-3}	8.24×10^{-6}
	OKHA	1.90×10^{-4}	5.01×10^{-3}	5.63×10^{-6}
	IKHA	5.61×10^{-5}	7.25×10^{-4}	4.56×10^{-7}
	CEQKHA	9.22×10^{-6}	5.20×10^{-4}	6.79×10^{-8}
f_6	KHA	3.82×10^{-4}	9.33×10^{-3}	7.17×10^{-5}
	OKHA	2.44×10^{-4}	1.49×10^{-3}	2.82×10^{-5}
	IKHA	3.45×10^{-4}	7.61×10^{-3}	2.75×10^{-5}
	CEQKHA	7.19×10^{-5}	5.17×10^{-3}	6.71×10^{-6}

从表 1 的统计结果能够看出, 在设定的循环迭代次数范围内, 几种磷虾改进算法 (OKHA、IKHA、CEQKHA), 相比基本磷虾算法 (KHA) 收敛性能都有了一定程度的提高. 另外, 从表 1 还可以看出, 笔者提出的 CEQKHA 在大部分函数上的进化性能都优于其他几种进化算法. 这是由于笔者采用了协同进化框架结构, 并应用量子势阱机制, 提升了个体在整个搜索空间中勘探和开采能力, 因而寻优效果整体优于其他几种改进算法.

为了充分进行算法对比,以 f_5 函数为例进行进一步详细分析,取一次迭代进化收敛如图 2 所示. 从图 2 可以看出,OKHA 和 IKHA 分别采用了对立搜索方式和局部搜索算子,提高了算法的基本进化性能;而本文算法不仅采用了协同进化框架,而且采用了量子编码提高了个体的全局和局部搜索能力. 另外,基本磷虾算法由于搜索的盲目性,初始进化效果不佳,OKHA 和 IKHA 由于采用了对立搜索方式及局部搜索设计,因而提高了进化能力;而本文的 CEQKHA 寻优迭代效果较好,能够利用较少的迭代次数就能锁定在最优值附近搜索,因而无论是方差和均值性能均优于其他对比算法.

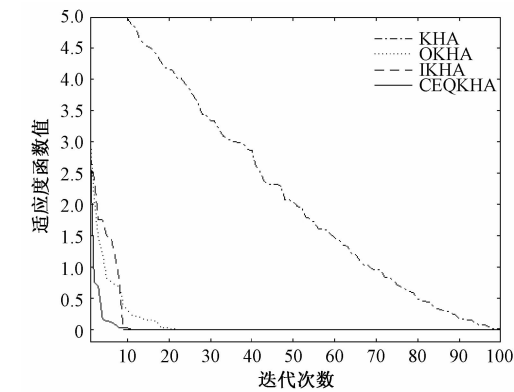


图 2 f_5 函数仿真对比结果

Fig.2 Comparison results for function f_5

4.2 多模函数对比分析

将本文算法 CEQKHA 以多峰基准函数 $f_7 \sim f_9$ 为例进行仿真对比分析,其中 3 个多峰函数的全局最优值均为 0,函数维数均设置为 20. 对比算法也分别是基本磷虾算法(KHA)、对立搜索磷虾算法(OKHA)以及改进磷虾算法(IKHA). 4 种算法分别运行 30 次,对其最优值、平均值及标准差进行统计对比,同时进行单侧 t 检验,并构造检验统计量:

$$t = \frac{M_1 - M_2}{\sqrt{\frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2}} \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}},$$

式中: n_1 和 n_2 表示运行次数; S_1 和 S_2 表示对比方差; M_1 和 M_2 为对比均值.

检验假设为: $H_0: \mu_1 \geq \mu_2, H_1: \mu_1 < \mu_2, \mu_1$ 和 μ_2 理论最优均值, 给定显著性水平 0.05, 拒绝域为: $W = \{t > t_{0.05}(n_1 + n_2 - 2)\}$, 经查表后 $t_{0.05}(58) = 1.672$, 统计结果如表 2 所示.

从表 2 的统计结果能够看出, 当对多模函数进行优化时, 本文的 CEQKHA 相比其他几种

表 2 多模函数仿真对比分析

Tab.2 Comparison results for multimodal function

函数	算法	最优值	均值	标准差	$t_{0.05}$
f_7	KHA	6.13×10^{-2}	2.55×10^{-1}	7.63×10^{-2}	8.73
	OKHA	5.87×10^{-4}	5.90×10^{-3}	2.68×10^{-3}	4.17
	IKHA	4.27×10^{-4}	7.85×10^{-3}	2.19×10^{-3}	3.39
	CEQKHA	4.43×10^{-4}	4.52×10^{-3}	6.02×10^{-4}	—
f_8	KHA	5.45×10^{-2}	3.17×10^{-1}	6.53×10^{-2}	6.84
	OKHA	7.18×10^{-3}	3.10×10^{-2}	6.48×10^{-3}	2.99
	IKHA	3.55×10^{-3}	2.56×10^{-2}	8.28×10^{-3}	5.16
	CEQKHA	7.91×10^{-4}	4.23×10^{-3}	4.57×10^{-3}	—
f_9	KHA	6.14×10^{-3}	5.78×10^{-2}	2.80×10^{-3}	9.74
	OKHA	1.52×10^{-4}	7.66×10^{-3}	4.25×10^{-4}	2.62
	IKHA	3.09×10^{-4}	7.15×10^{-3}	6.57×10^{-4}	3.17
	CEQKHA	3.73×10^{-5}	1.35×10^{-4}	5.79×10^{-5}	—

进化算法,在大部分函数上都能取得较好的收敛结果. 从多模函数优化对比结果可以看到,算法的进化能力得到了考验,对算法的全局开采能力提出了更高的要求. 同时表 2 还给出单侧 t 检验结果,当显著性水平为 0.05 时, t 的计算结果均大于 1.672, 因此拒绝假设 $H_0: \mu_1 \geq \mu_2$, 接受 $H_1: \mu_1 < \mu_2$, 说明本文算法平均值优于其他对比算法.

4.3 高维函数对比分析

利用高维基准函数 $f_{10} \sim f_{12}$ 为例进行仿真分析,其中函数维数均设置为 100, 3 个高维函数的全局最优值均为 0. 对比算法也仍然是基本磷虾算法(KHA)、对立搜索磷虾算法(OKHA)以及改进磷虾算法(IKHA). 4 种算法分别运行 30 次,对平均值、标准差和平均运行时间进行统计,结果如表 3 所示.

表 3 高维函数仿真对比分析

Tab.3 Comparison results for high-dimension functions

函数	算法	最优值	均值	标准差
f_{10}	KHA	7.19×10^{-2}	4.55×10^{-1}	5.42×10^{-1}
	OKHA	6.39×10^{-3}	5.98×10^{-2}	7.85×10^{-3}
	IKHA	5.64×10^{-3}	3.09×10^{-2}	6.16×10^{-3}
	CEQKHA	1.07×10^{-3}	9.81×10^{-3}	2.08×10^{-3}
f_{11}	KHA	9.27×10^{-2}	8.32×10^{-1}	5.57×10^{-1}
	OKHA	5.43×10^{-3}	2.55×10^{-2}	4.79×10^{-3}
	IKHA	6.38×10^{-3}	6.64×10^{-2}	5.86×10^{-3}
	CEQKHA	1.78×10^{-3}	7.14×10^{-3}	1.35×10^{-3}
f_{12}	KHA	7.24×10^{-2}	6.30×10^{-1}	2.67×10^{-1}
	OKHA	3.80×10^{-3}	1.29×10^{-2}	8.15×10^{-2}
	IKHA	5.22×10^{-3}	3.48×10^{-2}	9.41×10^{-2}
	CEQKHA	1.49×10^{-3}	6.37×10^{-3}	4.25×10^{-3}

当函数的维数达到 100 维时,对于几种进化算法来说,寻找到最优值都会变得较为困难. 从表

3 的统计结果能够看出,整体寻优精度都会有所下降,特别是基本磷虾算法(KHA),函数寻优性能下降明显,性能指标相比其他几种改进的磷虾算法差距较大,甚至有时无法收敛到全局最优值附近.从所有性能指标的对比可以看出,本文算法仍然能够取得较为优越的寻优效果,显示出算法具有较好的鲁棒性.笔者采用的协同进化搜索框架和量子进化行为增强了算法的全局勘探和局部开采能力,使得 CEQKHA 相比其他改进的磷虾算法,如 OKHA 和 IKHA,能够获得更优解.

4.4 算法进化行为影响分析

4.4.1 进化结构影响分析

对笔者提出的 CEQKHA,当不采用协同进化机制时,将算法标记为 CEQKHA_{wce}(CEQKHA without cooperative evolution),将 KHA、CEQKHA_{wce} 和 CEQKHA,利用 $f_{13} \sim f_{15}$ 函数进行对比分析,以验证协同进化框架对算法运行结果的影响,各种算法分别独立运行 30 次,函数维数均为 20,对平均值、标准差和平均运行时间进行统计,如表 4 所示.

表 4 协同进化对算法影响对比分析

Tab.4 Comparison results of cooperative evolution

函数	算法	均值	标准差	运行时间/s
f_{13}	KHA	3.82×10^{-3}	8.97×10^{-2}	1.15
	CEQKHA _{wce}	7.57×10^{-4}	3.86×10^{-3}	3.01
	CEQKHA	3.16×10^{-4}	6.58×10^{-4}	4.16
f_{14}	KHA	5.11×10^{-3}	9.42×10^{-2}	0.94
	CEQKHA _{wce}	5.06×10^{-4}	5.23×10^{-3}	3.17
	CEQKHA	3.83×10^{-4}	9.81×10^{-4}	4.29
f_{15}	KHA	5.13×10^{-3}	8.62×10^{-2}	1.89
	CEQKHA _{wce}	7.43×10^{-4}	7.15×10^{-3}	3.12
	CEQKHA	2.74×10^{-4}	6.78×10^{-4}	4.28

从表 4 的进化结果可以看到,CEQKHA 由于采用了协同进化框架结构和量子进化机制,因此其寻优迭代效果优于 CEQKHA_{wce} 和 KHA,虽然 CEQKHA_{wce} 没有采用协同进化框架结构,但由于量子进化机制优良的进化性能,其寻优效果还是优于 KHA,显示出本文算法良好的进化性能.从表 4 也可以看到,协同进化框架在提高了算法进化性能的同时,也相应地增加了算法的运行时间,因此适用于对时间要求不高而对精度要求较高的场合.

4.4.2 位置更新方式影响分析

文献[13]提出建立最优粒子集合,并采用轮盘赌方式选择粒子作为势阱中心,用于产生下一代个体,笔者没有采用这种方式,而是采用式(26)和式(27)进行个体更新.将笔者采用的磷虾

个体更新方式,以 $f_{16} \sim f_{18}$ 函数为仿真对象,函数维数均为 20,与文献[13]采用的更新方式进行仿真对比.各种算法独立运行 30 次,其统计结果如表 5 所示,其中采用文献[13]更新方式的算法标记为 CEQKHA_{rw}(CEQKHA with roulette wheel).

表 5 位置更新方式对算法影响分析

Tab.5 Comparison results of position refresh

函数	算法	均值	标准差	运行时间/s
f_{16}	KHA	5.78×10^{-4}	9.23×10^{-5}	1.70
	CEQKHA _{rw}	7.13×10^{-5}	6.17×10^{-6}	4.14
	CEQKHA	6.26×10^{-5}	2.64×10^{-6}	4.87
f_{17}	KHA	4.28×10^{-3}	5.57×10^{-4}	1.35
	CEQKHA _{rw}	7.22×10^{-4}	9.16×10^{-5}	2.67
	CEQKHA	3.50×10^{-4}	3.01×10^{-5}	2.69
f_{18}	KHA	6.51×10^{-3}	6.32×10^{-3}	0.77
	CEQKHA _{rw}	1.39×10^{-3}	5.17×10^{-4}	2.14
	CEQKHA	7.68×10^{-4}	4.47×10^{-5}	2.58

从表 5 的统计数据可以看到,笔者提出的 CEQKHA 在大部分函数上的平均值和标准差都略优于 CEQKHA_{rw},只是在时间性能上处于劣势;KHA 的进化性能相对较差,但其运行时间最短.可以看出,本文算法的进化性能得到了提高,但同时也注意到,为了单纯追求算法的寻优效果,运行时间呈现倍数级增长,对应用时机和场合提出了一定的要求.

5 结论

笔者提出了一种协同进化量子磷虾算法(CEQKHA),将进化种群划分为主种群和辅种群,采用具有量子进化的行为模式,将当前最优个体设置为量子势阱中心,并在主种群和辅种群内采用不同的个体更新方式.对所提出的 CEQKHA 进行了收敛性分析和函数仿真分析,显示出所提出算法的优良性能和应用前景.但同时也注意到,受制于进化算法的特性和基本规律,寻找一种既能保证收敛精度又能保证收敛速度的“全优”算法是很困难的.因此,提高算法的运行速度,降低算法的系统开销,将是以后发展方向.

参考文献:

[1] GANDOMI A H, ALAVVVI A H. Krill herd: A new bio-inspired optimization algorithm[J]. Communications in nonlinear science and numerical simulation, 2012, 17(12): 4831-4845.
[2] GUO L H, WANG G G, GANDOMI A H, et al. A new improved krill herd algorithm for global numerical

- optimization[J]. *Neurocomputing*, 2014, 138(8): 392–402.
- [3] SNEHA S, PROVAS K R. Oppositional krill herd algorithm for optimal location of capacitor with reconfiguration in radial distribution system[J]. *Electrical power and energy systems*, 2016, 74(12): 78–90.
- [4] GUO L H, WANG G G, GANDOMI A H, et al. Chaotic krill herd algorithm[J]. *Information sciences*, 2014, 274(8): 17–34.
- [5] JENSI R, WISELIN G J. An improved krill herd algorithm with global exploration capability for solving numerical function optimization problems and its application to data clustering[J]. *Applied soft computing*, 2016, 46(C): 230–245.
- [6] LI P, TANG Y G, HUA C H, et al. An improved krill herd algorithm: krill herd with linear decreasing step[J]. *Applied mathematics and computation*, 2014, 234(5): 356–367.
- [7] NIU P F, CHEN K, MA Y P, et al. Model turbine heat rate by fast learning network with tuning based on ameliorated krill herd algorithm[J]. *Knowledge-based systems*, 2017, 118(2): 80–92.
- [8] WANG G G, GANDOMI A H, AMIR H A. An effective krill herd algorithm with migration operator in biogeography-based optimization[J]. *Applied mathematical modeling*, 2014, 38(s9–10): 2454–2462.
- [9] PARAJITA M, PROVAS K R, MUKHERJEE V. Transient stability constrained optimal power flow using oppositional krill herd algorithm[J]. *Electrical power and energy systems*, 2016, 83(12): 283–297.
- [10] PRASAD S, VINOD K. Optimal allocation of measurement devices for distribution state estimation using multiobjective hybrid PSO-Krill herd algorithm[J]. *IEEE transactions on instrumentation and measurement*, 2017, 66: 1–14.
- [11] 程适, 陈俊凤, 孙奕菲, 等. 数据驱动的发展式头脑风暴优化算法综述[J]. *郑州大学学报(工学版)*, 2018, 39(3): 22–28.
- [12] 李阳阳, 焦李成. 求解 SAT 问题的量子免疫克隆算法[J]. *计算机学报*, 2007, 30(2): 176–183.
- [13] HOSSEIN N P. A quantum-inspired gravitational search algorithm for binary encoded optimization problems[J]. *Engineering applications of artificial intelligence*, 2015, 40(4): 62–75.
- [14] 方伟, 孙俊, 谢振平, 等. 量子粒子群优化算法的收敛性分析及控制参数研究[J]. *物理学报*, 2010, 59(6): 3686–3694.
- [15] NAJMEH S J, SALWANI A, ABDUL R H. Multi-population cooperative bat algorithm-based optimization of artificial neural network model[J]. *Information sciences*, 2015, 294(2): 628–644.
- [16] SUN J, FENG B, XU W B. Particle swarm optimization with particles having quantum behavior[C]//Proc of Congress on Evolutionary Computation. New York: IEEE, 2004:325–331.
- [17] MOHADESEH S M, HOSSEIN N P, MALIHE M F. A quantum inspired gravitational search algorithm for numerical function optimization[J]. *Information sciences*, 2014, 267(5): 83–100.
- [18] 曹茂俊, 李盼池, 尚福华. 量子行为引力搜索算法[J]. *控制与决策*, 2016, 31(9): 1678–1684.

A Quantum Behaved Krill Herd Algorithm and Its Simulation Analysis

LIU Zhen, LIU Wenbiao, LU Huajie

(College of Coastal Defense Force, Naval Aviation University, Yantai 264001, China)

Abstract: Aiming at the low efficiency and easy trap into local optimum for the basic krill herd algorithm, a new quantum behave krill herd algorithm was proposed. It was named as cooperative evolution quantum krill herd algorithm (CEQKHA) based on cooperative evolution and quantum computation. The population could be divided into two parts, such as main population and sub-population, which could evolve independently and exchange fine individuals. The position of krill herd could be updated by using quantum activity. The best position of krill herd could be set as the center of potential well of delta potential well, and the position of krill herd in the different population could evolve in different way. The convergence of the algorithm was also deduced. Simulation on benchmark functions proved that CEQKHA could get better results and perform better than other algorithms.

Key words: krill herd algorithm; cooperative evolution; quantum; multi-population; potential well