



文章编号:1001-9081(2019)03-0663-05

DOI:10.11772/j.issn.1001-9081.2018081661

## 基于动态压力控制算子的磷虾群算法

沈 莹, 黄樟灿\*, 谈 庆, 刘 宁

(武汉理工大学 理学院, 武汉 430070)

(\* 通信作者电子邮箱 [hunagzc@whut.edu.cn](mailto:hunagzc@whut.edu.cn))

**摘要:**针对基础磷虾群(KH)算法在求解复杂函数优化问题时局部搜索能力差、求解精度低、收敛速度慢、容易陷入局部最优等问题,提出一种基于动态压力控制算子的磷虾群算法(DPCKH)。该算法将一种新的动态压力控制算子加入了标准磷虾群算法,使其处理复杂函数优化问题更有效。动态压力控制算子通过欧氏距离量化了多个不同优秀个体对目标个体的诱导效应,进而在优秀个体附近加速产生新磷虾个体,提高了磷虾个体的局部探索能力。通过比较蚁群算法(ACO)、差分进化算法(DE)、磷虾群算法(KH)、改进的磷虾群算法(KHLD)和粒子群算法(PSO),DPCKH 算法在 7 个测试函数上的结果表明,DPCKH 算法与 ACO 算法、DE 算法、KH 算法、KHLD 算法和 PSO 算法相比有着更强的局部勘测能力,其开采能力更强。

**关键词:**磷虾群算法; 动态压力控制算子; 函数优化; 开采能力; 探索能力

中图分类号: TP18 文献标志码:A

### Krill herd algorithm based on dynamic pressure control operator

SHEN Ying, HUANG Zhangcan\*, TAN Qing, LIU Ning

(College of Science, Wuhan University of Technology, Wuhan Hubei 430070, China)

**Abstract:** Aiming at the problem that basic Krill Herd (KH) algorithm has poor local search ability and insufficient exploitation capacity on complex function optimization problems, a Krill Herd algorithm based on Dynamic Pressure Control operator (DPCKH) was proposed. A new dynamic pressure control operator was added to the basic krill herd algorithm, which made it more effective on complex function optimization problems. The dynamic pressure control operator quantified the induction effects of several different outstanding individuals on the target individual through Euclidean distance, accelerating the production of new krill individuals near the excellent individuals and improving the local exploration ability of krill individuals. Compared to ACO (Ant Colony Optimization) algorithm, DE algorithm, KH algorithm, KHLD (Krill Herd with Linear Decreasing step) algorithm and PSO (Particle Swarm Optimization) algorithm on 7 benchmark functions, DPCKH algorithm has stronger local exploration and exploitation ability.

**Key words:** Krill Herd (KH) algorithm; dynamic pressure control operator; function optimization; exploitation capacity; exploration capacity

群智能优化算法<sup>[1-2]</sup>是求解复杂的实际优化问题的一种计算方法。该类算法从仿生学的角度,模拟自然界中生物生活习性在定义域中快速寻找目标问题的最优解,具有可扩展性、自适应性和并行性<sup>[3]</sup>等多种优点。目前,群智能优化算法被广泛地应用于求解复杂的优化问题中,包括有:蚁群优化(Ant Colony Optimization, ACO)算法<sup>[4]</sup>、粒子群优化(Particle Swarm Optimization, PSO)算法<sup>[5]</sup>、人工蜂群(Artificial Bee Colony, ABC)算法<sup>[6]</sup>、遗传算法(Genetic Algorithm, GA)<sup>[7]</sup>、差分进化(Differential Evolution, DE)算法<sup>[8]</sup>等。

Gandomi 等<sup>[9]</sup>通过对南极磷虾群(Krill Herd, KH)的生活习性的观察和研究,并于 2012 年提出了磷虾群(KH)算法,该群智能优化算法<sup>[10]</sup>通过模仿磷虾的活动方式,能够在算法演化的初期快速收敛,找到表现优秀的可行解。该算法的缺点是磷虾个体随着演化的发展,缺乏全局搜索能力,算法容易陷入局部最优解。针对该问题,相关研究工作者通过优化算法参数和演化策略,改进算法的搜索能力。文献[11]引入差

分演化算子,提出了改进的磷虾群算法(hybrid Differential Evolution Krill Herd, DEKH);文献[12]中提出的改进算法通过线性递减步长对因子进行缩放,设计完成改进的磷虾群算法(Krill Herd with Linear Decreasing step, KHLD);文献[13]中设计了一种基于混沌映射的动态参数优化方法对当前种群中的最优值进行更新,从而实现对磷虾群算法的改进;文献[14]中的改进算法对粒子群与磷虾群两种不同类型的群智能优化算法进行了融合,对两种群智能算法规则下进行个体的演化,并分别保留其中的优秀个体。上述对于传统的 KH 算法的改进主要从参数优化和演化规则两个方面展开研究,均一定程度上改进了算法的性能。然而,已有的改进算法仅仅从适应度函数的角度出发,通过改进参数和适应度评价规则来实现对传统算法的改进,并没有考虑其中优秀个体周围环境中优秀个体对种群演化的影响,有一定的局限性。

为了进一步提升 KH 算法的搜索能力和收敛性能,本文设计了一种动态压力控制算子对演化规则进行控制,实现对

收稿日期:2018-08-13;修回日期:2018-10-12;录用日期:2018-10-17。

作者简介:沈莹(1994—),女,浙江嵊州人,硕士研究生,主要研究方向:群智能计算; 黄樟灿(1960—),男,浙江嵊州人,教授,博士,主要研究方向:智能计算、图像处理; 谈庆(1991—),男,湖南长沙人,硕士研究生,主要研究方向:智能计算; 刘宁(1994—),女,浙江嵊州人,硕士研究生,主要研究方向:智能计算。



磷虾群算法的改进。本文提出的基于动态压力控制算子的磷虾群算法(Krill Herd algorithm based on Dynamic Pressure Control operator, DPCKH),首先利用传统KH算法进行全局搜索,评价并保留潜在的优秀个体;然后,通过本文设计的动态压力控制算子在种群中产生的新个体与旧个体之间进行评价选择,保留优秀的磷虾个体。其中,动态压力控制算子通过在当前种群优秀个体的邻域内寻找相似个体,来形成潜在优秀个体种群,从而增强算法的搜索能力,提高算法的收敛速度。最后,本文中通过7个测试函数对提出的改进算法进行了验证,并选取了2种算法进行了对比。实验结果充分证明了本文设计的改进算法的有效性和高精度。

## 1 KH 算法

磷虾群(KH)算法是由Gandomi和Alavi在2012年提出的一种模仿南极磷虾群生存活动的新型群智能算法。磷虾个体的位置移动主要受到三个因素影响:1)其他磷虾个体的诱导运动;2)觅食活动;3)随机扩散活动。

磷虾个体的移动方向用拉格朗日模型建模:

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

其中  $\mathbf{X}_i = (x_i^1, x_i^2, \dots, x_i^{NV}) \in \mathbb{R}^{NV}$  且对于任意一个分量  $x_i^j$ ,有  $x_i^j \in [LB_j, UB_j]$ ;  $NV$  表示  $\mathbf{X}_i$  的维度;  $\mathbf{N}_i, \mathbf{F}_i, \mathbf{D}_i$  分别代表磷虾个体运动方向对应的诱导、觅食、随机扩散三类活动,它们都是与时间  $t$  相关的。三个因素的构造公式如下:

$$\mathbf{N}_i = N_{\max}^{\max} \alpha_i + w_n \mathbf{N}_i^{\text{old}} \quad (2)$$

$$\mathbf{F}_i = V_f \beta_i + w_f \mathbf{F}_i^{\text{old}} \quad (3)$$

$$\mathbf{D}_i = D_{\max} (1 - 1/t_{\max}) \delta \quad (4)$$

其中:  $N_{\max}, V_f, D_{\max}$  分别表示最大诱导速度、觅食速度、最大随机扩散速度;  $\alpha_i, \beta_i, \delta$  分别表示诱导方向、觅食方向、随机扩散方向;  $w_n, w_f$  分别表示诱导惯性权重和觅食惯性权重;  $t, t_{\max}$  分别表示当前迭代次数和最大迭代次数;  $\mathbf{N}_i^{\text{old}}, \mathbf{F}_i^{\text{old}}$  分别表示上一次迭代的诱导活动方向向量和觅食活动方向向量。诱导方向  $\alpha_i$  定义如下:

$$\alpha_i = \alpha_i^{\text{local}} + \alpha_i^{\text{target}} \quad (5)$$

$$\alpha_i^{\text{local}} = \sum_{j=1}^{NN} \hat{K}_{i,j} \hat{X}_{i,j} \quad (6)$$

$$\hat{X}_{i,j} = (X_j - X_i) / (\|X_j - X_i\| + \varepsilon) \quad (7)$$

$$\hat{K}_{i,j} = (K_i - K_j) / (K^{\text{worst}} - K^{\text{best}}) \quad (8)$$

$$d_{s,i} = \frac{1}{5N} \sum_{j=1}^N \|X_i - X_j\| \quad (9)$$

其中:  $\alpha_i^{\text{target}}$  是最优磷虾个体产生的诱导方向,  $\alpha_i^{\text{local}}$  则是附近磷虾群体产生的诱导方向,  $K^{\text{best}}$  和  $K^{\text{worst}}$  指目前迭代次数下磷虾群体中最优个体的适应度和最差个体的适应度,  $X_i$  表示第  $i$  个磷虾当前的位置,  $K_i$  表示第  $i$  个磷虾当前的适应度值,  $d_{s,i}$  表示第  $i$  个磷虾个体的敏感邻域半径,  $NN$  表示第  $i$  个磷虾个体敏感邻域内的磷虾个体数,  $\varepsilon$  表示正无穷小量。

觅食方向  $\beta_i = \beta_i^{\text{food}} + \beta_i^{\text{best}}$  的定义如下:

$$\beta_i = \beta_i^{\text{food}} + \beta_i^{\text{best}} \quad (10)$$

其中  $\beta_i^{\text{food}}$  为食物对第  $i$  只磷虾的吸引力,可以表示为:

$$X_{\text{food}} = \sum_{i=1}^N \frac{X_i}{K_i} / \sum_{i=1}^N \frac{1}{K_i} \quad (11)$$

$$\beta_i^{\text{food}} = 2(1 - I/I_{\max}) \hat{K}_{i,\text{food}} \hat{X}_{i,\text{food}} \quad (12)$$

$\beta_i^{\text{best}}$  为第  $i$  只磷虾迄今为止适应度最优的状态,对当前第  $i$  只磷虾所造成的影响为:

$$\beta_i^{\text{best}} = \hat{K}_{i,\text{best}} \hat{X}_{i,\text{best}} \quad (13)$$

磷虾个体在  $t$  到  $t + \Delta t$  的位置更新公式如下:

$$X_i(t + \Delta t) = X_i(t) + \Delta t \frac{dX_i}{dt} \quad (14)$$

$$\Delta t = Ct \sum_{j=1}^{NV} (UB_j - LB_j) \quad (15)$$

其中:  $\Delta t$  是位置更新的步长;  $Ct$  是步长限制因子,取值为  $[0, 2]$  内的常数;  $X_i(t)$  是迭代前一次磷虾粒子所处的位置。为了提高算法性能,文献[15]指出使用交叉算子来增强算法的搜索能力。交叉算子公式如下:

$$X_{i,m} = \begin{cases} X_{i,m}, & R_{i,m} < C_r \\ X_{r,m}, & R_{i,m} \geq C_r \end{cases} \quad (16)$$

其中:  $C_r$  为交叉概率阈值  $r \in \{1, 2, \dots, i-1, i+1, \dots, m\}$ ,  $R_{i,m}$  表示对于第  $i$  个个体  $X_i$  中第  $m$  个组分发生交叉运算的概率。根据式(12)所示,当  $R_{i,m} \geq C_r$  时,发生交叉运算,第  $i$  个个体的第  $m$  个组分与第  $r$  个个体进行交叉;否则,当  $R_{i,m} < C_r$  时保持不变。通过交叉算子,种群个体的状态出现更多可能性,从而增强了算法搜索能力。

## 2 本文算法

在通常的KH算法中,由式(5)可知,磷虾个体诱导方向  $\alpha_i$  由附近磷虾的诱导方向  $\alpha_i^{\text{local}}$  和最优个体的诱导方向  $\alpha_i^{\text{target}}$  组合构成。这里的  $\alpha_i^{\text{target}}$  指代种群中最优秀的一个磷虾个体对当前研究的目标磷虾个体产生的诱导方向。对于当前研究的目标磷虾个体而言,比自身优秀的磷虾个体可能有多个,由于没有考虑全局优秀方向的多样性,所以仅由一个优秀个体全局感知方向是存在缺陷的。为了克服多样性的不足,提高标准KH算法的开采能力,动态压力控制算子被加入了KH算法。动态压力控制算子从多个优秀个体与目标个体之间的距离入手,更加侧重于全局搜索,这使算法可以避免早熟。

DPCKH算法中,动态压力控制算子主要分为筛选操作和动态压力控制操作两部分。首先对磷虾群体中所有个体进行编号,磷虾种群  $S$  定义如下:

$$S = \{X_1, X_2, \dots, X_n\} \quad (17)$$

其中:  $S$  表示磷虾种群,  $X_i \in S$  表示磷虾个体。其次进行筛选操作,计算磷虾种群  $S$  对应的适应度函数值,定义为集合  $T$ :

$$T = \{f(X_1), f(X_2), \dots, f(X_n)\} \quad (18)$$

其中  $f(\cdot)$  是适应度函数。对集合  $T$  进行排序,选取排名前  $k\%$  的磷虾个体构成优秀个体集合,定义为集合  $S_1$ :

$$S_1 = \{X_{k1}, X_{k2}, \dots, X_{km}\} \quad (19)$$

假设当前研究目标磷虾是  $X_0 \in S$ ,则可以对应得到优秀个体诱导集合  $S_2$ :

$$S_2 = \begin{cases} S_1 \setminus \{X_0\}, & X_0 \in S_1 \\ S_1, & X_0 \notin S_1 \end{cases} \quad (20)$$

其中:  $S_2$  表示优秀磷虾个体构成的诱导集合,这个集合是不包含  $X_0$  本身的。至此筛选操作结束,接下来进行动态压力控制操作。对于当前研究目标磷虾  $X_0$ ,其全局优秀方向由集合  $S_2$  与  $X_0$  共同决定。定义集合  $S_2$  中元素与  $X_0$  的欧氏距离构成的集合为  $D$ :

$$D = \{d_1, d_2, \dots, d_r\} \quad (21)$$

其中:  $d_r$  表示  $X_r \in S_2$  与  $X_0$  的欧氏距离。然后对集合  $D$  进行由大到小排序,得到下标的集合  $Ind$ :

$$Ind = \{ind_1, ind_2, \dots, ind_r\} \quad (22)$$

其中:  $ind_r \in \{1, 2, \dots, r\}$ 。假设当前迭代次数为  $Cen$ ,目标最大迭代次数为  $MaxGen$ ,则定义动态压力控制权重的变化范围为



{ $wa, wb$ }:

$$p_x = \frac{\lfloor Gen - MaxGen/2 \rfloor}{MaxGen/2} \times (1 - 1/C) + 1/C \quad (23)$$

$$wa = \begin{cases} C \times px, & Gen < MaxGen/2 \\ 1, & Gen \geq MaxGen/2 \end{cases} \quad (24)$$

$$wb = \begin{cases} 1, & Gen < MaxGen/2 \\ C \times px, & Gen \geq MaxGen/2 \end{cases} \quad (25)$$

其中: $C$ 表示限制常数,取值范围为[0,0.2], $wa$ 对应 $D$ 中距离最大 $X_i$ 的权值, $wb$ 对应 $D$ 中距离最小 $X_i$ 的权值。其他优秀个体权值按照距离由大到小线性插值。至此可以得出 $S_2$ 对应的权重集合 $W$ :

$$W = \{w_1, w_2, \dots, w_r\} \quad (26)$$

其中: $w_r$ 表示 $X_r \in S_2$ 与 $X_0$ 的对应方向上的动态压力权值。下面计算 $X_0$ 与 $X_r$ 分别对应的适应度函数值,假设 $F(X_0) > F(X_r)$ ,则产生新解操作如下:

$$Y_r = X_0 + (X_r X_0 / |X_r X_0|) \times w_r \times rand \quad (27)$$

其中 $rand$ 是(0,1]的伪随机数。 $Y_r$ 是通过动态压力控制算子产生的新磷虾个体。对于优秀个体诱导集合 $S_2$ ,一次轮换会产生 $r$ 个新的潜在优秀磷虾个体,定义:

$$K_0 = \{Y_1, Y_2, \dots, Y_r\} \quad (28)$$

其中 $K_0$ 是 $X_0$ 在优秀个体诱导集合 $S_2$ 下产生的潜在优秀磷虾个体集合,则对于种群 $S$ 而言,定义集合 $K$ 为其产生的新个体的集合:

$$K = \{K_1, K_2, \dots, K_n\} \quad (29)$$

则在精英策略下更新后的磷虾种群 $\bar{S}$ 定义如下:

$$\bar{S} = best(S, K) \quad (30)$$

其中: $best$ 表示在种群 $S$ 和种群 $K$ 中选取最优秀的 $n$ 个磷虾个体保存到 $\bar{S}$ 中。

动态压力控制算子在DPC KH 算法演化初期主要起到局部搜索的作用,此时距离较近的优秀个体权值较大,其诱导作用较强,对算法本身起到加速作用。当DPC KH 算法演化到中后期时,距离较远的优秀个体权值较大,其诱导作用更强,由于新个体产生的搜索区域更大,因此跳出局部最优的能力更强,所以增强了算法的全局寻优能力。

动态压力控制算子的详细计算方法见算法1。

### 算法1 动态压力控制算子。

```

1) begin
2)   k = 1
3)   Calculate the size of population as the popsize
4)   while(popsize > 3)
5)     Calculate the fitness with S
6)     Select the excellent individual induction set S2
7)     Calculate Euclidean distance matrix between X(k) and
       the elements of S2
8)     Calculate the weight set W based on S2
9)     Calculate the size of S2
10)    t = 1
11)    while(t < the size of S2)
12)      Calculate the fitness with X(k) and X(t)
13)      if(fitness(X(k)) > fitness(X(t)))
14)        Y(k) = X(i) + rand × W(t) × (X(k) - X(t)) /
           |X(t) - X(k)|
15)      else
16)        Y(k) = X(t) + rand × W(t) × (X(t) - X(k)) /
           |X(t) - X(k)|
17)    end
18)    save Y into K(t)

```

19)  $t = t + 1$

20) end

21)  $k = k + 1$

22) end

23) Compare the  $X$  and  $K$ , save the better individuals

24) end

其中: $popsize$ 指的是种群里磷虾个体的数目, $rand$ 表示[0,1]中的伪随机数, $X$ 是原始种群中的磷虾个体, $K$ 是通过近邻搜索算子新产生的磷虾个体。

在动态压力控制算子中,对于函数值最小化问题,如果新产生的磷虾个体适应度值比较差,旧的个体将被保留;如果新产生的个体适应度值比较好,则新的个体将被保留下来。然后DPC KH 算法进入下一次演化。

在DPC KH 算法中,标准的KH 算法被用于全局搜索,其目的在于提供优秀个体的目标区域。在对可行域进行初步筛选之后,动态压力控制算子作为一个局部算子被引入。在算法演化初期,动态压力控制算子主要体现了目标个体近距离优秀个体的诱导作用,这种性质能够提高算法初期的收敛速度。进一步的,在算法演化的中后期,动态压力控制算子主要体现了目标个体远距离优秀个体的诱导作用,这种性质能够提高算法跳出局部最优的能力,使得算法具备更强的全局寻优能力。简而言之,DPC KH 算法平衡了磷虾个体在演化过程中开采与探索的矛盾。

将动态压力控制算子加入KH 算法之中,得到DPC KH 算法。该算法的流程如图1所示。

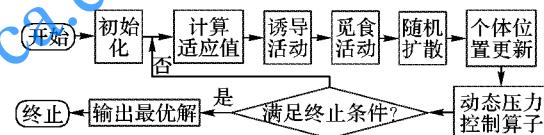


图1 DPC KH 算法流程

Fig. 1 Flow chart of DPC KH algorithm

## 3 仿真实验与结果分析

### 3.1 测试函数与实验参数

为了详细检验分析DPC KH 算法在函数优化问题下的性能,本文选取标准KH 算法、DPC KH 算法、KHLD 算法以及文献[5]中提到的PSO 算法进行比较。实验选取了7个典型的基准测试函数(见表1),其中: $f_1, f_2$ 是高维单峰函数, $f_3$ 是高维多峰函数, $f_4 \sim f_7$ 是低维函数并且仅有少数几个局部极小值。

表1 测试函数的维度  
Tab. 1 Dimensions of test functions

函数	函数名称	维度	邻域	最小值
$f_1$	Generalized Rosenbrock's Function	5	$[-30, 30]^D$	0
$f_2$	Step Function	5	$[-100, 100]^D$	0
$f_3$	Generalized Schwefel's Problem 2.26	5	$[-500, 500]^2$	-2094.9
$f_4$	Kowalik's Function	4	$[-5, 5]^D$	0.003 075
$f_5$	Six-Hump Camel-Back Function	2	$[-5, 5]^D$	-1.031 628
$f_6$	Branin Function	2	$[-5, 10] \cup [0, 15]$	0.398
$f_7$	Hartman's Function 2	6	$[0, 1]^D$	-3.32

实验参数设置:KH 算法与DPC KH 算法采用文献[14]中的参数进行设置,其中最大诱导速度 $N_{max} = 0.01$ ,最大觅食速度 $V_f = 0.02$ ,最大随机扩散速度 $D_{max} = 0.005$ ,KH 和



DPC KH 算法中  $C_t = 1$ , DPC KH 算法中  $k = 10$ ,  $C = 0.2$ 。对于 PSO 算法,所有的参数设置按照文献[5] 中设置,对于 KHLD 算法,其参数按照文献[12] 中进行设置。

### 3.2 实验结果

所有的算法均在 Matlab 2016a 软件上进行实现。其中种

群规模设置为 50,最大迭代次数设置为 1000,每种算法在对应的测试函数上运行独立重复运行 10 次,计算其平均最优适应度(Mean)、标准差(Std)以及最差的最优适应度(Worst)。关于测试函数的维度详见表 1,对应各种算法的运行结果如表 2 所示。

表 2 各算法结果  
Tab. 2 Results of different algorithms

函数	维度	指标	ACO	DE	PSO	KH	KHLD	DPC KH
$f_1$	5	Mean	7.60E + 01	1.88E + 01	3.6191E + 00	2.9676E + 00	1.4073E + 00	1.0584E - 03
		Std	9.91E + 02	1.93E + 01	1.5804E + 00	3.3177E + 00	5.0385E - 01	1.4306E - 03
		Worst	9.61E + 01	6.54E + 00	5.9378E + 00	1.0232E + 01	1.9472E + 00	3.8654E - 03
$f_2$	5	Mean	3.64E + 00	4.23E + 01	1.0000E + 00	0.0000E + 00	0.0000E + 00	0.0000E + 00
		Std	5.11E + 00	1.10E + 02	0.0000E + 00	0.0000E + 00	0.0000E + 00	0.0000E + 00
		Worst	2.44E + 00	1.86E + 01	1.0000E + 00	0.0000E + 00	0.0000E + 00	0.0000E + 00
$f_3$	5	Mean	4.91E + 01	5.56E + 01	1.0971E + 03	2.4872E + 02	2.0135E + 02	7.1100E + 01
		Std	6.28E + 02	8.74E + 01	1.7429E + 02	1.9301E + 02	1.3733E + 02	1.2730E + 02
		Worst	2.67E + 02	7.90E + 01	1.2823E + 03	5.3298E + 02	4.7375E + 02	3.5532E + 02
$f_4$	4	Mean	2.10E - 01	1.57E - 01	2.5733E - 02	2.6335E - 03	8.1450E - 04	3.0749E - 04
		Std	4.32E - 01	3.64E - 01	4.1411E - 02	6.2300E - 03	1.5471E - 04	2.7689E - 10
		Worst	6.89E - 01	1.21E - 01	1.1665E - 01	2.0363E - 02	1.2255E - 03	3.0749E - 04
$f_5$	2	Mean	2.67E - 03	1.17E - 06	1.0000E - 04	4.0753E - 10	1.6143E - 10	1.3307E - 10
		Std	6.06E - 03	2.54E - 04	0.0000E + 00	3.2520E - 10	2.5341E - 10	2.9317E - 10
		Worst	3.44E - 03	2.12E - 04	1.0000E - 04	1.0732E - 09	8.5031E - 10	9.6082E - 10
$f_6$	2	Mean	1.77E - 10	8.23E - 09	7.3848E + 00	1.7125E - 09	9.5749E - 10	2.3318E - 11
		Std	8.45E - 10	7.14E - 08	9.3622E - 16	2.2069E - 09	1.9147E - 09	3.5591E - 11
		Worst	4.65E - 10	8.43E - 08	7.3848E + 00	7.2090E - 09	6.1788E - 09	1.1789E - 10
$f_7$	6	Mean	1.64E + 00	2.05E + 00	2.8461E - 01	4.7497E - 02	7.1246E - 02	1.1874E - 02
		Std	2.44E + 00	2.41E + 00	2.3791E - 01	6.1319E - 02	6.1319E - 02	3.7550E - 02
		Worst	5.43E + 00	2.25E + 00	6.2973E - 01	1.1874E - 01	1.1874E - 01	1.1874E - 01

从表 2 中不难发现,DPC KH 算法相比标准 KH 算法、标准 PSO 算法、ACO 算法和 DE 算法而言在最大迭代次数内拥有更高的精度,其跳出局部最优的能力更强。对比 Mean、Std、Worst 三个指标发现,DPC KH 算法在这三个方面均领先于其他三个算法,说明 DPC KH 算法不仅在优化精度上领先其他算法,而且在算法的鲁棒性上也具备一定优势。Mean 和 Std 的优秀表现说明了在有限迭代次数内,DPC KH 算法的全局寻优能力优于其他算法,而 Std 的优秀表现则说明了 DPC KH 算法更加稳定。

为进一步探讨 DPC KH 算法的全局搜索能力与局部开采能力,针对高维单峰函数、高维多峰、低维少局部极值三类测试函数,分别提取 DPC KH 算法、KH 算法、KHLD 算法、PSO 算法的平均误差演化对比结果如图 2 所示。

高维度单峰函数 Generalized Rosenbrock's Function 和 Generalized Schwefel's Problem 2.26 测试结果如图 2(a)和(b)所示。PSO 算法的收敛速度最快,但是也容易陷入局部最优,导致其寻优精度有限。DPC KH 算法不仅收敛速度优于标准 KH 算法和 KHLD 算法,而且寻优精度也高于后两者。说明 DPC KH 算法具有较强的全局探索能力,能够有效地跳出局部最优。高维多峰函数 Kowalik's Function 的测试结果如图 2(c)所示,PSO 算法依旧是很快陷入了局部最优,标准 KH 算法与 KHLD 算法则一定程度上改善了 PSO 算法容易陷入局部最优的问题,但是两者无论是收敛速度还是优化精度都劣于 DPC KH 算法。低维少极值函数 Six-Hump Camel-Back Function、Branin Function 和 Hartman's Function 2 的测试结果如图 2(d)~图 2(f)所示。PSO 算法很快地陷入了局部最优,

而 DPC KH 算法无论是在收敛速度还是收敛精度上都优于标准 KH 算法和 KHLD 算法。仔细对比(a)、(d)与(e),可以发现 DPC KH 算法的演化过程中,前期由于较近优秀个体的诱导作用主导,存在一个较快的收敛速度,能够高效地找到潜在优秀个体所在的区域,而中后期由于较远优秀个体的诱导作用主导,提高了算法跳出局部最优的能力,加强了算法全局探索的作用。总的来说,基于动态压力控制算子的 DPC KH 算法在整个演化过程中,跳出局部最优的能力强于 PSO 算法、KH 算法与 KHLD 算法。动态压力控制算子克服了标准 KH 算法局部寻优能力不足的缺点,提高了算法的开采能力。

### 4 结语

本文针对 KH 算法在处理复杂函数优化问题上局部搜索能力差、开采能力不足导致的收敛速度慢、收敛精度有限的现象,提出了基于动态压力控制算子的磷虾群算法(DPC KH)。DPC KH 算法从优秀磷虾个体的诱导效应入手,通过不同距离下的动态压力权重控制,在优秀个体附近加速产生新的优秀个体,充分考察了优秀磷虾个体对目标磷虾个体的影响。进而克服了标准 KH 算法中局部搜索能力不足、开采能力弱的问题。基于 7 个测试函数在多个维度的测试结果表明,DPC KH 算法相比标准 KH 算法、KHLD 算法、PSO 算法不仅有着更强的全局搜索能力,寻优精度更高,而且收敛速度更快,稳定性更好。动态压力控制算子让 DPC KH 算法相较与普通 KH 算法有着更强的局部勘测能力。下一步的研究目标是继续探讨磷虾群算法在不同参数、不同压力控制下优化效果。

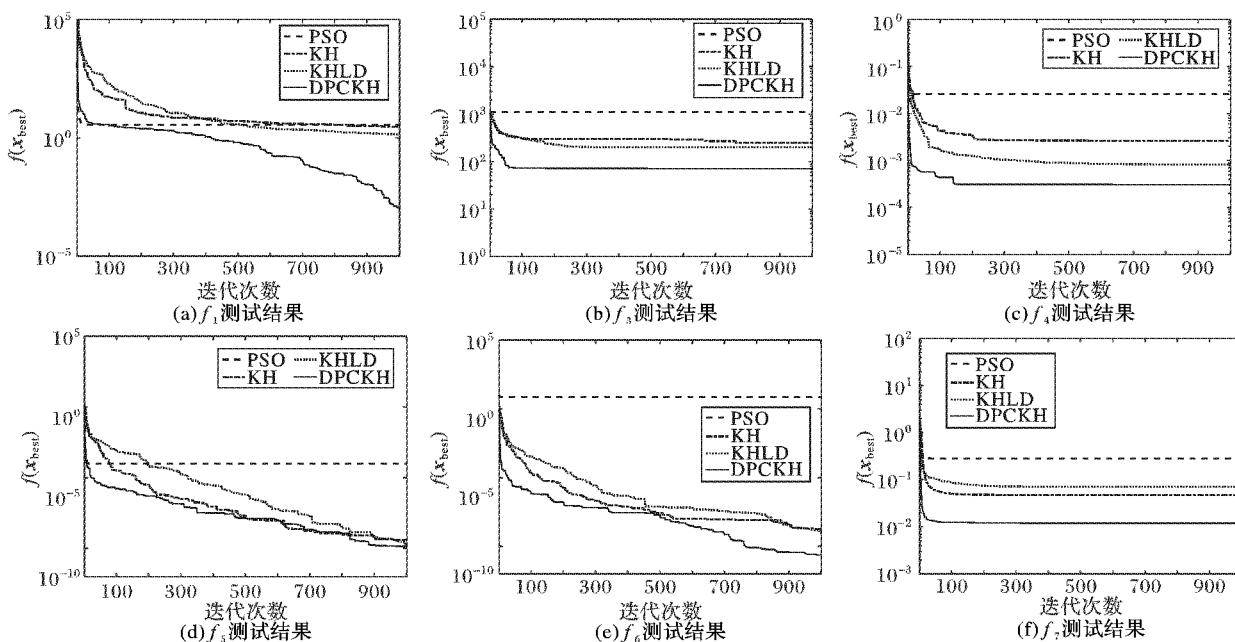


图2 各函数测试结果

Fig. 2 Test results of different functions

## 参考文献 (References)

- [1] 林诗洁,董晨,陈明志,等.新型群智能优化算法综述[J].计算机工程与应用,2018,54(12):1–9.(LIN S J, DONG C, CHEN M Z, et al. Summary of new group intelligent optimization algorithms [J]. Computer Engineering and Applications, 2018, 54(12): 1–9.)
- [2] 程述立,汪烈军,秦继伟,等.群智能算法优化的结合熵的最大类间方差法与脉冲耦合神经网络融合的图像分割算法[J].计算机应用,2017,37(12):3528–3535.(CHENG S L, WANG L J, QIN J W, et al. Image segmentation algorithm based on fusion of group intelligent algorithm optimized OTSU-entropy and pulse coupled neural network [J]. Journal of Computer Applications, 2017, 37(12): 3528 – 3535.)
- [3] SALEEM M, di CARO G A, FAROOQ M. Swarm intelligence based routing protocol for wireless sensor networks: survey and future directions [J]. Information Sciences, 2010, 181(20): 4597 – 4624.
- [4] QU M. Lunar soft-landing trajectory of mechanics optimization based on the improved ant colony algorithm [J]. Applied Mechanics and Materials, 2015, 3748(721): 446 – 449.
- [5] FAYCAL H, ANIS L, ANIS S, et al. A new images segmentation method based on modified particle swarm optimization algorithm [J]. International Journal of Imaging Systems and Technology, 2013, 23 (3): 265 – 271.
- [6] PAN Q-K. An effective co-evolutionary artificial bee colony algorithm for steelmaking-continuous casting scheduling [J]. European Journal of Operational Research, 2016, 250(3): 702 – 714.
- [7] 肖振久,孙健,王永滨,等.基于果蝇优化算法的小波域数字水印算法[J].计算机应用,2015,35(9):2527–2530.(XIAO Z J, SUN J, WANG Y B, et al. Wavelet domain digital watermarking method based on fruit fly optimization algorithm [J]. Journal of Computer Applications, 2015, 35(9): 2527 – 2530.)
- [8] 刘宝,董明刚,敬超.改进的排序变异多目标差分进化算法[J].计算机应用,2018,38(8):2157 – 2163.(LIU B, DONG M G, JING C. Multi-objective differential evolution algorithm with improved ranking-based mutation [J]. Journal of Computer Applications, 2018, 38(8): 2157 – 2163.)
- [9] GANDOMI A H, ALAVI A H. Krill herd: a new bio-inspired optimization algorithm [J]. Communications in Nonlinear Science and Numerical Simulation, 2012, 17(12): 4831 – 4845.
- [10] HOFMANN E E, HASKELL A G E, KLINCK J M, et al. Lagrangian modelling studies of Antarctic krill (*euphausia superba*) swarm formation [J]. ICES Journal of Marine Science, 2004, 61 (4): 617 – 631.
- [11] WANG G-G, GANDOMI A H, ALAVI A H, et al. Hybrid krill herd algorithm with differential evolution for global numerical optimization [J]. Neural Computing and Applications, 2014, 25(2): 297 – 308.
- [12] LI J, TANG Y, HUA C, et al. An improved krill herd algorithm: krill herd with linear decreasing step [J]. Applied Mathematics and Computation, 2014, 234(10): 356 – 367.
- [13] WANG G-G, GUO L, GANDOMI A H, et al. Chaotic krill herd algorithm [J]. Information Sciences, 2014, 274: 17 – 34.
- [14] WANG G-G, GANDOMI A H, ALAVI A H, et al. A hybrid method based on krill herd and quantum-behaved particle swarm optimization [J]. Neural Computing and Applications, 2016, 27(4): 989 – 1006.
- [15] 刘沛,高岳林,郭伟.基于自然选择和随机扰动的改进磷虾群算法[J].小型微型计算机系统,2017,38(8):1845 – 1849.(LIU P, GAO Y L, GUO W. Improved krill herd algorithm based on natural selection and random disturbance [J]. Journal of Chinese Computer Systems, 2017, 38(8): 1845 – 1849.)

**SHEN Ying**, born in 1994, M. S. candidate. His research interests include swarm intelligent computation.

**HUANG Zhangcan**, born in 1960, Ph. D., professor. His research interests include intelligent computation, image processing.

**TAN Qing**, born in 1991, M. S. candidate. His research interests include intelligent computation.

**LIU Ning**, born in 1994, M. S. candidate. Her research interests include intelligent computation.