文章编号：1001-506X(2012)07-1484-09

基于细致化仿生的改进粒子群优化算法

王兴元，张　鹏
（山东大学管理学院，山东 济南 250100）

摘　要：粒子群优化(particle swarm optimization，PSO)算法基本思想是试图通过模拟鸟群觅食中的迁徙和聚集等行为获得连续非线性函数的最佳值，其仿生算法产生于对鸟群寻食过程中飞行方向与飞行速度等的隐喻。近年对粒子群算法经典算法的研究，虽然在速度及精度上有所改进，但由于缺乏细致化仿生(precise bionic metaphor，PBM)，改进效果并不太明显。通过在 PSO 算法中引入飞鸟寻食细致化行为特征隐喻，即在算法中同时导入满意粒子局地细致化寻优和探索粒子随机寻优过程，进而提出了一种新的基于细致化仿生的改进 PSO 算法；对改进算法和经典算法进行了性能比较，结果显示所提算法在收敛速度和求解精度方面较经典算法有很大程度的改善。

关键词：粒子群优化；改进粒子群优化；满意粒子；探索粒子；细致化仿生

中图分类号：TP 301.6　　**文献标志码**：A　　**DOI**：10.3969/j.issn.1001-506X.2012.07.33

Improved particle swarm optimization based on precise bionic metaphor

WANG Xing-yuan，ZHANG Peng
(*School of Management*，*Shandong University*，*Jinan 250100*，*China*)

Abstract：The basic idea of particle swarm optimization (PSO) is to obtain the optimum value of the continuous nonlinear function by simulating birds behavior such as the direction and the speed of the flight in migration and foraging aggregation. In recent years，the results improvement of the classical PSO algorithm are not obvious because of the lack of precise bionic metaphor (PBM). By introducting PBM into the PSO，a new improved PBM-PSO is set up. The results show that the improved particle swarm optimization converges more quickly and gets a more accurate solution than the classical algorithm.

Keywords：particle swarm optimization (PSO)；improved particle swarm optimization (IPSO)；satisfied particle；explorer particle；precise bionic metaphor (PBM)

0　引　言

粒子群优化(particle swarm optimization，PSO)算法是文献[1]提出的一种基于群体行为求解连续非线性函数最优值的思想。PSO 算法基本思想是试图通过模拟动物群体觅食中的迁徙和聚集等行为获得连续非线性函数的最值，该算法求得的最值不一定是理论上最优值，但可以令人满意。同其他算法相比较，PSO 算法需要设置的参数很少，能够迅速、低成本地获得较好结果，它在许多工程与管理问题的研究和实践中获得了成功的应用。

自 PSO 算法提出以来，很多学者对其进行了不断改进，主要包括：

(1) 初始解的优化。如文献[2]提出，有策略地选择粒子初始位置可以提高 PSO 算法的性能；文献[3－4]通过新的粒子群初始化方法优化了 PSO 算法。

(2) 公式参数的优化。如文献[5－10]对惯性权重、加速系数进行了优化；文献[11]对随机数进行了优化。

(3) 公式形式的优化。如文献[12]提出通过增加收敛因子提高 PSO 算法的性能；文献[13－14]通过优化边界提高 PSO 算法收敛性；文献[15－19]通过优化粒子位置更新规则优化 PSO 算法，并有很多学者提出利用扰动优化 PSO 算法[20-27]。

(4) 粒子群体划分与信息传递优化。如文献[28－30]通过设置具有信息交互能力的多群体对 PSO 算法进行优化；文献[31－33]提出利用其他粒子信息、隔代信息等优化粒子寻优。

(5) 算法杂交。如文献[34－35]借鉴遗传算法优化 PSO 算法；文献[36－37]借鉴混沌理论优化 PSO 算法；

收稿日期：2011－07－27；**修回日期**：2012－03－13。
基金项目：国家自然科学基金(70872062)资助课题
作者简介：王兴元(1962－)，男，教授，博士研究生导师，主要研究方向为管理科学与工程。E-mail：wangxingyuan@126.com

文献[38]则综合 PSO 算法和蚁群算法提出了粒子群蚁群优化算法；文献[39]将二进制 PSO 算法与局部爬山搜索策略相结合给出了一种求解可满足性问题的新算法。

(6) 数值计算视角的改进。如文献[40]提出了非对称并行异步 PSO 算法。

(7) 拓展 PSO 算法的用途。如文献[41]提出通过取整的方法，将离散粒子群算法用于整数规划；文献[42]提出了离散二元版本 PSO 算法；文献[43]提出了模糊离散 PSO 算法用以解决旅行商问题。

(8) 生态模仿。如文献[44]通过模仿群体领土争端改进了 PSO 算法用以多峰函数优化问题。

PSO 算法优异的性能来自于其仿生学基础，尽管目前对于 PSO 算法的改进取得了许多进展，但大多数还是通过对鸟群群体飞行寻食行为的观察与仿生。一群鸟在一个区域内觅食，意图找到食物最丰富的位置。这些鸟随机分布在这一区域中，它们并不清楚哪里食物最丰富，但它们彼此之间可以食物寻找情况进行通信，即鸟儿们能够知道其他鸟所处位置以及这些位置食物丰富状况。每只鸟在飞到某一位置之后，对这一位置食物丰富情况进行判断，之后继续寻找是否有更好的食物点，其飞临的新位置主要依据 3 点确定：①飞行的惯性，即上一次寻食的前进方向；②群体经验的影响，即到目前为止，整个群体发现的食物最丰富的位置；③是个体经验的影响，即到目前为止，个体所发现的食物最丰富的位置。

经典 PSO 算法的迭代求解过程为：

步骤 1　设置各个粒子初始速度和位置。将各个粒子当前适应度作为个体最优适应度，将各个粒子当前位置作为个体最优位置，选取适应度最佳的粒子，将该粒子的适应度作为群体最优适应度，将该粒子位置作为群体最优位置，设置加速系数与随机数。

步骤 2　进行一次迭代。依据公式求解粒子群各个粒子的新位置；计算所有粒子新位置适应度；如果其当前适应度比其个体最优适应度更优，则将当前适应度作为其个体最优适应度，同时将当前位置作为其个体最优位置，否则不进行个体最优适应度和个体最优位置的修改。

步骤 3　选取当前适应度最佳的粒子。如果该粒子的适应度比群体最优适应度更优，则将该粒子的适应度作为群体最优适应度，同时将该粒子位置作为群体最优位置，否则不进行群体最优适应度和群体最优位置的修改。

步骤 4　判断是否满足迭代终止条件。如果满足，则迭代过程终止，输出群体最优适应度与群体最优位置，否则转到步骤 2 继续迭代。

然而，由于经典 PSO 算法中的仿生只是考虑鸟群中的个体鸟依据自身位置、经验及与其他鸟的通讯等信息确定飞行方向与寻食点，没有模仿个体鸟落地后的细致化啄食行为，如个体鸟连续的摆动前行行为或地上跳跃行为等。现实世界中的飞鸟当发现食物后，鸟群中的个体鸟落在一定地点上，然后按着视线围绕当前位置摆动与前行的方式进行啄食与继续寻食。当确无食物时，它会飞走落在其他地方继续按固定规则寻食。本文在分析现有研究的基础上，根据飞鸟细致化生物行为特征，引入满意粒子局地细化寻优和引入探索粒子全局寻优，提出了一种改进的 PSO 方法。该算法在吸收 PSO 经典算法的基础上，克服经典算法只是计算落脚点数值的规则，围绕落脚点细致化搜寻临近点的数值状态。由于算法从一个落脚点开始采用细致化网状搜寻，最终形成一条带状搜寻轨迹，加上探索粒子的随机干预，较易得到复杂问题方程的最优解。算例结果显示本算法收敛速度与计算精度较经典算法均有明显提高。

1　基于细致化仿生的改进算法

1.1　改进方法

PSO 算法是一种仿生算法，其优异的性能得益于生物行为隐喻，通过进一步细致化仿生（precise bionic metaphor, PBM），可能使 PSO 算法获得更好的性能。本文在分析现有研究的基础上，根据两种飞鸟寻食行为特征提出了一种基于 PBM 的改进 PSO 算法。

1.1.1　引入满意粒子局地细化寻优方法

人们在某处发现一块古代陶片，会猜测附近可能有更多的文物，进而在周边区域仔细搜索；在某棵果树上找到一个美味的果实，可能会继续在这棵果树上寻求美味的果实而不是马上走开；鸟群发现某一较好的取食位置后，会在附近继续觅食寻求更好的位置。受这些现象的启发，本文引入满意粒子局地细化寻优的方法。“满意粒子”是指对当前位置适应度感到满意的粒子，这里“满意”指粒子认为当前其适应度同其他粒子适应度相比居于领先地位（比如位于前 5%），满意粒子推测当前位置的附近可能存在能够达到足够好适应度的位置，因此这些粒子会在其周边搜索，寻找适应度更高的位置，即进行“局地寻优”。局地寻优为：以当前位置为中心划定一个封闭区域，即“局地”；对区域进行网格式细化，探索顶点处是否有更优适应度，如果不存在更优值则在当前位置进行缩小区域和网格间距的更细致局地寻优，即细化寻优；如果存在更优值：当更优值位置处于局地边界时，认为高适应度位置可能不在局地内，因此移动至更优值位置进行范围不变的局地寻优，不进行细化寻优的目的是防止因缩小范围导致无法紧密追寻高适应度位置，当更优值位置不处于边界，认为高适应度位置可能在局地内，此时移动至更优值位置进行细化寻优；局地寻优停止与否由细化程度确定，当细化程度已经达到既定好的细化阈值要求时，继续局地寻优，否则停止局地寻优。

如果某次寻优中，满意粒子局地寻优后发现其找到的位置自己曾经到达过（两个点各个维度坐标差小于细化阈值要求），则跳出至一随机位置以防止陷入局部极值。

也有学者提出了利用梯度等信息追击更优点的方

法[45-48]，但这与本文提出局地细致化寻优的思想有着本质的区别。主要有两点：一是梯度等算法类似爬虫在平面上二维寻优，容易被眼前的优劣所左右而偏离最优点。本文提出的改进方法对于局地进行了网格划分，探索不同距离网格点，模仿了三维环境下飞鸟对于地面的寻食过程，这可以避免被眼前优劣左右，之后的比较研究也证明了这种类三维寻优的优势。二是有些梯度算法需要知道欲求极值函数的导函数形式，本文提出的改进方法只要知道特定点的函数值即可，无需知道欲求极值函数的导函数形式。

1.1.2　引入探索粒子

人类社会与生物群体中都存在着探索者，比如在人们认识、探索自然的过程中，有些人不喜欢循规蹈矩，按照既有线路前进，而是喜欢思维上进行跳跃，行动上另辟蹊径以寻奇探幽，其行为并不受其他人或历史经验影响，而这些人往往可能有重大贡献；生物界尽管存在鸟、鱼等群体觅食行为，但同时也存在某些个体单独行动的现象。受这些现象的启发，本文欲在粒子群中引入一定比例“探索者”，即探索粒子。探索粒子是指行为并不受个体行为惯性、个体最优适应度和群体最优适应度影响，但其行为结果会影响其他粒子的粒子。算法设计上，探索者的新位置完全随机确定，等概率地分布于空间中，探索者参与满意粒子局地细化寻优。

目前有不少学者提出了通过加入扰动或(随机)变异操作改进 PSO 算法性能，但这和本文的探索粒子有着显著区别，本文提出的探索粒子完全随机定位，并非依赖于位置公式或速度公式迭代结果。另外本文提出的探索粒子不存在速度概念，其位置并非是速度的函数，而是完全随机给定；从范围上，随机粒子位置是整个求解空间，而非$[-X_{\max},X_{\max}]$。

1.2　改进算法

改进算法以经典 PSO 算法为基础，在经典 PSO 算法完成一次迭代以后，选取所有粒子(包含探索者粒子)中具有较高适应度的粒子进行局地寻优，这部分粒子的数量由粒子总数量乘以一比例参数确定；局地寻优粒子以其所在点为中心，对一定空间进行网格式划分，计算网格点处适应度，划分空间大小以及划分份数由参数确定；寻优粒子比较不同网格点的适应度，将具有最高适应度的网格点作为更优点，移至更优点，并更新个体最优值和个体最优位置；寻优粒子检测目前网格细分程度是不是已经达到了设定的阈值，如果已经达到，则无需继续进行局地寻优，此时应更新群体最优值和群体最优位置等；如果没有达到，则需要继续进行局地寻优，但此时应该根据更优点的位置确定进一步局地寻优的范围参数：当更优点为边界网格点时，为了防止成比例缩小局地寻优范围带来的前进距离有限($p>1$ 时，$\sum\limits_{i=1}^{\infty}\dfrac{1}{p^{i}}=C$，$C$ 为常数)，从而可能无法紧密追踪更高适应度位置，这时应保持局地寻优范围不变，继续进行局地寻优；当更优点不是边界网格点时，应该缩小局地范围进行细化局地寻优。局地寻优完毕后，所有粒子检测本次迭代最终位置和上次迭代最终位置是否相同(这里认为足够接近即为相同，比如当两点各维度坐标之差的绝对值都小于网格细分程度阈值时，则认为两个位置相同)，如果相同，则可能陷入了局地极值，应该跳出至随机位置，否则不跳动；等待其他粒子局地寻优完毕后，进入下一次迭代。改进算法思路如图 1 所示。

图 1　改进算法框图

1.3 改进算法说明

1.3.1 变量说明

设 d 表示搜索空间维数；n 表示粒子群粒子总数；t 表示迭代次数；$\boldsymbol{x}^{i,t}=[x_1^{i,t},x_2^{i,t},\cdots,x_d^{i,t}]$表示第 i 个粒子第 t 次迭代中所处位置向量；$\boldsymbol{v}^{i,t}=[v_1^{i,t},v_2^{i,t},\cdots,v_d^{i,t}]$表示第 i 个粒子第 t 次迭代中运动速度向量；pBest^i 表示第 i 个粒子个体最优适应度值；$\boldsymbol{p}^{i,t}=[p_1^{i,t},p_2^{i,t},\cdots,p_d^{t}]$表示第 i 个粒子第 t 次迭代中取得个体最优适应度值的位置向量；gBest 为群体最优适应度值；$\boldsymbol{p}^{t}=[p_1^{t},p_2^{t},\cdots,p_d^{t}]$表示第 t 次迭代中取得群体最优适应度值的位置向量；r 表示探索粒子数量比例；c_1,c_2 表示加速度系数，r_1,r_2 表示随机变量；f 表示适应度函数，并设定求解问题为 $\max f$；H 表示探索粒子集合，K 表示非探索粒子集合；rand (d)表示随机函数，它能随机产生 d 维空间中的一个坐标；q 表示满意粒子比例，将粒子按适应度排序，认为居于前 q 的粒子是满意的，会进行局地细化寻优；M^t 表示第 t 次迭代中满意粒子集合；$\boldsymbol{g}=[g_1,g_2,\cdots,g_d]$表示粒子局地细化寻优中在各个维度上探索范围半径向量；$\boldsymbol{s}=[s_1,s_2,\cdots,s_d]$表示对于 g 的网格细化向量，s_j 为整数，表示将第 j 维搜索范围半径均匀划分的份数，局地寻优时选取网格细化后粒子摩尔邻居网格节点位置进行适应度更优值的探测，以二维空间为例，g 和 s 的含义如图 2 所示；$\boldsymbol{e}=[e_1,e_2,\cdots,e_d]$表示满意粒子局地细化寻优细化精度阈值向量，其含义为在追寻更优适应度时，在空间细化上至少应该达到的精度。

图 2　参数 g 与 n 含义示意图

1.3.2 初始化粒子群

在 d 维搜索空间中创建 n 个粒子组成粒子群。设置各个粒子初始位置 $\boldsymbol{x}^{i,0}$ 和速度 $\boldsymbol{v}^{i,0}$，取 $\mathrm{pBest}^i=f(\boldsymbol{x}^{i,0})$，$\boldsymbol{p}^{i,0}=\boldsymbol{x}^{i,0}$，令 $u=\arg\max\limits_i\{\mathrm{pBest}^i\}$，取 $\mathrm{gBest}=\mathrm{pBest}^u$，$\boldsymbol{p}^0=\boldsymbol{p}^{u,0}$，设置 c_1,c_2，设置 r_1,r_2，设置 $\boldsymbol{e}$。

随机选取 $n\times r$ 个粒子(如果有小数则取整)加入探索粒子集合 H，剩余粒子加入非探索粒子集合 K。

1.3.3 迭代计算

设第 t 次迭代已经完成，本次迭代为第 $t+1$ 次，迭代过程描述如下(应该注意位置更新时需要检查粒子是否越界，如果发生越界，则将粒子置于设定空间内一随机位置)：

(1) 计算粒子新位置

$$\boldsymbol{x}^{i,t+1}=\begin{cases}\mathrm{rand}\,(d)，粒子属于探索粒子集合H\\ \boldsymbol{x}^{i,t}+\boldsymbol{v}^{i,t+1}，粒子属于非探索粒子集合K\end{cases}\tag{1}$$

其中

$$\boldsymbol{v}^{i,t+1}=\boldsymbol{v}^{i,t}+c_1\boldsymbol{r}_1(\boldsymbol{p}^{i,t}-\boldsymbol{x}^{i,t})+c_2r_2(\boldsymbol{p}^{t}-\boldsymbol{x}^{i,t})\tag{2}$$

(2) 更新个体最优位置与个体最优值

$$\boldsymbol{p}^{i,t+1}=\begin{cases}\boldsymbol{x}^{i,t+1},\ \mathrm{pBest}^i<f(\boldsymbol{x}^{i,t+1})\\ \boldsymbol{p}^{i,t},\ \mathrm{pBest}^i\geqslant f(\boldsymbol{x}^{i,t+1})\end{cases}\tag{3}$$

$$\mathrm{pBest}^i=\begin{cases}f(\boldsymbol{x}^{i,t+1}),\ \mathrm{pBest}^i<f(\boldsymbol{x}^{i,t+1})\\ pBest^i,\ \mathrm{pBest}^i\geqslant f(\boldsymbol{x}^{i,t+1})\end{cases}\tag{4}$$

(3) 确定满意粒子，并进行局地细化寻优

选择当前适应度 $f(\boldsymbol{x}^{i,t+1})$最高的 $n\times q$(如果有小数则取整)个粒子加入集合 M^{t+1}；设置局地寻优范围向量参数 $\boldsymbol{g}^{i,t+1}=\boldsymbol{g}$，即 $\boldsymbol{g}_j^{i,t+1}=\boldsymbol{g}$；对 M^{t+1}中的任一粒子有

步骤 1　确定具有最优适应度的局地细化探测位置参数，有

$$[k_1^*,k_2^*,\cdots,k_d^*]=\underset{[k_1,k_2,\cdots,k_d]}{\arg}\max\left\{f(\boldsymbol{x}^{i,t+1}+[\frac{k_1}{s_1}g_1^{i,t+1},\frac{k_2}{s_2}g_2^{i,t+1},\cdots,\frac{k_d}{s_d}g_d^{i,t+1}])\right\}\tag{5}$$

式中，$k_j=0,\pm1,\pm2,\cdots,\pm s_j$；$j=1,2,\cdots,d$。

为了之后简化算法描述，令 $\boldsymbol{a}=[\frac{k_1^*}{s_1}g_1^{i,t+1},\frac{k_2^*}{s_2}g_2^{i,t+1},\cdots,\frac{k_d^*}{s_d}g_d^{i,t+1}]$。

步骤 2　判断粒子是否需要更新位置；如果需要，则更新粒子位置、个体最优适应度和个体最优位置。

当 $k_1^*=k_2^*=\cdots=k_d^*=0$ 时，表明最优适应度位置为粒子当前所在点，此时不需要更新位置；

当不满足 $k_1^*=k_2^*=\cdots=k_d^*=0$ 时，需要进行相应更新，即

$$\boldsymbol{x}^{i,t+1}=\boldsymbol{x}^{i,t+1}+\boldsymbol{a}\tag{6}$$

$$\boldsymbol{p}^{i,t+1}=\begin{cases}\boldsymbol{x}^{i,t+1},\ \mathrm{pBest}^i<f(\boldsymbol{x}^{i,t+1})\\ \boldsymbol{p}^{i,t+1},\ \mathrm{pBest}^i\geqslant f(\boldsymbol{x}^{i,t+1})\end{cases}\tag{7}$$

$$\mathrm{pBest}^i=\begin{cases}f(\boldsymbol{x}^{i,t+1}),\ \mathrm{pBest}^i<f(\boldsymbol{x}^{i,t+1})\\ \mathrm{pBest}^i,\ \mathrm{pBest}^i\geqslant f(\boldsymbol{x}^{i,t+1})\end{cases}\tag{8}$$

步骤 3　检查满意粒子细化寻优终止条件是否满足以适时结束局地细化寻优过程。

如果满足任意$\frac{g_j^{i,t+1}}{s_j}<e_j(j=1,2,\cdots,d)$，表明搜索已经足够细致，局地细化寻优结束，转至步骤 4。

步骤 4　依据更优点位置设置局地寻优范围，转至①继续进行局地寻优。

局地寻优范围设置方法为

①如果存在任一 $k_j^*=s_j(j=1,2,\cdots,d)$，则表明探测到

的更优适应度位置处于局地边缘，此时局地寻优范围参数 $\boldsymbol{g}^{i,t+1}$ 保持不变；

②如果不存在任一 $k_j^* = s_j(j=1,2,\cdots,d)$，则表明探测到的更优适应度位置不在局地边缘而在内部，此时需要缩小局地寻优范围参数 $\boldsymbol{g}^{i,t+1}$，对 $g_j^{i,t+1}(j=1,2,\cdots d)$ 重新赋值，即

$$g_j^{i,t+1} = \begin{cases} g_j^{i,t+1}, & \dfrac{g_j^{i,t+1}}{s_j} < e_j \\ \dfrac{g_j^{i,t+1}}{s_j}, & \dfrac{g_j^{i,t+1}}{s_j} \geqslant e_j \end{cases} \tag{9}$$

(4) 更新 gBest 及 $\boldsymbol{p}^t$

满意粒子是适应度最优的一批粒子，它们在局地寻优后，适应度只能更好而不会变差，故全局最优适应度必然出自某个满意粒子适应度。

故选择所有满意粒子，令 $u=\underset{i}{\arg\max}\{\text{pBest}^i\}$，则

$$\boldsymbol{p}^t = \begin{cases} \boldsymbol{p}^{u,t}, & \text{gBest} < \text{pBest}^u \\ \boldsymbol{p}^t, & \text{gBest} \geqslant \text{pBest} \end{cases} \tag{10}$$

$$\text{gBest} = \begin{cases} \text{pBest}^u, & \text{gBest} < \text{pBest}^u \\ \text{gBest}, & \text{gBest} \geqslant \text{pBest}^u \end{cases} \tag{11}$$

(5) 判断粒子本次迭代到达位置和上次迭代到达位置是否相同，如果相同，则粒子可能陷入局地极值，此时应该跳至寻优空间内一随机位置。这里“位置相同”被定义为两个位置坐标向量各维度距离均小于细化寻优阈值。即如果满足

$$| x_j^{i,t+1} - x_j^{i,t} | < e_j,\ j = 1,2\cdots,d$$

则

$$\boldsymbol{x}^{i,t+1} = \text{rand}\,(d) \tag{12}$$

(6) 如果迭代次数达到要求或者 gBest 已经满足要求，则算法结束，输出 gBest 和 $\boldsymbol{p}^t$。

2　改进算法与经典算法的性能比较

2.1　性能比较的内容与方法

PSO 算法的特点与优势在于收敛速度快、求解的结果精度很高，能够在短时间内找到问题的满意解。作为一种改进，本文提出的算法应该在收敛速度和结果精度上有所进步，所以对于性能主要考察两点：

(1) 新算法是否收敛更好，即能够快速的找到更为满意的解；

(2) 新算法是否精度更好，即经过一段可以接受、相对充分时间，新算法能否获得更满意的解。

参考其他研究，可以设置一个复杂函数，分别利用两种方法求解最值，进行比较以考察新算法性能。为了能够使结果更具有说服力，应该注意以下几点：

(1) 基本算法完全相同。新算法是在经典算法基础上添加满意粒子局地寻优过程和探索者粒子获得的，其本身包含经典算法，因此新算法中的经典算法部分和经典算法在描述、实施(程序编码)应该完全一致，确保其性能变化是由引入满意粒子局地寻优过程和探索者粒子带来的。

(2) 相同参数设置完全相同。新算法本身包含经典算法，其需要设置的参数包含经典算法所有参数，这些参数应该完全相同。需要指出的是，新算法和经典算法中地位相同的随机数应该采用相同算法、相同种子的伪随机数以确保结果可比。

2.2　性能比较分析实例

2.2.1　问题设定

本研究欲通过求解 f 函数在 $0\leqslant x\leqslant 100, 0\leqslant y\leqslant 100$ 内的最大值比较经典算法和改进算法的性能，其表达式为

$$f(x,y) = \left(\cos\left(\frac{x}{10}+2\right)\sin\left(\frac{y}{10}+1\right)+4\right)\cdot \left(\sin(x)\cos(y)+5\right) \tag{13}$$

f 函数在指定范围内含有众多极值点，其图像如图 3 所示。

图 3　函数 f 图像

2.2.2　性能比较结果

算法实现上，本研究基于多主体建模平台 Repast Simphony 编码实现经典 PSO 算法和改进 PSO 算法。Repast 是由芝加哥大学社会科学计算实验室开发研制的一款优秀的多主体建模工具，是目前多主体建模的主流工具之一，在复杂适应性系统仿真方面取得了许多成功的应用。算法的几个主要参数设置如下(由于改进算法基于经典算法，运算中改进算法与经典算法相同的参数取值相同)：粒子空间范围横纵坐标为 0～100，粒子数量设置为 5，初始位置随机指定，初始速度为 2，加速度系数为 2。改进算法独有的几个参数设置如下：探索因子比重为 0.3，满意粒子比重为 0.3，局地细化寻优探索范围半径为 4，网格细化份数为 3，寻优细化精度阈值为 10^{-8}。计算机仿真硬件方面，使用一台笔记本(电脑)，其 CPU 为 IntelT4200，内存 2 GB；计算机软件方面，操作系统为 Windows7，RePast Simphony 版本为 2.0。

(1) 从收敛时间考虑改进算法的性能提升情况。将随机数种子分别设置为 1,2,…,10 对问题进行 10 次求解，相同运算时间内经典算法和本文提出改变算法的求解优化结

果最小值、最大值、均值及最值相应坐标(解)如表 1 所示。可以看出，由于引入了局地细化寻优和探索性粒子，相同运算时间内改进算法获得的优化结果明显优于经典算法。10 次运算中，改进算法 5 s 内取得的最差优化结果(29.800 71)亦已经超过了经典算法 100 s 内获得的最好结果(29.798 96)。而平均来看，改进算法 0.5 s 内取得的最优结果均值(29.794 60)已经超过了经典算法 100 s 内获得的最优结果均值(29.791 67)。

表 1　相同运算时间内经典算法和改进算法 10 次求解优化结果最小值、最大值、均值及最值相应坐标(解)

算法	运算时间/s	优化结果最小值	优化结果最小值取得点坐标	优化结果最大值	优化结果最大值取得点坐标	优化结果均值
经典算法	0.5	28.778 45	[77.137 433 894 468 22,31.516 548 887 038 187]	29.699 52	[11.197 302 549 235 156,34.547 013 835 445 82]	29.470 32
改进算法		29.782 40	[11.046 041 619 227 317,34.660 156 677 500 44]	29.800 41	[73.837 963 793 428 32,97.429 339 449 633 91]	29.794 60
经典算法	1	28.916 90	[45.532 316 144 163 644,6.868 738 209 025 715]	29.699 52	[11.197 302 549 235 156,34.547 013 835 445 82]	29.543 54
改进算法		29.790 98	[73.832 351 489 155 54,97.481 983 974 543 41]	29.800 41	[73.837 963 793 428 32,97.429 339 449 633 91]	29.796 21
经典算法	5	29.581 67	[45.344 938 719 485 55,69.323 165 555 649 98]	29.766 19	[76.937 979 380 824 69,37.642 954 830 406 06]	29.720 35
改进算法		29.800 71	[11.001 441 378 571 434,97.421 997 619 651 88]	29.800 73	[73.832 297 212 074 69,97.419 661 468 354 69]	29.800 72
经典算法	10	29.676 73	[45.405 423 558 332 75,6.434 782 965 015 251]	29.772 18	[10.903 247 598 777 044,34.543 876 870 505 684]	29.744 73
改进算法		29.800 71	[11.001 441 378 571 434,97.421 997 619 651 88]	29.800 73	[42.416 387 840 575 915,3.171 875 988 766 295 5]	29.800 73
经典算法	50	29.757 31	[73.901 145 165 895 63,97.307 536 396 681 06]	29.794 33	[11.049 175 014 545 643,97.432 905 783 587 77]	29.778 72
改进算法		29.800 73	[73.832 661 005 502 63,97.419 674 752 722 61]	29.800 73	[73.832 297 212 074 69,97.419 661 468 354 69]	29.800 73
经典算法	100	29.766 31	[76.980 198 138 098 19,37.715 618 491 208 99]	29.798 96	[10.980 680 165 212 043,97.437 447 193 481 73]	29.791 67
改进算法		29.800 73	[73.832 297 212 074 69,97.419 661 468 354 69]	29.800 73	[11.000 433 511 141 686,97.419 655 555 176 39]	29.800 73

(2) 从收敛代数考虑改进算法性能提升情况。统计上述 10 次求解过程相同迭代次数经典算法和改进算法求解优化结果最小值、最大值、均值及最值相应坐标(解)如表 2 所示，最值、均值与迭代次数关系的折线图如图 4 所示。可以看出，由于引入了局地细化寻优和探索性粒子，相同迭代次数改进算法获得的优化结果明显优于经典算法。10 次运算中，改进算法 1 000 次迭代取得的最差优化结果(29.800 71)亦已经超过了经典算法 10 000 次迭代获得的最好结果(29.790 58)。而平均来看，改进算法 100 次迭代取得的最优结果均值(29.794 60)已经明显超过了经典算法 10 000 次迭代获得的最优结果均值(29.770 31)。

表 2　相同迭代次数经典算法和改进算法 10 次求解优化结果最小值、最大值、均值及最值相应坐标(解)

算法	迭代次数	优化结果最小值	优化结果最小值取得点坐标	优化结果最大值	优化结果最大值取得点坐标	优化结果均值
经典算法	10	26.577 08	[10.796 875 746 744 206,41.944 579 348 378 88]	29.460 89	[73.572 229 885 011 74,97.154 698 885 171 27]	28.255 18
改进算法		29.066 46	[102.019 857 843 881 3,69.567 106 425 298 93]	29.800 40	[73.843 497 945 736 25,97.422 280 933 156 03]	29.599 12
经典算法	100	28.657 19	[73.415 254 384 680 99,40.317 885 485 768 564]	29.699 52	[11.197 302 549 235 156,34.547 013 835 445 82]	29.341 18
改进算法		29.782 40	[11.046 041 619 227 317,34.660 156 677 500 44]	29.800 41	[73.837 963 793 428 32,97.429 339 449 633 91]	29.794 60
经典算法	1 000	29.501 83	[76.681 174 382 930 93,37.903 692 467 630 67]	29.764 97	[76.913 616 442 004 1,37.643 509 965 140 12]	29.700 12
改进算法		29.800 71	[11.001 441 378 571 434,97.421 997 619 651 88]	29.800 73	[42.416 387 840 575 915,3.171 875 988 766 295 5]	29.800 72
经典算法	10 000	29.757 31	[73.901 145 165 895 63,97.307 536 396 681 06]	29.790 58	[10.974 463 182 485 595,34.645 864 257 085 48]	29.770 31
改进算法		29.800 73	[73.832 707 968 760 72,97.419 656 567 690 57]	29.800 73	[11.000 433 511 141 686,97.419 655 555 176 39]	29.800 73

图 4　最值、均值与迭代次数的关系

3　结　论

本文通过对飞鸟寻食行为的细致化仿生，引入满意粒子局地细化寻优法和探索粒子可以改进 PSO 算法的性能，之后的实验证实了改进算法的有效性。本研究的不足之处在于没有对于算法进行有针对性的优化，比如二维空间细化寻优中更优点位于边界时，将有超过一半的探测点是重复探测；另外，没有采用更全面的指标性能与更多类型的优化方法进行全面比较研究。

下一步的研究方向为本改进算法的优化，确定其性能水平；将本方法与其他算法进行组合应用；将本文算法在不同的工程及管理优化领域进行推广应用等。

参考文献：

[1] Kennedy J, Eberhart R. Particle swarm optimization[C]// *Proc. of the IEEE International Conference on Neural Networks*, 1995: 1942 - 1948.

[2] Richards M, Ventura D. Choosing a starting configuration for particle swarm optimization[C]// *Proc. of the IEEE International Joint Conference on Neural Networks*, 2004: 2309 - 2312.

[3] 肖理庆，邵晓根，石天明，等. 利用改进粒子群算法整定 PID 参数[J]. 计算机应用，2010，30(6)：1547 - 1549.(Xiao L Q, Shao X G, Shi T M, et al. Tuning PID parameters with improved particle swarm optimization[J]. *Journal of Computer Applications*, 2010, 30(6): 1547 - 1549.)

[4] 金义雄，程浩忠，严健勇，等. 改进粒子群算法及其在输电网规划中的应用[J]. 中国电机工程学报，2005，25(4)：46 - 50，70. (Jin Y X, Cheng H Z, Yan J Y, et al. Improved particle swarm optimization method and its application in power transmission network planning[J]. *Proceedings of the CSEE*, 2005, 25(4): 46 - 50, 70.)

[5] Koshino M, Murata H, Kimura H. Improved particle swarm optimization and application to portfolio selection[J]. *Electronics and Communications in Japan* (*Part 3*), 2007, 90(3): 13 - 25.

[6] 左旭坤，左家圣. 基于改进粒子群的永磁同步电机速度控制器设计[J]. 组合机床与自动化加工技术，2010，52(10)：44 - 47. (Zuo X K, Zuo J S. Speed controller design in PMSM based on improved particle swam optimization algorithm[J]. *Modular Machine Tool & Automatic Manufacturing Technique*, 2010, 52(10): 44 - 47.)

[7] 罗豪，雷友诚. 基于改进粒子群算法的 PID 控制器参数优化[J]. 计算机仿真，2009，26(9)：156 - 159. (Luo H, Lei Y C. Optimization of PID controller paramerters based on improved particle swarm algorithms[J]. *Computer Simulation*, 2009, 26(9): 156 - 159.)

[8] 陈家照，罗寅生. 改进粒子群三维空间路径规划研究[J]. 计算机工程与应用，2010，46(33)：39 - 42. (Chen J Z, Luo Y S. Research on improved particle swarm optimization for path planning in 3-D space[J]. *Computer Engineering and Applications*, 2010, 46(33): 39 - 42.)

[9] Ratnaweera A, Halgamuge S K, Watson H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients [J]. *IEEE Trans. on Evolutionary Computation*, 2004, 8(3): 240 - 255.

[10] 杨雪榕，梁加红，陈凌，等. 多邻域改进粒子群算法[J]. 系统工程与电子技术，2010，32(11)：2453 - 2458. (Yang X R, Liang J H, Chen L, et al. Multi-neighborhood improved particle swarm optimization algorithm[J]. *Systems Engineering and Electronics*, 2010, 32(11): 2453 - 2458.)

[11] 李淼. 基于函数变换的改进混沌粒子群优化[J]. 计算机应用研究，2010，27(11)：4105 - 4107，4113. (Li Y. Improved chaotic particle swarm optimization based on function transform [J]. *Application Research of Computers*, 2010, 27(11): 4105 - 4107, 4113.)

[12] Eberhart R C, Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization[C]// *Proc. of the Congress on Evolutionary Computation*, 2010: 84 - 88.

[13] Mikki S, Kishk A. Improved particle swarm optimization technique using hard boundary conditions[J]. *Microwave and Optical Technology Letters*, 2005, 46(5): 422 - 426.

[14] 杨智，陈志堂，范正平，等. 基于改进粒子群优化算法的 PID 控制器整定[J]. 控制理论与应用，2010，27(10)：1345 - 1352. (Yang Z, Chen Z T, Fan Z P, et al. A tuning of PID controller based on improved particle-swarm-optimization[J]. *Control Theory & Applications*, 2010, 27(10): 1345 - 1352.)

[15] Ali M M, Kaelo P. Improved particle swarm algorithms for global optimization[J]. *Applied Mathematics and Computation*, 2008, 196(2): 578 - 593.

[16] 李娟，饶妮妮，廖瑞华，等. 基于改进粒子群算法的 ad hoc 网络移动模型研究[J]. 电子学报，2010，38(1)：222 - 227. (Li J, Rao N N, Liao R H, et al. Mobility model based on the improved-PSO algorithm for ad hoc network[J]. *Acta Electronica Sinica*, 2010, 38(1): 222 - 227.)

[17] 张建科，刘三阳，张晓清. 改进的粒子群算法[J]. 计算机工程与设计，2007，28(17)：4215 - 4219.（Zhang J K，Liu S Y，Zhang X Q. Improved particle swarm optimization[J]. *Computer Engineering and Design*，2007，28(17)：4215 - 4219.）

[18] 曾建潮，崔志华. 一种保证全局收敛的 PSO 算法[J]. 计算机研究与发展，2004，41(8)：1333 - 1338.（Zeng J C，Cui Z H. A guaranteed global convergence particle swarm optimizer[J]. *Journal of Computer Research and Development*，2004，41(8)：1333 - 1338.）

[19] 姚金杰，韩焱. 基于改进自适应粒子群算法的目标定位方法[J]. 计算机科学，2010，37(10)：190 - 192.（Yao J J，Han Y. Research on target localization based on improved adaptive velocity particle swarm optimization algorithm[J]. *Computer Science*，2010，37(10)：190 - 192.）

[20] 王红玲，郑纲，何剑锋. 基于改进粒子群算法的生鲜农产品配送路径优化研究[J]. 安徽农业科学，2010，38(31)：17961 - 17962，17985.（Wang H L，Zheng G，He J F. Study on optimization of distribution routing of fresh agricultural product based on improved particle swarm optimization algorithm[J]，*Journal of Anhui Agricultural Sciences*，2010，38(31)：17961 - 17962，17985.）

[21] Wei J，Yun C X，Ji X Q. An improved particle swarm optimization algorithm with disturbance[C]// *Proc. of the IEEE International Conference on Systems，Man and Cybernetics*，2004：5900 - 5904.

[22] 王维刚，倪红梅. 改进粒子群算法在管壳式换热器优化设计中的应用[J]. 动力工程学报，2010，30(12)：947 - 951.（Wang W G，Ni H M. Application of IPSO algorithm to optimal design of shell-and-tube heat exchangers[J]. *Journal of Chinese Society of Power Engineering*，2010，30(12)：947 - 951.）

[23] 胡勇. 用随机模式和调整机制改进粒子群优化算法[J]. 重庆邮电大学学报(自然科学版)，2010，22(1)：99 - 102.（Hu Y. Using random pattern and regulation mechanism to improve PSO algorithm[J]. *Journal of Chongqing University of Posts and Telecommunications* (*Natural Science Edition*)，2010，22(1)：99 - 102.）

[24] 黄纯，罗伟原，江辉. 基于增量式 PID 的改进粒子群算法[J]. 湖南大学学报(自然科学版)，2009，36(12)：35 - 39.（Huang C，Luo W Y，Jiang H. Modified particle swarm optimization based on increment PID[J]. *Journal of Hunan University* (*Natural Science*)，2009，36(12)：35 - 39.）

[25] 付国江，王少梅，刘舒燕，等. 改进的速度变异粒子群算法[J]. 计算机工程与应用，2006，43(13)：48 - 50，105.（Fu G J，Wang S M，Liu S Y，et al. An improved velocity mutation particle swarm optimizer[J]. *Computer Engineering and Applications*，2006，43(13)：48 - 50，105.）

[26] 李陶深，陈松乔，杨明，等. 改进的粒子群优化算法在 qos 选播路由中的应用[J]. 小型微型计算机系统，2010，31(1)：68 - 71.（Li T S，Chen S Q，Yang M，et al. Application of an improved particle swarm optimization algorithm in qos anycast routing[J]. *Journal of Chinese Computer Systems*，2010，31(1)：68 - 71.）

[27] 孙坚，丁永生，郝矿荣. 基于改进粒子群优化算法的并联机构位置正解法[J]. 制造业自动化，2010，32(4)：92 - 95.（Sun J，Ding Y S，Hao K R. An improved particle swarm optimization algorithm based positioning method for parallel manipulator[J]. *Manufacturing Automation*，2010，32(4)：92 - 95.）

[28] Blackwell T，Branke J. Multi-swarm optimization in dynamic environments[J]. *Applications of Evolutionary Computing*，2004，3005：489 - 500. doi:10.1007/978 - 3 - 540 - 24653 - 4_50.

[29] Jiang Y，Hu T，Huang C，et al. An improved particle swarm optimization algorithm[J]. *Applied Mathematics and Computation*，2007，193(1)：231 - 239.

[30] Jiang Y，Liu C，Huang C，et al. Improved particle swarm algorithm for hydrological parameter optimization[J]. *Applied Mathematics and Computation*，2010，217(7)：3207 - 3215.

[31] Zhao B，Guo C X，Bai B R，et al. An improved particle swarm optimization algorithm for unit commitment[J]. *International Journal of Electrical Power & Energy Systems*，2006，28(7)：482 - 490.

[32] 赵波，曹一家. 电力系统机组组合问题的改进粒子群优化算法[J]. 电网技术，2004，28(21)：6 - 10.（Zhao B，Cao Y J. An improved particle swarm optimization algorithm for power system unit commitment[J]. *Power System Technology*，2004，28(21)：6 - 10.）

[33] 王涛，俞承芳. 一种改进的粒子群算法在 PCD 板元件检测中的应用[J]. 微电子学与计算机，2007，24(12)：213 - 216.（Wang T，Yu C F. A modified particle swarm optimization algorithm and its application in detecting printed circuit board components[J]. *Microelectronics & Computer*，2007，24(12)：213 - 216.）

[34] Higashi N，Iba H. Particle swarm optimization with Gaussian mutation[C]// *Proc. of the Swarm Intelligence Symposium*，2003：72 - 79.

[35] Zhang C，Sun J，Zhu X，et al. An improved particle swarm optimization algorithm for flowshop scheduling problem[J]. *Information Processing Letters*，2008，108(4)：204 - 209.

[36] Liu B，Wang L，Jin Y H，et al. Improved particle swarm optimization combined with chaos[J]. *Chaos，Solitons & Fractals*，2005，25(5)：1261 - 1271.

[37] Xiang T，Liao X，Wong K W. An improved particle swarm optimization algorithm combined with piecewise linear chaotic map[J]. *Applied Mathematics and Computation*，2007，190(2)：1637 - 1645.

[38] Shelokar P S，Siarry P，Jayaraman V K，et al. Particle swarm

and ant colony algorithms hybridized for improved continuous optimization[J]. *Applied Mathematics and Computation*, 2007, 188(1): 129-142.

[39] 贺毅朝，刘坤起. 求解 SAT 问题的改进粒子群优化算法[J]. 计算机工程与设计，2006，27(15)：2731-2733,2758.(He Y C, Liu K Q. Improved particle swarm optimizers for solving SAT problem[J]. *Computer Engineering and Design*, 2006, 27(15): 2731-2733, 2758.)

[40] Koh B I, George A D, Haftka R T, et al. Parallel asynchronous particle swarm optimization[J]. *International Journal for Numerical Methods in Engineering*, 2006, 67(4): 578-595.

[41] Laskari E C, Parsopoulos K E, Vrahatis M N. Particle swarm optimization for integer programming[C]// *Proc. of the Congress on Evolutionary Computation*, 2002: 1582-1587.

[42] Kennedy J, Eberhart R C. A discrete binary version of the particle swarm algorithm[C]// *Proc. of the IEEE International Conference on Systems, Man, and Cybernetics*, 1997: 4104-4109.

[43] Pang W, Wang K P, Zhou C G, et al. Fuzzy discrete particle swarm optimization for solving traveling salesman problem[C]// *Proc. of the 4th International Conference on Computer and Information Technology*, 2004: 796-800.

[44] Jang H S, Chang H I, Sang Y K, et al. An improved particle swarm optimization algorithm mimicking territorial dispute between groups for multimodal function optimization problems[J]. *IEEE Trans. on Magnetics*, 2008, 44(6): 1046-1049.

[45] Noel M M, Jannett T C. Simulation of a new hybrid particle swarm optimization algorithm[C]// *Proc. of the 36th Southeastern Symposium on System Theory*, 2004: 150-153.

[46] Chen S, Mei T, Luo M, et al. Identification of nonlinear system based on a new hybrid gradient-based PSO algorithm[C]// *Proc. of the International Conference on Information Acquisition*, 2007: 265-268.

[47] Ghaffari M M, Farmahini F A, Faraji D R, et al. An efficient hybrid swarm intelligence-gradient optimization method for complex time Green's functions of multilayer media[J]. *Progress in Electromagnetics Research*, 2007, 77: 181-192. doi: 10.2528/PIER07072504

[48] 王俊伟，汪定伟. 一种带有梯度加速的粒子群算法[J]. 控制与决策，2004，19(11)：1298-1300，1304.(Wang J W, Wang D W. Particle swarm optimization algorithm with gradient acceleration[J]. *Control and Decision*, 2004, 19(11): 1298-1300, 1304.)