

基于扩散机制的双种群粒子群优化算法*

徐星^{1,2}, 李元香², 吴昱²

(1. 景德镇陶瓷学院信息工程学院, 江西景德镇 333300; 2. 武汉大学软件工程国家重点实验室, 武汉 430072)

摘要: 为了避免标准粒子群优化算法(PSO)过早收敛的缺点,把热力学中的扩散现象引入到 PSO 算法的改进当中,提出了基于扩散机制的双种群粒子群优化算法(DPSO)。DPSO 算法中定义了粒子的扩散能、种群的温度和粒子的扩散概率三个概念,两个群体中的粒子在进化过程中根据粒子的扩散概率被选入到各自种群的扩散池中,从而实现两个种群之间信息的交换和共享。通过解决典型的多峰、高维函数优化问题来证实 DPSO 算法的有效性,实验结果表明 DPSO 比标准 PSO 具有更高的性能。

关键词: 粒子群优化算法; 扩散机制; 多种群; 热力学

中图分类号: TP301.6 **文献标志码:** A **文章编号:** 1001-3695(2010)08-2882-04

doi:10.3969/j.issn.1001-3695.2010.08.018

Bi-swarm particle swarm optimization based on diffusion mechanism

XU Xing^{1,2}, LI Yuan-xiang², WU Yu²

(1. College of Information & Engineering, Jingdezhen Ceramic Institute, Jingdezhen Jiangxi 333300, China; 2. State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, China)

Abstract: In order to avoid premature convergence, this paper applied the diffusion phenomenon of thermodynamics to improve standard particle swarm optimization (PSO), and proposed a modified PSO algorithm, the bi-swarm particle swarm optimization based on diffusion mechanism (DPSO). In DPSO algorithm, defined the diffusion energy of the particle, the temperature of the swarm and the diffusion probability of the particle. During the evolution of DPSO, chose the particle of each swarm into the diffusion pool of each swarm. The diffusion pool of both swarms exchanged and shared information. Experimental results on multi-modal, high-dimensional numerical optimization problems show that DPSO outperforms standard PSO.

Key words: particle swarm optimization(PSO); diffusion mechanism; multi-swarm; thermodynamics

0 引言

粒子群优化(PSO)算法是由 Kennedy 等人^[1]于 1995 年提出的一种基于群智能的全局优化进化算法,其直观背景来源于动物行为学和社会心理学,是对简单生物社会系统的模拟。粒子群优化算法流程简单且易编程实现,它既有传统演化算法深刻的背景,同时又有自己独特的优化性能,因此一经提出,立刻引起国内外自然计算研究者的关注并获得迅速发展,在函数优化、神经网络训练、模式分类、模糊系统控制以及其他诸多工程领域都得到了广泛的应用。粒子群算法和演化算法、模拟退火算法以及蚁群算法等其他元启发式优化算法一样,都存在着早熟收敛的问题,极大地影响了算法的稳定性。鉴于这些元启发式算法本身都是计算机学科和其他学科交叉研究的成果,所以为了避免算法的过早收敛,提高算法的性能,一些研究者通过将粒子群算法与物理、数学和生物等其他学科进行交叉研究,或者与其他计算智能方法相融合,以突破其自身局限,提高算法总体性能^[2]。

为了避免算法过早地陷入局部最优解,本文提出了一种新颖的基于扩散现象的粒子群优化算法(DPSO)。实验结果表明,DPSO 算法具有良好的性能,能够有效地解决一些典型的、

复杂的全局优化问题。

1 标准粒子群优化算法

在标准 PSO 算法中,每个优化问题的解被看成是搜索空间的一个粒子,算法初始化为一群随机粒子(随机解),然后通过迭代找到最优解。在每一次的迭代中,粒子通过跟踪两个极值来更新自己。在找到这两个最优值后,粒子通过下式来更新自己的速度和位置:

$$v_{id}^{t+1} = \omega v_{id}^t + R_1 c_1 (P_{id} - x_{id}^t) + R_2 c_2 (p_{gd} - x_{id}^t) \quad (1)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (2)$$

其中: $i=1,2,\dots,M$, M 是该群体中粒子的总数; v_{id}^t 为粒子 i 第 t 次迭代后的速度矢量的第 d 维分量; x_{id}^t 为粒子 i 第 t 次迭代后的位置矢量的第 d 维分量; P_{id} 为粒子 i 最好位置的第 d 维分量; p_{gd} 为种群当前最好位置的第 d 维分量; R_1 和 R_2 是服从 $U(0,1)$ 分布的随机数; c_1 和 c_2 是学习因子,通常取 $c_1 = c_2 = 2$; ω 是惯性因子。

在每一维,粒子都有一个最大限制速度 V_{max} ,如果某一维的速度超过设定的 V_{max} ,那么这一维的速度就被限定为 V_{max} 。从社会学的角度来看,式(1)的第一部分称为记忆项,表示过去的速度对现在的影响;第二部分称为自身认知项,是从当前

收稿日期: 2010-01-17; 修回日期: 2010-03-08 基金项目: 国家自然科学基金资助项目(60773009); 国家“863”计划资助项目(2007AA01Z290)

作者简介: 徐星(1984-),男,湖北武汉人,博士,主要研究方向为智能计算、模式识别(whuxx84@yahoo.com.cn); 李元香(1962-),男,教授,博导,主要研究方向为演化计算、并行计算; 吴昱(1982-),女,博士研究生,主要研究方向为智能计算、金融计算。

点指向粒子自身最好点的矢量,表示粒子的动作来源于自己经验的部分;第三部分称为群体认知项,是从当前点指向种群最好点的矢量,反映了粒子间的协同合作和知识共享。粒子就是通过自己的经验和同伴中最好的经验来决定下一步的运动。

2 扩散运动

2.1 扩散现象

当在房间的某处打开一瓶香水,慢慢在其他地方可以闻到香味;将装有两种不同气体的两个容器连通,经过一段时间,两种气体就在这两个容器中混合均匀;在厨房里做饭炒菜,屋外也能闻到饭菜的香味。这些现象在物理学里面被称之为扩散现象^[3]。扩散现象不仅在气体里有,在液体和固体中也存在,只不过由于液体分子与固体分子之间的作用力较大,其扩散速度要小于气体分子的扩散速度。例如在清水中滴入一滴黑墨水,在静止的状态下可以看到,原本清亮的水逐渐变成墨黑;做汤的时候,滴进几滴酱油,即使不搅拌,整个汤里也会逐渐均匀地染上酱油的色泽。这种气味和颜色的均匀化过程不是由于物质的搅动或对流造成的,而是由于物质微粒(分子、原子或离子)的扩散造成的。扩散会造成物质的迁移,会使浓度均匀化,而且温度越高,扩散进行得越快。

在固体中也存在扩散,并且它是固体中物质传输的惟一方式,因为固体不能像气体或液体那样通过流动来进行物质传输。固体分子间的作用力很大,绝大多数分子只能在各自的平衡位置附近振动,这是固体分子热运动的基本形式。但是在一定温度下,固体中也总有一些分子的速度较大,具有足够的能量脱离平衡位置。这些分子不仅能从一处移到另一处,而且有的还能进入相邻物体,这就是固体发生扩散的原因。固体的扩散在金属的表面处理和半导体材料生产上很有用处,如钢件的表面渗碳法(提高钢件的硬度)、渗铝法(提高钢件的耐热性)都利用了扩散现象;在半导体工艺中利用扩散法渗入微量的杂质,以达到控制半导体性能的目的^[4]。

2.2 扩散定律

扩散定律是描述分子扩散导致的传质过程的定律,这一定律由德国物理学者菲克(A. Fick)于 1855 年在实验的基础上提出的。扩散定律是由 A. Fick 提出的,故又称为菲克(Fick)定律,它包括 Fick 第一定律和 Fick 第二定律。第一定律用于稳态扩散,即扩散过程中各处的浓度及浓度梯度不随时间变化;第二定律用于非稳态扩散,即扩散过程中各处的浓度和浓度梯度随时间发生变化。

1) Fick 第一定律

Fick 第一定律是 A. Fick 通过实验导出的。Fick 第一定律指出,在稳态扩散过程中,扩散流量 J 与浓度梯度成正比:

$$J = -D \frac{dc}{dx} \quad (3)$$

其中: D 为扩散系数,是描述扩散速度的重要物理量,它表示单位浓度梯度条件下,单位时间单位截面上通过的物质流量;式中的负号表示物质沿着浓度降低的方向扩散。式(3)的意思是,如果某个事物的空间分布是不均匀的,就会造成流动(再分配),而引起的物质流正比于该事物的梯度。梯度是物质流动的驱动力。

2) Fick 第二定律

Fick 第一定律仅适用于稳态扩散,但实际上稳态扩散的情

况是很少的,大部分属于非稳态扩散。这就要应用 Fick 第二定律。Fick 第二定律是由第一定律推导出来的。在非稳态扩散过程中,若 D 与浓度无关,则 Fick 第二定律的表达式为

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} \quad (4)$$

其中: $c=f(t,x)$, $\frac{dc}{dt} \neq 0$, t 为扩散时间。扩散第二方程反映的是扩散物质浓度与时间和空间位置之间的定量关系,且这个方程不能直接应用,必须结合具体的初始条件和边界条件才能求出积分解。

2.3 扩散系数

扩散系数是扩散定律非常重要的物理量,它与当前物体的温度、物质的成分和结构、浓度等具有一定的关系。式(5)是由瑞典化学家 Arrhenius 根据实验结果提出的化学反应的速率常数与温度之间的关系式,称为 Arrhenius 公式^[5]或 Arrhenius 速率定律。在一定条件下,扩散系数也可用式(5)表示:

$$D = D_0 e^{-\frac{Q}{RT}} \quad (5)$$

其中: D_0 为扩散常数; Q 为扩散激活能(粒子克服势垒从原来的位置扩散到相邻位置上所需的能量称为激活能); R 为气体常数; T 为热力学温度。

3 融合扩散机制的双种群粒子群优化算法(DPSO)

3.1 相关工作

扩散现象是自然界中普遍存在的现象,在人类社会中也存在类似的迁移迁徙现象。例如,游牧民族为了克服恶劣的自然环境和复杂的社会环境中的不利因素,在部落首领的带领下进行有目的的群体迁移;在自然界的生物群中,当生物的物种密度过大时,该物种有自动分散并寻找新的生存栖息地以繁衍子代的习性。王铁奇等人^[6]模拟游牧民族的迁徙机制,设计了一种迁移策略,并把这种策略与粒子群优化算法集成起来,提出了迁移粒子群优化算法集成(migration particle swarm optimization ensemble, EMPSO)。该算法中采用了多种群的思想,各个种群进行独立的并行搜索,经过一定的迭代次数之后,从所有种群中选出最好的 P_g ;然后替换最后一个种群的最优个体,接着从 i 个种群随机选择部分个体来替换第 $i-1$ 个种群中的部分个体,以实现种群之间的信息交流;最后把 EMPSO 算法用于计算生物学中的生物序列模式识别,并取得了较好的实验结果。赫然等人^[7]认为粒子的速度与粒子群种群的密度具有一定的关系,因此借鉴生物根据生存密度自动分家的特性,提出了一种自适应逃逸粒子群优化算法(particle swarm optimization based on self-adaptive escape velocity)。该算法在单模态和多模态函数优化问题上都取得了理想的实验结果。马海平等^[8]根据自然界中的生物迁徙现象,提出了一种物种迁移优化算法(species migration optimization)。该算法借鉴了其他演化计算的思想,通过物种迁移来实现信息的共享和交换,从而完成演化过程。

以上研究人员在模拟自然科学和社会科学中的迁移或扩散机制时,或多或少地使用了多种群的思想,正是由于多种群能有效地提高算法的性能。早在 20 世纪 80、90 年代,演化计算领域的先驱们在设计并行演化算法时提出了迁移模型。迁移模型是一种群体级的并行模型,其中两种比较典型的迁移方

式是岛屿模型和脚踏石模型。在设计和改进粒子群优化算法时,很多研究人员也把多种群思想引入到 PSO 算法当中来。Lovbjerg 等人^[9]在 PSO 算法中最早引入了子种群的概念。Bergh 等人^[10]提出了协同(cooperative)PSO 算法,其基本思想是用 K 个相互独立的粒子群分别在 D 维目标搜索空间中的不同维度方向上进行搜索,即使用多种群分别来优化决策向量的不同片段。王元元等人^[11]也提出了一种多种群协同进化的 PSO 算法,其基本思想是各种群独立进化,周期性地共享交换信息。其中采用了两种共享交换策略,这两种策略类似于岛屿模型和领域模型。陶新民等人^[12]借鉴阴性免疫算法和文化算法的思想设计了一种改进的多种群协同进化的粒子群优化算法,利用免疫算法的思想来实现解空间的均匀划分,借鉴文化算法的思想来对超级个体的优胜区域进行更新。黄芳等人^[13]受并行演化算法的启示,提出了基于岛屿模型的多种群并行粒子群优化算法。刘卓倩等人^[14]提出了一种三群协同粒子群优化算法(TSC-PSO),三个种群中其中两个为优群,另外一个为差群,TSC-PSO 算法在执行时若全局极值更新停滞,则任选其中一优群与差群的粒子进行交换,但个体极值不交换。夏桂梅等人^[15]提出了双群体随机粒子群优化算法,当其中一个种群中出现停止粒子时,通过另外一个种群随机产生一个粒子来替换此粒子;如果另外一个种群出现停止粒子时,亦通过类似的方法替换停止粒子。

3.2 DPSO 算法的思想和流程

受扩散现象和多种群思想的启发,本文采用双种群来模拟扩散机制,提出了一种基于扩散机制的双种群粒子群优化算法(double particle swarm optimization based on diffusion, DPSO),类比于扩散理论中的扩散激活能和温度的概念,给出了 DPSO 算法中粒子的扩散能、种群温度和粒子的扩散概率三个定义,接着在这三个定义的基础上详细地介绍了 DPSO 算法的核心思想和算法流程。

定义 1 粒子的扩散能。物理学中物体由于运动而具有的能叫动能,它的大小是运动物体的质量和速度平方乘积的二分之一。类比于物理学中的动能,粒子克服势垒从原来的位置扩散到其他位置上所具有的能量称为扩散能,本文忽略粒子的质量,扩散能的大小定义为粒子速度矢量的每一维分量平方和二分之一。

$$Q_i = \frac{1}{2} \sum_{j=1}^{\text{Dim}} V_{ij}^2 \quad (6)$$

其中:Dim 是粒子在搜索空间的维数; i 是种群中粒子的下标; V_{ij} 是粒子 i 速度矢量的第 j 维分量。

定义 2 种群温度。在物理学中,温度是用来表示物体冷热程度的物理量,微观上来讲是物体分子热运动的剧烈程度。从分子运动论观点看,温度是物体分子平均动能的一种表现形式,分子运动愈快,物体愈热,则温度愈高;分子运动愈慢,物体愈冷,则温度愈低。温度是大量分子热运动的集体表现,含有统计意义。对于个别分子来说,温度是没有意义的。类比于物理学中的温度,定义了粒子群的种群温度。

$$T = \frac{\sum_{i=1}^M Q_i}{M} \quad (7)$$

其中: M 为种群的粒子数量,即种群大小。从式(7)可以看出,种群的温度是种群粒子扩散能的算术平均。

定义 3 粒子的扩散概率。其定义为 1 减去扩散系数与

扩散常数的比值。

$$P_i = 1 - \frac{D}{D_0} = 1 - \frac{D_0 e^{-\frac{Q_i}{T}}}{D_0} = 1 - e^{-\frac{Q_i}{T}} \quad (8)$$

其中: Q_i 是粒子 i 的扩散能; T 是种群的温度;气体常数 R 设置为 1。粒子的扩散能和当前种群的温度共同决定了粒子扩散的概率。当粒子的扩散能大于种群温度时,粒子将具有较大的扩散概率;反之,当粒子的扩散能小于种群温度时,粒子将具有较小的扩散概率。

DPSO 算法中采用了双种群,分别用种群 A 和 B 表示,并且对两个种群的操作完全一致。在算法的每一迭代中,根据种群 $A(B)$ 中每个粒子的速度矢量计算得到每个粒子的扩散能,然后根据所有粒子的扩散能计算当前迭代的种群 $A(B)$ 温度,再根据式(8)计算得到每个粒子的扩散概率值,生成服从均匀分布的随机数。如果此随机数小于粒子的扩散概率值,则将此粒子放入到种群 $A(B)$ 的扩散池当中,在扩散池中随机选择两个粒子,生成一个差异向量对种群 $A(B)$ 中的全局极值进行一个扰动。如果扰动后的向量优于另一种群 $B(A)$ 中的全局极值,则替换其全局极值。基于扩散机制的双种群粒子群优化算法(DPSO)的具体流程如下:

- a) 分别初始化种群 A 和 B 中的所有粒子(种群规模分别为 M),包括位置、速度;
- b) 评价种群 A 和 B 中每个粒子的适应值;
- c) 分别更新种群 A 和 B 中粒子的个体历史极值,以及种群 A 和 B 的全局极值,分别用 p_g^A 和 p_g^B 表示;
- d) 根据式(6)计算种群 A 和 B 中所有粒子的扩散能;
- e) 根据式(7)计算种群 A 和 B 的温度;
- f) 根据式(8)计算种群 A 和 B 中所有粒子的扩散概率;
- g) 对于种群 A
 - for($i=0; i < M; i++$)
 - {
 - if($\text{rand}() < P_i$)
 - 将种群 A 中的粒子 i 放入到扩散池 A 中
 - }
- h) 对于种群 B (步骤 g) 和 h) 可同时进行
 - for($j=0; j < M; j++$)
 - {
 - if($\text{rand}() < P_j$)
 - 将种群 B 中的粒子 j 放入到扩散池 B 中
 - }
- i) 如果扩散池 A 中的粒子数目大于等于 2,则从中随机选择两个粒子 m 和 n 作为扩散粒子,粒子 m 和 n 生成一个差异向量 $\eta(x_m - x_n)$ 。对全局极值 p_g^A 添加一个随机扰动,得到一个临时的向量 $p_g^A + \eta(x_m - x_n)$ 。如果此向量优于种群 B 的全局极值 p_g^B ,则替换之,否则不替换;同时,如果扩散池 B 中的粒子数目大于等于 2,则从中随机选择两个粒子 a 和 b 作为扩散粒子,粒子 a 和 b 生成一个差异向量 $\eta(x_a - x_b)$ 。对全局极值 p_g^B 添加一个随机扰动,得到一个临时的向量 $p_g^B + \eta(x_a - x_b)$ 。如果此向量优于种群 A 的全局极值 p_g^A ,则替换之,否则不替换。通过此步骤来实现两个种群之间信息的交流和扩散。
- j) 输出种群 A 和 B 的全局极值 p_g^A 和 p_g^B 中的最优者;
- k) 根据式(1)(2)调整种群 A 和 B 中粒子的速度及位置;

1)未满足算法终止条件则转 b),否则 DPSO 算法结束。

4 实验仿真与结果分析

在本文的实验中,使用了五个经典的测试函数,这些函数经常被国内外很多学者用来测试优化算法的性能和可靠性。本文提出的 DPSO 算法和标准粒子群算法同时求解优化问题,以对比考察各种算法的性能。

测试函数如下:

$$F_1(x) = \sum_i x_i^2, -100 \leq x_i \leq 100$$

$$F_2(x) = \frac{1}{4000} \sum_i x_i^2 - \prod_i \cos(\frac{x_i}{\sqrt{i}}) + 1, -100 \leq x_i \leq 100$$

$$F_3(x) = \sum_i (x_i^2 - 10 \cos(2\pi x_i) + 10), -100 \leq x_i \leq 100$$

$$F_4(x) = -20 \exp(-0.2 \sqrt{\frac{1}{N} \sum_i x_i^2}) - \exp(\frac{1}{N} \sum_i \cos(2\pi x_i)) + 20 + e, -32 \leq x_i \leq 32$$

$$F_5(x) = \frac{(\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5)}{[1 + 0.001(x_1^2 + x_2^2)]^2} - 0.5, -100 \leq x_i \leq 100$$

其中: F_1 为单峰函数; F_2 、 F_3 和 F_4 为多峰函数,存在多个局部最优解,并且其局部最优解的个数随着维数的增加呈指数增加,这四个函数均在(0,0,...,0)处取得全局最优解; F_5 在(0,0)点处取得全局极小值-1,而在距该点约3.14 范围内隆起的无穷多个次全局极小点,次全局极小值约为-0.990 284。由于 F_5 的强烈振荡性质以及它的全局最优解被次最优解所包围的特性,使得一般算法难以找到它的全局最优解。

标准 PSO 算法中,种群大小均设置为 40,惯性权重 ω 从 1.0 线性递减到 0,速度最大值为搜索区间的一半。DPSO 中两个种群大小分别为 20,惯性权重 ω 和速度最大值 PSO 算法取值完全一致。对于 F_1 、 F_2 、 F_3 和 F_4 四个函数,维数分别为 10、20 和 30 时,最大迭代次数分别设置为 1 000、2 000 和 3 000。对于测试函数 5,最大迭代次数设置为 1 000。DPSO 和 PSO 两种算法在所有的测试例上都运行 50 次,所有实验均在一台 Pentium4 2.0 GHz CPU/1 GB 内存的计算机上进行。

表 1 中列出了在五个测试函数上 DPSO 和 PSO 算法的性能统计信息,包括成功收敛的比率(算法获得的最优解与真实最优解的值小于 $1e-10$ 即认为收敛)、50 次中最差的、平均的以及最好的结果和标准差。图 1~4 给出了在 F_1 、 F_2 、 F_3 和 F_4 四个函数维数为 30 时 DPSO 和 PSO 算法的最好适应值变化曲线。

从表 1 和图 1~4 中所展现的实验结果,可作出如下分析和判断:从表 1 可以看出,对于五个测试函数 DPSO 方法得到的最优结果的平均值和标准差以及收敛比率基本上要优于 PSO 算法所获得的结果,特别是随着维数从 10 维增加到 20 维、30 维时,DPSO 的性能明显提高,在函数 F_1 和 F_4 上表现尤为明显;在函数 F_2 和 F_3 上,随着维数的增加,DPSO 的性能提高虽不是特别显著,但是仍然好于 PSO 算法;在函数 F_5 上,相比于 PSO 算法,DPSO 算法具有较好的稳定性和鲁棒性,在 50 次运行中均收敛到全局最优解。图 1~4 显示在算法执行的中前期,PSO 和 DPSO 算法的收敛曲线基本重合,说明两者的收敛速度很接近;然而随着算法逐渐执行到中后期,PSO 算法基本上陷入停滞,而 DPSO 还具有一定的进化能力,说明 DPSO 比 PSO 算法有更大的机会跳出局部最优解,这正是由于 DPSO 引入的扩散机制起到的关键作用。

表 1 两种算法的实验统计结果

函数	维数	算法	最优解	最差解	均值	标准差	收敛比率
F_1	10	PSO	9.07e-53	3.56e-37	7.16e-39	5.04e-38	50/50
	10	DPSO	2.08e-44	3.85e-39	5.06e-40	1.12e-39	50/50
F_1	20	PSO	5.62e-35	1.53e-20	9.81e-22	3.78e-21	50/50
	20	DPSO	3.2e-37	1.18e-31	5.52e-33	1.98e-32	50/50
F_1	30	PSO	5.48e-26	3.18e-17	6.36e-18	1.42e-17	50/50
	30	DPSO	5.47e-31	1.45e-27	5.03e-28	1.47e-27	50/50
F_2	10	PSO	3.69e-02	1.62e-01	8.06e-02	3.52e-02	0/50
	10	DPSO	0	1.40e-01	6.92e-02	3.43e-02	2/50
F_2	20	PSO	0	8.57e-02	2.62e-02	2.12e-02	4/50
	20	DPSO	0	3.44e-02	1.51e-02	1.62e-02	12/50
F_2	30	PSO	0	5.16e-02	1.23e-02	1.46e-02	10/50
	30	DPSO	0	2.21e-02	7.63e-03	7.73e-03	15/50
F_3	10	PSO	1.989918	11.93950	4.815644	2.577000	0/50
	10	DPSO	0.9949591	8.954627	5.487021	2.836235	0/50
F_3	20	PSO	7.959673	43.77809	21.90898	7.822651	0/50
	20	DPSO	8.954632	39.79829	23.62486	8.605334	0/50
F_3	30	PSO	31.83867	1.01e+04	1.05e+03	3.16e+03	0/50
	30	DPSO	27.85884	67.65699	49.05140	11.58457	0/50
F_4	10	PSO	5.88e-16	7.69e-15	4.07e-15	8.76e-16	50/50
	10	DPSO	4.14e-15	7.69e-15	4.21e-15	5.02e-16	50/50
F_4	20	PSO	7.69e-15	2.13e-12	6.42e-12	4.49e-11	50/50
	20	DPSO	7.69e-15	1.12e-14	9.39e-15	2.18e-15	50/50
F_4	30	PSO	4.65e-13	1.54e-08	1.66e-09	4.85e-09	45/50
	30	DPSO	1.83e-14	3.61e-14	2.51e-14	5.66e-15	50/50
F_5	2	PSO	-1.00000	-0.9902841	-0.990284	2.94e-3	44/50
	2	DPSO	-1.00000	-1.00000	-1.00000	4.48e-16	50/50

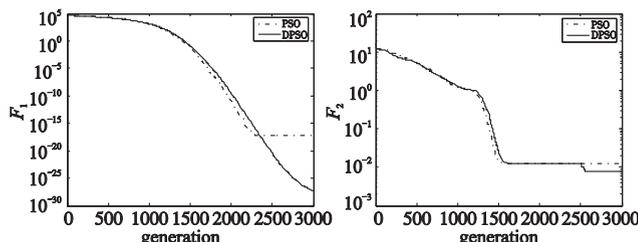


图 1 在 F_1 上两算法的收敛曲线

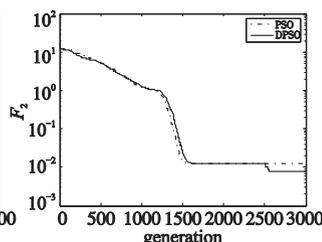


图 2 在 F_2 上两算法的收敛曲线

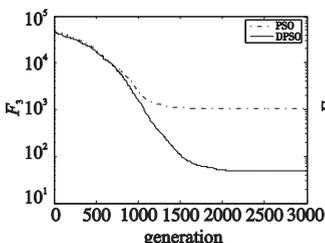


图 3 在 F_3 上两算法的收敛曲线

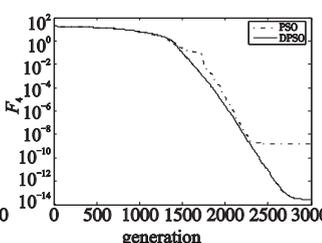


图 4 在 F_4 上两算法的收敛曲线

5 结束语

本文首先介绍了自然界中大量存在的扩散现象,然后介绍了描述扩散现象的第一和第二扩散定律以及扩散系数公式。类比于热运动中的扩散激活能和温度概念,在粒子群优化算法中引入了粒子的扩散能、种群温度和粒子扩散概率等定义,并提出了基于扩散机制的双种群粒子群优化算法(DPSO)。该算法引入了一个非常重要的概念——扩散池,两个种群中的粒子根据各自的扩散概率放入到扩散池当中,通过扩散池中的粒子来实现种群之间信息的扩散和交流共享。通过与标准 PSO 算法进行比较,发现 DPSO 算法在初期与 PSO 算法具有相近的收敛行为,但是到了中后期,当 PSO 陷入停滞状态时,DPSO 还具有良好的进化能力。下一步研究的重点是借鉴扩散机制设计其他的扩散策略。

这种方法是不可靠的。从表 3 可以看出,二阶不变矩方法所得到的结果是一致的,这与文献[11]中所给的实际结果完全相同,即匹配率为 100%。从结果中可以看出,基于 φ_2 的方法对应的弹头的矩(最小距离)误差更小,所以采用这种方法更有效。

表 1 数据库 1 中子弹次棱的相关系数比

子弹 t2 的次棱 c1	子弹 t1 的次棱			
	c1	c2	c3	c4
c1	0.812 4	0.903 4	0.852 2	0.925 5
c2	0.907 3	0.943 7	0.945 0	0.963 5
c3	0.947 0	0.790 4	0.790 4	0.852 4
c4	0.946 7	0.824 4	0.837 1	0.920 3

注:t1 为编号 1203959 枪的一发子弹,t2 为它的另外一发子弹。

表 2 数据库 1 中子弹的 φ_1 和 φ_2 二阶不变矩的结果

t1 的次棱顺序	c1 c2 c3 c4	c2 c3 c4 c1	c3 c4 c1 c2	c4 c1 c2 c3
t2 的次棱顺序	c1 c2 c3 c4			
1203959 号枪的 φ_1	0.005 482	0.003 105	0.001 868	0.004 872
1203959 号枪的 φ_2	0.000 060	0.000 050	0.000 013	0.000 036
1504519 号枪的 φ_1	0.017 673	0.015 363	0.016 105	0.014 086
1504519 号枪的 φ_2	0.000 131	0.000 088	0.000 100	0.000 080
1811345 号枪的 φ_1	0.001 612	0.000 578	0.002 147	0.001 799
1811345 号枪的 φ_2	0.000 038	0.000 033	0.000 067	0.000 035
1812492 号枪的 φ_1	0.004 631	0.002 001	0.003 287	0.005 048
1812492 号枪的 φ_2	0.000 068	0.000 022	0.000 053	0.000 085
1923252 号枪的 φ_1	0.011 675	0.009 147	0.003 999	0.018 660
1923252 号枪的 φ_2	0.000 099	0.000 127	0.000 060	0.000 186
1928033 号枪的 φ_1	0.007 671	0.002 873	0.005 601	0.010 159
1928033 号枪的 φ_2	0.000 147	0.000 051	0.000 139	0.000 173

表 3 数据库 2 中的子弹匹配结果

匹配程度由高到低的 前 5 位弹头编号						匹配程度由高到低的 前 5 位弹头编号					
t01	t20	t11	t21	t08	t16	t12	t11	t20	t16	t14	t03
t02	t20	t11	t21	t16	t03	t13	t20	t11	t16	t17	t08
t03	t07	t02	t13	t09	t19	t14	t07	t13	t15	t04	t19
t04	t11	t20	t16	t08	t17	t15	t11	t20	t16	t08	t17
t05	t07	t13	t19	t01	t18	t16	t07	t13	t15	t04	t06
t06	t20	t11	t21	t16	t08	t17	t07	t13	t19	t04	t15
t07	t20	t11	t16	t17	t14	t18	t20	t11	t08	t16	t17
t08	t07	t13	t04	t19	t15	t19	t20	t11	t17	t08	t16
t09	t20	t11	t07	t17	t07	t20	t07	t13	t15	t04	t19
t10	t07	t13	t11	t21	t20	t21	t07	t06	t13	t01	t04
t11	t07	t13	t15	t04	t19	t22	t20	t11	t15	t19	t07

综上所述,进行子弹匹配的方法,得出最有效的方法即基于二阶

不变矩的子弹匹配方法。其中基于 φ_2 二阶不变矩的方法误差很小,将其应用于数据库 2 中 22 弹头痕迹的匹配之中,并得到了每发子弹相似度从高到低的前 5 位子弹(表 3)。

5 结束语

本文在消除测量过程中弹头痕迹的误差(平移误差和转动误差)后,对二阶不变矩方法在子弹痕迹匹配中的应用进行了研究,并通过与相似度的方法进行了比较。实验表明,基于二阶不变矩的方法误差很小,而且匹配率很高。将此方法应用到第九届全国研究生数学建模中,得到了赛题中所给的 22 发弹头痕迹匹配程度从高到低的前 5 位子弹。

注:本文的研究成果获第六届(2009)全国研究生数学建模竞赛三等奖

参考文献:

- [1] 李小虎,赖杨威,孙文英,等.初探枪弹痕迹三维自动识别系统[J]. 刑侦技术,2008,1(9):34-36.
- [2] 潘自勤.对枪弹弹头痕迹特征自动检索技术的探讨[J]. 警察技术,2003,1(6):29-31.
- [3] 丘维声.解析几何[M].北京:北京大学出版社,2003:61-62.
- [4] 章毓晋.图像工程(中册)图像分析[M].2版.北京:高等教育出版社,2003:101-102.
- [5] HU M K. Visual pattern recognition by moment invariants[J]. IEEE Trans on Information Theory,1962,8(2):179-187.
- [6] 潘自勤.对枪弹弹头痕迹特征自动检索技术的探讨[J]. 警察技术,2003(6):35-37.
- [7] 寿海涛,尹继尧,于海英,等.余山地磁信号的小波消噪处理[J]. 地震地磁观测与研究,2009,10(3):85-89.
- [8] 刘清,孙颖.用实时小波分析消除噪声的动态测量误差补偿[J]. 南京师范大学学报:工程技术版,2006,33(6):101-103.
- [9] 董付国,杜萍.图像中值滤波快速计算的符号检验法[J]. 计算机工程与应用,2009,45(19):163-165.
- [10] 张晓强,高莉,于洪珍.基于 Focuss 的自适应去噪误差消除方法[J]. 清华大学学报:自然科学版,2007,42(Z2):1848-1852.
- [11] 全国研究生数学建模组委会.第六届全国研究生数学建模 B 题[EB/OL].(2009-09-23)[2009-09-24].http://www.shumo.com/home/html/481.html.

(上接第 2885 页)

参考文献:

- [1] KENNEDY J, EBERHART R C, SHI Yu-hui. Swarm intelligence[M]. San Francisco: Morgan Kaufmann, 2001.
- [2] 徐星,李元香,姜大志,等.一种基于分子动力理论的改进粒子群优化算法[J]. 系统仿真学报,2009,21(7):1904-1907.
- [3] 梁斌,董会宁,虞有康.普通物理学[M].北京:机械工业出版社,2009.
- [4] 宋晓岚,黄学辉.无机材料科学基础[M].北京:化学工业出版社,2006.
- [5] 周道品,赵永金,张廷金. Arrhenius 公式与活化能[J]. 石河子大学学报:自然科学版,1995(4):76-80.
- [6] 王铁奇,邱德华,胡桂武.基于迁徙策略的 PSO 集成及其在序列模式识别中的应用[J]. 衡阳师范学院学报,2008,29(3):21-25.
- [7] 赫然,王永吉,王青,等.一种改进的自适应逃逸微粒群算法及实验分析[J]. 软件学报,2005,16(12):2036-2044.

- [8] 马海平,陈子栋,潘张鑫.一类基于物种迁移优化的进化算法[J]. 控制与决策,2009,24(11):1620-1624.
- [9] LOVBJERG M, RASMUSSEN T K, KRINK T. Hybrid particle swarm optimiser with breeding and subpopulations[C]//Proc of Genetic and Evolutionary Computation Conference. [S. l.]: ACM Press, 2001: 469-476.
- [10] BERGH F Van den, ENGELBRECHT A P. A cooperative approach to particle swarm optimization [J]. IEEE Trans on Evolutionary Computation,2004,8(3):225-239.
- [11] 王元元,曾建潮,谭瑛.多种群协同进化的微粒群算法[J]. 计算机工程与设计,2007,28(15):3661-3664.
- [12] 陶新民,徐晶,杨立标,等.改进的多种群协同进化微粒群优化算法[J]. 控制与决策,2009,24(9):1406-1411.
- [13] 黄芳,樊晓平.基于岛屿群体模型的并行粒子群优化算法[J]. 控制与决策,2006,21(2):175-179.
- [14] 刘卓倩,顾幸生,陈国初.三群协同粒子群优化算法[J]. 华东理工大学学报:自然科学版,2006,32(7):754-757.
- [15] 夏桂梅,曾建潮.双群体随机微粒群算法[J]. 计算机工程与应用,2006,42(24):46-48.