第38卷 第17期
Vol.38 No.17

计 算 机 工 程
Computer Engineering

2012年9月
September 2012

·人工智能及识别技术· 文章编号：1000—3428(2012)17—0178—04 文献标识码：A 中图分类号：TP312

基于模拟退火的差分变异群搜索优化算法

郑慧杰[1,2]，刘 弘[1,2]，郑向伟[1,2]，孙玉灵[1,2]

(1. 山东师范大学信息科学与工程学院，济南 250014；
2. 山东省分布式计算机软件新技术重点实验室，济南 250014)

摘 要：标准群搜索优化算法易陷入局部最优。为此，引入模拟退火策略和差分进化算子，使算法跳出局部极值点，变异和迭代同时进行，并保持前期搜索速度快的特性。测试结果证明，改进算法的全局收敛能力明显提高，个体具有良好的人工智能性，能够真实模拟群体行为。

关键词：群搜索优化算法；群体动画；差分进化；局部最优

Group Search Optimization Algorithm of Differential Variation Based on Simulated Annealing

ZHENG Hui-jie[1,2], LIU Hong[1,2], ZHENG Xiang-wei[1,2], SUN Yu-ling[1,2]

(1. School of Information Science and Engineering, Shandong Normal University, Jinan 250014, China;
2. Shandong Provincial Key Laboratory for Novel Distributed Computer Software Technology, Jinan 250014, China)

【Abstract】 To reduce the possibility of falling into local optimum, metroplis rule and differential evolution operator is introduced in Group Search Optimization algorithm, which makes the variation of the algorithm get rid of the shackles of the local extreme advantage, maintain the pre-fast search feature, and improve global search capabilities. During varation and iterative, merit-based evolution improves optimization performance. Test results point that the ability to improve the global convergence of the algorithm is significantly improved, meanwhile, simulation results show that the individual has a good artificial intelligence, which can simulate group behavior toralistically.

【Key words】 Group Search Optimization(GSO) algorithm; group animation; Differential Evolution(DE); local optimum

DOI: 10.3969/j.issn.1000-3428.2012.17.049

1 概述

计算机动画[1]是目前计算机应用的一个重要方面。随着计算机软硬件技术的迅速发展，传统计算机动画创作技术不断完善，但群体动画的创作一直是计算机动画创作的难点。传统的关键帧技术需要动画师手工设定每个个体的具体路径和动作，耗费大量时间和人力。群智能算法的发展为群体动画的制作创造了新的方法和途径。典型的群体智能算法如蚁群算法[2]、粒子群优化(Particle Swarm Optimization, PSO)算法[3]及人工蜂群算法[4]已被广泛应用到群体动画中。

文献[5]基于群居动物如鸟、鱼、狮子等的觅食行为提出了群搜索优化(Group Search Optimization, GSO)算法。本文对其进行改进，并应用于群体动画中，通过群搜索优化算法生成群体中每个个体的运动路径[6]，与文献[3]相比，本文为群体动画的实现提供了更加快速、逼真的方法，极大提高了动画制作效率。同时，不仅实现了单个群体活动，而且在源源不断出现群体的情况下，实现了2个群体之间的对穿行为。

2 基于模拟退火的差分变异群搜索算法

标准GSO算法框架主要建立在PS模型[5]上，利用生物学的视觉搜索原理进行搜索，将群成员分为3类：发现者，加入者和游荡者。在每次迭代时，当代全局最优者即为发现者，加入者向发现者靠近，游荡者则在一定区域内随机游荡。所有个体都是平等的，每一个群体成员都可以在这3种角色中切换。

2.1 差分进化算子变异

差分进化(Differential Evolution, DE)算法是一种全局并行搜索算法[7]。差分进化也有类似遗传算法的变异、交叉和选择等操作，其中变异操作定义如下：

$$v_{i,G+1} = x_{r_1,G} + F \cdot (x_{r_2,G} - x_{r_3,G}) \quad (1)$$

其中，$r_1, r_2, r_3 \in \{1,2,\cdots,NP\}$；$F$是位于[0,2]的一个常数。

本文提出的变异算子是针对群成员的位置x_i的，其

基金项目：国家自然科学基金资助项目(6097004)；教育部博士点基金资助项目(20093704110002)；山东省自然科学基金资助项目(ZR2010QL01)
作者简介：郑慧杰(1987－)，女，硕士研究生，主研方向：进化计算，计算机辅助设计；刘 弘，教授、博士生导师；郑向伟，副教授、博士；孙玉灵，硕士研究生
收稿日期：2011-09-27 **修回日期**：2011-12-19 **E-mail**：dang920521dang@126.com

中，第 p 维变异定义如下：

$$\text{if}(rand() < CR \text{ or } p == k) \text{ then}$$

$$Trail_i = x_i + F \cdot (P_{1,p} + P_{2,p} - P_{3,p} - P_{index,p}) \tag{2}$$

其中，CR 为自定义的变异概率；k 是[1,D]之间的一个随机数，以保证 x_i 至少有一维产生变异；F 是位于[0,2]的一个常数；$P_{1,p}$、$P_{2,p}$、$P_{3,p}$ 是从群成员中随机选取元素的值；$P_{index,p}$ 是发现者的位置。只有当由 $Trail_i$ 得到的适应值优于变异前 x_i 的适应值时，x_i 才被 $Trail_i$ 取代，否则仍保留 x_i。这种情况可视为变异不成功，下次继续变异，直到微粒跳出目前的局部最优。

改进的群搜索优化算法在运行初期，差分变异不起作用，只有在发现者连续 G 代没有改善时，才对发现者的位置引入差分变异操作，使算法摆脱局部极值点的束缚，同时又保持前期搜索速度快的特性。

2.2　模拟退火机制

Metroplis 准则[8]是先将粒子相对位置表征的初始状态作为固体的当前状态，在这种状态下的能量为 E_i。之后，该粒子在摄动装置随机选取下，位移发生一个随机的微小变化，得到一个新的状态下的能量 E_j。如果 E_i 小于 E_j，则接受这一变化；如果 E_i 大于 E_j，则要根据固体处于该状态的几率 P 来判断：

$$p = \exp(E_i - E_j / KT) \tag{3}$$

其中，K 是 Boltzmann 常数；T 是温度。

2.3　基于模拟退火的差分变异群搜索算法 MDGSO

在 GSO 算法中，基本的扫描策略是将视觉扫描到的区域作为 N 维空间，在 3D 空间中通过最大追求角和最大追求距离进行扫描，见图 1。

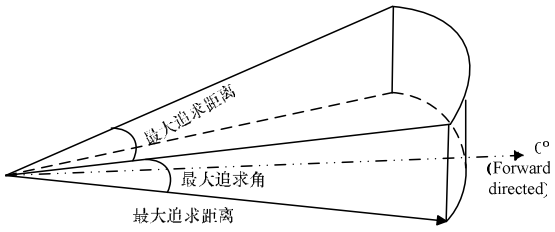


图 1　3D 空间视觉扫描的区域

在一个 N 维搜索空间里，第 i 个个体在第 K 次迭代的位置为 $X_i^k \in R^n$，角度为 $\theta_i^k = (\theta_{i1}^k, \theta_{i2}^k, \cdots, \theta_{i(n-1)}^k) \in R^{n-1}$，角度的方向为 $D_i^k(\theta^k) = (d_{i1}^k, d_{i2}^k, \cdots, d_{in}^k) \in R^n$，其中：

$$\begin{aligned} d_{i1}^k &= \prod_{p=1}^{n-1} \cos(\theta_{ip}^k) \\ d_{ij}^k &= \sin(\theta_{i(j-1)}^k) \prod_{p=1}^{n-1} \cos(\theta_{ip}^k) \\ d_{in}^k &= \sin(\theta_{i(j-1)}^k) \end{aligned} \tag{4}$$

根据群成员角色职能，针对发现者、加入者和游荡者，介绍算法流程。

(1)第 K 代中的发现者 X_p 的行为如下：

发现者将从零度角处开始扫描，然后逐步地随机抽样三点在扫描区域中扫描。

在 0°前方这点：

$$X_z = X_p^k + r_1 l_{\max} D_p^k(\theta^k) \tag{5}$$

右侧：

$$X_r = X_p^k + r_1 l_{\max} D_p^k(\theta^k + r_2 \theta_{\max} / 2) \tag{6}$$

左侧：

$$X_l = X_p^k + r_1 l_{\max} D_p^k(\theta^k - r_2 \theta_{\max} / 2) \tag{7}$$

其中，$r_1 \in R^1$ 是一个正态分布的随机数，其平均值为 0，标准差为 1；$r_2 \in R^{n-1}$ 是均匀分布在(0,1)内的随机数。

发现者在最好的资源中找到最优点。如果该最优点比在当前位置拥有更好的资源，那么它将飞往这一点，否则，会停在当前位置并且转向一个新的角度。如果发现者在 a 次迭代后不能找到一个更好的位置，它将回到 0°。

若全局最优连续 G 代没有改善，则发现者极有可能陷入某一局部极值点，从而导致寻优性能下降。为解决该问题，本文算法采用式(2)进行差分变异。若 G 太小，则可能使发现者随机改变位置，从而失去最优性；若 G 太大，则会使群体成员长期处于停滞状态，削弱引入差分算子变异的操作。通过测试实验，G 取值在 10~20 比较合适。在本文实验中，G 为 15，变异率 CR 取值为 0.1，F 取值为 0.6。

(2)在第 K 次迭代时，第 i 个加入者向发现者靠近：

$$X_i^{k+1} = X_i^k + r_3(X_p^k - X_i^k) \tag{8}$$

其中，$r_3 \in R^n$ 是均匀分布在(0,1)间的随机数。

对于 X_i^{k+1} 是否要作为本次迭代的新位置，根据引入的 Metropolis 准则的优值增量 $\Delta f = f(X_i^{k+1}) - f(X_i^k)$ 计算得出，其中，$f(X)$ 为目标函数。若 Δf <0，则接受 X_i^{k+1} 作为第 i 个群成员第 k 次迭代时的新位置，否则，以一定概率接受，即式(9)，从而既能预测到更好的移动位置，又能达到有效跳出局部极小值点的目的。

$$X_i^{k+1} = \begin{cases} X_i^{k+1} & \text{if}(\exp(f(X_i^{k+1}) - f(X_i^k)/MaxIter) > rand) \\ & \text{or}(f(X_i^{k+1}) < f(X_i^k)) \\ X_i^k & \text{otherwise} \end{cases} \tag{9}$$

(3)对于游荡者，随机走动策略是搜索随机分布资源的最有效策略。在第 K 次迭代中，产生一个随机角度 θ_i^k：

$$\theta_i^{k+1} = \theta_i^k + r_2 \alpha_{\max} \tag{10}$$

其中，$\alpha_{\max}$ 是最大的转向角。

选定一个随机距离：

$$l_i = a r_1 l_{\max} \tag{11}$$

移动到一个新的位置：

$$X_i^{k+1} = X_i^k + l_i D_i^k(\theta^{k+1}) \tag{12}$$

对于新产生的 X_i^{k+1}，如果 $f(X_i^{k+1}) - f(X_i^k) > 0$，则不选择更新，否则，进行更新。

3　算法分析

3.1　测试函数

Sphere 函数和 Rosenbrock 函数用于测试算法对单模态的优化能力，Griewank 函数、Rastrign 函数和 Schwefel 函数被用于测试算法对多模态函数的优化能力。

Sphere 函数：

$f_1(x)=\sum_{i=1}^{n}x_i^2$, $-100\leqslant x_i\leqslant 100$

$\min(f_1)=0$

Rosenbrock 函数：

$f_2(x)=\sum_{i=1}^{n-1}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$, $-30\leqslant x_i\leqslant 30$

$\min(f_2)=0$

Rastrigin 函数：

$f_3(x)=\sum_{i=1}^{n}(x_i^2-10\cos(2\pi x_i)+10)^2$, $-5.12\leqslant x_i\leqslant 5.12$

$\min(f_3)=0$

Griewank 函数：

$f_4(x)=\frac{1}{4\ 000}\sum_{i=1}^{10}x^2-\prod_{i=1}^{10}\cos(\frac{x_i}{\sqrt{i}})+1$, $-600\leqslant x_i\leqslant 600$

$\min(f_4)=0$

Schwefel 函数：

$f_5(x)=-\sum_{i=1}^{n}(x_i\sin(\sqrt{|x_i|}))$, $-500\leqslant x_i\leqslant 500$

$\min(f_5)=-418.98\times n$

3.2 实验结果及算法比较

在进行测试时，设种群大小为 20，每个测试函数运行 30 次，每次运行 1 000 代。表 1、表 2 分别显示了函数 $f_1(x)$~$f_5(x)$在 30 维和 3 维的情况下运行 50 次所得结果的平均值和标准差。

表 1 5 个函数在 30 维情况下运行 50 次的结果平均值和标准差

函数	平均值/标准差	PSO 算法	标准 GSO 算法	MDGSO 算法
$f_1(x)$	平均值	9.2 016e−31	9.112 0e−9	1.147 0e−10
	标准差	2.87e−32	1.509e−8	3.347 5e−10
$f_2(x)$	平均值	53.259 0	49.109 3	11.691 4
	标准差	188.634 5	31.217 1	82.670 8
$f_3(x)$	平均值	95.172 9	2.673 6	1.129 4e−2
	标准差	46.478 3	1.640 4	7.85e−2
$f_4(x)$	平均值	0.472 94	0.055	0.060 9
	标准差	0.170 887	0.035 8	0.029 2
$f_5(x)$	平均值	−6.454 9e+3	−1.003e+4	−1.124 9e+4
	标准差	972.772 5	3.001 4e+4	1.262 8e+3

表 2 5 个函数在 3 维情况下运行 50 次的结果平均值和标准差

函数	平均值/标准差	PSO 算法	标准 GSO 算法	MDGSO 算法
$f_1(x)$	平均值	5.65	0.018 8	1.147 0e−18
	标准差	20.35	0.042 3	3.347 5e−18
$f_2(x)$	平均值	105.17	34.530 0	7.671 7
	标准差	1.7e+02	70.670 6	20.662 4
$f_3(x)$	平均值	55.18	0.268 7	1.051 8e−31
	标准差	12.31	0.437 9	5.664 2e−31
$f_4(x)$	平均值	2.9e−02	0.323 3	0.285 3
	标准差	3.1e−02	0.166 6	0.142 6
$f_5(x)$	平均值	−11.441	−8.809e+02	−1.215 1e+03
	标准差	0.394 5	47.438	41.898 2

由表 1 可看出，标准 GSO 算法在 30 维情况下表现出比 PSO 算法更好的寻优性能，且寻优效果稳定。MDGSO 算法得到的结果更接近已知的最优值，并且其平均值低于标准 GSO 算法的平均值，说明 MDGSO 算法不仅保持了标准 GSO 算法在高维情况下具有的优势，而且自身优化能力有较大的增强。同时，MDGSO 算法标准差较小，说明 50 次实验中每次优化结果变化不大，优化性能比较稳定。MDGSO 算法比标准 GSO 算法都有更好的数据结果，算法的稳定性较高，为将 MDGSO 算法运用于三维动画制作奠定了良好的基础。同时，通过以下对标准 GSO、标准 PSO 和 MDGSO 的曲线对比，表明 MDGSO 算法具有较好的寻优性能。

图 2~图 6 表明，随着时间的推移，MDGSO 算法在持续能力上与 GSO 算法的差距有逐渐变大的趋势，且表现出较好的寻优性和收敛性。在函数 $f_4(x)$下，标准 PSO 算法表现出较好数据结果，但 MDGSO 算法相对于标准的 GSO 算法，仍然提高了群搜索算法的寻优能力，算法收敛性有较明显的改进。

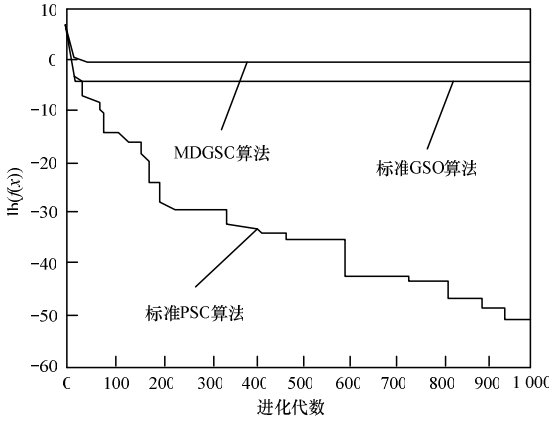


图 2 $f_1(x)$的优化曲线对比

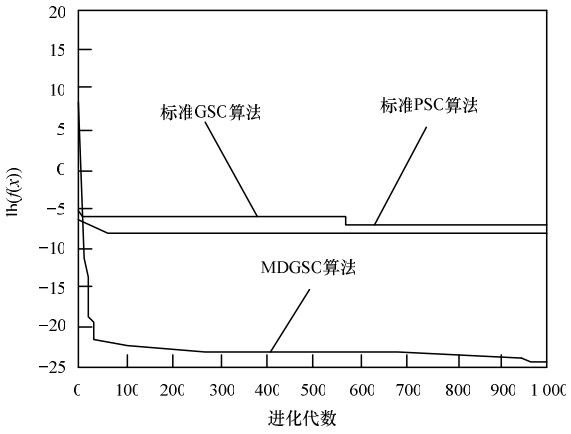


图 3 $f_2(x)$的优化曲线对比

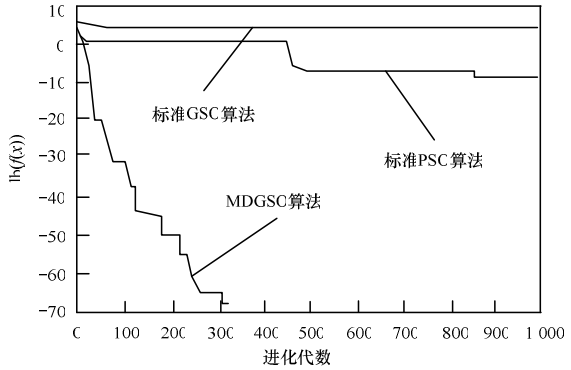


图 4 $f_3(x)$的优化曲线对比

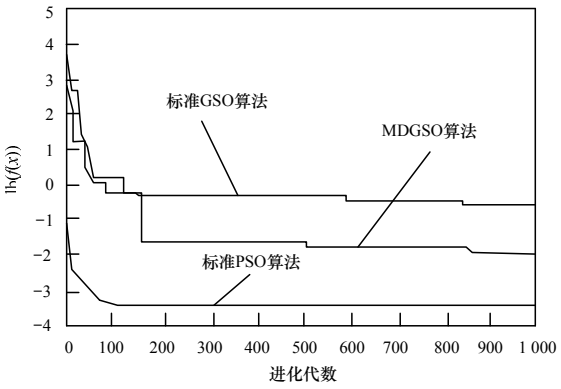


图 5　$f_4(x)$的优化曲线对比

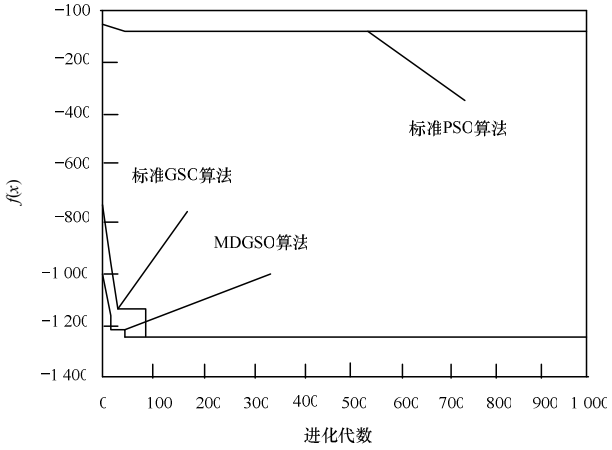


图 6　$f_5(x)$的优化曲线对比

4　仿真实验

针对上述分析与研究，仿真实验在 Microsoft Visual Studio.NET 2003 平台下，基于 ACIS 和 HOOPS 环境将 MDGSO 算法用于群体动画中。根据动画需求分析，本文模拟源源不断的人群在一个城市小区中通过人群个体与障碍物之间的碰撞避免以及人群个体之间的碰撞避免到达目标点的情形。在 Maya 动画中，可以根据动画需要导入动画场景，最后将粒子运动轨迹导入 Maya 动画中。

实验效果如图 7、图 8 所示。其中，2 个粒子作为特殊人群，穿梭源源不断的大量人群；长方形模拟的是现实中的建筑物。图 7 为初始化视图。图 8 为粒子运行若干代之后的视图。为了更清晰显示粒子的运动轨迹，图 9 为各粒子的运动轨迹。可以看出，在源源不断出现的粒子群体情况下，各个粒子都能成功地避免障碍物和个体之间的碰撞，顺利快速地到达目标位置。

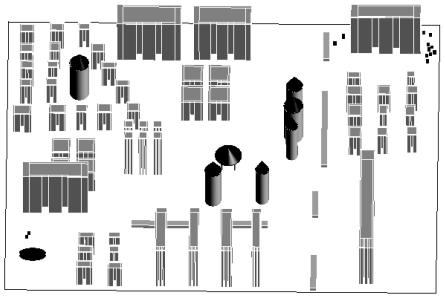


图 7　粒子群初始状态

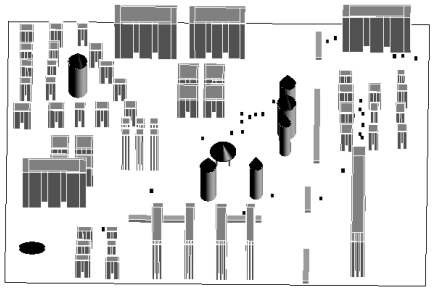


图 8　粒子运动若干代后状态

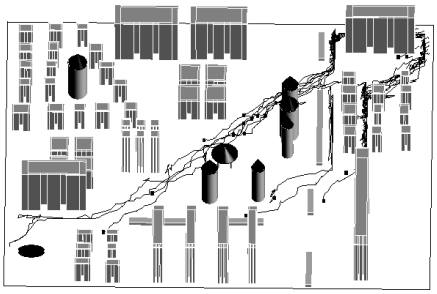


图 9　粒子运动轨迹

5　结束语

GSO 是一种新的智能计算优化方法，本文针对群成员的早熟停滞问题提出的 MDGSO 算法不仅继续保持了算法独特的角色平等转换、视觉搜索方式，而且在低维上也有较好的寻优效果，并在碰撞避免等情况下将其应用于群体动画中，为群体动画的实现提供了新的思路和方法。GSO 诞生至今已被实践证明是一种有效的方法，但是在日后工作中无论在理论研究、提高算法性能及应用于实践等方面还有很多工作和问题需要解决。

参考文献

[1] 饶云波, 陈雷霆, 周　骏, 等. 计算机群体动画中的真实感行为综述[J]. 计算机应用, 2010, 30(3): 571-578.

[2] Colorni A, Dorigo M, Maniezzo V. Distributed Optimization by Ant Colonies[C]//Proceedings of the 1st European Conference on Artificial Life. Paris, France: [s. n.], 1991: 134-142.

[3] 聂　晶, 刘　弘, 王　琪. 基于粒子群算法的群体动画研究与实现[J]. 计算机工程, 2009, 35(4): 210-214.

[4] Camazine S, Sndyd J. A Model of Collective Nectar Source by Honey Bees: Self-organization Through Simple Rules[J]. Journal of Theoretical Biology, 1991, 149(4): 547-571.

[5] He S, Wu Q H, Saunders J R. Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior[J]. IEEE Transactions on Evolutionary Computation, 2009, 13(5): 973-990.

[6] 李　雪, 刘　弘, 常　亮. 基于模拟退火机制的多微粒群协同进化算法[J]. 计算机应用研究, 2009, 26(1): 71-73.

[7] 吕新欣. Maya 高级角色动画教程[M]. 北京: 北京大学出版社, 2007.

[8] 莫愿斌, 刘贺同. 优化算法的信息原理与群搜索[J]. 计算机工程与设计, 2008, 29(4): 909-913.

编辑　张正兴