

改进粒子群优化算法研究

王 勇 张 伟 陈 军 韦鹏程
(重庆教育学院计算机系 重庆 400067)

摘 要 提出一种新的粒子群优化(Particle Swarm Optimization, PSO)算法,将微调(Fine-Tuning)机制导入 PSO 算法中,可提高算法在最优区域局部搜寻的能力,改善 PSO 在搜寻末期,粒子相似度过高的缺陷。最后用 2 种不同复杂程度的函数为例,比较本算法与 PSO 算法的最优化能力。结果显示,本算法在搜寻成功率及平均收敛时间、平均收敛代数的性能表现上皆优于 PSO 算法。

关键词 粒子群优化,微调机制,多极值函数

中图法分类号 TP301.6 **文献标识码** A

Study of Improved Particle Swarm Optimization

WANG Yong ZHANG Wei CHEN Jun WEI Peng-cheng

(Computer Department of Chongqing Education College, Chongqing 400067, China)

Abstract This paper intends to develop an improved particle swarm optimization (PSO) algorithm. The proposed method will introduce "Fine-Tuning" into the PSO algorithm which can promote the ability of local search to modify the defects of high similarity of individual particles on the late period of search following PSO algorithm. At last the performance of the improved PSO and PSO will be compared by optimizing five massively multimodal functions with varying complexities. The results show that the performance of the improved PSO is better than PSO on search success rate, average convergence time and average convergence generations.

Keywords Particle swarm optimization, Fine-tuning mechanism, Multimodal functions

1 引言

PSO^[1]的概念源自群体行为理论,启发于观察鸟群或鱼群行动时,能通过个体间特别的信息传递方式,使整个团体朝预期较适合的方向、目标而去,这个概念是模仿生物行为的反应来寻求完成群体最大利益的方法。PSO 的主要运算公式可用 2 个式子表示:

$$V_{id}^j = V_{id}^{j-1} + \phi_1 \times \text{rand}() \times (P_{id} - x_{id}^j) + \phi_2 \times \text{rand}() \times (P_{gd} - x_{id}^j) \quad (1)$$

$$x_{id}^{j+1} = x_{id}^j + V_{id}^j \times \Delta t \quad (2)$$

其中 i 表示第 i 个粒子; d 表示所处的维度; x_{id} 表示粒子在空间中所处的位置; V 表示粒子移动的速度; j 为演化的代际数; ϕ_1 与 ϕ_2 为 2 个大于零的控制参数,又称为认知参数或群居参数,大多数文献将其设为 2; $\text{rand}()$ 为一个介于 $[0, 1]$ 的常态分布随机函数; P_{id} 表示粒子个体本身所达到的最优解; P_{gd} 为族群总体所找到的最优解; Δt 为时间差,一般设为 1。依据此公式的计算方式,每一个粒子将会向 P_{id} 及 P_{gd} 合成向量的方向移动,以增加改善 x 的机会,同时又设计了随机(stochastic)的干扰量,以避免掉入局部最优(local optimum)。由于其

实作简单又具备良好的性能,PSO 至今已成为广为熟知的最优化技术。

但是,如果遇到某些特殊的最优化问题,比如目标函数的全域最优值与局部最优值所包含极值的目标函数适应值相差不大,但两者于设计空间所处的位置相距甚远,则在 PSO 的架构与基本概念中,可显现算法的性能表现,此表现与族群初始状态的分布有关。若初始状态使得族群有朝局部最优区域收敛的趋势时,族群便有可能无法跳出局部区域而落入陷阱,此为 PSO 面临这种情况时的潜在劣势。因此本文将在 PSO 中加入微调机制,以提高族群在搜寻末期最优区域局部搜寻的能力。

2 微调机制

由 PSO 流程可知,当计算完所有的粒子后,将得到 pbest (数量大小视总族群数而定)以及一个 gbest;在微调机制中,以 gbest 为对象进行微调,以期能在 gbest 邻近的微调区域中搜寻比 gbest 更佳解。

2.1 微调准则

我们知道,PSO 算法的性能表现相当优异,因此微调机

到稿日期:2009-02-20 返修日期:2009-05-16 本文受重庆市教委项目(No. KJ071502,071501,070409,080809, 081501,081502),重庆市科委自然科学基金项目(CSTC,2008BB2199)资助。

王 勇(1972-),男,硕士,讲师,主要研究方向为群体智能、数据挖掘与最优化技术;张 伟(1970-),男,博士后,教授,主要研究方向为信息安全、计算智能与数据挖掘;陈 军(1971-),男,硕士,副教授,主要研究方向为信息安全;韦鹏程(1975-),男,博士,副教授,主要研究方向为信息安全、混沌理论。

制的加入,同时必须兼顾 PSO 本身原有的优势。微调机制并非每一世代都要执行,仅在满足特定的条件下才进行微调运算,本文拟采用适应函数的方向导数值为判定的基准。假设 f 为 X 的函数, $X=[x_1, x_2, \dots, x_n]$, f 在 g 方向的方向导数定义如下:

$$D_g f = \frac{\Delta f}{\|X\|} \quad (3)$$

在某一代 j , 族群的总体最优设计值 g_{best} , 其对应的目标函数适应值 f_g , 在 j 代的 g_{best} 位于设计空间中的位置, 可表示为:

$$g_{best}^j = x_1^j e_{x_1} + x_2^j e_{x_2} \quad (4)$$

假设族群在经过了 p 次迭代后, g_{best} 来到新的位置, 式(5)代表这 2 个位置的位移向量 g ; 式(6)为两者目标函数适应值之差 Δf : $g = (x_1^{j+p} - x_1^j) \vec{u} + (x_2^{j+p} - x_2^j) \vec{v}$ (5)

$$\Delta f = f_g^{j+p} - f_g^j \quad (6)$$

可知 $D_g f$ 就是总体最优函数适应值 f_g 沿 g 方向的变化率, 利用式(3)即可得适应函数 f_g 在 $j+p$ 代的方向导数值:

$$D_g f = \frac{\Delta f}{\|g\|} \quad (7)$$

当算法于最小化问题时, 取方向导数判定值为正, 将式(6)改写为:

$$\Delta f = f_g^j - f_g^{j+p} \quad (8)$$

于是本算法在最小化问题时, 由式(8)便可使方向导数判定值恒为正。

综合以上的关系可知, 由此 $D_g f$ 可得到族群在最优值搜寻的情况。族群在最优值的搜寻时, 除非族群中的粒子搜寻到比 g_{best} 更佳的位置, g_{best} 才会进行更换, 否则族群将会保留 g_{best} 的记忆, 不做任何变动。我们便可判断出族群经过 p 次迭代后 f_g 变化的程度, 也就是说, 经过计算后获得的 $D_g f$ 较大时, 表示族群正进行全域性的搜寻, 处于较大的变动情况; 反之, 族群中粒子的分布较集中于目前族群所处的最优区域内, 进行区域性的搜寻, 族群的移动也较为缓慢。算法便可用 $D_g f$ 为指针, 判断算法是否启动微调运算。

在执行微调机制前, 使用者需预先决定世代间隔 p 与 f_g 方向导数判定值 (criterion of directional derivative, D_c^f) 的大小, 作为启动微调运算的判断条件。也就是族群在经过 p 次迭代后, f_g 方向导数值 $D_g f$ 小于或等于 D_c^f 时, 族群将略过正规的迭代执行过程, 转而进入微调区域进行微调运算; 此时将族群中所有的粒子重新定义为微调粒子群, 其微调粒子数目与原族群粒子数相同。

根据本文的测试结果及经验, 建议将世代间隔 p 与方向导数判定值 D_c^f 分别设定为 10 与 0.3~0.5, 作为判断是否启动微调运算的准则。

2.2 微调运算

微调机制核心观念主要在以 g_{best} 为中心, 经由事先所定义的方式构成微调区域 (fine-tuning block), 在此区域内随机产生微调粒子群, 计算对应每一个微调粒子的目标函数适应值后, 取出其中的最优微调粒子, 将其与 f_g 进行比较, 若此微调粒子的适应值优于 f_g , 则此微调粒子将成为新的 g_{best} 。

以一个简单的二维问题来说明, 令 g_{best} 位于边长为 l 的正方形区域的中心点, 则随机均匀分布于此正方形区域内之粒子在任一维的坐标为:

$$x_{id} = g_{best_d} + l \times [\text{rand}() - 0.5] \quad (9)$$

l 值可由式(10)决定:

$$l = \frac{\|g_{best} - x_s\|}{\sqrt{2}} \quad (10)$$

式(10)中之 $\|g_{best} - x_s\|$ 表示本代下 g_{best} 所处位置与最接近 g_{best} 目标函数适应值 f_g 粒子所处位置 x_s 两点间的距离。

考虑多维度的最优化问题, 则式(10)可改写为:

$$l = \frac{\|g_{best} - x_s\|}{\sqrt{n}} \quad (11)$$

将式(11)代入式(9), 则微调粒子在任一维度之位置可表示为:

$$x_{id} = g_{best_d} + l \times [\text{rand}() - 0.5] \quad (12)$$

由式(12)随机产生微调粒子, 并计算其目标函数适应值 f_i 再进行排序后, 获得最优微调粒子与其对应的目标函数适应值, 若其优于 f_g , 则此微调粒子将成为新的 g_{best} , g_{best} 的函数适应值也作相应替换。

3 算法描述

本算法在最小化问题执行程序中将以下面所述的步骤进行迭代:

1) 给定适当比例之随机粒子数 q 、执行微调运算的条件世代间隔 p 、方向导数判定值 D_c^f 及族群总数 n 。

2) 初始时, 在 d 维设计空间中随机产生 n 个粒子, 由这些粒子数所构成的群体称为族群 (population), 每个粒子所处的位置 x_i^j 代表设计空间中的一个随机解。

3) 当 $c = p + 1$ 时检查 f_g 的方向导数值 $D_g f$ 是否小于或等于 D_c^f ; 若为真, 则略过步骤 4)~9), 跳到步骤 9) 进行微调运算; 反之, 则不进行微调运算, 族群仍旧遵循步骤 4)~8) 进行最优值搜寻。

4) 计算粒子 i 于 d 维空间中的目标函数适应值 (fitness of particle i , f_i)。

5) 将粒子的函数适应值 f_i 与其本身所搜寻过的个体最优值 (fitness of pbest, $f_{i,p}$) 进行比较, 若此粒子现在的函数适应值优于 $f_{i,p}$, 则将 pbest 替换成该粒子现在之位置 x_i , 而 $f_{i,p}$ 亦替换成现在的 f_i 。反之, 则略过步骤 6), 依步骤 7) 进行粒子的速度与位置更新。

6) 在 $f_{i,p}$ 比较且替换后, f_i 将与整个族群所搜寻过的总体最优适应值 (fitness of gbest, f_g) 进行比较, 若 $f_{i,p}$ 优于 f_g , 则 g_{best} 将替换到该粒子现在的位置 x_i , 而 f_g 也替换到其对应的函数适应值 f_i 。

7) 粒子 x_i 根据式(1)与式(2)进行速度与位置更新。

8) 若 i 小于族群总数 n , 重回步骤 4), 进行下一个粒子的计算, 直至计算完族群中所有的粒子。

9) 若算法满足步骤 3) 的条件, 则对 g_{best} 执行微调机制。在以 g_{best} 为中心的微调区域内, 微调粒子数根据式(12), 随机于此区域内均匀分布产生。否则, 略过步骤 9)~10)。

10) 对每一个微调粒子 x_i 计算其目标函数适应值 f_i 并进行排序, 取出最优微调粒子与其所对应的最优微调函数适应值; 若其优于 f_g , 则原先族群的总体最优函数适应值 f_g 与总体最优设计值 g_{best} 均分别替换至此最优微调粒子对应的

(下转第 280 页)

识别在不可控制条件下的准确性和有效性,对人脸图像首先进行一系列简单实用而又有效的预处理,采用一种优化的新型 LBP 纹理提取算法,即 LTP 算法,其提取的纹理特征增强了对光照变化和噪声的鲁棒性。最后对纹理特征空间应用 PCA 降维和 Fisher 准则最优鉴别分类,在标准人脸库上的测试结果验证了此方法的有效性和正确性。

参 考 文 献

[1] Ahonen T, Pietikainen H A. Face description with local binary patterns: Application to face recognition [J]. IEEE TPAMI, 2006, 28(12)

[2] Tan Xiaoyang, Triggs B. Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions [M]. AMF, 2007

[3] Zhao Jiali, Wang Haitao, Ren Haibing, et al. LBP Discriminant Analysis for Face Verification [C] // Proceedings of the 2005 Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). 2005

[4] Zhang Wenchao, Shan Shiguang, Gao Wen, et al. Local Gabor Binary Pattern Histogram Sequence (LGBPHS): A Novel Non-Statistical Model for Face Representation and Recognition [C] // Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV'05). 2005

[5] Pietikainen M. Face Analysis Using Local Binary Patterns [M]. Machine Vision Group, University of Oulu, Finland

[6] Heusch G, Rodriguez Y, Marcel S. Local Binary Patterns as an Image Preprocessing for Face Authentication [C] // IEEE International Conference on Automatic Face and Gesture Recognition. 2006; 11-16

[7] 郑宇杰, 於东军, 杨静宇, 等. 一种给予 ICA 和 LDA 组合的人脸识别新方法. 2006

[8] 王月华. 基于 LBP 特征空间研究及其在自动人脸识别中应用 [D]. 南京: 南京航空航天大学, 2006

[9] 杨梅. 基于 LBP 和混合 2DPCA+PCA 的人脸识别研究 [D]. 南京: 南京理工大学, 2007

(上接第 259 页)

函数适应值与其在求解空间中所处位置。

11) 当总体最优函数适应值的误差小于或等于所要求的函数适应值误差或最大迭代数到达时, 则停止迭代, 算法输出总体最优函数适应值 f_g 与总体最优设计值 g_{best} , 即为算法执行最优化结果。反之, 重回步骤 3), 继续进行迭代。

4 试验结果

在当前许多改进的 PSO 算法中, PSO-CF^[2] 得到的评价较高, 广为多数研究者所引用。因此本文选择以 PSO-CF 与本文所提出的算法进行平均收敛时间、平均收敛代数及成功率等性能比较与分析。使用的函数如表 1 所列。

表 1 测试函数表

函数名	函数表达式	全局最优值
Rastrigrin	$f_1(x) = 10N + \sum_{i=1}^N (x_i^2 - 10 \cos(2\pi x_i))$	$x_i = 0, f(x) = 0$
Griewank	$f_2(x) = \frac{1}{4000} \sum_{i=1}^N x_i^2 - \prod_{i=1}^N \cos(\frac{x_i}{i}) + 1$	$x_i = 0, f(x) = 0$

在参考文献[3]中, 作者经过验证, ϕ_1 与 ϕ_2 取值为 2.8 与 1.3, PSO 得到的效率最高, 因此, 本文也如此取值。本文对每一个测试函数的最佳化问题都重复运行 100 次。试验结果比较如表 2、表 3 所列。

表 2 函数 1 测试结果比较表

	PSO-CF	本算法	改进率 %
平均计算时间(s)	8.96	6.08	32.14
平均收敛代数	848.59	586.34	30.94
成功率 (%)	96	100	4.16

表 3 函数 2 测试结果比较表

	PSO-CF	本算法	改进率 %
平均计算时间(s)	13.57	9.12	32.79
平均收敛代数	1252.34	994.65	20.57
成功率 (%)	93	98	5.37

结束语 本文所提出的改良 PSO 算法在演算过程中适

时加入微调机制, 提高了族群在搜索末期最优区域局部搜寻的能力, 改善了 PSO 所存在的劣势, 并提高了算法的各项性能。若能算法可靠度的方向进一步深入研究, 在算法不同的参数组合运用更多不同的测试函数或工程最优化问题进行算法可靠度的评估, 或是将此法运用于不同的实际最优化问题上, 这些都可以作为往后 PSO 算法改良的研究目标。本文所提出的方法并不能完全改善 PSO 所有的劣势, 但希望藉由我们的一点工作能对 PSO 算法发展有所帮助, 以使 PSO 算法更趋成熟稳健。

参 考 文 献

[1] Kennedy J, Eberhart R C. Particle swarm optimization [C] // Proceedings of the IEEE International Conference on Neural Networks. Perth, Australia, IEEE, Piscataway, NJ, USA, 1995, 4: 1942-1948

[2] Clerc M, Kennedy J. The particle swarm: Explosion, stability, and convergence in a multimodal complex space [C] // Proceedings of the Congress of Evolutionary Computation. Vol. 6, Washington DC, IEEE, Piscataway, NJ, USA, 2000; 58-73

[3] Li-Ping Z, Huan-Jun Y, Shang-Xu H. Optimal choice of parameters for particle swarm optimization [J]. Journal of Zhejiang University: Science, 2005(6A): 528-534

[4] Xiaohui H, Eberhart R C, Shi Y. Recent advances in particle swarm [C] // Proceedings of the 2004 Congress on Evolutionary Computation. Institute of Electrical and Electronics Engineers Inc., 2004, 1: 90-97

[5] Kennedy J, Eberhart R C, Shi Y. Swarm intelligence [M]. San Francisco: Morgan Kaufmann Publishers, 2001

[6] Eberhart R C, Kennedy J. A new optimizer using particle swarm theory [C] // Proceedings of the 6th International Symposium on Micro Machine and Human Science. Nagoya, Japan, IEEE, Piscataway, NJ, USA., 1995; 39-43

[7] 黄贤英, 张丽芬. 基于粒子群优化的模糊聚类算法 [J]. 重庆工学院: 自然科学版, 2008, 22(11): 120-123