

一种改进快速稳定的多目标优化算法*

樊纪山¹, 刘冠蓉², 王 鲁², 时忠伟³

(1. 淮海工学院 电子工程系, 江苏 连云港 222005; 2. 武汉理工大学 计算机学院, 湖北 武汉 430070; 3. 山东宏嘉工程造价咨询所, 山东 潍坊 261041)

摘要: 多目标优化问题属于高维的搜索空间, 用一些传统方法来优化这些问题会导致较高的时间复杂性。为了解决该问题, 使用了粒子群优化算法 (PSO), 同时将 ϵ -dominance 的概念应用到 PSO 中。该方法在实验过程中取得了良好的效果。其运算速度快, 而且最终优化的点数可以得到控制。

关键词: 多目标优化; 粒子群算法; 支配多目标优化

中图分类号: TP301.6 文献标志码: A 文章编号: 1001-3695(2007)04-0052-02

Improved Fast Robust PSO Multi-objective Algorithm

FAN Ji-shan¹, LIU Guan-rong², WANG Lu², SHI Zhong-wei³

(1. Dept. of Electronic Engineering, Huaihai Institute of Technology, Lianyungang Jiangsu 222005, China; 2. College of Computer Science, Wuhan University of Technology, Wuhan Hubei 430070, China; 3. Shandong Hope-Star Engineering Cost Consultation Ltd., Weifang Shandong 261041, China)

Abstract: Multi-objective optimization problems always have large and complex search space. Using traditional exact methods solve multi-objective problems may results high time-complex. To solve the problem, the Particle Swarm Optimization (PSO) algorithm was used. The algorithm used ϵ -dominance method. During experiment, it has better results than CMOPSO, at the same time, it runs fast, and the optimal solutions can be controlled.

Key words: multi-objective optimization; particle swarm optimization algorithm; ϵ -dominance multi-objective optimization

优化问题是人们在改造世界时经常会遇到的一类普遍的问题。当人们考虑一个优化问题时, 目的是求得目标的最优解。而多目标优化问题更是很普遍的问题^[1]。多目标优化问题起源于许多实际复杂系统的设计、建模和规划问题。这些系统所在的领域包括工业制造、城市运输、资本运算、森林管理、水库管理、新城市的规划和美化、能量分配等。几乎每个现实生活中重要的决策问题都要在考虑不同约束的同时处理若干相互冲突的目标, 这大大增加了问题的复杂程度^[2]。因此, 研究一种快速、稳定、实用的多目标优化算法具有相当重要的现实意义。本文是使用粒子群优化算法 (PSO) 来解决多目标优化问题, 同时将 ϵ -dominance 的概念应用到 PSO 中。粒子群优化是近几年发展起来的群聚智能算法。该算法是基于群体中的各个粒子能够从过去的经历和其他粒子的经历得到有效信息这个假设之上的。通过实验发现, 该算法与 ϵ -dominance 一起使用能使优化过程更加稳定、快速, 而且实验的效果也很理想。

1 多目标优化的基本概念

1.1 多目标优化问题

一个通常的多目标问题包括一个含有 n 个参数 (自变量) 的集合, 一个 k 个目标函数的集合, 一个 m 个约束的集合和目标函数和约束都是自变量的函数^[3]。

$$\text{Maximize: } y = f(x) = (f_1(x), f_2(x), \dots, f_k(x))$$

$$\text{Subject to: } g(x) = (g_1(x), g_2(x), \dots, g_m(x)) \leq 0$$

$$\text{Where: } x = (x_1, x_2, \dots, x_n) \in X$$

$$y = (y_1, y_2, \dots, y_k) \in Y$$

其中, x 是自变量向量, y 是目标向量, X 是自变量空间, Y 是目标空间, 约束 $g(x) \leq 0$ 确定了可行解。

1.2 Pareto 支配关系

对于任意两个自变量向量 a 和 b 。

$$a: b \text{ (} a \text{ dominates } b \text{) if } f(a) < f(b)$$

$$a: _ b \text{ (} a \text{ Weakly dominates } b \text{) if } f(a) \leq f(b)$$

$$a \sim b \text{ (} a \text{ is indifferent to } b \text{) if } f(a) \neq f(b) \wedge f(b) \neq f(a)$$

1.3 ϵ -dominance 概念^[4]

如果对于所有的 $i \in [1, 2, \dots, m]$ 均满足 $(1 + \epsilon) \times f_i \leq g_i$, 则认为 f ϵ -dominance 支配 g 。表示形式为 $f \epsilon$ -dom g 。其中 $\epsilon > 0$ 。

1.4 Pareto 优化

根据目标域的偏序关系, 就可以得出 Pareto 优化的定义: 一个自变量向量 $x \in X$, 考虑 X 的一个子集 A , 如果! $\exists a \in A$ $a: x$, x 被称为是对应于集合 A 的 Pareto 最优。就是说, 自变量向量 a 如果称为对应于集合 A 的 Pareto 最优解或非劣最优解, 那么, 在任何一个目标上它都不能继续得到优化而并没有引起任何其他维目标的劣化。

收稿日期: 2005-10-28; 修返日期: 2006-03-01 基金项目: 国家“863”计划资助项目 (2002AA1Z1490)

作者简介: 樊纪山 (1975-), 男, 硕士, 主要研究方向为机器学习、网络 (fjsszw2005@126.com); 刘冠蓉 (1945-), 男, 副教授, 硕士, 主要研究方向为分布计算; 王鲁 (1981-), 男, 硕士研究生, 主要研究方向为机器学习; 时忠伟 (1979-), 女, 本科, 主要研究方向为工程设计优化。

2 粒子群算法

粒子群优化(Particle Swarm Optimization, PSO)^[5,6]也是基于群体的方法,最早由 Kennedy 和 Eberhart 提出。它最初是由鸟类迁徙觅食的模型而得到的。在该模型中,群体中的每个个体都有给予一定的内部信息和外部信息而控制自己行为的能力。也就是说,每个个体都有一定的感知能力,能够感知自己周围的局部最好位置的个体和整个群体的全局最好位置的个体存在,并根据当前状态和所获得的信息,调整自己的下一步行为,从而使整个群体表现出一定的智能性。在解决优化问题时,每个个体的所在位置可被相应地看成一个潜在的解,经过反复迭代最终得到所需的全局最优解。下面给出粒子群算法的伪代码:

```

创建并初始化一个 n 维空间的 PSO: M
初始化 r1 和 r2, 每个粒子的 x, v
repeat
for 每个粒子 i [1, ..., M]:
利用 vj,t+1 = vj,t + c1r1(pj,ti - xj,ti) + c2r2(pj,tg - xj,ti)
xj,t+1 = xj,t + vj,t+1; 计算 xj,t+1i 和 vj,t+1i
如果 f(xj,t+1i) 优于 f(xj,ti), 就用 xj,t+1i 更新 xj,ti;
用群体中的最优 pj,t+1i 更新 pj,ti
until 满足停止条件
    
```

其中, 每个个体由当前位置 x 和速度 v 组成; 其能感知的最优位置由 p 表示; $p_{j,t}^{i,g}$ 表示全局最优位置; $p_{j,t}^{i,l}$ 表示局部最优位置。这里 $j=1, 2, \dots, n$, 代表第 j 维搜索空间; $i=1, 2, \dots, M$ 代表第 i 个个体; t 为迭代次数; r_1 是收缩因子; r_2 是惯性因子; c_1 和 c_2 为两个正常数; r_1 和 r_2 为某个范围中的随机数。

3 ϵ -MOPSO 算法流程

在这个算法中关键的一步就是计算 ϵ -非支配解。其计算过程(图 1)的伪代码如下:

```

Input: a, b
Output:  $\epsilon$ -non-dominated solution( s)
if a; b then
return a
else if b; a then
return b
else if a; b then
return a
else
return a b
end if
    
```

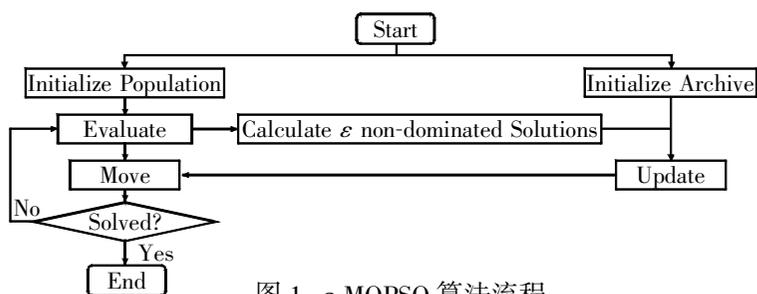


图 1 ϵ -MOPSO 算法流程

4 实验结果

对该算法验证其稳定性和快速性。在实验过程中主要与 CMOPSO^[7] 多目标优化算法进行比较。在测试时,用到的测试

函数是多目标算法测试比较常用的几个测试函数,即 ZDT1 ~ ZDT6。这几个测试函数是 Deb 在 1998 年指出多目标优化中比较困难的几个问题: 收敛到 Pareto 最优阵面; 在群体中维持个体多样性。对于问题, 在单目标演化优化中多峰函数、欺骗函数和孤立最优函数都是很著名的问题。为了获得好的 Pareto 最优阵面的分布特性,问题也是很重要的。然而, Pareto 最优阵面的某些特征会不利于多目标优化演化算法找到各种各样的 Pareto 最优解: 凸或非凸的、离散的、非均匀分布的。针对具有这些特性的 Pareto 最优解, Deb 设计出这六个相应的测试函数。其测试函数如表 1 所示。

表 1 测试函数

问题	n	变量值范围	目标函数	优化解集	注释
ZDT1	30	[0, 1]	$f_1(x) = x_1$ $f_2(x) = g(x) [1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i^2) / (n-1)$	$x_1 \in [0, 1]$ $x_i = 0, i=2, \dots, n$	凸函数
ZDT2	30	[0, 1]	$f_1(x) = x_1$ $f_2(x) = g(x) [1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i^2) / (n-1)$	$x_1 \in [0, 1]$ $x_i = 0, i=2, \dots, n$	非凸函数
ZDT3	30	[0, 1]	$f_1(x) = x_1$ $f_2(x) = g(x) [1 - \sqrt{x_1/g(x)} - x_1/g(x) \sin(10x_1)]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i^2) / (n-1)$	$x_1 \in [0, 1]$ $x_i = 0, i=2, \dots, n$	凸非连续
ZDT4	10	$x_1 \in [0, 1]$ $x_i \in [-5, 5], i=2, 3, \dots, n$	$f_1(x) = x_1$ $f_2(x) = g(x) [1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^n [x_i^2 - 10 \cos(4x_i)]$	$x_1 \in [0, 1]$ $x_i = 0, i=2, \dots, n$	非凸
ZDT6	10	[0, 1]	$f_1(x) = 1 - \exp(-4x_1) \sin^6(4x_1)$ $f_2(x) = g(x) [1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i^2) / (n-1)^{0.25}$	$x_1 \in [0, 1]$ $x_i = 0, i=2, \dots, n$	非凸非均匀分布

在这个算法中对于不同的测试函数取不同的值。在其他设置参数相同的情况下, ϵ 的值不同计算时间就不同, 得到的优化点个数就不同。 ϵ 的取值越大, 得到的优化点就越少。这就是本算法的关键所在, 而且最终得到的优化点的个数等于 $1/\epsilon$ 。

实验参数的设置如表 2 所示。

表 2 实验参数

算法	种群大小	归档大小	Box 划分	惯性因子	C_1	C_2	r_1	r_2
CMOPSO	100	100	30	0.4	1	1	$U(0, 1)$	$U(0, 1)$
MOPSO	100	100	30	0.4	1	1	$U(0, 1)$	$U(0, 1)$

MOPSO 算法中 ϵ 的取值, 对于不同的测试函数取不同的值。如图 2 所示 $\epsilon=0.05$ 时, 比较的结果; 当 $\epsilon=0.0001$ 时, 其实验结果如图 3 所示。为了说明此算法的优越性, 在以下几个测试函数的验证过程中就不再设 $\epsilon=0.0001$ 的情况了。

从图 2 中可以明显地看出来, 当 ϵ 的取值很小时, 其最终优化的点数就越多。如图 3 当 $\epsilon=0.0001$ 时, 其优化结果比 CMOPSO 算法的优化结果要好一些。这两个图是对 ZDT1 进行测试。

以下分别对策略树函数 ZDT2、ZDT3、ZDT4 和 ZDT6 进行测试。其测试结果分别如图 4、5 所示。

对于 ZDT2, $\epsilon=0.03$; 对于 ZDT3, $\epsilon=0.001$; 对于 ZDT4, $\epsilon=0.065$; 对于 ZDT6, $\epsilon=0.075$ 。可以看出, 对于不同 ϵ 的取值, 实验结果优化的点的数目就不同。(下转第 76 页)

2.3 Avatar虚拟教学助理

Avatar是一个二维的卡通形象,可以分别在学生端和教师端显示。Avatar接收情感认知模型的输出信息,向学生提供改变自身学习状态的建议;向教师提供学生的学习状态,如多少学生表示出对课程的厌倦,或表现出积极情绪的学生个数。如果需要教师调整教学,则给出调整建议。此外Avatar还可以实现老师与学生之间直接的信息交流。

3 系统实现

本系统采用 Visual C++ 6.0,数据库采用 SQL Server 实现。Avatar提示界面如图6所示。表情识别模块采用几何特征定位和肤色模型方法。姿态识别模块采用肤色密度分布法和特征验证法。Avatar虚拟教学助理采用 Windows COM 组件实现,教师和学生的信息传输采用局域网的 B/S 结构。系统实现中的部分效果图如图7所示。

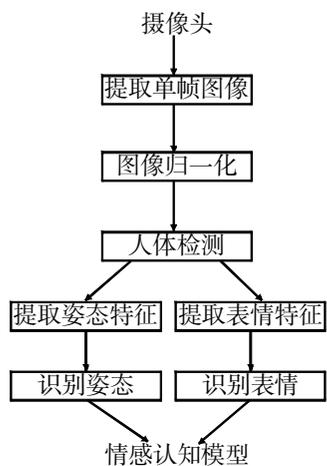


图5 表情信息识别模块功能图



图6 Avatar提示界面



图7 表情识别效果图

4 结束语

该系统通过情感认知模型对学生表情的处理,实时地向教师反映学生的学习心理状态、状态评价值以及状态不佳的原因,并向教师和学生提供改进学习效果的建议。本系统可以帮助教师实时了解学生的学习状态,实现了电子教学中师生的情感互动;弥补了现代电子教育方式缺乏情感交流的缺陷;有助于提高电子教学或网络教学的教学效率,有利于实现真正意义上的个性化、人性化计算机教学系统。该系统可与 E-Learning 教学平台配合使用,也可以独立用于电子教室教学。今后将在此基础上进行更深入的研究:系统的情感认知模型、人脸检测和表情识别算法;设计更加形象的虚拟教师助理;在系统中加入语音识别等新的技术;研制更实用的人性化教学系统。并在此基础上研究开发应用于其他领域的情感交互信息系统,如情感会议或在电子政务和电子商务等领域实现人性化的电子销售助理和虚拟服务助理。

参考文献:

[1] 王有智, 欧阳仑. 心理学基础——原理与应用 [M]. 北京: 首都经济贸易大学出版社, 2003: 374.

[2] 丁德成, 张伟, 师梅梅. 现代脑科学与教育 [J]. 陕西师范大学学报: 自然科学版, 2004, 32(专辑): 189-193.

[3] 王志良. 人工心理学——关于更接近人脑工作模式的科学 [J]. 北京科技大学学报, 2000, 22(5): 478-481.

[4] 孟昭兰. 普通心理学 [M]. 北京: 北京大学出版社, 1994: 403.

[5] EKMAN P. An argument for basic emotions [J]. **Cognition and Emotion**, 1992, 6(3-4): 169-200.

[6] 吴江霖. 心理学概论 [M]. 广州: 广东高等教育出版社, 2003: 186-188.

(上接第 53 页)

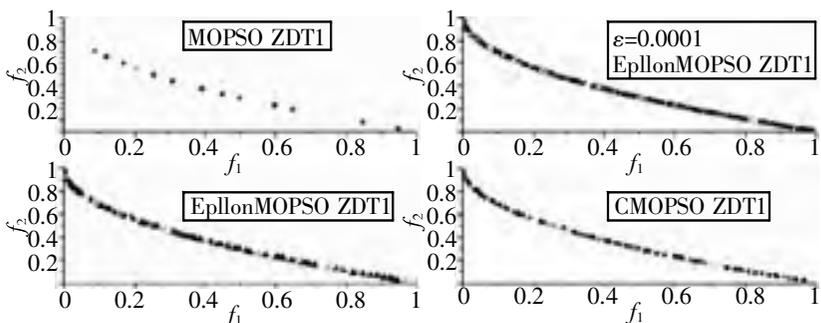


图2 $\epsilon=0.05$ 时的比较结果

图3 $\epsilon=0.0001$ 时的比较结果

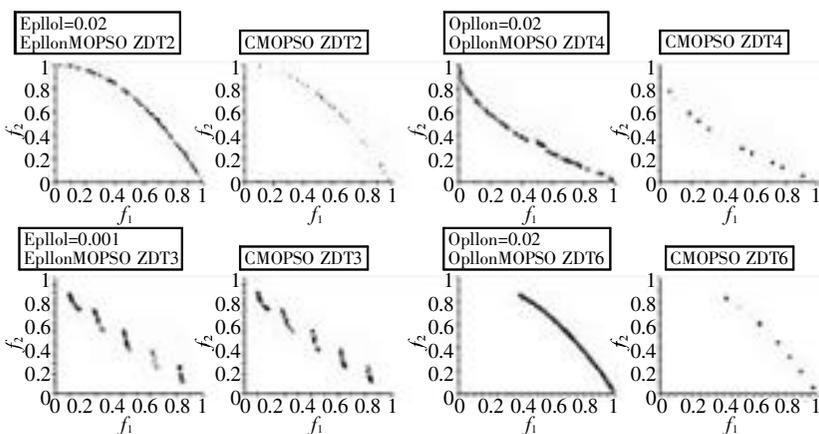


图4 显示 ZDT2 和 ZDT3 的实验结果

图5 显示 ZDT4 和 ZDT6 的实验结果

5 结束语

多目标优化问题是一个比较热门的研究课题。不同的优化算法有不同的效果,但最终的目的就是如何控制算法搜索的收敛性、解的多样性和计算时间最小化。本文提出的 -MOP-

SO算法能满足上述要求,而且还能有效地控制优化解的个数。这就为决策者提供了一种很好的解决方法。因而,此算法具有有效的使用价值。

参考文献:

[1] 谢涛, 陈火网, 康立山. 多目标优化的演化算法 [J]. 计算机学报, 2003, 26(8): 997-1003.

[2] 玄光男, 程润伟. Genetic algorithm and engineering optimization [M]. 北京: 清华大学出版社, 2004: 76-83.

[3] SRINIVAS N, LKALYANMOY D. Multi-objective optimization using the nondominated sorting in genetic algorithms [J]. **Evolutionary Computation**, 1994, 2(3): 221-248.

[4] ZITZLER E D. Combining convergence and diversity in evolutionary multi-objective optimization evolutionary computation [J]. **Massachusetts Institute of Technology Evolutionary Computation**, 2002, 10(3): 208-346.

[5] KENNEDY J, EBERHART R C. A discrete binary version of the particle swarm algorithm: proc. of Conf. on Systems, Man, and Cybernetics [C]. Piscataway: IEEE Service Center, 1997: 4104-4109.

[6] EBERHART R C, KENNEDY J. A new optimizer using particle swarm theory: proceedings of the 6th International Symposium on Micro Machine and Human Science, Nagoya, Japan [C]. Piscataway: IEEE Service Center, 1995: 39-43.

[7] GREGORIO T P, CARLOS A C. Using clustering techniques to improve the performance of a multi-objective particle swarm optimizer [C] // KALYANMOY D, et al. Genetic and evolutionary computation conference, part I: lecture Notes in Computer Science vol. 3102. Seattle: Springer-Verlag, 2004: 237-255.