

粒子群优化算法及其与遗传算法的比较

沈 艳¹, 郭 兵², 古天祥¹

(1. 电子科技大学机械电子工程学院 成都 610054; 2. 四川大学计算机学院 成都 610041)

【摘要】粒子群优化算法是根据鸟群觅食过程中的迁徙和群集模型而提出的用于解决优化问题。该文讨论粒子群优化算法的基本原理和实现步骤,分析了该算法中各参数的设置。通过一个测试函数,对粒子群优化算法与遗传算法进行了比较,结果表明粒子群优化算法在找寻最优解效率上好于遗传算法。

关键词 群集智能; 粒子群; 遗传算法; 优化

中图分类号 O224 文献标识码 A

Particle Swarm Optimization Algorithm and Comparison with Genetic Algorithm

SHEN Yan¹, GUO Bing², GU Tian-xiang¹

(1. School of Electromechanical Engineering, UEST of China Chengdu 610054; 2. School of Computer, Sichuan University Chengdu 610041)

Abstract Particle swarm optimization, rooting from simulation of swarm of bird, solves optimization problem. Firstly, discusses particle swarm optimization algorithm principle and step of implementation, and then analyzes each of parameter. Particle swarm optimization algorithm compares with genetic algorithm through the same mathematic function. The comparative result indicates that Particle swarm optimization algorithm can obtain the optimum solutions more easily than genetic algorithm and it is a good optimization method with strong competition.

Key words warm intelligence; particle swarm; genetic algorithm; optimization

自然界经过数万年的演化,以最优的方式及其规律存在。“师法自然”,人们受自然界的运行发展机制的启发,建立和发展了一系列的优化算法,如遗传算法(Genetic Algorithm, GA)、模拟退火、人工神经网络等,这些算法已广泛应用于工程中,解决许多研究过程中遇到的困难。

随着人们对鸟类的群集(Flocking)行为的研究,如大雁飞行自动排成人字形、蝙蝠在洞穴中快速飞行却互不相碰等。1995年,文献[1,2]对鸟群觅食过程中的迁徙和群集进行模拟:鸟群觅食时,从一地到另一地的迁徙过程中,总是有一只鸟对食源的大致方向具有较好的洞察力,同时,在找寻食源的途中,它们通过一套自己独有的方式随时相互传递着信息,特别是较好的信息。在“好消息”的指引下,导致鸟群“一窝蜂”的奔向食源,达到在食源的群集。粒子群优化算法从中得到启示并用于解决优化问题。

1 粒子群优化算法

1.1 基本原理

在粒子群优化(Particle Swarm Optimization, PSO)算法中,一只鸟称为“粒子”,解群相当于一个鸟群,一地到另一地的迁徙相当于解群的进化,“好消息”相当于解群每代进化中的最优解,食源相当于全局最优

解。

假设在一个 N 维目标搜索空间中,有 m 个粒子组成一个群落,其中第 i 个粒子在 N 维空间里的位置 $X_i=(x_{i1}, x_{i2}, \dots, x_{iN})^T$, $i=1, 2, \dots, m$; 飞行速度 $V_i=(v_{i1}, v_{i2}, \dots, v_{iN})^T$, $i=1, 2, \dots, m$; 适应值 $fitness_i=f(X_i)$, 则 $Pbest_i$ 和 $X_i^{Pbest}=(x_{i1}^{Pbest}, x_{i2}^{Pbest}, \dots, x_{iN}^{Pbest})^T$ 分别为第 i 个粒子曾经达到的最大适应值及其对应的位置。 $Gbest$ 为在群体所有粒子经历过的最好位置,其索引号为 g 。则对每一代,其第 d 维根据如下方程变化:

$$v_{id} = wv_{id} + c_1r_1(x_{id}^{Pbest} - x_{id}) + c_2r_2(x_{gd}^{gbest} - x_{id}) \quad (1)$$

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

式中 v_{id} 为粒子 i 飞行速度矢量的第 d 维分量; x_{id} 为粒子 i 位置矢量的第 d 维分量; r_1, r_2 为 $[0, 1]$ 之间的随机数; c_1, c_2 为加速度系数; w 为惯性权重。式(1)中,首项为粒子先前的速度; $c_1r_1(x_{id}^{Pbest} - x_{id})$ 项为“认知项(Cognitive Term)”,该项与粒子的认知经验相关; $c_2r_2(x_{gd}^{gbest} - x_{id})$ 项为“社会项(Social Term)”,它代表粒子间的信息共享与合作。式(2)为粒子 i 的新坐标位置。它们共同决定粒子 i 下一步的运动位置。

1.2 实现步骤

粒子群优化算法的实现步骤为:(1) 初始化粒子群。设群体规模为 m ,在允许的范围内随机设置粒子的初始位置和速度;(2) 评价每个粒子的适应值,即计算每个粒子的目标函数 $fitness_i$;(3) 对所有的 $i \in \{1, 2, \dots, m\}$,如果 $fitness_i > Pbest_i$,则令 $fitness_i = Pbest_i$, $X_i^{Pbest} = X_i$,如果 $fitness_i > gbest$,则重新设置 $gbest$ 的索引号 g ;(4) 根据式(1)、(2)调整每一个粒子的位置和速度;(5) 检查终止条件。如果达到最大迭代次数 gen_{max} 或最好解停滞不再变化,终止迭代,否则返回(2)。

粒子群优化算法主要包括^[3]:(1) 粒子以随机的方式在整个问题空间中流动并且可以对自己所处的环境进行评价(计算适应度)。(2) 每个粒子均可以记忆自己到过的最好位置和感知邻近粒子已达到的最好位置。(3) 在改变速度的时候同时考虑自己到过的最好位置和邻近粒子已达到的最好位置。

1.3 参数分析

PSO算法参数包括:群体规模 m ,惯性权重 w ,加速度系数 c_1, c_2 ,最大代数 gen_{max} 。

(1) 惯性权重 w :它使粒子保持运动惯性,使其有扩展搜索空间的趋势,有能力探索新的区域。如果 $w=0$,由于速度本身没有记忆性,只取决于粒子当前位置和其历史最好位置 $Pbest$ 和 $gbest$,所以,粒子群将收缩到当前的全局最好位置,更像一个局部算法;如果 $w > 0$,微粒有扩展搜索空间的趋势,即有全局搜索能力。因此,对于较大的 w 值有利于跳出局部极小点,而较小的 w 值有利于算法收敛。(2) 加速度系数 c_1, c_2 :该项参数用于调整粒子的自身经验与社会经验在其运动中所起的作用,表示将每个粒子推向 $Pbest$ 和 $gbest$ 位置的统计加速项的权重。低的值允许粒子在被拉回前可以在目标区域外徘徊,而高的值则导致粒子突然冲向或越过目标区域。如果 $c_1=0$,则粒子没有认知能力。在粒子的相互作用下,不仅能到达新的搜索空间,但也容易陷入局部极值点;如果 $c_2=0$,粒子间没有社会信息共享,其算法变成一个多起点的随机搜索;如果 $c_1=c_2=0$,粒子将一直以当前的速度飞行,直到到达边界。通常 c_1, c_2 的范围在 $0 \sim 4$ 之间。(3) 粒子个数和维数:粒子个数一般取 $20 \sim 40$,对于大部分的问题10个粒子已经足够取得好的结果,对于比较难的问题或者特定类别的问题,粒子数可以取到100或200。

2 举例与比较

2.1 测试函数

本文采用测试函数 $f(x) = x \sin(10\pi x) + 1.0$, $x \in [-1, 2]$,测试函数如图1所示。图中, $f(x)$ 是一个多峰函数,它在区间 $[-1, 2]$ 上的最优解为: $x=1.8515$, $f(x)=2.8512$ 。

2.2 参数对粒子群优化算法影响

(1) c_1, c_2 的不同取值对PSO的影响如图2所示,图中, $c_1=c_2=0.5$,最优解出现在24代; $c_1=c_2=1$,最优解出现在23代; $c_1=c_2=1$,最优解出现在20代; $c_1=c_2=2$,最优解出现在17代; $c_1=c_2=2.5$,最优解出现在27代; $c_1=c_2=4$,最优解没有出现。(2) 惯性权重 w 的不同取值对PSO的影响如图3所示,图中, $w=1.05$ 时,粒子群优化算法的搜索效率和搜索精度高。

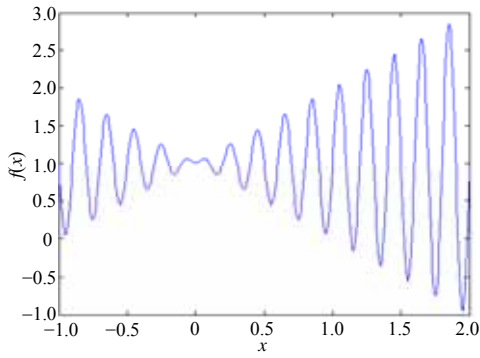


图1 测试函数图

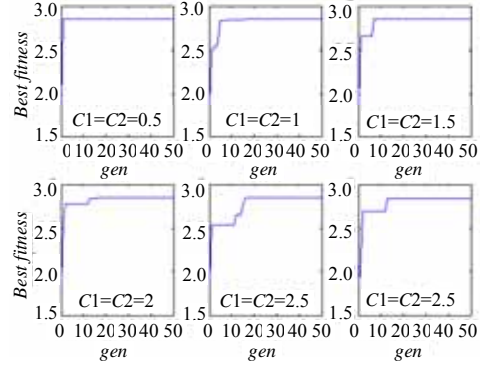


图2 c_1 、 c_2 的不同取值对PSO的影响

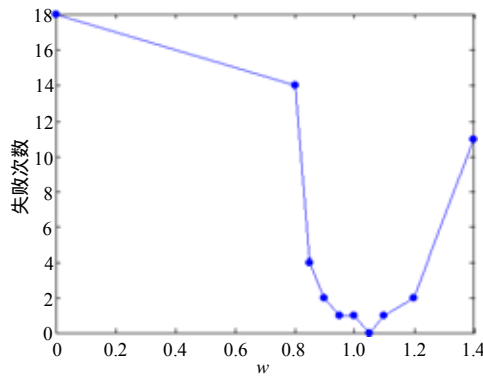


图3 w 不同取值对PSO的影响

2.3 粒子群优化算法与遗传算法比较

遗传算法的控制参数设定如下：群体规模大小20，个体串长度为22；杂交概率0.9，变异概率为0.01；最大遗传代数100；粒子群优化算法的控制参数设定为：群体规模大小20，最大进化代数：50， $c_1=c_2=2$ ；遗传算法与粒子群优化算法的解随进化代数的变化情况如图4、5所示，比较结果如图6所示。由图4~6，粒子群优化算法的最优解出现在17代，而遗传算法的最优解出现在49代，粒子群优化算法在找寻最优解效率上要好于遗传算法。

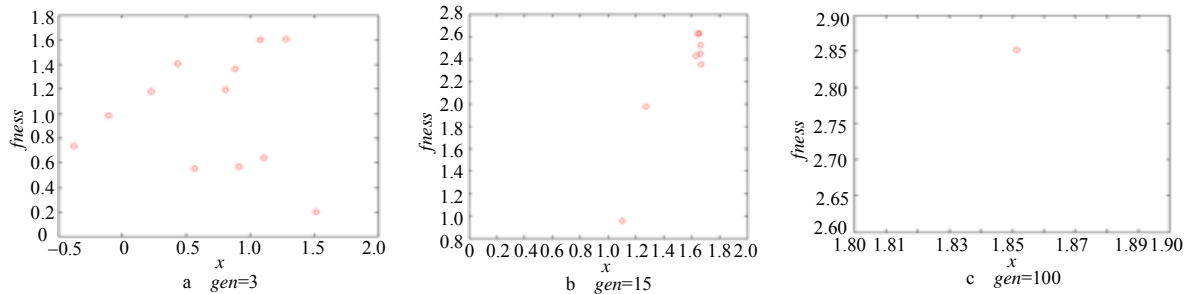


图4 遗传算法的解随进化代数的变化情况

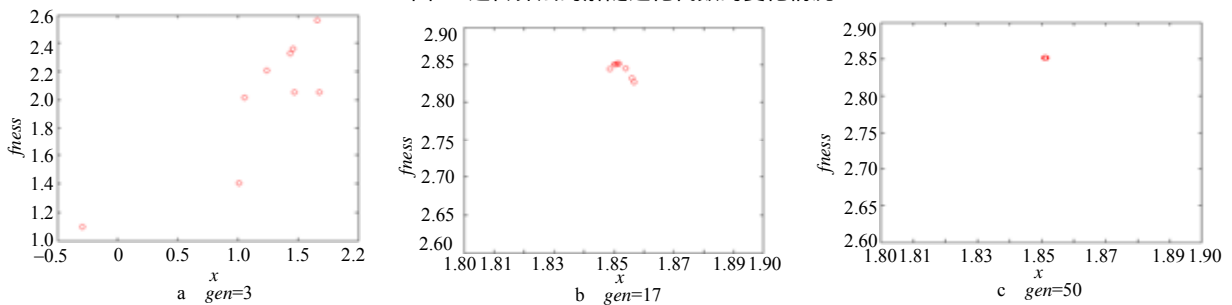


图5 粒子群优化算法的解随进化代数的变化情况

由此可见,虽然粒子群优化算法和遗传算法有很多共同之处:两者都随机初始化种群,都使用评价函数来衡量个体的优劣程度,并根据由评价函数得到的适应值来进行一定的随机搜索。但粒子群优化算法搜索速度快,在搜索性能上好于遗传算法,这是因为:(1)粒子群优化算法没有遗传操作,如交叉(Crossover)和变异(Mutation),而是利用个体在解空间中的随机速度来改变个体,其解群相对进化代数而言,表现出更强的随机性,其计算复杂度比遗传算法低。(2)粒子具有“记忆”的特性,它们通过“自我”学习和向“他人”学习,使其下一代解有针对性的从“先辈”那里继承更多的信息,从而能在较短的时间内找到最优解。(3)与遗传算法相比,粒子群优化算法的信息共享机制是很不同的:在遗传算法中,染色体互相共享信息,所以整个种群的移动是比较均匀的向最优区域移动;在粒子群优化算法中,信息流动是单向的,即只有 g_{best} 将信息给其他的粒子,这使得整个搜索更新过程跟随当前最优解。

从以上实验及分析可以看出粒子群优化算法在多数情况下,比遗传算法更快的收敛于最优解。

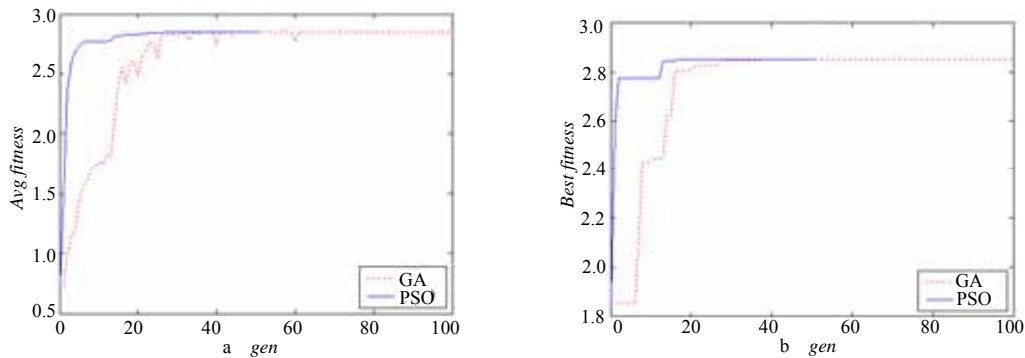


图6 粒子群优化算法与遗传算法的比较

3 结束语

粒子群优化算法是通过粒子间的相互作用发现复杂搜索空间中的最优区域,是一类随机全局优化技术。其优势在于简单容易实现而又功能强大。目前,粒子群优化算法最成功地运用是在进化神经网络方面,但对其的研究才刚起步,相对其他较成熟的进化算法,还没有形成系统的分析方法和一定的数学基础,且应用范围较小。随着研究的进一步深入,该算法将会应用到越来越多的领域中。

本文研究工作得到电子科技大学青年基金(jx04038)资助,在此表示感谢。

参 考 文 献

- [1] Kennedy J, Eberhart R C. Particle swarm optimization[C]. Proc. IEEE Int. Conf.on Neural Networks, Perth, WA, Australia, 1995. 1 942-1 948
- [2] Eberhart R C, Kennedy J A. A new optimizer using particle swarm theory[C]. Proc. The Sixth Int. Symposium on Micro Machine and Human Science, Nagoya, Japan, 1995. 39-43
- [3] Kennedy J, Eberhart R C, Shi Y. Swarm intelligence[M]. San Francisco: Morgan Kaufmann Publishers, 2001
- [4] Dautenhahn K. Book review: Swarm intelligence[J]. Genetic Programming and Evolvable Machines, 2002,3(1): 93-97

编 辑 漆 蓉