

• 人工智能 •

粒子群优化算法综述

黄少荣

(广东司法警官职业学院 信息管理系, 广东 广州 510520)

摘要: 为了进一步推广应用粒子群优化算法(PSO)并为深入研究该算法提供相关资料,在分析 PSO 基本原理和机制的基础上,从参数设置、收敛性、拓扑结构及与其它算法混合等方面对其发展历程和研究现状进行深入调查,论述了该算法的各种改进技术,并阐述了 PSO 在连续领域和离散领域的应用成果,最后对该算法未来发展趋势做出了展望。

关键词: 粒子群算法优化算法; 参数设置; 拓扑结构; 混合算法

中图分类号: TP301 文献标识码: A 文章编号: 1000-7024 (2009) 08-1977-04

Survey of particle swarm optimization algorithm

HUANG Shao-rong

(Department of Information Management, Guangdong Justice Police Vocational College, Guangzhou 510520, China)

Abstract: In order to promote the applications of particle swarm optimization algorithm (PSO), and provides the relevant information for the further research on this algorithm, a review on the recent progress of PSO is given. Based on the introduces of PSO's basic principles and mechanism, a thorough investigation on the research progress of PSO is given in aspects of parameter setting, convergence characteristic, topology, hybrid algorithm, and the applications in continuous and discrete domains. Finally, the future research issues of the PSO are given.

Key words: particle swarm optimization algorithm; parameters setting; topology; hybrid algorithm

0 引言

粒子群优化算法 (particle swarm optimization, PSO) 是由 Kennedy 和 Eberhart 在研究鸟类和鱼类的群体行为基础上于 1995 年提出的一种群智能算法^[1],其思想来源于人工生命和演化计算理论,模仿鸟群飞行觅食行为,通过鸟集体协作使群体达到最优。

PSO 是进化计算的一个分支,是一种基于迭代的优化工具,系统初始化为一组随机解,通过迭代搜寻最优值。PSO 的原理和机制简单,仅仅通过更新速度和位置来不断进化到全局最优解,无需梯度信息,可调参数少,算法容易实现且运行效率高。自其提出以来,受到众多研究者的探讨和改进,并且被运用到了越来越广泛的领域之中。

本文全面介绍 PSO 算法的基本原理、发展和应用,为 PSO 的推广应用和深入研究提供相关的资料。

1 粒子群算法原理

PSO 中,每个优化问题的解看作搜索空间中的一只鸟(即粒子),所有的粒子都有一个被优化的函数决定的适应值并且有一个速度决定它们飞翔的方向和速率,粒子们追随当前的最优粒子在解空间中搜索。算法首先初始化一群随机粒子,然后通过迭代找到最优解。在每一次迭代中,粒子通过跟踪

两个“极值”即个体极值和全局极值来更新自己的速度与位置。在 D 维目标搜索空间中,由种群数为 m 的粒子组成粒子群,其中,第 i 个粒子在第 d 维的位置为 x_{id} ,其飞行速度为 v_{id} ,该粒子当前搜索到的最优位置为 $p_{id}(pBest)$,整个粒子群当前的最优位置为 $p_{gd}(gBest)$ 。

速度与位置更新公式如下

$$v_{id+1} = v_{id} + c_1 \times rand() \times (p_{id} - x_{id}) + c_2 \times rand() \times (p_{gd} - x_{id}) \quad (1)$$

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

式中 $rand()$ ——[0,1]范围内变化的随机数 c_1 和 c_2 为加速系数。

PSO 的算法框架如下所述,图 1 给出了 PSO 的算法流程。

(1)初始化所有的个体(粒子),初始化它们的速度和位置,并且将个体的历史最优 $pBest$ 设为当前位置,而群体中最优的个体作为当前的 $gBest$ 。

(2)在当代的进化中,计算各个粒子的适应度函数值。

(3)如果该粒子当前的适应度函数值比其历史最优值要好,那么历史最优将会被当前位置所替代。

(4)如果该粒子的历史最优比全局最优要好,那么全局最优将会被该粒子的历史最优所替代。

(5)对每个粒子按照公式(1)和公式(2)对速度和位置进行更新。

(6)进化代数增加 1,如果还没有到达结束条件,转到(2),否则输出 $gBest$ 并结束。

收稿日期: 2009-01-26; 修订日期: 2009-03-11。

作者简介: 黄少荣 (1976 -), 女, 广东饶平人, 硕士, 讲师, 研究方向为计算机应用和智能算法。E-mail: huangshaorong@163.com

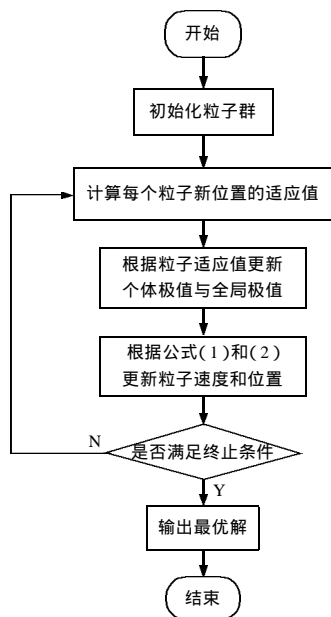


图1 PSO 流程

2 粒子群算法的发展

传统 PSO 算法具有：局部搜索能力较差，搜索精度不高；容易落入局部最优；搜索性能对参数具有依赖性；算法后期易震荡等缺点。

自提出以来，很多研究者从参数设置、收敛性、拓扑结构、与其它算法融合等角度对传统 PSO 进行研究，并针对其不足提出了各种改进，以提高算法性能。

2.1 参数设置

PSO 中的可调参数有 ω 、 c_1 和 c_2 、 V_{\max} 、种群规模等，这些参数的设置对 PSO 的性能有重要影响，对其设置原则进行研究将是一个广阔而富有挑战的领域。

(1) 惯性权重 ω

Shi 等人首次将惯性权重 ω 引入到 PSO 的速度更新公式中^[2]，如公式(3)所示。 ω 保持粒子的运动惯性，使其有扩展搜索空间的趋势，获得较好的求解效果，其后的研究一般都以该模型为基础。Shi 等还指出，较大的 ω 有利于群体在更大的范围内进行搜索，而较小的 ω 能够保证群体最终收敛到最优位置，因此提出了一个 ω 随着进化代数线性递减的模型，如公式(4)所示。

$$v_{id+1} = \omega v_{id} + c_1 \times \text{rand}() \times (p_{id} - x_{id}) + c_2 \times \text{rand}() \times (p_{gd} - x_{id}) \quad (3)$$

$$\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \times \frac{i}{\text{Iter}_{\max}} \quad (4)$$

Chatterjee 等则提出了非线性变化惯性权重的 PSO 算法^[3]，提高算法的收敛速度。而使用模糊系统来动态调节的 ω 值和随机的 ω 值(设为 $\omega = 0.5 + \text{rand}(0, 1)/2$)也分别被使用。王俊伟等综合分析了常数和可变 ω 对算法性能的影响并提出了 ω 的设置原则^[4]。Zhan 等通过对 PSO 的进化状态进行判定和划分，提出了一种基于进化因子的自适应控制的惯性权重。由于自适应的惯性权重能够根据算法的搜索信息而提供合适的取值，因此能够获得更快的收敛速度和更好的求解精度^[5]。

(2) 加速系数 c_1 和 c_2

c_1 和 c_2 代表了将每个粒子拉向 $pBest$ 和 $gBest$ 位置的随机加速项的权重，分别指示着个体的“认知能力”和群体的“社会引导”功能，一般被设为相同的值，最常见的是 2.0。Suganthan 最早使用了非 2.0 的加速系数，参照时变惯性权重的设置方法，让 c_1 和 c_2 同步时变，随时间线性递减，发现特定加速系数对特定的问题有较好的效果，但是没有找到一般的规律^[6]。Ratnaweera 等提出了异步时变的加速系数，让 c_1 和 c_2 在优化过程中随时间进行不同的变化，效果并不明显，但由于混合了其它技术而可以取得对算法的改善^[7]。另一个常见的取值为 1.49445，它可以确保 PSO 算法的收敛，但在处理多峰问题时易落入局部最优。彭宇等利用方差分析方法分析惯性权重和加速系数的设置对算法性能的影响并提供参数设置的指导原则^[8]。但是，以上的这些调整方案并不能针对具体的进化过程和进化状态而对参数做出有效的控制，为了实现更加有效的调整方案，Zhan 等提出了一种使用聚类的方法对进化状态进行判断，并且对加速系数进行相应调整的算法改进方案，并且取得了比较明显的改进^[9]。进一步地，使用进化状态估计方法的自适应参数控制 PSO 具有更加快速的收敛能力和更高的求解精度^[5,9]。

(3) 最大速度 V_{\max}

V_{\max} 决定粒子在一个循环中最大的移动距离，在每一维粒子的速度都会被限制在 V_{\max} 内，如果某一维更新后的速度超过 V_{\max} ，那么这一维的速度就被限定为 V_{\max} 。原始 PSO 没有 ω ，需要 V_{\max} 来限制速度的范围，采用 ω 后， V_{\max} 可以省略。不过，即使速度的大小可以由 ω 控制，但是公式(2)中的位置更新必须合法，每次更新后要检查新位置是否在问题空间内，否则必须修正，重新随机设定或者限定在边界。

另外，粒子群的规模(粒子数 m)与问题的规模相关，一般设定在 20~50 之间，对于复杂问题 m 可以取到 100~200。算法中止条件一般为最大循环数或最小错误要求。

2.2 算法的收敛性

Clerc 等最早开始研究 PSO 的数学基础和收敛性^[10]，在分析 PSO 算法的工作机制及其收敛性基础上引入压缩因子 k 以保证 PSO 的收敛，速度更新公式变为如公式(5)~(6)所示，加入压缩因子 PSO 可看作为惯性权重 PSO 的一个特例。Bergh 证明最初版本的 PSO 算法不能收敛于全局最优，提出一种保证收敛于局部最优解的 PSO 算法^[11]。曾建潮等则提出了一种能保证全局收敛的 PSO 算法^[12]。此外，Trelea^[13] 和 Kadirkamanathan 等^[14] 也分别对 PSO 进行了数学分析，而且逐步从静态分析过渡到动态系统分析，这些研究对 PSO 基本机理的认识和发展具有重要意义。

$$v_{id+1} = k \times [v_{id} + c_1 \times \text{rand}() \times (p_{id} - x_{id}) + c_2 \times \text{rand}() \times (p_{gd} - x_{id})] \quad (5)$$

$$k = \frac{2}{|2 - \phi - \sqrt{\phi^2 - 4\phi}|}, \quad c_1 + c_2 = \phi > 4 \quad (6)$$

2.3 拓扑结构研究

种群拓扑结构分全局版($gBest$)模型和局部版($lBest$)模型， $gBest$ 模型中，粒子同时跟踪自己的历史最优值与全局最优值来更新速度和位置。 $lBest$ 模型中，粒子根据自己的历史最优值和领域内粒子的最优值对速度和位置进行更新，粒子 i 的领域随着迭代次数的增加而递增，最后扩展到整个群，这时即变

成 $gBest$ 模型。Kennedy 研究了拓扑结构对搜索性能的影响,表明小规模拓扑结构对求解复杂问题有利,而大规模的拓扑结构则适用于求解简单问题^[15]。潘峰等使用了不同连接拓扑的 PSO 进行实验,结果表明,没有一种邻近群拓扑对所有基准函数都是最合适的,具体选择哪种拓扑结构与具体问题有关^[16]。为了避免一些由固定拓扑结构带来的缺点,Hu 和 Eberhart 提出了一种动态的拓扑结构解决方案^[17]。针对 $gBest$ 模型收敛快但易陷入局部最优和 $lBest$ 模型收敛慢但不易陷入局部最优的特点,有学者提出将两者相结合的方法,其中速度更新的公式如(7)所示。其中 n 是一个 $[0,1]$ 区间的实数,表示全局版速度所占的比重。

$$v = n \times v(gBest \text{ 版}) + (1-n) \times v(lBest \text{ 版}) \quad (7)$$

2.4 与其它算法混合

在对 PSO 的改进中,很多学者提出了将 PSO 混合其它算法或技术以改进算法性能的建议,其中最多的是 PSO 与遗传算法混合,根据不同遗传算子可以生成 3 种不同混合算法:

(1)与选择算子结合:Angeline 于 1999 年首次将选择算子混合到 PSO 中,通过保留具有较高适应值的粒子进行复制来加快收敛速度^[18]。结合思想:按粒子适应度大小赋予每个粒子一个被选中的概率,根据概率对粒子进行选择用于生成下一代。这种混合算法具有更强的搜索能力,收敛速度快,但易陷入局部最优。

(2)与变异算子结合:Ratnaweera 等将变异算子引入到了 PSO 中以保证群体的多样性^[19]。结合思想:测试所有粒子与当前最优距离,当距离小于一定值时,以变异率让粒子随机初始化。钟文亮等针对 PSO 在优化多峰函数时容易早熟的缺点,在 PSO 中引入启发性变异机制,扩展了算法的搜索区域,提高了算法的速度和精度且不容易陷入局部最优^[20]。

(3)与交叉算子结合:LØvbjerg 等将饲养和分组成群的思想引入了 PSO^[21],提出了带有繁殖和子种群的混合算法,在每轮迭代中随机选择一定的粒子作为父代,随机两两交叉以产生具有新的空间坐标和速度的子代粒子取代父代以保持种群规模,并且把种群分为若干子种群,交叉可以在同一子种群内进行,也可在不同种群间进行。该混合算法使粒子增强搜索能力,易于跳出局部最优。

另外,Zhang 等将寿命的思想融入到 PSO 中,为全局最优粒子设置一个生命周期,当生命周期结束的时候,将重新选择全局最优粒子。该方法很好地解决了 PSO 容易落入局部最优的缺点,并且在多峰函数的优化取得了显著的效果^[22]。高鹰等把模拟退火思想引入到具有杂交和高斯变异的 PSO 中,有效避免算法陷入局部最优,提高算法的收敛速度和精度^[23]。协同 PSO 以及协同混合 PSO 也被提了出来,将粒子的 D 维分到 D 个粒子群中,每个粒子群优化一维向量,评价适应值时再把这些分量合并为一个完整向量。

3 粒子群算法的应用

PSO 最早应用于非线性连续函数的优化和神经网络的训练,目前的应用已经涉及了功率电子电路的设计^[24],电磁学^[25],经济调度^[26],医学图像配准^[27],多目标优化^[28],系统设计^[29],机器学习与训练^[30],数据挖掘^[31],模式识别与分类^[32],信号控制^[33],

Steiner 树问题^[34],旅行商问题^[35],多播路由问题^[36],车间调度^[37]等各个领域。

3.1 连续领域的应用

作为连续领域的优化方法,PSO 基本上能够胜任所有的应用。许多实际工程与实践问题本质上是函数优化问题,或者这些问题本身就是要求进行参数的设计与优化,可以转换为函数优化问题进行求解,PSO 非常适合这类问题的求解。

3.2 离散领域的应用

传统 PSO 是针对连续领域的优化,但通过改变 PSO 问题的形式或对模型进行修正,例如速度和位置的更新公式,可将 PSO 运用到离散领域中,比较典型的解决方案有二进制编码和顺序编码^[38]。二进制 PSO 由 Kennedy 提出,将粒子的每一维及粒子本身的历史最优、全局最优限制为 1 或 0,不限制速度。位置更新时,设定一个阈值,当速度高于该阈值时,粒子位置取 1,否则取 0。顺序编码 PSO 由 Hu 等提出,用以解决排列问题,其编码规则与遗传算法的顺序编码相同,速度则定义为粒子变化的概率。

Zhong 等通过引入变异因子 C_3 并修改位移模型,设计了一种改进离散粒子群算法 C3DPSO,在 TSP 问题中取得成功应用^[35]。Zhan 等将离散 PSO 应用于多播路由问题中^[36]。高海兵等人则提出了广义 PSO 模型,使其适用于离散领域^[39]。

4 结束语

PSO 作为一种新的进化计算方法,由于其本身简单的概念、清晰的生物机制、容易的编程实现、高效的运行效率,已经越来越受到人们的重视,而且应用也越来越广泛。目前对 PSO 的研究大多注重于实际应用,对算法内部机理的研究比较贫乏,数学基础薄弱,模型中位置和速度的构造设计不成熟,参数设置也缺乏确切的理论依据,通常依靠经验来设定。而且像将 PSO 离散化、拓扑结构的改进、与其它各种先进技术融合之类的尝试和研究工作也远远不够,这些都是 PSO 在本身不断被完善、应用的前提下必须突破的问题。在众多研究者的不断努力下,我们相信 PSO 将会越来越完善,发展和应用前景会更加光明。

参考文献:

- [1] Kennedy J, Eberhart R C. Particle swarm optimization[C]. Proc of the IEEE International Conference on Neural Networks. Piscataway, NJ: IEEE Service Center, 1995: 1942-1948.
- [2] Shi Yuhui, Eberhart R C. A modified particle swarm optimizer [C]. Proc of the IEEE International Conference on Evolutionary Computation. Piscataway, NJ: IEEE Service Center, 1998: 69-73.
- [3] Chatterjee A, Siarry P. Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization[J]. Computers and Operations Research, 2006, 33(3): 859-871.
- [4] 王俊伟, 汪定伟. 粒子群算法中惯性权重的实验与分析[J]. 系统工程学报, 2005, 20(2): 194-198.
- [5] Zhan Z H, Zhang J. Adaptive particle swarm optimization[C]. The Sixth International Conference on Ant Colony Optimization and

- Swarm Intelligence,2008:227-234.
- [6] Suganthan P N. Particle swarm optimizer with neighborhood operator[C]. Proc of the Congress on Evolutionary Computation. Piscataway, NJ: IEEE Service Center, 1999: 1958-1962.
- [7] Ratnaweera A, Halgamuge S K, Watson H. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[J]. IEEE Transaction on Evolutionary Computation, 2004, 8(3): 240-255.
- [8] 彭宇, 彭喜元, 刘兆庆. 微粒群算法参数效能的统计分析[J]. 电子学报, 2004, 32(2): 209-213.
- [9] Zhan Z H, Xiao J, Zhang J, et al. Adaptive control of acceleration coefficients for particle swarm optimization based on clustering analysis[C]. Singapore: Proceeding IEEE Congress on Evolutionary Computation, 2007: 3276-3282.
- [10] Clerc M, Kennedy J. The particle swarm-explosion, stability and convergence in a multidimensional complex space [J]. IEEE Transaction on Evolutionary Computation, 2002, 6(2): 58-73.
- [11] Van den Bergh F, Engelbrecht A P. A new locally convergent particle swarm optimizer[C]. Hammamet: Proc of the IEEE International Conference on Systems, Man and Cybernetics, 2002: 96-101.
- [12] 曾建潮, 崔志华. 一种保证全局收敛的 PSO 算法 [J]. 计算机研究与发展, 2004, 41(8): 1333-1338.
- [13] Trelea I C. The particle swarm optimization algorithm: Convergence analysis and parameter selection[J]. Information Processing Letters, 2003, 85(6): 317-325.
- [14] Kadirkamanathan V, Selvarajah K, Fleming P J. Stability analysis of the particle dynamics in particle swarm optimizer [J]. IEEE Transaction on Evolutionary Computation, 2006, 10(3): 245-255.
- [15] Kennedy J, Mendes R. Population structure and particle swarm performance[C]. Proc IEEE CEC, 2002: 1671-1676.
- [16] 潘峰, 陈杰. 粒子群优化算法模型分析[J]. 自动化学报, 2006, 32(3): 369-378.
- [17] Hu X, Eberhart R C. Multiobjective optimization using dynamic neighborhood particle swarm optimization [C]. IEEE Int Congress on Evolutionary Computation. Piscataway, NJ: IEEE Service Center, 2002: 1677-1681.
- [18] Angeline P J. Using selection to improve particle swarm optimization[C]. Proc of the IEEE International Conference on Evolutionary Computation. Piscataway, NJ: IEEE Service Center, 1998: 84-89.
- [19] Ratnaweera A, Halgamuge S K, Watson H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[C]. IEEE Trans Evol Comput, 2004: 240-255.
- [20] 钟文亮, 张惠森, 张军, 等. 带启发性变异的粒子群优化算法[J]. 计算机工程与设计, 2008, 29(13): 3402-3406.
- [21] Løvbjerg M, Rasmussen T K, Krink T. Hybrid particle swarm optimizer with breeding and subpopulations[C]. Proc of the GECCO, 2001: 469-476.
- [22] Zhang J, Lin Y. A particle swarm optimizer with lifespan for global optimization on multimodal functions[C]. HongKong: IEEE Congress on Evolutionary Computation, 2008: 2439-2445.
- [23] 高鹰, 谢胜利. 基于模拟退火的粒子群优化算法[J]. 计算机工程与应用, 2004, 40(1): 47-50.
- [24] Zhang J, Shi Y, Zhan Z H. Power electronic circuits design: A particle swarm optimization approach[C]. SEAL, 2008: 605-614.
- [25] Ciuprina G, Ioan D, Munteanu I. Use of intelligent-particle swarm optimization in electromagnetism[J]. IEEE Trans on Magnetics, 2002, 38(2): 1037-1040.
- [26] Victoire TAA, Jeyakumar A E. Reserve constrained dynamic dispatch of units with valve-point effects [J]. IEEE Trans Power Syst, 2005, 20(3): 1273-1282.
- [27] Wachowiak M P, Smolikova R, Zheng Y F. An approach to multimodal biomedical image registration utilizing particle swarm optimization[J]. IEEE Trans Evol Comput, 2004, 8(3): 289-301.
- [28] Messerschmidt L, Engelbrecht A. Learning to play games using a PSO-based competitive learning approach[J]. IEEE Trans Evol Comput, 2004, 8: 280-288.
- [29] Gaing Z L. A particle swarm optimization approach for optimum design of PID controller in AVR system[J]. IEEE Trans Energy Conversion, 2004, 19(2): 384-391.
- [30] Franken N, Engelbrecht A P. Particle swarm optimization approaches to coevolve strategies for the iterated prisoner's dilemma[J]. IEEE Trans Evol Comput, 2005, 9(6): 562-579.
- [31] Sousa T, Silva A, Neves A. A particle swarm data miner [C]. Springer-Verlag Berlin Heidelberg, 2003: 43-53.
- [32] Sousa T, Silva A, Neves A. Particle swarm based data mining algorithms for classification tasks [C]. Parallel Computing, 2004: 767-783.
- [33] Donelli M, Azaro R, Natale FGB, et al. An innovative computational approach based on a particle swarm strategy for adaptive phased-arrays control[J]. IEEE Trans Antennas and Propagation, 2006, 54(3): 888-898.
- [34] Zhong W L, Huang J, Zhang J. A novel particle swarm optimization for the Steiner tree problem in graphs[C]. HongKong: IEEE Congress on Evolutionary Computation, 2008: 2460-2467.
- [35] Zhong W L, Zhang J, Chen W N. A novel discrete particle swarm optimization to solve traveling salesman problem[C]. Singapore: IEEE Congress on Evolutionary Computation, 2007: 3283-3287.
- [36] Zhan Z H, Zhang J. Discrete particle swarm optimization for multiple destination routing problems [C]. Evo Workshops, 2009: 117-122.
- [37] 王凌, 刘波. 微粒群优化与调度算法[M]. 北京: 清华大学出版社, 2008: 8-10.
- [38] 汪定伟, 王俊伟, 王洪峰, 等. 智能优化方法[M]. 北京: 高等教育出版社, 2007: 233-236.
- [39] 高海兵, 周驰, 高亮. 广义粒子群优化模型 [J]. 计算机学报, 2005, 28(12): 1980-1987.