

基于子区域的粒子群优化算法研究

曾嘉俊^{1,2}, 刘志刚¹, 黄元亮², 刘新东²

(1. 西南交通大学电气工程学院, 成都 610031; 2. 暨南大学电气自动化研究所, 广东 珠海 519070)

摘 要: 针对粒子群优化(PSO)算法在寻优时容易陷入局部最优的不足, 提出一种基于子区域的 PSO 算法。将搜索空间划分成若干个子区域, 在各个子区域中均使用标准 PSO 算法进行寻优, 通过比较各个子区域的全局最优解, 从而得出整个搜索空间的全局最优。与标准 PSO 算法及自适应变异 PSO 算法的比较结果表明, 该算法能降低在寻优过程中陷入局部最优的概率, 具有较强的寻优能力。

关键词: 粒子群优化算法; 局部最优; 全局最优; 子区域

Research of Particle Swarm Optimization Algorithm Based on Sub-region

ZENG Jia-jun^{1,2}, LIU Zhi-gang¹, HUANG Yuan-liang², LIU Xin-dong²

(1. School of Electric Engineering, Southwest Jiaotong University, Chengdu 610031, China;

2. Electric Automation Institute, Jinan University, Zhuhai 519070, China)

【Abstract】 Aiming at problem that Particle Swarm Optimization(PSO) algorithm falls into local optimum easily, this paper presents a PSO algorithm based on sub-region. It makes the search space some sub-region, uses the PSO algorithm to optimize in each region, compares these sub-region global optimums and finds out the search space global optimums. Results compared with standard PSO algorithm and adaptive mutation PSO algorithm show that this algorithm can reduce the probability of optimizing which falls into local optimum, and it has strong optimization ability.

【Key words】 Particle Swarm Optimization(PSO) algorithm; local optimum; global optimum; sub-region

DOI: 10.3969/j.issn.1000-3428.2011.14.069

1 概述

粒子群优化算法(Particle Swarm Optimization, PSO)是智能优化算法中的一种。它的优点是收敛速度快、容易实现并且需要的参数少。目前, PSO 算法已被广泛应用于目标函数优化、动态环境优化、神经网络训练等领域^[1-6]。但是, PSO 算法发展与研究时间尚短, 在理论上存在明显的缺陷, 即当 PSO 应用于高维复杂的问题优化时, 会造成过早收敛、容易陷入局部最优的现象。因此, 不能保证算法收敛到全局最优。这是因为 PSO 算法早期收敛速度较快, 到了后期其结果缺乏有效的机制使算法逃离极小点^[2-3]。综上所述, 如何提高 PSO 算法搜索出全局最优解的概率是很重要的。本文阐述了传统粒子群优化算法, 并指出该算法的局限性。针对其局限性, 提出一种基于子区域的粒子群优化算法, 并应用到实例中进行验证。

2 标准粒子群优化算法

标准粒子群算法^[4]由传统粒子群算法^[1]改进而来。其数学描述如下: 设搜索空间为 D 维, 总粒子数为 n , 第 i 个粒子位置表示为 $x_i=(x_{i1}, x_{i2}, \dots, x_{iD})$, 第 i 个粒子迄今为止搜索到的最优位置为 $pbest=(p_{i1}, p_{i2}, \dots, p_{iD})$, 整个粒子群迄今为止搜索到的最优位置为 $gbest=(g_1, g_2, \dots, g_D)$, 第 i 个粒子的位置变化率(速度)为向量 $v_i=(v_{i1}, v_{i2}, \dots, v_{iD})$ 。每个粒子的每一维速度和位置按式(1)和式(2)进行变化^[3]:

$$v_{id}(t+1)=\omega(t) \times v_{id}(t)+c_1 \times r_1 \times (p_{id}(t)-x_{id}(t))+c_2 \times r_2 \times (p_{gd}(t)-x_{id}(t)) \quad (1)$$

$$x_{id}(t+1)=x_{id}(t)+v_{id}(t+1) \quad 1 \leq i \leq n, 1 \leq d \leq D \quad (2)$$

其中, c_1 、 c_2 为正常数, 称为加速因子, c_1 调节粒子飞向自身最好位置方向的步长, c_2 调节粒子飞向全局最好位置方向;

r_1 、 r_2 为 $[0,1]$ 之间的随机数。

目前, 采用较多的惯性权重是线性递减权值^[5], 即:

$$\omega(t)=(\omega_{ini}-\omega_{fin})(T_{max}-t)/T_{max}+\omega_{fin} \quad (3)$$

其中, T_{max} 为最大进化代数; ω_{ini} 为初始惯性权值; ω_{fin} 为进化至最大代数时的惯性权值。典型取值 ω_{ini} 取 0.9, ω_{fin} 取 0.4。

3 基于子区域的粒子群优化算法

3.1 粒子群优化算法的局限性

粒子群优化算法最初的定义是目标粒子(即被影响的粒子)根据自身的最优位置和邻居集(在拓扑结构中目标粒子相邻的所有粒子的总和)中最优个体的位置动态调整自身的搜索速度, 在解空间中寻优。在传统 PSO 算法中, 粒子逐一搜索它的邻居以确定最优解粒子, 从而用最优解粒子的信息校正自己的搜索路线。

标准 PSO 算法是建立在以下假设上^[6]: 在 t 时刻找出的最优邻居, 比其他所有邻居都处于一个更好的搜索区域, 这个区域包含或更靠近真正的全局最优解。然而, 这个假设未必成立, 因为这时的最优邻居未必最靠近全局最优解, 有可能处于一个局部极值的周围, 只是其适应值暂时处于最小, 从而被认定为最优粒子。而真正靠近全局最优解的也许是适

基金项目: 国家自然科学基金资助项目(51007074); 教育部新世纪优秀人才支持计划基金资助项目(NECT-08-0825); 铁道部科技研究开发计划基金资助项目(2009G009-C)

作者简介: 曾嘉俊(1987—), 男, 硕士研究生, 主研方向: 粒子群优化算法; 刘志刚, 教授、博士; 黄元亮, 副教授、博士; 刘新东, 讲师、博士

收稿日期: 2011-02-25 **E-mail:** tod105@163.com

应值排在第 2 位乃至更后面的粒子。仅凭单个“最优邻居”的引导,而忽略其他邻居,则信息来源单一,搜索区域的真实信息可能被遗漏。

传统 PSO 算法局限性示意图如图 1 所示,这是一个求解一维函数最小值的例子,4 号粒子为全局最优解,2 号和 3 号粒子适应值都比 1 号粒子小。从图 1 可以看出,粒子陷入局部最优,因此,1 号粒子被误认为全局最优解。

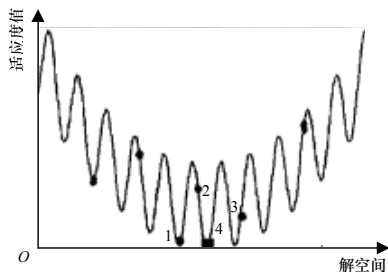


图 1 传统 PSO 算法局限性示意图

由图 1 可以知道,对于复杂的解空间,传统的 PSO 算法通常容易陷入局部极值,而误认为是全局最优。因此,有必要寻求一种降低陷入局部极值的概率的改进 PSO 算法。

3.2 基于子区域的粒子群优化算法的基本思想

针对图 1 进行划分:在理想前提下,假设图 2 中 12 个区域的上下限均是目标函数中的极大值点。因此,在这 12 个区域中,均只含有一个最小值点。若对这 12 个区域均使用标准 PSO 算法进行寻优,将得到的 12 个最小值点进行一次比较,可以得到全局最优解。

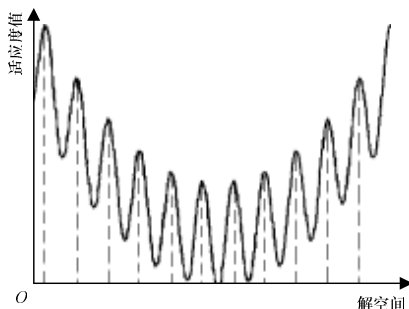


图 2 将解空间划分成多个区域的示意图

通过上述分析可以得到以下推断:若目标函数具有 N 个极小值点,则在理想情况下,若将目标函数划分成 $M(M \geq N)$ 个区域,且每个区域有且只有一个最小值。然后,将得到的 M 个最小值点进行比较,那么可以得到目标函数的全局最优解。

3.3 基于子区域的粒子群算法的数学模型

在实际生活中,目标函数通常很复杂,特别是在维数高的情况下,无法得知目标函数的极小值个数。但是,在实际情况中,极点数是有限的。因此,只要将区间划分足够多,那么可以将目标函数划分为若干个有且只有一个极小值的子区域。

对于 D 维的搜索空间,设第 i 维 x_i 的上下限为 $[x_{imin}, x_{imax}]$,将这个区间或分若干子区间, $[x_{imin}, x_{i1}], [x_{i1}, x_{i2}], \dots, [x_{in}, x_{imax}]$ 。其中, $x_{imin} \leq x_{i1} \leq x_{i2} \leq \dots \leq x_{in} \leq x_{imax}$ 。由上述可以得出,若将 $x_1, x_2, \dots, x_D$ 分别划分成 $N_1, N_2, \dots, N_D$ 个子区间,则这个搜索空间一共划分成 $N = \prod_{i=1}^D N_i$ 个子区域。

数学描述如下:设搜索空间为 D 维,总粒子数为 n ,第 i 个粒子位置表示为 $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ 。其中, $x_{i1}, x_{i2}, \dots, x_{iD}$

的范围分别为 $[x_{i1min}, x_{i1max}], [x_{i2min}, x_{i2max}], \dots, [x_{iDmin}, x_{iDmax}]$,并将 $x_1, x_2, \dots, x_D$ 分别划分成 $N_1, N_2, \dots, N_D$ 个子区间。

设在某个子区域 Z 中, $[x_{i1g}, x_{i1k}], [x_{i2g}, x_{i2k}], \dots, [x_{iDg}, x_{iDk}]$ 分别为 $[x_{i1min}, x_{i1max}], [x_{i2min}, x_{i2max}], \dots, [x_{iDmin}, x_{iDmax}]$ 的一个子区间。其中, $x_{i1}, x_{i2}, \dots, x_{iD}$ 分别属于 $[x_{i1g1}, x_{i1k1}], [x_{i2g2}, x_{i2k2}], \dots, [x_{iDgD}, x_{iDkD}]$ 。第 i 个粒子迄今为止搜索到最优位置为 $pbest_Z = (p_{i1}, p_{i2}, \dots, p_{iD})$,整个粒子群迄今为止搜索到最优位置为 $gbest_Z = (g_1, g_2, \dots, g_D)$,第 i 粒粒子的位置变化率(速度)为向量 $v_i = (v_1, v_2, \dots, v_D)$ 。对所有子区域均进行 PSO 算法后,将所有子区域的 $gbest_Z$ 进行比较,其中,最小的值为全局最优解 $gbest$ 。

根据式(1)与式(2),结合上述基于子区域的粒子群算法的思想,得到如下数学模型:

$$v_{id}(t+1) = \omega(t) \times v_{id}(t) + c_1 \times r_1 \times (p_{id}(t) - x_{id}(t)) + c_2 \times r_2 \times (p_{gd}(t) - x_{id}(t)) \quad (4)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad 1 \leq i \leq n, 1 \leq d \leq D \quad (5)$$

在式(4)和式(5)中, $x_{idg} \leq x_{id}(t) \leq x_{idk}$, $x_{idg} \leq x_{id}(t+1) \leq x_{idk}$ 。其中, $[x_{idg}, x_{idk}]$ 为 $[x_{imin}, x_{i1}], [x_{i1}, x_{i2}], \dots, [x_{in}, x_{imax}]$ 中的一个子区间。

基于子区域的粒子群优化算法的具体步骤如下:

(1)将搜索空间划分成 N 个子区域,分别为子区域 $r_1, r_2, \dots, r_N$ 。

(2)初始化, $k=1$ 。

(3)在子区域 r_k 中,进行初始化,对粒子群进行位置与速度的设定。设定学习因子 c_{k1}, c_{k2} ,最大进化代数 T_{kmax} ,当前进化代数 $t_k=1$ 。同时,将定义空间中随机产生 m 个粒子 $x_{k1}, x_{k2}, \dots, x_{km}$,即种群 $X_k(t)$ 。同样,随机产生粒子位移变化 $v_{k1}, v_{k2}, \dots, v_{km}$,即位移变化矩阵 $V_k(t)$ 。

(4)评价种群。计算每个粒子的适应值 $F_k(X_i)$ 。

(5)比较粒子当前的适应值 $F_k(X_i)$ 和自身历史最优值 $pbest_k$ 。如果 $F_k(X_i)$ 优于 $pbest_k$,则令 $pbest_k$ 为当前的适应值 $F_k(X_i)$,并令 $pbest_k$ 的位置 $pbest_k$ 为 n 维空间中当前的位置。

(6)比较粒子当前的适应值 $F_k(X_i)$ 和种群最优值 $gbest_k$ 。如果 $F_k(X_i)$ 优于 $gbest_k$,则令 $gbest_k$ 为当前的适应值 $F_k(X_i)$, $gbest_k$ 对应的序号为当前粒子序号。

(7)按照式(4)与式(5)更新粒子速度和位置,产生新种群 $X_k(t+1)$ 。

(8)检查适应值是否达到精度。如果达到或进化代数达到 T_{kmax} ,则循环结束,到步骤(9);否则, $t=t+1$,返回步骤(4)。

(9)检查 k 是否大于 N 。如果大于 N ,则循环结束,到步骤(10);否则, $k=k+1$,返回步骤(3)。

(10)将 N 个子区域中的种群最优值 $gbest_k$ 进行比较,得出全局最优解 $gbest$ 。

4 实例分析

为检验本文给出的基于子区域的粒子群优化算法的有效性,将标准粒子群算法、自适应变异粒子群算法以及基于子区域的粒子群算法分别应用到同一个目标函数中。

目标函数如下:

$$\min f(x) = -c_1 \times \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{j=1}^n x_j^2}\right) - \exp\left(\frac{1}{n} \sum_{j=1}^n \cos(2\pi x_j)\right) + c_1 + e \quad x_j \in [-5, 5], j=1, 2, \dots, n \quad (6)$$

这是一个无约束优化问题,其中, $c_1=20$; $e=2.718\ 28$ 。若 $n=2$,则全局最优在 $(0,0)$ 处,此时 $f(0,0)=0$ 。该函数称为 Ackley 函数,该函数有着许多局部最优。图 3 为 Ackley 函数的示意图。

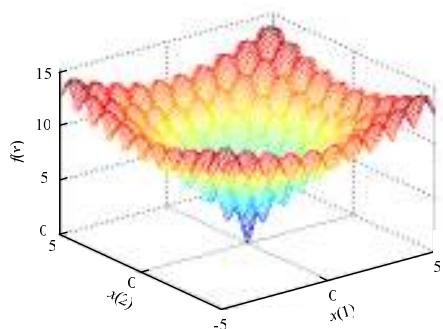


图3 Ackley函数示意图

在本算例中, 对于标准粒子群算法以及自适应变异粒子群算法, 群体个数设置为 20, 加速因子 c_1 、 c_2 取 1.494 45, 最大迭代次数取 200。另外, 自适应变异粒子群算法的粒子有 20% 的概率变异成一个随机值。对于基于子区域的粒子群优化算法, 群体个数设置为 20, 加速因子 c_1 和 c_2 取 1.494 45, 最大迭代次数取 50。将 x_1 划分成 $[-5, 0]$ 、 $[0, 5]$ 2 个子区间, x_2 划分成 $[-5, 0]$ 、 $[0, 5]$ 2 个子区间, 即将搜索空间分为 4 个子区域。对于 3 种算法, 权重均按式(3)计算得出, 其中, ω_{ini} 取 0.9; ω_{fin} 取 0.4。粒子速度的范围均是 $[-1, 1]$ 。由上述可以得出, 在搜索过程中, 3 种算法的群体个数和最大迭代次数的设置对于两者而言是公平的, 只是基于子区域的粒子群优化算法将粒子散落到 4 个子区域中。程序在 Matlab7.0 中运行, 具体过程如下:

(1) 运用标准粒子群算法对目标函数进行寻优。结果为: 全局最优解在 $(0.040\ 7, 0.029\ 8)$ 处, 即 $x_1=0.040\ 7$ 、 $x_2=0.029\ 8$ 。此时, 全局最优 g_{best} 为 0.209 2。标准粒子群算法的适应度曲线如图 4 所示。

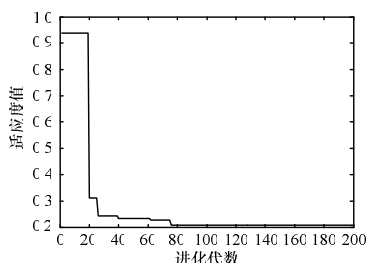


图4 标准粒子群优化算法的适应度曲线

(2) 运用自适应变异粒子群算法对目标函数进行寻优, 结果为: 全局最优解在 $(0.013, -0.018\ 9)$ 处, 即 $x_1=0.013$ 、 $x_2=-0.018\ 9$ 。此时, 全局最优 g_{best} 为 0.076 9。自适应变异粒子群算法的适应度曲线如图 5 所示。

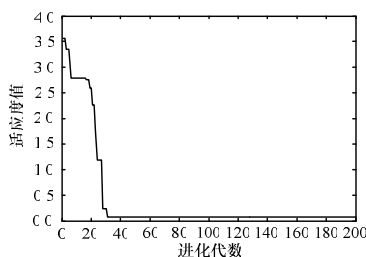
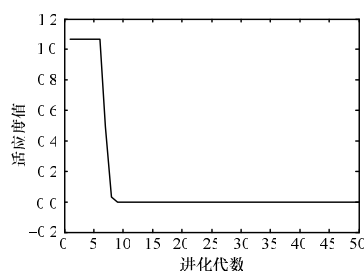


图5 自适应变异粒子群优化算法的适应度曲线

(3) 运用基于子区域粒子群算法对目标函数进行寻优, 结果为: 当 x_1 、 x_2 的范围为 $[-5, 0]$ 时, 基于子区域粒子群算法寻优后得出全局最优位置在 $(0, 0)$ 处, 即 $x_1=0$ 、 $x_2=0$ 。此时, 全局最优 g_{best1} 为 $-1.828\ 5 \times 10^{-6}$ 。子区域都为 $[-5, 0]$ 时本文算法的适应度曲线如图 6 所示。

图6 子区域都为 $[-5, 0]$ 时本文算法的适应度曲线

此后, 子区域分别为 $[-5, 0]$ 和 $[0, 5]$ 、 $[0, 5]$ 和 $[-5, 0]$ 以及 $[0, 5]$ 和 $[0, 5]$ 时, 基于子区域的粒子群优化算法寻优后得出全局最优位置均在 $(0, 0)$ 处, 即 $x_1=0$ 、 $x_2=0$ 。此时, 各个子区域全局最优均为 $-1.828\ 5 \times 10^{-6}$ 。限于篇幅, 只罗列出子区域分别为 $[-5, 0]$ 和 $[-5, 0]$ 时的适应度曲线。通过在 4 个子区域进行寻优后, 将 4 个子区域的最优解进行比较, 得到目标函数的全局最优解在 $(0, 0)$ 处, 全局最优 g_{best} 为 $-1.828\ 5 \times 10^{-6}$ 。

理论上, 目标函数在 $(0, 0)$ 处的值为 0。由于自然对数 e 的取值以及程序运算的精度等原因, 导致运行结果不为 0。但是 $-1.828\ 5 \times 10^{-6}$ 非常小, 因此可以将其近似看成 0。

通过将 3 种算法分别应用到上述算例中, 可以看出, 在群体个数和最大迭代次数的公平设置下, 基于子区域的粒子群优化算法的寻优能力比前面 2 种算法好。这主要是因为基于子区域的粒子群优化算法能够将粒子尽可能分布在搜索空间的各个区域中, 降低陷入局部最优的概率, 从而保证粒子尽可能寻找出全局最优。因此, 基于子区域的粒子群优化算法具有更强的寻优能力, 能保证粒子尽可能不陷入局部最优并找出全局最优。但由于分开多次寻优, 因此基于子区域的粒子群优化算法所需时间比前面 2 种算法长。

5 结束语

本文提出的基于子区域的粒子群优化算法是将搜索空间划分成若干个子区域, 在各个子区域中均使用标准 PSO 算法进行寻优, 再比较各个子区域的全局最优解得出全局最优。对于维数高、局部极值多的目标函数, 基于子区域的粒子群优化算法优势更明显。需要指出的是, 在实际情况当中, 目标函数通常非常复杂, 如何合理划分子区域以及设置最大迭代次数和群体个数, 尽可能使基于子区域的粒子群优化算法实现高效寻优是今后需要研究的内容。

参考文献

- [1] Angeline P J. Evolutionary Optimization Versus Particle Swarm Optimization: Philosophy and Performance Differences[C]//Proc. of the 7th Annual Conference on Evolutionary Programming. Washington D. C., USA: [s. n.], 1998: 256-260.
- [2] 焦巍, 刘光斌. 基于多样性反馈的粒子群优化算法[J]. 计算机工程, 2009, 35(22): 202-204.
- [3] 马翠, 周先东, 杨大地. 分层粒子群优化算法[J]. 计算机工程, 2009, 35(20): 194-196.
- [4] Shi Yuhui, Eberhar R. A Modified Particle Swarm Optimizer[C]//Proc. of International Conference on Evolutionary Computation. [S. l.]: IEEE Press, 1998.
- [5] Shi Yuhui, Eberhart R C. Parameter Selection in Particle Swarm Optimization[C]//Proc. of the 7th Annual Conference on Evolutionary Programming. New York, USA: Springer-Verlag, 1998: 591-600.
- [6] 方峻. 粒子群算法及其应用研究[D]. 成都: 电子科技大学, 2006.

编辑 陆燕菲