

文章编号 :1002-0411 (2004) 05-0513-05

# 粒子群优化算法的分析与改进

张丽平,俞欢军,陈德钊,胡上序

(浙江大学智能信息工程研究所,浙江 杭州 310027)

**摘要:**分析了惯性权值对粒子群优化(PSO)算法优化性能的影响,进而提出选择惯性权值的新策略.在随机选取惯性权值的同时,自适应地调整随机惯性权值的数学期望,有效地调整算法的全局与局部搜索能力.测试表明基于随机惯性权(RIW)策略的 PSO 算法,其全局搜优的速率与精度有明显提高.\*

**关键词:**粒子群;优化算法;随机;惯性权;策略

**中图分类号:** TP18

**文献标识码:** A

## Analysis and Improvement of Particle Swarm Optimization Algorithm

ZHANG Li-ping, YU Huan-jun, CHEN De-zhao, HU Shang-xu

(Institute of Intelligent Information Engineering, Zhejiang University, Hangzhou 310027, China)

**Abstract:** The effects of inertia weight on particle swarm optimization (PSO) performance are analyzed. A novel method of selecting inertia weight in PSO is developed, which can tune the expectations of inertia weights adaptively when the inertia weights are randomly selected and lead to effectively balance between the local and global search ability. Results of the two benchmark functions indicate that the PSO algorithm based on the strategy of random inertia weight (RIW) has been significantly improved on both optimization speed and computational accuracy.

**Keywords:** particle swarm; optimization algorithm; random; inertia weight; strategy

## 1 引言(Introduction)

优化是科学研究、工程技术和经济管理等领域的重要研究课题.最优化问题常可表示为实函数的全局寻优,即:

$$f(x^*) = \min_{x \in R^n} f(x), \quad \forall x \in R^n \text{ s. t. } g(x) \leq 0 \quad (1)$$

其中  $x$  为  $n$  维定义空间  $R^n$  中的向量,或视为该空间的点,  $x^*$  为搜寻空间的全局最优点,  $f(x)$  是目标函数,  $g(x)$  为约束条件集合.最优化算法通常可分为确定性和随机性两种.确定性算法按函数变化的机理寻优,搜索速率较高,过程可再现,其不足在于寻优结果与初值有关,一般只能搜得局部最优.因此开发能实现全局优化的算法一直是多方关注的重要课题.近年来,从自然界很多自适应优化现象中得到启示,人们研究开发了基于随机搜索的模拟退火、进化计算等现代优化技术,在理论与应用方面都有长足发展.

粒子群优化(Particle Swarm Optimization, PSO)算法也为其中之一<sup>[1~3]</sup>,是由 Kennedy 和 E-

berhart 通过对鸟群、鱼群和人类社会某些行为的观察研究,于 1995 年提出的一种新颖的进化算法.鉴于 PSO 的发展历史尚短,它在理论基础与应用推广上都还存在一些问题,有待解决.本文拟针对基本 PSO 算法求解多峰函数优化问题时易陷入局部极小的缺点,提出一种新的惯性权选取策略,即随机惯性权(Random Inertia Weight, RIW)策略,以期对基本 PSO 算法做出改进.

## 2 基本粒子群优化算法(Standard PSO algorithm)

与其它进化算法相类似,PSO 算法也将通过个体间的协作与竞争,实现复杂空间中最优解的搜索. PSO 先生成初始种群,即在可行解空间中随机初始化一群粒子,每个粒子都为优化问题的一个解,并由目标函数为之确定一个适应值(Fitness Value).每个粒子将在解空间中运动,并由一个速度决定其方向和距离.通常粒子将追随当前的最优粒子而动,并

\* 收稿日期:2004-02-23  
基金项目:国家自然科学基金资助项目(20076041)

经逐代搜索,最后得到最优解.在每一代中,粒子将跟踪两个极值,一个为粒子本身迄今找到的最优解,代表粒子自身的认知水平;另一个为全种群迄今找到的最优解,代表社会认知水平<sup>[3]</sup>.

设第  $i$  个粒子位置为  $x_i = (x_{i1}, x_{i2}, \dots, x_{in})^T$ , 其速度为  $v_i = (v_{i1}, v_{i2}, \dots, v_{in})^T$ . 它的个体极值为  $p_i = (p_{i1}, p_{i2}, \dots, p_{in})^T$ , 种群的全局极值为  $p_g = (p_{g1}, p_{g2}, \dots, p_{gn})^T$ . 按追随当前最优粒子的原理, 粒子  $x_i$  将按(2)、(3)式改变速度和位置.

$$v_{id}(t+1) = v_{id}(t) + c_1 r_1 (p_{id}(t) - x_{id}(t)) + c_2 r_2 (p_{gd}(t) - x_{id}(t)) \quad (2)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (3)$$

其中  $d=1, 2, \dots, n, i=1, 2, \dots, S$ ,  $S$  为种群规模,  $r_1$  和  $r_2$  为均匀分布于  $[0, 1]$  之间的随机数,  $c_1$  和  $c_2$  为加速度限定因子,通常取为 2. 为使粒子速度不致过大,可设定速度上限  $V_{\max}$ , 即当(2)式的  $|v_{id}| > V_{\max}$  时,取  $|v_{id}| = V_{\max}$ ; 此后, Shi Y 又增设了惯性权因子  $w$ , 将(2)式改变为:

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

由(3)、(4)式组成的迭代算法被认为是基本 PSO 算法. 对于惯性权因子, Shi Y 建议采用线性递减权 (Linearly Decreasing Weight, LDW) 策略<sup>[4,5]</sup>, 即:

$$w(t) = (w_{\text{ini}} - w_{\text{end}})(T_{\max} - t) / T_{\max} + w_{\text{end}} \quad (5)$$

其中  $t$  为当前进化代数,  $T_{\max}$  为最大进化代数,  $w_{\text{ini}}$  为初始惯性权值,  $w_{\text{end}}$  为进化至最大代数时的惯性权值. 试验表明权值  $w$  将影响 PSO 的全局与局部搜优能力<sup>[4]</sup>,  $w$  值较大, 全局搜优能力强, 局部搜优能力弱, 反之, 则局部搜优能力增强, 而全局搜优能力减弱. 线性惯性权的引入使 PSO 可以调节算法的全局与局部搜优能力, 但还有两个缺点: 其一, 迭代初期局部搜索能力较弱, 即使初始粒子已接近于全局最优点, 也往往错过, 而在迭代后期, 则因全局搜索能力变弱, 而易陷入局部极值. 其二, (5) 式的最大迭代次数  $T_{\max}$  较难预测, 从而将影响算法的调节功能. 最近 Shi Y 又提出采用模糊系统调节权值, 需要专家知识建立模糊规则<sup>[6]</sup>, 在复杂系统被优化前, 领域知识往往贫乏而难以获取, 故其实现比较复杂.

### 3 粒子群优化算法收敛性分析 (Convergence analysis of PSO algorithm)

本节将推导惯性权值与粒子速度的关系, 分析它对 PSO 算法收敛性的影响, 以改进算法的寻优性能. 为简便计, 不妨设自变量空间为 1 维的, 且种群只有 1 个粒子, 此时将有  $p_i = p_g$ . 又令  $v_1 = r_1 c_1$ ,  $v_2 = r_2 c_2$ ,  $v = v_1 + v_2$ , 记  $y_i = p_i - x_i$ , 则(3)、(4)式将简化为:

$$y(t+1) = -w v(t) + (1 - w) y(t) \quad (6)$$

$$v(t+1) = w v(t) + y(t) \quad (7)$$

对(6)、(7)式进行迭代消去  $y$  可得:

$$v(t+2) + (1 - w) v(t+1) + w v(t) = 0 \quad (8)$$

将(8)式连续化, 可导出二阶微分方程:

$$\frac{\partial^2 v}{\partial t^2} + \ln(e_1 e_2) \frac{\partial v}{\partial t} + \ln(e_1) \ln(e_2) v = 0 \quad (9)$$

其中  $e_1$  和  $e_2$  为二次方程(10)的根.

$$v^2 + (1 - w) v + w = 0 \quad (10)$$

即  $e_1 = \frac{w+1-\sqrt{(w+1)^2-4w}}{2}$ ,  $e_2 = \frac{w+1+\sqrt{(w+1)^2-4w}}{2}$ , 式中  $\Delta = (w+1)^2 - 4w$ . 故

微分方程(9)的通解为:

$$v(t) = k_1 e_1^t + k_2 e_2^t \quad (11)$$

于是有:

$$y(t) = [k_1 e_1^t (e_1 - w) + k_2 e_2^t (e_2 - w)] / \quad (12)$$

其中  $k_1 = \frac{-v(0) - (w - e_2) v(0)}{e_2 - e_1}$ ,  $k_2 = \frac{v(0) + (w - e_1) v(0)}{e_2 - e_1}$ .

(11)和(12)式反映了粒子速度和位移在进化中的演变情况, 这种演变又与  $e_1$  和  $e_2$ , 亦即与  $w$  和  $\Delta$  密切相关. 显然在  $\max(|e_1|, |e_2|) < 1$ ,  $t \rightarrow \infty$  时, 有  $v(t), y(t) \rightarrow 0$ , 即算法收敛. 若  $w > 0.5(1 + \sqrt{\Delta}) - 1$ , 则有  $\max(|e_1|, |e_2|) < 1$ ; 而在  $w > 0.5(c_1 + c_2) - 1$  时, 必有  $w > 0.5(1 + \sqrt{\Delta}) - 1$  成立, 故算法一定收敛.

基于上述思想, Clerc M 引入限定因子, 构成 PSO 的限定因子法 (Constriction Factor Method, CFM)<sup>[7]</sup> 如(13)式所示, 以保证算法收敛.

$$v_{id} = (v_{id} + \omega_1 (p_{id} - x_{id}) + \omega_2 (p_{gd} - x_{id})) \quad (13)$$

其中  $\omega$  可用(14)式确定:

$$= \frac{2}{\sqrt{2 - \sqrt{2 - 4}}} \quad (14)$$

式中  $\omega > 4$ . CFM 法一般选择  $\omega$  为 4.1, 则  $\omega$  为 0.7298, 这相当于  $\omega$  为定值 0.7298,  $c_1 = c_2 = 1.49445$  的基本 PSO 方法<sup>[8]</sup>, 此时 CFM 法收敛. 而采用 LDW 策略时,  $\omega$  和  $(c_1 + c_2)$  都是可变的, 将不一定完全满足收敛条件.

若 PSO 算法过早收敛, 如 (11) 与 (12) 式所示, 粒子速度将下降至 0, 粒子群将趋于当前的极值  $p_i$  与  $p_g$ , 而它们往往为局部极值, 尚未达到全局最优. 因此对 PSO 算法的改进不能着眼于收敛性, 而应调节算法的搜索范围, 以及全局和局部搜索能力. 鉴于惯性权值对粒子速度的影响, 本文提出随机地选择惯性权值的策略 RIW, 以提高 PSO 算法的全局搜优性能与效率.

#### 4 随机惯性权(RIW)策略(Random inertia weight strategy)

Shi Y 提出的 LDW 策略也是通过惯性权值的线性下降调节全局与局部搜索能力, 由于惯性权值的这种变化较为单一, 因此对搜索能力的调节也就有限, 使其搜索过程不一定能适应实际问题的复杂情况. 本文提出的 RIW 策略, 意在改变这种单一的调节模式, 使之较好地适应复杂的实际环境. 为此, 先定义最优适应值变化率  $k$ , 如式 (15) 所示:

$$k = \frac{f(t) - f(t-10)}{f(t-10)} \quad (15)$$

其中  $f(t)$  为种群在第  $iter$  代时的最优适应值,  $f(t-10)$  为种群在第  $(t-10)$  代时的最优适应值. 则  $k$  表示在进化 10 代内最优适应值的相对变化率, 惯性权  $\omega$  将按 (16) 式取随机值, 且其数学期望值将随  $k$  而变.

$$\begin{cases} \omega = c_1 + r/2.0, & k \geq 0.05 \\ \omega = c_2 + r/2.0, & k < 0.05 \end{cases} \quad (16)$$

其中  $r$  为均匀分布于  $[0, 1]$  之间的随机数, 当  $k \geq 0.05$  时, 期望值  $E(\omega) = c_1 + 0.25$ ; 当  $k < 0.05$  时,  $E(\omega) = c_2 + 0.25$ , 且令  $c_1 > c_2$ . 该选择策略有两个特点: 其一, 随机地选取  $\omega$  值, 由 (4) 式可见, 这将使粒子历史速度对当前速度的影响为随机的; 其二,  $\omega$  的数学期望值将随最优适应值的变化率自适应地调整, 从而可以更灵活地调节全局搜索与局部搜索能力. 另外, 惯性权取值的随机性, 在一定程度上与遗传算法的变异算子较为相似<sup>[9]</sup>, 这将有助于保持种

群的多样性.

对粒子速度按 (11) 式进行仿真, 初始速度取为  $v(0) = 5.0$ ,  $v(1) = 5\omega + 5$ . 惯性权值  $\omega$  分别采用 RIW 与 LDW 两种策略, 即分别按 (16) 与 (5) 式取值, 各参数选择为:  $c_1 = 0.5$ ,  $c_2 = 0.4$ ,  $\omega_{ini} = 0.9$ ,  $\omega_{end} = 0.4$ ,  $T_{max} = 100$ ,  $c_1$  和  $c_2$  分别取均匀分布于  $[0, 2]$  之间的随机数. 粒子速度演变的曲线示于图 1 与图 2.

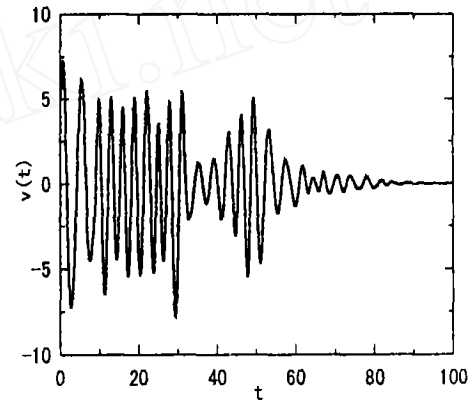


图 1 RIW 法速度进化过程

Fig. 1 Evolution process of velocity using RIW strategy

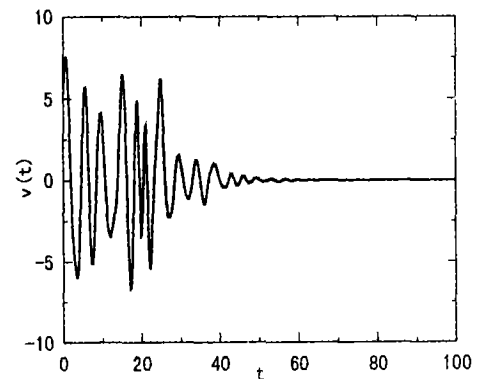


图 2 LDW 法速度进化过程

Fig. 2 Evolution process of velocity using LDW strategy

从图 1 和图 2 可以看出, LDW 算法的进化初期粒子速度较大, 粒子群的搜索范围较广, 即全局搜优能力强, 但影响了局部区域的搜优能力. 在进化代数达到 40 时, 粒子速度明显减小, 粒子群将对局部区域进行较强的搜索, 而全局搜索能力大为减弱. 最后, 粒子速度逐渐衰减到 0, 搜优进程相应终止. 该图也反映出: 基于本文导出的 (11) 式所得结果与 Shi Y 等的试验观测基本一致. 与此相对照, RIW 算法的粒子速度并不单调地下降, 而是有所起伏, 在进化初期粒子速度也较大, 当进化至 30 代时, 速度逐

渐衰减,在进化超过 40 代后速度又开始增加,然后又逐渐衰减. 这样的起伏特性在 PSO 进化中有重要意义,使算法的全局搜索与局部搜索在更长的进程中均衡地展开,尤其对于多峰函数的寻优更为有效.

5 性能测试(Performance test)

本文拟用 2 个经典的测试函数: Rosenbrock 函数和 Rastrigin 函数,对 RIW、LDW 和 CFM 三种 PSO 算法的寻优性能进行测试,并作比较.

Rosenbrock 函数是一个单峰函数,各变量之间有很强的耦合性. 它的全局极小点在  $x^* = (1, 1, \dots, 1)$  处,全局极小值  $f(x^*) = 0$ .

$$f(x) = \sum_{d=1}^n (100(x_{d+1} - x_d^2)^2 + (x_d - 1)^2) - 10 < x_d < 10 \tag{17}$$

Rastrigin 函数是一个多峰函数,有很多正弦凸起的局部极小点,各变量相互独立. 它的全局极小点在  $x^* = (0, 0, \dots, 0)$  处,全局极小值  $f(x^*) = 0$ .

$$f(x) = \sum_{d=1}^n (x_d^2 - 10\cos(2x_d) + 10) - 5.12 < x_d < 5.12 \tag{18}$$

对上述两个测试函数均取 10 维进行测试,三种算法种群粒子数取 30. RIW 和 LDW 的速度上限  $V_{\max}$  取各函数初始范围的上限,  $c_1 = c_2 = 2.0$ . 对 RIW:  $\omega_1 = 0.5, \omega_2 = 0.4$ . 对 LDW:  $w_{\text{ini}} = 0.9, w_{\text{end}} = 0.4$ . 对 CFM:  $\omega_1 = \omega_2 = 2.05, \omega_3 = 0.7298$ .

测试以两种方式进行,先给定寻优应达到的精度  $\epsilon = 5.0$ ,即  $|F_{\text{best}} - F^*| < \epsilon$ ,其中  $F_{\text{best}}$  为算法搜优达到的适应值,  $F^*$  为全局极小值,试验三种算法所需的进化代数. 对三种算法分别进行 50 次试验,对 LDW 设定  $T_{\max} = 10000$ ,表 1 给出了三种算法所需进化代数的平均值. 其次,给定最大的进化代数为 2000,试验三种算法寻优所达到的精度. 对三种算法亦分别进行 50 次试验,表 2 给出了三种算法所达到精度的平均值.

表 1、表 2 的数据表明,对于单峰 Rosenbrock 函数,CFM 法的优化性能最强,在给定的精度下,它需要的进化代数最少,而在给定的进化代数下,它所达到的精度最高. 它的搜优性能强,缘于它的收敛性好,能迅速有效地靠近函数唯一的最优值. 与其相比,RIW 法的优化性能次之,LDW 法的最差,这归因于 LDW 法要求预先设定最大进化代数,若预定过大,惯性权值下降很慢,将影响其局部搜索能力. 对于存在多个局部极小值的 Rastrigin 函数,CFM

法的优化性能将变得极差,50 次试验中有 47 次进化了 10000 代,也未达到规定的精度. 同时,在给定的进化代数下,它的寻优精度也相当差. 这表明收敛性好的 CFM 法,全局搜优性能很弱. 对此,RIW 法和 LDW 法因惯性权值的变化,具有跳出局部极小的能力,尤其是 RIW 法,其优化性能更强,效率更高,它对粒子群全局搜索和局部搜索能力的调节最有效.

表 1 给定精度下各方法所需进化代数比较

Tab. 1 Comparison of evolution generations among different methods at certain accuracy

函数名称	RIW	LDW	CFM
Rosenbrock	1185	4439	387
Rastrigin	826	3646	9407 *

\*在进化了 10000 代后,仍有 47 次未达到给定精度.

表 2 给定进化代数时各方法达到精度比较

Tab. 2 Comparison of accuracy among different methods at certain evolution generations

函数名称	RIW	LDW	CFM
Rosenbrock	3.11101	3.16685	0.72004
Rastrigin	2.28843	3.08526	13.11356

此外,还应用三种方法对 30 维的 Rosenbrock 函数和 Rastrigin 函数进行搜优试验,图 3 和图 4 为其最优适应值随进化代数变化的曲线,从中也可以得到与上述相同的结论.

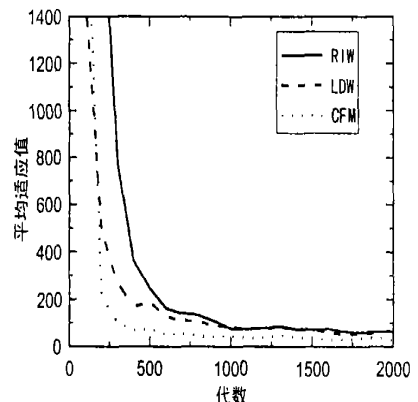


图 3 三种方法对 Rosenbrock 函数性能比较

Fig. 3 Performance comparison of Rosenbrock function among different methods

6 结论(Conclusion)

PSO 算法从生物群体的社会行为中得到启示,

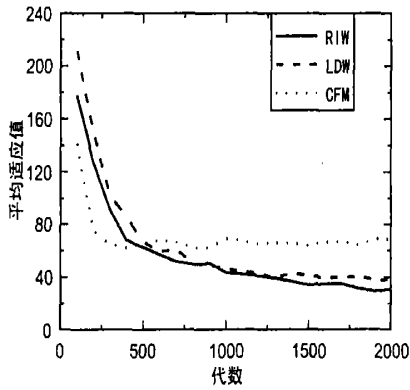


图4 三种方法对 Rastrigin 函数性能比较

Fig. 4 Performance comparison of Rastrigin function among different methods

是一种新颖的智能计算优化方法,其实现简易,效果良好,也有待于进一步改进和完善.本文导出了惯性权值与粒子速度的关系式,从数学的角度分析了 PSO 算法的优化性能,进而提出随机惯性权值选择策略,在随机选取惯性权值的同时,将自适应地调整随机惯性权的期望值,以此有效地调节算法的全局搜索与局部搜索能力,并保持种群多样性.通过经典函数的测试,显示出基于 RIW 策略的 PSO 法对多峰问题的优化十分有效,在优化速率与精度上明显优于其它 PSO 方法.

### 参 考 文 献 (References)

- [1] Kennedy J, Eberhart R C. Particle swarm optimization [A]. Proceedings of IEEE International Conference on Neural Networks [C]. Piscataway, NJ: IEEE Press, 1995. 1942 ~ 1948.
- [2] Eberhart R C, Kennedy J. A new optimizer using particle swarm theory [A]. Proceedings of the Sixth International Symposium on Micro Machine and Human Science [C]. Nagoya, Japan: IEEE Press, 1995. 39 ~ 43.
- [3] Eberhart R C, Simpson P K, Dobbins R W. Computational Intelligence PC Tools [M]. Boston, MA: Academic Press Professional, 1996.
- [4] Shi Y, Eberhart R C. A modified particle swarm optimizer [A]. Proceedings of the IEEE Congress on Evolutionary Computation [C]. Piscataway, NJ: IEEE Press, 1998. 303 ~ 308.
- [5] Shi Y, Eberhart R C. Empirical study of particle swarm optimization [A]. Proceedings of the IEEE Congress on Evolutionary Computation [C]. Piscataway, NJ: IEEE Press, 1999. 1945 ~ 1950.
- [6] Shi Y, Eberhart R C. Fuzzy adaptive particle swarm optimization [A]. Proceedings of the IEEE Congress on Evolutionary Computation [C]. Seoul, Korea: IEEE Press, 2001. 101 ~ 106.
- [7] Clerc M, Kennedy J. The particle swarm - explosion, stability, and convergence in a multidimensional complex space [J]. IEEE Transactions on Evolutionary Computation, 2002, 6 (1): 58 ~ 73.
- [8] Eberhart R C, Shi Y. Comparing inertia weight and constriction factors in particle swarm optimization [A]. Proceedings of the IEEE Congress on Evolutionary Computation [C]. San Diego, CA: IEEE Press, 2000. 84 ~ 88.
- [9] Miranda V, Fonseca N. EPSO-best-of-two-worlds meta-heuristic applied to power system problems [A]. Proceedings of the IEEE Congress on Evolutionary Computation [C]. Honolulu, Hawaii, USA: IEEE Press, 2002. 1080 ~ 1085.

### 作者简介

张丽平(1976 - ),男,博士研究生.研究领域为建模与优化,信息智能处理等.

俞欢军(1969 - ),男,博士,副教授.研究领域为计算机网络及应用,软件工程等.

陈德钊(1943 - ),男,教授,博士生导师.研究领域为神经网络,信息智能处理等.