

张彦超, 王晓丽, 苏奎. 一种增强型双种群粒子群算法的设计与实现[J]. 智能计算机与应用, 2024, 14(5): 194-198. DOI: 10.20169/j.issn.2095-2163.240526

# 一种增强型双种群粒子群算法的设计与实现

张彦超<sup>1</sup>, 王晓丽<sup>2</sup>, 苏奎<sup>1</sup>

(1 牡丹江医学院 医学影像学院, 黑龙江 牡丹江 157011; 2 牡丹江医学院 现代教育技术中心, 黑龙江 牡丹江 157011)

**摘要:** 粒子群算法作为一种具有收敛速度快、易于实现和参数调节少等优点的群体智能算法, 被广泛应用于函数与组合优化、机器学习等众多领域。在粒子群及其众多修改算法中惯性权重选择对算法性能起到较大的作用。为此, 在经典粒子群算法上针对惯性权重提出了改进策略, 设计了一种增强型双种群粒子群算法(EDUPSO)。当粒子在进化中将较小的惯性因子赋予到此次进化到最优位置的种群, 将较大的惯性因子赋予到此次没有进化到当前最优位的种群。通过此思路提出了算法的设计和实现方式, 并通过多个不同的测试函数分析了该算法与其他经典改进算法在性能上的差异, 通过测试的结果可以看出相对于经典的粒子群算法及经典改进算法, 此算法在搜索的速度、稳定性及精度上都有明显的优势。

**关键词:** 粒子群算法改进; 神经网络优化; 无线定位优化; 人工智能与数据挖掘

中图分类号: TP301.6; TP18

文献标志码: A

文章编号: 2095-2163(2024)05-0194-05

## Design and implementation of an enhanced dual-population Particle Swarm Optimization algorithm

ZHANG Yanchao<sup>1</sup>, WANG Xiaoli<sup>2</sup>, SU Kui<sup>1</sup>

(1 College of Medical Imaging, Mudanjiang Medical University, Mudanjiang 157011, Heilongjiang, China;

2 Modern Educational & Technical Center, Mudanjiang Medical University, Mudanjiang 157011, Heilongjiang, China)

**Abstract:** Particle Swarm Optimization is widely used in many fields such as function and combinatorial optimization, machine learning, etc., due to its advantages of fast convergence, easy implementation and less parameter adjustment. For PSO and its many modified algorithms, the selection of inertia weight has a significant impact on the algorithm's performance. Therefore, an improved strategy is proposed for the inertia weight in the classical PSO algorithm, which led to the development of Enhanced Dual-Particle Swarm Optimization (EDUPSO). In EDUPSO, smaller inertia factors are assigned to the population that evolves to the best position during evolution, while larger inertia factors are assigned to the population that did not reach the current optimal position. The algorithm's design and implementation approach are developed using this idea, and the algorithm's performance is analyzed in comparison to other classical improved algorithms using various test functions. The results of the tests show that, relative to the classical PSO algorithm and other classical improved algorithms, this algorithm has significant advantages in terms of search speed, stability and accuracy.

**Key words:** PSO optimization; neural network optimization; wireless positioning optimization; artificial intelligence and data mining

## 0 引言

粒子群(Particle Swarm Optimization, PSO)算法是一种以启发式搜索为主的群体智能算法<sup>[1-3]</sup>。目前粒子群算法在函数与组合优化<sup>[4]</sup>、机器学习<sup>[5]</sup>、自然语言的处理<sup>[6]</sup>、无线定位优化<sup>[7]</sup>、数据挖掘<sup>[8]</sup>等众多领域都有广泛的应用。受到鸟群捕食行为的

启发, 该算法将问题中的每个潜在解看作一个粒子, 并将其放置在问题的解空间中。每个粒子的位置表示解的位置, 速度表示解的更新方向。通过粒子间的信息交流和合作, 粒子们在解空间中搜索最优解。在搜索过程中, 粒子会记住自己找到的最优解, 并尝试在下一搜索中优化自己的位置和速度。与蚁群算法、遗传算法等计算智能的其他优化算法类似, 粒

**基金项目:** 牡丹江市应用技术与开发计划项目(HT2022JG130); 牡丹江医学院教育教学改革项目(MYPY20210009)。

**作者简介:** 张彦超(1972-), 男, 硕士, 讲师, 主要研究方向: 水下UVU协同算法, 智能影像工程; 王晓丽(1983-), 女, 硕士, 讲师, 主要研究方向: 深度学习, 数据挖掘。

**通讯作者:** 苏奎(1981-), 男, 硕士, 讲师, 主要研究方向: 智能网络。Email: suensk@163.com

收稿日期: 2023-04-11

子群算法中初始粒子的随机与进化的不确定性带来搜索结果上的不同。国内外学者对该算法进行了多种方式的改进,比如惯性权重线性递减<sup>[9]</sup>、惯性权重指数递减<sup>[10]</sup>、引导粒子朝向<sup>[11]</sup>、将差分进化的思路与粒子群算法结合<sup>[12]</sup>等。本文借鉴了上述针对惯性权重的修改思路,提出了一种增强型双种群粒子群算法(Enhanced Dual Particle Swarm Optimization, EDPSO),当粒子在进化中将较小的惯性因子赋予到此次进化到最优位置的种群,将较大的惯性因子赋予到此次未能进化到当前最优位的种群,通过不同惯性种群的相互协调来寻找最优解。在阐述算法前,首先介绍了经典的粒子群算法及2种用于对比的改进粒子群算法。在此基础上对算法进行了改进,并对增强型双种群粒子群算法加以研究及实现。文章最后还进行了对比实验。

## 1 经典粒子群算法及其改进算法

### 1.1 经典粒子群算法

经典粒子群算法(Basic Particle Swarm Optimization, BPSO)<sup>[1]</sup>主要包含3个部分:

(1)初始化。随机初始化一定数量的粒子,每个粒子表示搜索空间中的一个解,并为每个粒子赋予一个随机的速度和位置。

(2)粒子更新。每个粒子根据当前位置和速度,通过评估目标函数的值来计算其适应度,并更新其位置和速度。粒子的速度和位置更新受到自身历史最佳位置和整个粒子群历史最佳位置的影响。设有优化函数如下:

$$\begin{cases} \min f(x_1, x_2, \dots, x_D) \\ \text{s.t. } R_{1d} \leq x_d \leq R_{2d}, d = 1, 2, \dots, D \end{cases} \quad (1)$$

粒子群由  $n$  个随机生成的粒子构成,粒子位置  $x_d$  表示优化函数在  $D$  维搜索空间中的潜在解,则第  $i$  个粒子位置和速度的更新公式可依次写为:

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

$$x_{id} = x_{id} + v_{id} \quad i = 1, 2, \dots, n; d = 1, 2, \dots, D \quad (3)$$

其中,  $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ ,  $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ ,  $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ , 分别表示粒子  $i$  在  $D$  维空间中的最优位置、位置向量和速度向量;  $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$  表示群体的最优位置;学习因子  $c_1, c_2$  表示非负常数;  $r_1, r_2$  表示均匀分布随机数,且  $r_1, r_2 \in [0, 1]$ 。

(3)终止条件。当达到预先设定的停止条件时,BPSO 算法停止并返回最优解。

### 1.2 惯性权重线性递减与指数递减粒子群算法

惯性权重线性递减算法(Linearly decreasing inertia Weight Particle Swarm Optimization, LWPSO)是经典粒子群算法的一种改进版。同经典粒子群算法一样,惯性权重线性递减算法的每个粒子都代表了一个候选解并根据其历史最优解和全局最优解不断地更新自身的位置和速度。不同的是惯性权重线性递减算法中的惯性权重是随着迭代次数不断线性下降的,这样迭代初期可以加快全局搜索能力,而在迭代后期加强局部搜索能力。LWPSO 算法的惯性权重描述可以表示为:

$$w = w_{\max} - (w_{\max} - w_{\min}) \times k/G \quad (4)$$

其中,  $w_{\max}$ ,  $w_{\min}$  分别表示  $w$  的最大值与最小值,  $G, k$  分别表示最大和当前的迭代数。

同理,惯性权重指数递减算法(Inertia Weight with Exponential Decrease Particle Swarm Optimization, EDPSO)在经典粒子群算法的基础上将惯性权重随迭代次数的下降而呈指数下降,在迭代后期进行快速收敛以达到全局搜索能力。惯性权重指数递减算法的惯性权重描述可以表示为:

$$w = w_{\min} + (w_{\max} - w_{\min}) \times \exp(- (4 \times k/G)^2) \quad (5)$$

其中,  $w_{\max}$  和  $w_{\min}$  分别表示  $w$  的最大值与最小值,  $G$  和  $k$  分别表示最大和当前的迭代数。

由于惯性权重的指数递减方式,该算法能够更好地平衡探索和提高效率,算法的适用性取决于适当选择递减指数和学习因子  $c_1$  和  $c_2$ 。

## 2 增强型双种群粒子群算法设计与实现

为充分应用惯性因子对粒子群影响特点,考虑采用2个惯性因子不同的种群协调进化的双种群算法实现粒子寻优。该算法采用了以上方法的折中。当粒子在进化中将较小的惯性因子赋予到此次进化到最优位置的种群,将较大的惯性因子赋予到此次未能进化到当前最优位的种群,通过惯性种群互相协调配合寻求最优解,种群的改变受群体最优位置影响改变以完成信息的共享。算法流程详述如下。

设  $t$  为种群,  $n$  为种群粒子个数;  $\omega^t$  表示  $t$  的权重,  $\omega_{\max}$  表示较大权重,  $\omega_{\min}$  表示较小权重。  $x'_{id}(k)$  表示在  $k$  次迭代时  $i$  粒子的维数是  $d$  的坐标,  $v'_{id}(k)$  表示在  $k$  次迭代下维数是  $d$  的  $i$  粒子速度;函数  $fitness(x'_i(k))$  表示在  $k$  次迭代下  $i$  粒子适应度;  $p_{gd}(k)$  表示最优粒子维数是  $d$  的坐标;  $p'_{id}(k)$  表示粒子进化中曾经出现的维数是  $d$  的最优坐标。

(1)初始化过程

- 步骤1 随机初始化  $x'_{id}(1)$ ;
- 步骤2 随机初始化  $v'_{id}(1)$ ;
- 步骤3 计算  $fitness(x'_i(1))$ , 令  $p'_i(1) = x'_i(1)$ ;
- 步骤4  $p_g(1) = \{x'_i(1) \mid x'_i(1) \in \min(fitness(x'_i(1)))\}$

(2) 循环迭代过程

- 步骤1 更新  $v'_i(k)$  和  $x'_i(k)$ ;
  - 步骤2 计算  $fitness(x'_i(k))$ , 如果  $fitness(x'_i(k)) < fitness(p'_i(k))$ , 则  $p'_i(k) = x'_i(k)$ ;
  - 步骤3 如果  $fitness(x'_i(k)) < fitness(p_g(k))$ , 则  $p_g(k) = x'_i(k)$ ;
  - 步骤4 如果  $fitness(x_i^1(k)) < fitness(x_i^2(k))$ , 则  $\omega^1 = w_{min}, \omega^2 = w_{max}$ ;  
如果  $fitness(x_i^1(k)) > fitness(x_i^2(k))$ , 则  $\omega^1 = w_{max}, \omega^2 = w_{min}$ ;  
如果  $fitness(x_i^1(k)) = fitness(x_i^2(k))$ , 则  $\omega^1 = \omega^2 = 0.7$ ;
  - 步骤5 不满足终止条件则转回步骤1;
- (3) 终止条件: 同 BPSO。

### 3 模拟仿真测试

研究采用9种测试函数对该算法与经典粒子群算法(BPSO)、惯性权重线性递减算法(LWPSO)、惯性权重指数递减算法(EPPO)进行对比以测试其性能,文中选用的测试函数具体如下。

(1) sphere:

$$f_1(x) = \sum_{i=1}^n x_i^2, -100 < x_i < 100$$

(2) Axis parallel hyper-ellipsoid :

$$f_2(x) = \sum_{i=1}^n i \cdot x_i^2, -100 < x_i < 100$$

(3) Rotated hyper-ellipsoid :

$$f_3(x) = \sum_{i=1}^n \left( \sum_{j=1}^i x_j \right)^2, -100 < x_i < 100$$

(4) Moved axis parallel hyper-ellipsoid :

$$f_4(x) = \sum_{i=1}^n 5i \cdot x_i^2, -100 < x_i < 100$$

(5) Rosenbrock :

$$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2], -100 < x_i < 100$$

(6) Rastrigin :

$$f_6(x) = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i) + 10], -100 < x_i < 100$$

(7) Griewank :

$$f_7(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, -100 < x_i < 100$$

(8) Sum of different power :

$$f_8(x) = \sum_{i=1}^n |x_i|^{(i+1)}, -100 < x_i < 100$$

(9) Ackley :

$$f_9(x) = -20 \exp\left[-2.0 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right] - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e, -100 < x_i < 100$$

在模拟仿真中对 EDUPSO 的种群1和2粒子个数  $n$  取15,对其他3个对比算法取种群粒子  $n$  为30。学习因子  $c_1, c_2 = 2$ , 维数  $D = 10$ ,  $\omega$  设为0.7,  $\omega_{max} = 0.9, \omega_{min} = 0.4, x_i \in [-100, 100]$ , 速度范围  $v_i \in [-100, 100]$ 。仿真时对各函数优化多次后对得到的结果取均值与标准差,仿真测试结果见表1。

表1 BPSO, WPSO, EPPO 和 EDUPSO 对比

Table 1 Comparison of BPSO, WPSO, EPPO and EDUPSO

函数	BPSO	LWPSO	EPPO	EDUPSO
$f_1$	6.988e-11(±1.696e-10)	6.257e-25(±2.101e-24)	2.521e-51(±1.104e-50)	3.247e-58(±1.996e-57)
$f_2$	2.342e-10(±5.047e-10)	2.288e-23(±1.318e-22)	1.996e-50(±7.191e-50)	1.982e-57(±9.535e-57)
$f_3$	0.0017(±0.004)	5.332e-08(±2.072e-07)	1.474e-15(±7.162e-15)	9.389e-17(±3.377e-16)
$f_4$	2.465e-09(±8.299e-09)	8.526e-23(±4.657e-22)	2.804e-48(±2.503e-47)	4.176e-56(±2.938e-55)
$f_5$	5.518(±1.0926)	4.422(±1.4338)	3.486(±1.1956)	3.158(±1.2075)
$f_6$	5.170(±3.1996)	3.597(±1.9471)	3.486(±1.8174)	3.860(±2.1084)
$f_7$	0.0003(±0.002)	0.0067(±0.012)	0.0041(±0.011)	0(±0)
$f_8$	1.007e-17(±7.361e-17)	1.004e-39(±8.157e-39)	4.128e-82(±3.404e-81)	1.529e-96(±7.462e-96)
$f_9$	0.014(±0.081)	0.084(±0.189)	0.156(±0.2471)	0.302(±0.297)

由表 1 可以看出在搜索的精度与稳定性上 EDUPSO 要优于前 3 种算法。图 1 为 BPSO、LWPSO、EPSO 及 EDUPSO 在进化中的适应度变化。图 1 中, 实线、虚线、点线和点划线依次为 EDUPSO、EPSO、LWPSO、BPSO 的曲线图。对上述测试的 9 种函数搜索效率由高到低依次为 EDUPSO、EPSO、LWPSO 和 BPSO。

通过 Rosenbrock 函数由 20 维增大到 80 维的一些数据验证了 EDUPSO 和其他对比算法在不同维数的性能。不同维数下 BPSO 和 EDUPSO 对比结果见表 2。当维数从 20 维增大到 80 维后, 基本粒子群算法在搜索时的稳定性和精度都会明显下降, 而 EDUPSO 算法的稳定性和精度则保持基本不变。

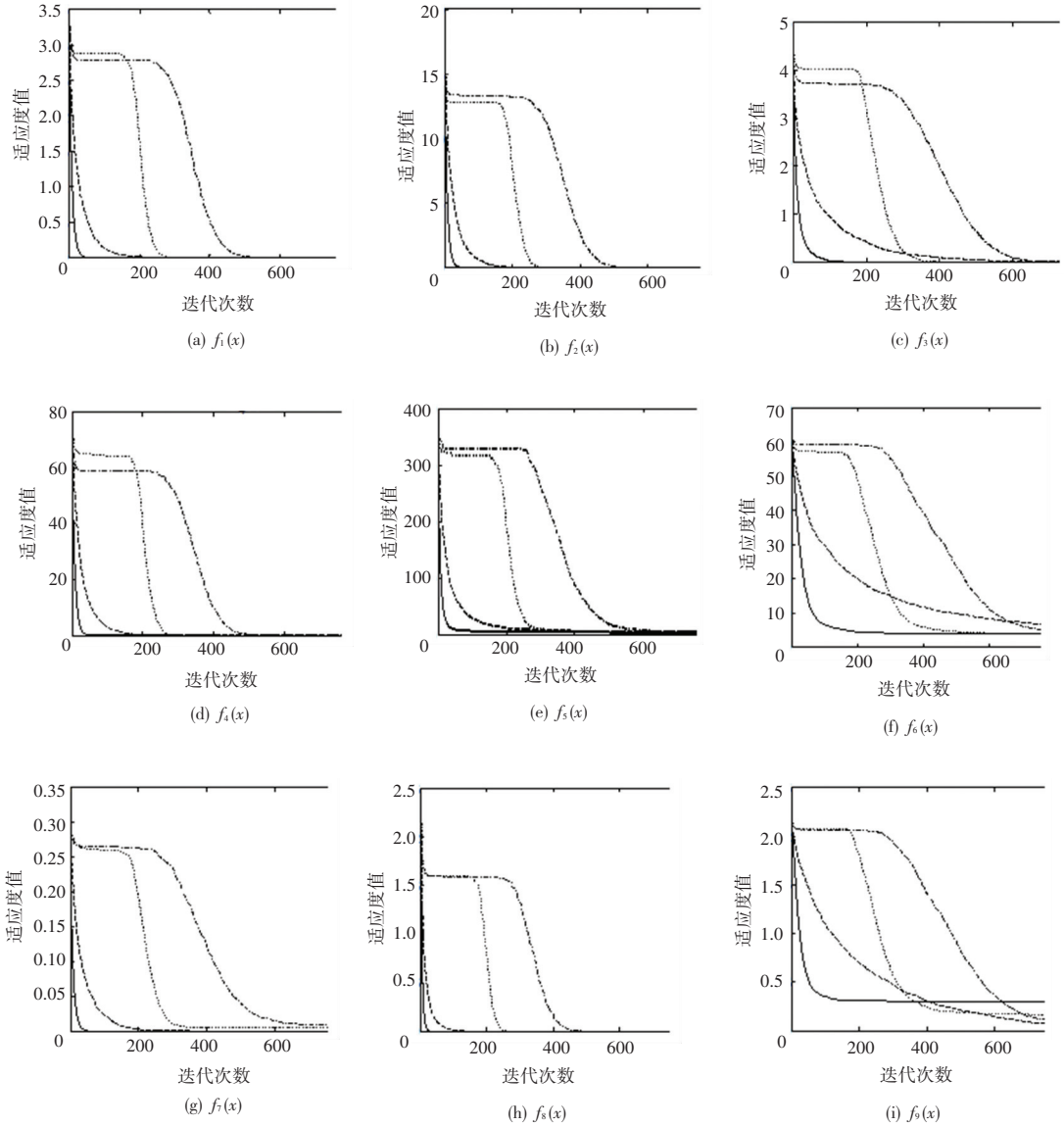


图 1 适应度曲线图

Fig. 1 Change curve of the fitness

表 2 不同维数下 BPSO 和 EDUPSO 对比结果

Table 2 Comparison of BPSO and EDUPSO in different dimensions

维数	BPSO	LWPSO	EPSO	EDUPSO
20	60.268(±49.728)	4.032(±1.407)	2.929(±0.975)	3.026 8(±1.147)
40	3 196.699(±1 784.725)	4.187(±1.119)	3.025(±0.882)	3.098 2(±1.342)
60	9 104.863(±1 806.536)	4.409(±1.215)	2.978(±1.149)	3.317 6(±1.394)
80	14 516.943(±2 476.530)	4.278(±1.330)	3.086(±1.074)	3.085 7(±1.298)

## 4 结束语

文中提出的增强型双种群粒子群算法 EDUPSO 通过粒子进化过程中改变惯性权重来加快粒子搜寻效率,当粒子在进化中将较小的惯性因子赋予到此次进化到最优位置的种群,将较大的惯性因子赋予到此次未能达到当前最优位的种群,通过不同惯性种群的相互协调来寻找最优解。算法性能对比分析测试表明了 EDUPSO 算法在无论搜索精度和稳定性还是在搜寻效率上相对于基本粒子群算法及其改进方法均有较大提高。

## 参考文献

- [1] KENNEDY J, EBERHART R. Particle swarm optimization[C]// Proceedings of ICNN'95 - International Conference on Neural Networks. Perth, Australia:IEEE, 1995, 4: 1942-1948.
- [2] POLI R, KENNEDY J, BLACKWELL T. Particle swarm optimization: An overview[J]. Swarm Intelligence, 2007, 1: 33-57.
- [3] SHI Y. Particle swarm optimization: developments, applications and resources [C]//Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546). Seoul, Republic of Korea:IEEE, 2001, 1: 81-86.
- [4] 秦桢,唐秋华,魏国前,等. 基于粒子群算法的装配公差组合优化[J]. 组合机床与自动化加工技术,2022(4):143-147.
- [5] 胡友彪,琚棋定. 基于粒子群优化随机森林的煤矿涌(突)水水源机器学习识别技术[J]. 煤炭科技,2022,43(4):52-60.
- [6] 徐健,职燕,刘源. 基于 PSO 优化孪生支持向量机的自然语言处理[J]. 南京理工大学学报,2023,47(1):103-108.
- [7] 莫树培,唐璠,李国良,等. 混沌粒子群优化神经网络的井下人员无线定位方法研究[J]. 传感技术学报,2020,33(3):456-463.
- [8] 柏世兵. 基于神经网络和粒子群优化的数据挖掘算法的研究[J]. 激光杂志,2017,38(3):88-92.
- [9] SHI Y, EBERHART R. A modified particle swarm optimizer [C]//1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360). Piscataway, USA: IEEE, 1998: 69-73.
- [10] CHEN Dong, WANG Gaofeng, CHEN Zhenyi. The inertia weight self-adapting in PSO[C]//Proceedings of the 7<sup>th</sup> World Congress on Intelligent Control and Automation. Chongqing, China:IEEE, 2008:5313-5316.
- [11] LI N J, WANG W J, HSU C C J, et al. Enhanced particle swarm optimizer incorporating a weighted particle[J]. Neuro-Computing, 2014, 124: 218-227.
- [12] SAYAH S, HAMOUDA A. A hybrid differential evolution algorithm based on particle swarm optimization for nonconvex economic dispatch problems[J]. Applied Soft Computing, 2013, 13(4): 1608-1619.