一种自适应调整的混合粒子群优化算法

胡堂清 校云鹏 刘彬 党永政

空军勤务学院，江苏徐州，221116；

摘要：为改善粒子群优化算法在运行过程中出现的早熟收敛、容易陷入局部最优解等缺陷，文中提出一种自适应惯性权重的粒子群算法，通过粒子适应度对惯性权重进行自适应调整，采用 s 型函数，实现对惯性权重的非线性调整，平衡算法的全局搜索与局部能力；引入差分进化算法中的交叉和变异操作，以保证种群的多样性，使粒子更快的收敛到全局最优。实验结果表明，与 SAPSO、IPSO 和 LPSO 算法相比较，改进后的算法无论是在收敛速度还是收敛精度上都明显提高。

关键词：粒子群算法；惯性权重；自适应；差分进化

DOI：10.69979/3041-0673.26.01.001

Kennedy 和 Eberhart 通过对群体捕食行为进行模拟研究，提出了粒子群优化算法（PSO）。与其他算法相比，粒子群优化算法凭借易于实现、程序简单明了以及可调参数少等特点，自诞生以来就吸引了学者们的目光，成为研究热点，并且已在实际的优化问题中得到了运用[1]。

粒子群算法并非完美无缺，它存在一些不足之处，比如容易出现早熟收敛的情况，而且在算法运行后期，也常常会陷入局部寻优的困境。针对这些问题，众多学者开展了广泛且深入的研究。在对粒子群算法（PSO 算法）进行改进的过程中，主要有两个方向：一是对算法中的参数进行调整，二是通过控制种群的多样性来提升算法的性能[2]。在粒子群优化算法的可调参数里，惯性权重是最为关键的一个。Shi 与 Eberhart 不仅将惯性权重引入到粒子群优化（PSO）算法中，还提出了线性递减的惯性权重这一概念[3-4]，提高了算法的收敛速度。Hu 等人提出一种自适应调整PSO算法的惯性权重和学习因子，实验表明该算法具有较好的鲁棒性[5]。黄延林等在 PSO 算法中引入混沌搜索，PSO 算法的全局搜索能力和寻优精度都有所提高[6]。赵志刚等人提出一种自适应惯性权重，对种群划分等级，不同等级的种群惯性权重不同[7]。对粒子群优化算法中粒子种群的多样性进行控制，也可以提高算法的性能的一种改进方法。Krink　T 等引入速度和位置变异[8]，Xie　X　F 等引入小概率重新初始化等方法增加种群的多样性[9]。

为提升优化算法的收敛速率与收敛准确度，本文构建了一种借助S型函数实现惯性权重自适应调节的混合粒子群算法（HAPSO）。依据当前粒子的自适应值与群体均值的比例，对惯性权重实施自适应调整；此外，为提升算法种群的多样性，还引入了差分进化算法中的交叉和变异操作。

研究中选用 8 个标准测试函数对 HAPSO 算法的性能进行验证，实验结果表明，相较于其他算法，本文提出的改进算法在性能上更具优势。

1 粒子群优化算法

粒子群算法江群体中的鸟被抽象的描述为没有质量和体积的“粒子”[10]。在一个 D 维的空间中有 n 个粒子组成的群体，每个粒子搜索时，会参考自身过往的最优搜索点以及种群内过往的最优搜索点，并在此前提下产生变动。粒子群里的第 i 个粒子，由三个 D 维向量构成：

目前位置：xi=(xi1,xi2,......,xiD)；

历史最优位置：pi=(pi1,pi2,......,piD)；

速度：Vi=(vi1,vi2,......,viD)；

这里 i=1,2,....,n。对于每一个粒子，其 d 维 $(1 \le d \le D)$ 根据如下等式变化：

$$\begin{cases} v_{id}^{t+1} = \omega v_{id}^{t} + c_1 r_1 \left(p_{id} - x_{id}\right) + c_2 r_2 \left(p_{gd} - x_{id}\right) \\ x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1} \end{cases} \tag{1}$$

其中，ω是粒子的速度系数，称为惯性权重，t 是当前迭代次数，d 是粒子维数。vid 和 xid 分别是第 t 次迭代时粒子 i 第 d 维的速度和位置。c1 是粒子在其历

史搜索中所找到的最佳值的权重系数，c2 是粒子群体在搜索过程找到的最优值的权重系数，r1 和 r2 是[0,1]内的随机数。pid 和 pgd 分别是个体最优及全局最优。

惯性权重ω反映了当前粒子对前代粒子速度的继承能力，较大的惯性加强算法的全局搜索能力，减小算法陷入局部寻优的可能性；惯性权重数值较小时，算法的局部搜索能力会得到增强。因此，在算法的初始搜索阶段，我们期望惯性权重能取较大的数值，在算法的后期取较小的惯性权重，以提高算法的整体寻优性能。

2 粒子群算法改进策略

2.1 自适应惯性权重

惯性权重ω是 PSO 算法里一个关键参数，其数值大小决定了粒子先前速度对当前速度的影响程度。在对 PSO 算法的诸多改进中，学者们更倾向于采用简单的惯性权重线性递减方法。惯性权重的线性调整，会对算法的搜索能力与寻优速率造成显著影响。在算法初期，ω值较大时，算法具备较强的全局搜索能力，能加快寻优速度，但也可能提高跳过全局最优的概率；而到了算法后期，ω值较小时，算法的局部搜索能力会增强，不过却容易陷入局部最优的困境。

为改善上述不足，本文构建了一种能够对惯性权重实施自适应调节的混合粒子群算法（HAPASO）。在算法每次进行寻优迭代时，先计算每个粒子的适应度值和本次迭代的最优适应度值，利用粒子自适应度和群体适应度的平均值的比值来对惯性权重公式进行调整。并引入 S 型函数，该函数的特性是在整个区间内呈递增态势，且在初始与末尾阶段递增速率较为缓慢。将 S 型函数引入以改进ω，可让算法在迭代初期，ω能在较长时间内维持较大数值，进而增强算法的全局搜索能力；而在迭代后期，ω能长时间保持较小数值，以此强化算法的局部搜索能力。经改进的惯性权重更新公式如下：

$$\omega=\omega_{\min}+\left(\omega_{\max}-\omega_{\min}\right)\frac{1}{1+e^{a-b\left(f_i/f_{avg}\right)}} \quad (2)$$

在粒子群算法里，粒子适应度值和群体平均适应度值的比例，可用于刻画该粒子在群体中的搜索状况。针对最小适应度值求解问题，若两者的适应度比例较大，意味着当前粒子与群体最优粒子相距较远，此时需增大惯性权重，以提升算法的全局搜索能力；若该比例较小，则表明当前粒子离群体最优粒子较近，应相应减小惯性权重，从而增强算法的局部搜索能力。本文改进惯性权重公式满足上述条件，使粒子的惯性权重随适应度比值的变化而自适应改变。

2.2 差分进化操作

针对粒子群算法在迭代后期出现种群多样性降低、容易陷入局部寻优的问题，本文引入差分进化操作，通过对粒子执行一系列操作来维持种群多样性，提升全局搜索能力，防止陷入局部最优。本文中，通过引入变异与交叉操作来对粒子的位置进行更新。

位置更新公式如下所示：

$$u_{i,j}=\begin{cases}x_{r_1,j}+F\left(x_{r_2,j}-x_{r_3,j}\right), rand<CR;\\ x_{r_1,j}, rand\geq CR\end{cases} \quad (3)$$

其中，xr1,j、xr2,j 和 xr3,j 分别代表三个随机个体，且满足 r1≠r2≠r3≠i；F 被称为缩放因子，本文将其设置为 0.5；CR 为交叉概率，本文取值为 0.05；rand 可生成 0 到 1 之间的随机数。当 rand 生成的随机数小于交叉概率时，采用变异操作对粒子的位置进行更新；当 rand 生成的随机数大于交叉概率时，粒子位置保持不变。在改进算法中引入这两种操作，能够随机地更新算法中粒子的位置，从而提高算法的种群多样性，增强算法的全局搜索能力，降低算法在迭代后期陷入局部寻优的可能性，提升算法对复杂问题的求解精度。

3 仿真实验及分析

3.1 测试函数选取及参数设置

为验证 HAPSO 算法的有效性，本文挑选了 8 个标准测试函数，用于对所提出的改进算法开展试验测试。

将 HAPSO 算法与线性递减惯性权重算法(LPSO)、非线性递减惯性权重(IPSO)、自适应惯性权重算法(SAPSO)进行比较。

聚知刊出版社
JZK publishing

表 1 测试函数

函数编号	范围	公式	fmin
f1	[-100,100]	$f_1(x)=\sum_{i=1}^{n} x_i^2$	0
f2	[-10,10]	$f_2(x)=\sum_{i=1}^{n} \lvert x_i\rvert+\prod_{i=1}^{n} \lvert x_i\rvert$	0
f3	[-100,100]	$f_3(x)=\sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)^2$	0
f4	[-100,100]	$f_4(x)=\max_i\{\lvert x_i\rvert,1\le i\le n\}$	0
f5	[-30,30]	$f_5(x)=\sum_{i=1}^{n-1}\left[100\left(x_{i+1}-x_i^2\right)+\left(x_i-1\right)^2\right]$	0
f6	[-100,100]	$f_6(x)=\sum_{i=1}^{n}\left(\left[x_i+0.5\right]\right)^2$	0
f7	[-1.28,1.28]	$f_7(x)=\sum_{i=1}^{n} ix_i^4+random[0,1)$	0
f8	[-5.12,5.12]	$f_9(x)=\sum_{i=1}^{n}\left[x_i^2-10\cos(2\pi x_i)+10\right]$	0

实验中不同的粒子群优化算法设置了相同的参数：最大迭代次数 tmax=1000，种群规模 N=50，搜索维数 D=15，学习因子 c1=c2=2。在惯性权重更新公式中 ωmax=0.9、ωmin=0.4、a=3.4、b=0.7。

3.2 仿真结果分析

重复对所选的 8 个函数在 15 维情况下进行 40 次测试，测试结果数据如表 2-表 9 所示：

表 2 函数 f1 的运行仿真结果对比

算法	最优值	最大值	平均值
IPSO	4.863E-35	9.341E-34	9.344E-35
SAPSO	2.201E-36	1.553E-35	6.396E-35
LPSO	5.022E-34	2.404E-31	3.256E-32
HAPSO	1.593E-38	1.4419E-35	1.476E-36

表 3 函数 f2 的运行结果仿真对比

算法	最优值	最大值	平均值
IPSO	3.395E-17	1.210E-15	4.929E-16
SAPSO	2.196E-20	7.794E-19	1.385E-19
LPSO	3.889E-15	2.848E-12	2.011E-13
HAPSO	2.569E-21	7.195E-19	1.827E-20

表 4 函数 f3 的运行仿真结果对比

算法	最优值	最大值	平均值
IPSO	1.517E-6	4.276E-5	2.276E-5
SAPSO	2.269E-7	4.443E-6	1.069E-6
LPSO	3.462E-6	9.0996E-5	1.276E-5
HAPSO	1.355E-8	3.894E-7	3.483E-8

表 5 函数 f4 的运行仿真结果对比

算法	最优值	最大值	平均值
IPSO	1.471E-6	1.090E-5	8.927E-6
SAPSO	9.524E-7	1.588E-5	3.589E-6
LPSO	1.923E-5	5.452E-5	7.882E-5
HAPSO	1.317E-6	6.897E-6	1.897E-6

表 6 函数 f5 的运行仿真结果对比

算法	最优值	最大值	平均值
IPSO	4.5889	9.332	9.1637
SAPSO	7.7199	9.123	8.2429
LPSO	8.8842	10.497	9.7965
HAPSO	0.1261	1.5723	0.2254

表 7 函数 f6 的运行仿真结果对比

算法	最优值	最大值	平均值
IPSO	0	2.216E-31	7.007E-32
SAPSO	0	9.244E-32	2.983E-32
LPSO	1.164E-30	1.407E-29	4.535E-30
HAPSO	0	0	0

表 8 函数 f7 的运行仿真结果对比

算法	最优值	最大值	平均值
IPSO	4.369E-3	6.507E-3	5.118E-3
SAPSO	1.468E-3	5.067E-3	3.902E-3
LPSO	4.201E-3	1.564E-2	7.482E-3
HAPSO	1.044E-3	5.760E-3	3.251E-3

表9 函数f8的运行仿真结果对比

算法	最优值	最大值	平均值
IPSO	5.9698	17.9092	9.3526
SAPSO	5.9698	12.9345	8.9546
LPSO	5.9698	20.8941	10.9445
HAPSO	4.9748	8.9445	5.3576

从表1-表8可以看出，LPSO算法寻优效果最差，其次是IPSO算法,SAPSO算法寻优性能略逊与改进算法HAPSO，新算法的收敛精度均优于其他3种粒子群算法，而且收敛的稳定性也有所提升。尤其是在对函数f6的测试中，改进算法每次都能寻求到最优解，说明HAPSO算法寻优性能强于其他算法。对函数f3和f5的寻优精度明显高于其他3类算法。从表中数据可以看出，对于所有测试函数，本文提出的HAPSO算法和自适应权重SAPSO算法都取得了较好的效果，但是改进算法的精度优于SAPSO算法是因为HAPSO算法引入差分进化操作，使粒子种群多样性提升，寻优精度更高。

4 结束语

本文提出一种新的基于S型函数自适应调整惯性权重的混合粒子群算法（HAPSO）。利用粒子的适应度值与群体平均适应度值的比值来非线性的对惯性权重进行调整，并且引入差分进化算法中的操作，增加群体的多样性。将改进后的算法和SAPSO、IPSO、LPSO这三种算法就测试函数展开针对性对比，从仿真结果能够发现，HAPSO算法的寻优性能得到显著改进，其收敛速率与求解准确度均有所提高。

参考文献

[1]刘建华.粒子群算法的基本理论及其改进研究[D].长沙：中南大学，2009：15-16.
[2]黄洋，鲁海燕，许凯波等.一种动态调整惯性权重的简化均值粒子群优化算法[J].小型微型计算机系统，2018,39(12):2590-2595.
[3]Kennedy J，Eberhart R C. Particle swarm optimization[C].Proceedings of the 1995 IEEE International Conference on Neural Networks,1995:1942-1948.
[4]Shi Y H，Eberhart R C. A modified particle swarm optimizer[C].Proceedings of IEEE Congress on Eveolutionary Computation,1998:69-73.
[5]Hu M，Wu T F，Weir J D.An adaptive particle swarm optimization with multiple adaptive methods[J].IEEE Transactions on Evolutionary Computation,2013,17(5):705-720.
[6]黄延林，戴雪峰，张卉等.改进PSO算法在多水源供水系统优化调度中的应用[J].中国给水排水，2013,29(23):65-66.
[7]赵志刚，林玉娇，尹兆远.基于自适应惯性权重的均值粒子群优化算法[J].计算机工程与科学，2016,38(3):501-506.
[8]Krink T，Vesterstorm J S，Right J.Particle swarm optimization with spatial particle extension[C].Proceedings of the IEEE International 1st Conference on Evolutionary Computation.Honolulu:IEEE Inc,2002:1474-1497.
[9]Xie X F,Zhang W J,Yang Z L.A dissipative particle swarm optimization[C].Proceedings of the IEEE International Conference on Evolutionary Computation.Honolulu,IEEE Inc,2002:1456-1461.
[10]张鑫，邹德旋，肖鹏等.自适应简化粒子群优化算法及其应用[J].计算机工程与应用，2019,55(8):250-251.

作者简介：胡堂清(1994.07-)，男，汉族，山东省临沂市人，硕研，助教，从事油料保障教学和科研工作。