

一种增加权重的蜘蛛猴优化算法

姜 爽, 林 丹

(天津大学 数学学院, 天津 300350)

摘 要: 本文提出了增加权重的蜘蛛猴算法(WSMO). 为进一步提高蜘蛛猴算法(SMO)的性能, 在本地领导者阶段和本地领导者决策阶段对蜘蛛猴个体的原位置引入线性递减的惯性权重, 该算法可以在迭代前期增加种群的多多样性, 在迭代后期增加局部搜索能力. 通过六个基准函数的数值实验结果表明改进的算法比原始蜘蛛猴算法在收敛速度、寻优精度和鲁棒性等均有改进, 特别是在求解多峰函数优化问题时算法性能的改善更加显著.

关键词: 蜘蛛猴算法; 惯性权重; 函数优化; 本地领导者

中图分类号: TP18

文献标识码: A

文章编号: 1000—7180(2018)10—0001—06

A Weighted Spider Swarm Intelligent Swarm Optimization Algorithm

JIANG Shuang, LIN Dan

(Department of Mathematics, Tianjin University, Tianjin 300350, China)

Abstract: This paper proposes a weighted spider monkey algorithm (WSMO). In order to further improve the performance of the spider-monkey algorithm (SMO), a linear decreasing inertia weight is introduced into the spider monkey individual's home position in the local leaders' and local leaders' decision-making stages. This algorithm can increase the diversity of the population in the early iteration and increase the local search capabilities in the later iteration. The experimental results of six benchmark functions show that the improved algorithm is better than the original spider monkey algorithm in terms of convergence speed, optimization accuracy and robustness, especially in the optimization of multimodal function optimization problems.

Key words: spider monkey algorithm; inertia weight; function optimization; local leader

1 引言

群体智能优化算法是近年来发展起来的非常活跃的研究领域. 群体智能优化算法通过模拟自然现象的物理规律及自然界各类生物种群的生活习性和行为特点进行寻优搜索, 在很多领域得到了广泛的应用. 蜘蛛猴算法 (Spider Monkey Optimization, SMO) 是 Bansal^[1] 等提出的一种新型群体智能优化算法. 在该算法中, 蜘蛛猴群根据裂变融合的社会制度, 通过群体分裂来进行觅食达到寻优目的. 数值实验证实该算法简单且高效, 已引起了不同研究领域人员的关注和研究. 文献[2]在本地领导者阶段和全球领导者阶段引入黄金分割搜索(GSS)过程取得了

较好的效果. 文献[3]提出了基于个体适应度的位置更新蜘蛛猴算法(FPSMO). 文献[4]提出老年蜘蛛猴算法(AMSMO), 通过对蜘蛛猴年龄进行分组, 导致群体分裂和融合的速度加快, 从而提高了收敛速率. 文献[5]等提出用锦标赛选择概率来代替个体适应度的选择概率, 避免了选择的单一性, 使得适应度低的个体也有机会更新自己位置. 文献[6]提出一种新的变体 FCSMO, 在全局领导者阶段和本地领导者决定阶段引入随迭代次数变化的加速系数. 文献[7]通过将蜘蛛猴算法和遗传算法结合, 设计出一种混合算法. 蜘蛛猴算法因其简单高效、参数少在电子、电气等领域^[8-9]得到了应用. 但是, 由于 SMO 算法提出时间相对较短, 其相关研究文献相对较少, 也

具有后期收敛速度慢、易陷入局部最优、稳定性差等缺点. 本文针对这些问题设计了 WSMO 改进算法, 在本地领导者阶段和本地领导者决策阶段对蜘蛛猴个体的原位置加入惯性权重. 通过对测试函数的数值仿真分析可知改进的算法的寻优精度、收敛速度和鲁棒性均优于原算法.

2 基本蜘蛛猴算法

蜘蛛猴算法是通过模拟现实生活中蜘蛛猴基于裂变-融合社会结构(FFSS)的觅食行为而衍生的一种群体智能优化算法. 主要思想是蜘蛛猴群被分成小组进行觅食以减少成员之间的竞争, 小组成员的位置通过全球领导者、本地领导者、小组成员的位置的反馈进行更新. 每个小组由母猴领导, 如果没有找到目标食物, 则再细分小组, 到最大组数后进行融合. 复制这样的过程直到找到目标食物. 蜘蛛猴算法主要分为以下几部分:

2.1 种群的初始化

最初种群产生一个数量为 N 的蜘蛛猴群每只猴子代表 D 维的向量, $SM_i (i = 1, 2, \dots, N)$ 代表群体中的第 i 个猴子. 每个代表被优化函数潜在的解. 每个 SM_i 按下式初始化:

$$SM_{ij} = SM_{\min j} + U(0, 1) \times (SM_{\max j} - SM_{\min j}) \quad (1)$$

$SM_{\min j}$ 和 $SM_{\max j}$ 是 $SM_i (i = 1, 2, \dots, N)$ 第 j 维的边界, $U(0, 1)$ 是 0 到 1 的随机数.

2.2 本地领导者阶段

在本地领导者阶段, 每个蜘蛛猴个体 SM_i 依据本地领导者经验和小组成员经验更新自己的位置. 根据新位置计算其适应度, 如果新位置的适应度值大于原位置的适应度值, 则更新位置. 第 i 个蜘蛛猴 SM 也是第 k 个小组的成员的位置更新公式如下:

$$SM_{\text{newij}} = SM_{ij} + U(0, 1) \times (LL_{kj} - SM_{ij}) + U(-1, 1) \times (SM_{rj} - SM_{ij}) \quad (2)$$

SM_{ij} 是第 i 个蜘蛛猴向量的第 j 维, LL_{kj} 是第 k 组本地领导者位置向量的第 j 维, SM_{rj} 第 r 个蜘蛛猴位置向量的第 j 维, 该个体是在第 k 组中随意选取且 $i \neq r$.

2.3 全局领导者阶段

完成本地领导者即开始全局领导者阶段, 在全局领导者阶段, 所有个体根据全局领导者和本地小组成员的经验来更新其位置. 位置更新公式如下:

$$SM_{\text{newij}} = SM_{ij} + U(0, 1) \times (GL_j - SM_{ij}) + U(-1, 1) \times (SM_{rj} - SM_{ij}) \quad (3)$$

GL_j 代表全局领导者位置向量的第 j 维. 在该阶段, 蜘蛛猴个体位置依据由其适应度计算得来的概率来进行更新. 这样, 一个适应度高的个体将会有更多的可能性使其进入下一个阶段.

$$\text{prob}_i = 0.9 \times \frac{\text{fitness}_i}{\max_fitness} + 0.1 \quad (4)$$

fitness_i 是第 i 个蜘蛛猴个体的适应度值, $\max_fitness$ 是小组中最大的适应度. 新产生的个体位置的适应度将会得到计算并与原位置的适应度作出比较, 从而接受更好的位置.

2.4 全局领导者学习阶段

在这个阶段, 通过应用贪婪选择来更新全球领导者的位置, 即选择人口中具有最佳适应度的 SM 的位置作为全球领导者的更新位置. 此外, 检查全球领导者的位置是否正在更新, 如果不是则 GlobalLimitCount 增加 1.

2.5 本地领导者学习阶段

在这个阶段, 通过应用该组中的贪婪选择来更新本地领导者的位置, 即选择该组中具有最佳适应度的 SM 的位置作为本地领导者的更新位置. 接下来, 将本地领导者的更新位置与旧领导者的位置进行比较, 如果本地领导者未更新, 则 LocalLimitCount 增加 1.

2.6 本地领导者决策阶段

如果任何当地领导者的位置没有更新到一个预先设置的 LocalLimitCount 的值, 那么该小组的所有成员都可以通过扰动率的大小来决定自动初始化或是通过使用来自全球领导者和当地领导者的组合信息来更新他们的位置.

$$SM_{\text{newij}} = SM_{ij} + U(0, 1) \times (GL_j - SM_{ij}) + U(0, 1) \times (SM_{ij} - LL_{kj}) \quad (5)$$

2.7 全局领导者决策阶段

在这个阶段, 全局领导者的位置更新次数如果没有更新到预定设置的 GlobalLimitCount 的值, 那么全局领导者将群体分成更小的群体. 首先, 将猴群分成两组, 然后分成三组, 直到达到最大组数, 此时启动本地领导者学习过程, 以选出新成立的小组中的本地领导者. 最多组成团队的情况下, 全局领导者的位置不更新, 全局领导者即将所有组合成一个组. 因此, 所提出的算法是受到蜘蛛猴群体觅食行为融合-裂变结构启发的一种新型智能群优化算法.

2.8 SMO 算法的伪代码

步骤 1 初始化种群规模, 本地领导者限制次数, 全局领导者限制次数, 扰动率.

步骤 2 计算适应度(判断个体距离食物源的距离).

步骤 3 依据贪婪准则选出本地领导者和全局领导者.

如果停止条件不被满足即进行如下步骤:
(1)为了找到最优解,在本地领导者阶段依据自身经验本地领导者经验,小组成员经来产生新位置.

(2)依靠小组成员的适应度值,采用贪婪选择策略.

(3)利用等式(4)来计算每个小组成员的机会值.

(4)通过自己的经验,全局领导者经验和团队成员的经验,为 prob_i 选定的所有小组成员更新位置.

(5)对所有小组成员进行贪婪选择,更新本地领导者和全局领导者的位置.

(6)如果任何本地领导者没有在预先设定的本地领导人限制次数 LocalLimitCount 之后更新它的位置,则重新指导该特定小组的所有成员用本地领导者决策阶段进行觅食.

(7)如果全局领导者没有没有在预先设定的全局领导者限制次数 GlobalLimitCount 之后更新它的位置,则利用全局领导者决策语句将改组分成更小的组.

3 增加权重的蜘蛛猴算法

探索和开发是智能群算法的两个关键属性,如何更好地平衡二者的关系是改进算法的关键.蜘蛛猴算法很好地平衡了其探索和开发能力,但仍有改进空间.本文为蜘蛛猴个体的原位置增加线性递减惯性权重.较大的惯性权重 w 有利于全局搜索,并增加种群的多样性,而较小的 w 则可以提高算法的局部开采能力,加快收敛速度.因此,本文采用线性递减权重策略,即从算法前期的较大值线性递减至一个较小值,以保证算法较好的前期全局搜索能力和后期局部搜索能力.本文在原始算法的本地领导者阶段和本地领导者决策阶段引入这种惯性权重的设置.

3.1 本地领导者阶段

原始蜘蛛猴算法在此阶段的每个蜘蛛猴 SM 依据本地领导者经验和小组成员经验更新自己的位置.该蜘蛛猴个体的原位置也有可能含有与最优解相关的、重要的信息,所以在该阶段为蜘蛛猴个体的原位置增加惯性权重.采用线性递减策略,可以在迭

代前期增加种群的多样性,在迭代后期增加局部搜索能力.改进的算法可以提高原算法的收敛速度和寻优精度.改进公式如下

$$w = w_{\max} - \frac{\text{iter}}{\text{maxcycle}} \times (w_{\max} - w_{\min}) \quad (6)$$

$$SM_{\text{new}ij} = w \times SM_{ij} + U(0,1) \times (LL_{kj} - SM_{ij}) + U(-1,1) \times (SM_{ij} - SM_{ij}) \quad (7)$$

式中, $w_{\max}$ 为惯性权重最大值; $w_{\min}$ 为惯性权重最小值; iter 为当前迭代次数; maxcycle 为最大迭代次数.

3.2 本地领导者决策阶段

该小组的所有成员通过扰动率的大小来决定自动初始化或是通过使用来自全局领导者和当地领导人的组合信息来更新他们的位置.当扰动率较大时,则通过使用来自全局领导者和本地领导者的组合信息来更新他们的位置.此时由于蜘蛛猴个体的原位置也包含着与食物源有关的重要信息,所以本文为蜘蛛猴个体的原位置增加惯性权重.且在迭代前期,可以保持良好的搜索空间,随着迭代次数的增加惯性权重逐渐减小,在后期可以保持良好的开发性能.提高了算法的收敛性.其改进公式如下:

$$SM_{\text{new}ij} = w \times SM_{ij} + U(0,1) \times (GL_j - SM_{ij}) + U(0,1) \times (SM_{ij} - LL_{kj}) \quad (8)$$

w 随着迭代次数的增加而减小.适当增加种群中蜘蛛猴的数目,能在迭代后期产生不可行解而跳出局部最优,加快收敛速度.

4 数值实验

为了测试改进的蜘蛛猴优化算函数的求解效果,本文在一台配置为 Intel Core M 370, 2.40 GHz, 2 G 内存的 PC 机上利用 MATLAB R2012a 运行环境实现该算法.

4.1 参数设置

当函数维数为 30 维时,种群规模 NP 设为 50;当函数维数增加到 100 维时,种群规模 NP 设置为 100;最大迭代次数 M 均为 2 000 次.

表 1 标准测试集函数

标准测试函数	编号	定义域	最优解
Sphere	F_1	$[-5, 12, 5, 12]$	0
Zakharov	F_2	$[-5, 12, 5, 12]$	0
Axis-parallel-hyper-ellipsoid	F_3	$[-5, 12, 5, 12]$	0
Rastrigin	F_4	$[-5, 12, 5, 12]$	0
Ackley	F_5	$[-30, 30]$	0
Griewank	F_6	$[-600, 600]$	0

最大组数 $MG=NP/10$; $w_{\max}=0.8, w_{\min}=0.4$; $GlobalLimit=NP$; $LocalLimit=D * NP$; 扰动率 $pr_{G+1}=pr_G+0.4/\maxcycle$ $pr_1=0.1$.

实验用改进前后的算法分别求解了 6 个测试函

数(如表 1)(30 和 100 维)进行 50 次实验取平均结果,如表 2、3 所示. 同时给出各测试函数在 30 维时的收敛曲线,如图 1 所示.

表 2 改进算法和原算法的对比结果 ($D=30, NP=50, M=2\ 000$)

函数	算法	最优值	最差值	均值	标准差
F_1	SMO	7.08047E-17	2.54834E-16	1.2757E-16	6.6447E-17
	WSMO	3.2214E-17	8.0136E-17	5.5494E-17	9.8034E-18
F_2	SMO	5.2286E-17	2.3771E-16	1.1912E-16	6.5469E-17
	WSMO	3.4218E-17	7.0166E-17	5.4445E-17	8.4327E-18
F_3	SMO	5.2265E-17	2.5107E-16	1.1805E-16	6.4601E-17
	WSMO	3.3639E-17	7.3983E-17	5.5510E-17	9.6886E-18
F_4	SMO	0	2.02635E-06	4.0599E-08	2.8656E-07
	WSMO	0	0	0	0
F_5	SMO	7.99361E-15	1.15515	0.0021	0.1634
	WSMO	4.4409E-15	4.4409E-15	4.4409E-15	0
F_6	SMO	0	0.0245733	0.0042	0.0086
	WSMO	0	0	0	0

表 3 改进算法和原算法的对比结果 ($D=100, NP=100, M=2\ 000$)

函数	算法	最优值	最差值	平均值	标准差
F_1	SMO	2.3686E-16	7.1501E-16	4.3987E-16	1.1532E-16
	WSMO	7.5425E-17	2.4736E-16	1.2952E-16	5.7266E-17
F_2	SMO	2.4457E-16	7.1548E-16	4.2242E-16	1.2281E-16
	WSMO	7.3825E-17	2.3986E-16	1.5210E-16	6.1629E-17
F_3	SMO	2.5069E-16	7.0027E-16	4.1414E-16	9.9068E-17
	WSMO	6.9341E-17	2.2984E-16	1.3827E-16	5.7702E-17
F_4	SMO	0	0.0221	4.4158E-04	0.0031
	WSMO	0	0	0	0
F_5	SMO	1.1546E-14	2.3441	1.3614	0.7490
	WSMO	4.4409E-15	7.9936E-15	7.9226E-15	5.0243E-16
F_6	SMO	0	0.0610	0.0049	0.0113
	WSMO	0	0	0	0

4.2 算法有效性测试

为检验改进算法的有效性,在指定种群规模和进化代数下,分别用改进前后的算法求解测试问题,结果见表 2 和表 3. 在从表中数据可以看出对于 F_1, F_2 和 F_3 单峰函数,在 30 维时平均求解精度均提高了 1 个数量级,标准差降低了 1 个数量级. 100 维时最优值提高了一个数量级,平均求解精度均比原算法高,标准差降低了 1 个数量级. 实验表明改进后的算法在低维时具有更好的求解精度和稳定性. 对于 F_4, F_5 和 F_6 等多峰函数,在 30 维时,对 F_4 函数改进后的算法的平均求解精度提高了 8 个数量级,并

且均求解到了最优值,标准差为 0 比用原算法求解的标准差降低了 7 个数量级. 对于 F_5 函数改进后的算法的平均求解精度提高了 12 个数量级,且标准差为 0. 对于 F_6 函数改进后算法均求解到了最优值,且标准差为 0. 在 100 维时 WSMO 算法仍然保持了很好的稳定性,随着维数的增加,对于多峰函数原始 SMO 算法的求解精度开始下降,且标准差增大表现出较大的不稳定性. 实验说明改进后的算法展现了更好的寻优效果和稳定性即 WSMO 算法在高维空间求解能力更强.

对于 F_4 和 F_6 函数,WSMO 算法的结果无论在

低维还是高维其最优值均为标准值且标准差为 0, 其寻优精度和稳定性

大大高于原始 SMO 算法. 对于 F_5 函数在低维数时其标准差为 0, 虽然在高维数时标准差增大, 但是仍比 SMO 算法产生的标准差小 15 个数量级. 由上所述, 改进后的 WSMO 算法提高了寻优能力和稳定性, 尤其提高了多峰函数的寻优能力和稳定性.

为了进一步对比 WSMO 算法与 SMO 算法在搜索精度、速度等优化性能上的差异, 给出两个算法在各个测试函数($D=30$)维时的进化曲线, 可以看出, WSMO 算法能够在较少的迭代次数中迅速捕获函数最优值, 搜索速度快, 同时具有更好的求解精度. 而原始 SMO 算法容易陷入局部最优无法得到更好的解. 对于 F_1 函数, 算法在大约 400 代后陷入了较长时间的局部最优, F_3 函数在 600 代到 1 500 代左右陷入局部最优, F_4 函数也陷入很长时间的局部最优. 而改进后的算法可以快速找到最优值, 几乎不会陷入局部最优的情况.

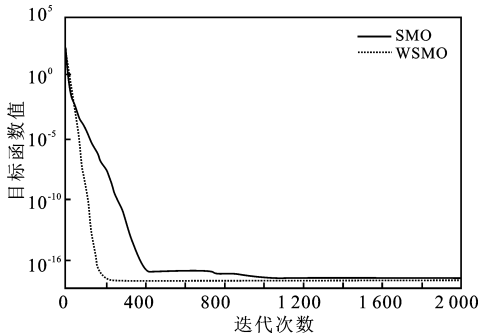


图 1 F_1 函数收敛曲线

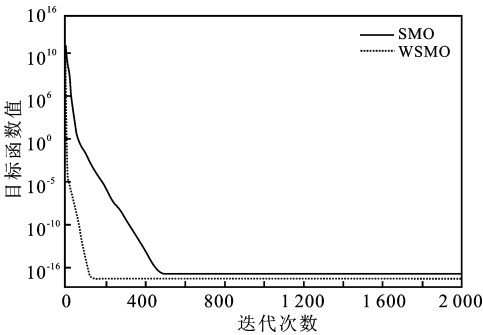


图 2 F_2 函数收敛曲线

4.3 算法的效率测试

为进一步对改进算法的效率进行测试, 在指定的解的精度下对改进前后算法进行对比测试实验.

分别为每个测试函数指定设置的收敛精度, 种群中个体数为 50, 通过 50 次实验比较 SMO 与 WS-

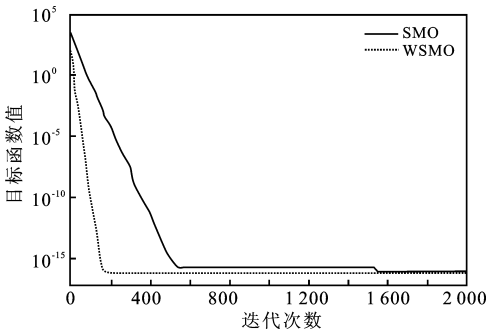


图 3 F_3 函数收敛曲线

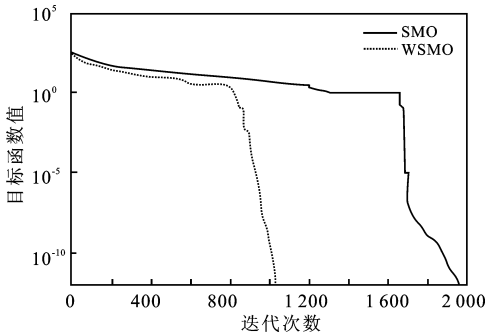


图 4 F_4 函数收敛曲线

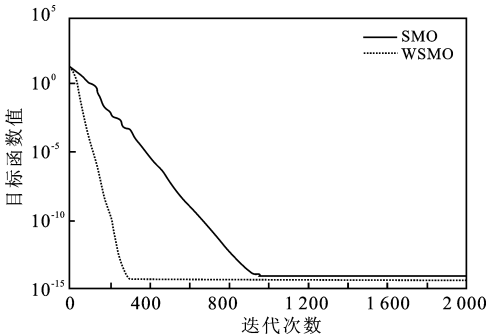


图 5 F_5 函数收敛曲线

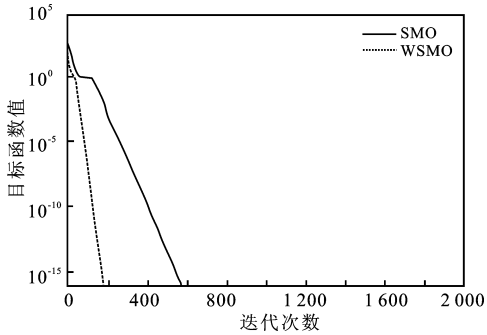


图 6 F_6 函数收敛曲线

MO 在函数 30 维时达到目标精度时的平均迭代次数、平均时耗和成功率(达到目标精度的实验次数/总实验次数). 特别地, 实验指定迭代次数上限为 2 000, 即算法经过 2 000 次迭代都未能达到指定精

度,认为算法不能达到目标精度. 数值实验结果如表 4 所示.

表 4 目标精度下算法的性能测试

函数编号	算法	平均迭代次数	平均耗时/s	精度	成功率
F_1	SMO	125	0.66	9E-05	100
	WSMO	43	0.34	9E-05	100
F_2	SMO	126	0.84	1.0E-05	100
	WSMO	51	0.44	1.0E-05	100
F_3	SMO	169	1.12	1.0E-05	100
	WSMO	61	0.54	1.0E-05	100
F_4	SMO	1159	3.18	1.0E-03	98
	WSMO	934	2.34	1.0E-03	100
F_5	SMO	472	1.94	1.0E-05	20
	WSMO	99	0.64	1.0E-05	100
F_6	SMO	815	3.46	1.0E-05	76
	WSMO	86	0.68	1.0E-05	100

由表 4 中数据可以很明显看出 WSMO 算法迭代次数远远少于 SMO 算法,只需要较少的迭代次数就能达到指定精度,节约了运行时间. 即 WSMO 算法在有更好的寻优精度的同时减少了迭代次数,加快了收敛速度. 从表中可知,对于单峰函数,两种算法的成功率均为 100%,但对于多峰函数来说,原始 SMO 算法对函数优化的成功率较低,且对于 F_5 函数优化的成功率不足 50%,但 WSMO 算法对 F_4 、 F_5 和 F_6 函数优化成功率均为 100%. 实验表明 WSMO 算法有更快的收敛速度,有效地解决了 SMO 算法成功率低的问题,取得了较好的效果.

4.4 与其他现有改进算法的对比

为了进一步验证本文改进算法的性能,实验首先将 WSMO 算法与文献[6]中的 FCSMO 算法进行对比. 统一参数设置为:函数维数为 30 维时,种群规模 NP 设为 50,扰动率 $pr \in [0.1, 0.5]$. 两种算法分别运行 50 次,实验结果如表 5 所示.

表 5 与参考文献中算法性能的对比

函数编号	算法	均值	标准差
F_1	FCSMO	1.0634E-16	5.4063E-17
	WSMO	5.5494E-17	9.8034E-18
F_2	FCSMO	9.4583E-17	3.8080E-17
	WSMO	5.4445E-17	8.4327E-18
F_3	FCSMO	1.3429E-16	7.0520E-17
	WSMO	5.5510E-17	9.6886E-18
F_4	FCSMO	0.3648	0.6653
	WSMO	0	0
F_5	FCSMO	0.0454	0.2286
	WSMO	4.4409E-15	0
F_6	FCSMO	0.0045	0.0094
	WSMO	0	0

从表中数据显示,WSMO 算法比 FCSMO 算法具有更好的寻优效果和鲁棒性.

5 结束语

蜘蛛猴算法因其简单易操作,求解精度高等特点有很好的发展潜力和应用前景. 本文通过为原蜘蛛猴个体的位置增加线性递减权重来对原始算法进行改进,提出了 WSMO 算法. 在标准测试函数上进行的数值实验结果表明 WSMO 算法在收敛速度、搜索精度及鲁棒性等方面均比原始算法有了明显优化. 特别地,对其中的多峰测试函数,原始 SMO 算法存在成功率较低的问题,而 WSMO 算法得无论是在成功率上还是在平均最优值上都有很大改善. 本文采用的是粒子群算法中的线性递减权重策略,将粒子群算法中的自适应权重、随机权重等引入 SMO 算法能否对改进 SMO 算法,还需进一步的研究和数值实验.

参考文献:

[1] Bansal J C, Sharma H, Jadon S S, et al. Spider monkey optimization algorithm for numerical optimization [J]. Memetic computing, 2014, 6(1): 31-47.

[2] Kumar S, Kumari R. Modified position update in spider monkey optimization algorithm [C] // Proceedings of International Journal of Emerging Technologies in Computational and Applied Sciences IJETCAS. [s. l.] 2014.

[3] Kumari R, Sharma V K. Fitness based position update in spider monkey optimization algorithm[J]. Procedia Computer Science, 2015(62): 442-449.

[4] Sharma A, Panigrahi B K. Ageist spider monkey optimization algorithm [J]. Swarm and Evolutionary Computation, 2016(28): 58-77.

[5] Gupta K, Deep K. Tournament selection based probability scheme in spider monkey optimization algorithm [M]. Berlin, Heidelberg: Springer Press, 2016.

[6] Agarwal N, Jain S C. Fast convergent spider monkey optimization algorithm [C] // Proceedings of Sixth International Conference on Soft Computing for Problem Solving. Springer, Singapore, 2017: 42-51.

[7] Agrawal A, Farswan P, Agrawal V, et al. On the hybridization of spider monkey optimization and genetic algorithms [C] // Proceedings of Sixth International Conference on Soft Computing for Problem Solving. Singapore, Springer, 2017: 185-196.

DBM-ELM. 因此,本文方法整体的估计精度高于其他的算法.

5 结束语

面对软测量模型中辅助变量之间相关性差使得主导变量估计精度不高等问题,本文提出了 ML-CRBM_s-ELM 软测量模型,首先利用 FCM 算法对数据进行属性簇划分,将划分得到多个属性簇输入至 MLCRBM_s 特征提取集中进行特征提取,将提取到的多个特征子集输入至 Blending 层进行特征融合得到新的特征集,最后输入至极限学习机中进行拟合,得到最后的拟合结果. 实验结果表明 ML-CRBM_s-ELM 模型相比于其他的传统算法,不仅有较高的拟合精度,而且提升了软测量模型的稳定性,具有较好的泛化能力.

参考文献:

- [1] Min H, Luo X L. Calibration of soft sensor by using Just-in-time modeling and AdaBoost learning method [J]. Chinese Journal of Chemical Engineering, 2016, 24(8):1038-1046.
- [2] 曹鹏飞, 罗雄麟. 化工过程软测量建模方法研究进展 [J]. 化工学报, 2013, 64(3):788-800.
- [3] 汤健, 赵立杰, 岳恒, 等. 湿式球磨机筒体振动信号分析及负荷软测量 [J]. 东北大学学报(自然科学版), 2010, 31(11):1521-1524.
- [4] 常鹏, 王普, 高学金, 等. 基于统计量模式分析的 MK-PLS 间歇过程监控与质量预报 [J]. 仪器仪表学报, 2014, 35(6):1409-1416.
- [5] 张振, 徐立中, 韩华, 等. 基于径向基神经网络的明渠流量软测量方法 [J]. 仪器仪表学报, 2011, 32(12):2648-2655.
- [6] Ackley D H, Hinton G E, Sejnowski T J. A learning

algorithm for boltzmann machines [J]. Cognitive Science, 1985, 9(1):147-169.

- [7] Ackley D H, Hinton G E, Sejnowski T J. A learning algorithm for Boltzmann machines [M]. Ablex Publishing Corp. 1985:147-169.
- [8] Melchior J, Fischer A, Wiskott L. How to center deep boltzmann machines [J]. Journal of Machine Learning Research, 2016, 17(99):61.
- [9] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: Theory and applications [J]. Neurocomputing, 2006, 70(1/3):489-501.
- [10] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: a new learning scheme of feedforward neural networks [C]// IEEE International Joint Conference on Neural Networks. IEEE, 2005:985-990.
- [11] Guo L, Hao J, Liu M. An incremental extreme learning machine for online sequential learning problems [J]. Neurocomputing, 2014, 128(27):50-58.
- [12] 严东, 汤健, 赵立杰. 基于特征提取和极限学习机的软测量方法 [J]. 控制工程, 2013, 20(1):55-58.
- [13] Saul L K, Roweis S T. Think globally, fit locally: unsupervised learning of low dimensional manifolds [J]. Journal of Machine Learning Research, 2003, 4(2):119-155.
- [14] 王宇红, 狄克松, 张姗, 等. 基于 DBN-ELM 的聚丙烯熔融指数的软测量 [J]. 化工学报, 2016, 67(12):5163-5168.

作者简介:

倪建华 男, (1992-), 硕士研究生. 研究方向为人工制智能与软测量建模. E-mail:1207790299@qq.com.

王 静 女, (1982-), 副研究员. 研究方向为机器学习和软测量建模.

(上接第 6 页)

- [8] Al Azza A A, Al Jodah A A, Harackiewicz F J. Spider monkey optimization: a novel technique for antenna optimization [J]. IEEE Antennas and Wireless Propagation Letters, 2016(15):1016-1019.
- [9] Singh U, Salgotra R. Optimal synthesis of linear antenna arrays using modified spider monkey optimization [J]. Arabian Journal for Science and Engineering, 2016, 41(8):2957-2973.

- [10] 杜振鑫, 王兆青. 一种改进的动态改变惯性权重的粒子群算法 [J]. 微电子学与计算机, 2011, 28(3):85-88.

作者简介:

姜 爽 女, (1992-), 硕士研究生. 研究方向为智能算法. E-mail:13302139306@163.COM.

林 丹 男, (1968-), 博士, 教授. 研究方向为智能算法、车辆调度问题.