

基于自适应权重和模拟退火的 鲸鱼优化算法

褚鼎立, 陈 红, 王旭光

(国防科技大学电子对抗学院, 61477 部队, 安徽合肥 230037)

摘 要: 针对鲸鱼优化算法容易陷入局部极值和收敛速度慢的问题, 提出了一种结合自适应权重和模拟退火的鲸鱼优化算法. 通过改进的自适应权重策略来调整算法的收敛速度, 通过模拟退火增强鲸鱼优化算法的全局寻优能力. 仿真实验中计算了 18 个测试函数, 对比了粒子群算法、海豚回声定位算法和标准鲸鱼算法并进行统计分析, 同时比较了单独结合自适应权重和模拟退火对鲸鱼优化的影响, 结果表明, 改进的算法在测试函数的极值计算中, 计算精度和收敛速度方面都有了明显提升, 验证了改进算法的有效性.

关键词: 智能优化算法; 鲸鱼优化算法; 自适应权重; 模拟退火算法

中图分类号: TP301.6 **文献标识码:** A **文章编号:** 0372-2112 (2019)05-0992-08

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2019.05.003

Whale Optimization Algorithm Based on Adaptive Weight and Simulated Annealing

CHU Ding-li, CHEN Hong, WANG Xu-guang

(61477 Bridge and Institute of Electronic Countermeasure, National University of Defense Technology, Hefei, Anhui 230037, China)

Abstract: Aiming at the problem that whale optimization algorithm is easy to fall into local extreme value and slow convergence speed, this paper proposes a whale optimization algorithm based on adaptive weight and simulated annealing. The improved convergence weight strategy is used to adjust the convergence speed of the algorithm, and the global optimization ability of the whale optimization algorithm is enhanced by simulated annealing. In the simulation experiment, 18 test functions were calculated and the genetic algorithm, the particle swarm optimization algorithm and the standard whale algorithm were compared and statistically analyzed. At the same time, the influence of the adaptive weight and simulated annealing on the whale optimization is compared. The results show that the improved algorithm has a significant improvement in the calculation of the extremum of the test function, and the effectiveness of the improved algorithm is verified.

Key words: intelligent optimization algorithm; whale optimization algorithm (WOA); adaptive weight; simulated annealing (SA)

1 引言

智能优化算法来源于对自然界生物行为的模拟, 广泛应用于模式识别、生产调度、系统控制、信号处理、计算机工程等众多领域. 具有代表性的智能优化算法有遗传算法^[1]、粒子群算法^[2]、免疫算法^[3]、蚁群算法^[4]、鱼群算法^[5]等. 这些算法在处理复杂优化问题上各自存在性能的缺陷, 如遗传算法编程实现比较复杂且对初始种群的选择有一定的依赖性、粒子群算法容

易产生早熟收敛并被证明算法不是全局收敛^[6]、蚁群算法和鱼群算法参数设置复杂, 如果参数设置不当, 容易偏离优质解. 后续学者也在算法基础上提出了一系列改进措施^[7,8], 避免算法陷入局部最优.

2016 年, 澳大利亚学者 Mirjalili 观测鲸鱼捕食行为后, 提出了一种新的智能优化算法——鲸鱼优化算法 (Whale Optimization Algorithm, WOA)^[9], 该算法通过模拟鲸鱼的捕食行为实现对复杂优化问题的寻优. 针对算法陷入局部极值和收敛速度问题, 学者们提出了一些改

进的方法.例如,文献[10]提出一种基于非线性收敛因子的改进鲸鱼优化算法,在搜索空间中利用对立学习策略进行初始化鲸鱼个体位置,并利用一种随机进化迭代次数非线性变化的收敛因子更新公式以协调 WOA 算法的探索 and 开发能力,该文献对高维度的测试函数进行计算,展现出良好的计算性能;文献[11]受文献[12]的启发,将惯性权重引入鲸鱼优化算法,以此提升算法的收敛速度,同时引入文献[13]提出的柯西高斯变异算子,进一步提升了算法的全局搜索能力,计算精度有了提高,但从文献[11]中的数据统计来看,虽然部分测试函数的性能有所提高,但对其中一部分测试函数的计算能力反而下降,依然存在容易陷入局部最优的问题.

本文在前人研究工作的基础上,针对鲸鱼优化算法陷入局部极值和收敛速度慢的问题,提出一种基于自适应权重和模拟退火(Simulated Annealing, SA)的鲸鱼优化算法.改进的算法通过一定概率接受劣质解来提升算法的全局勘探能力,同时对自适应权重做了改进工作,仿真结果表明,改进的算法(W-SA-WOA)相较于粒子群算法(PSO)、海豚回声定位算法(DE)^[14]以及标准鲸鱼优化算法(WOA)计算精度有了进一步提升.

2 标准鲸鱼优化算法

WOA 算法结构简便、调节参数少,其数学模型有三种:包围猎物、发泡网攻击和搜索捕食.下面是三种更新方式的介绍.

2.1 包围猎物

WOA 算法假设当前最佳的候选解是目标猎物位置或最接近目标猎物的位置,在定义最优位置后,其他鲸鱼将向最优位置靠近,位置更新的数学表达式为

$$X(j+1) = X^*(j) - A \cdot D \quad (1)$$

$$D = |C \cdot X^*(j) - X(j)| \quad (2)$$

式中: A 和 C 为系数变量; $X^*(j)$ 为当前最佳鲸鱼位置; $X(j)$ 表示当前鲸鱼位置; j 表示迭代次数.

计算系数变量 A 和 C :

$$A = 2a \times r - a \quad (3)$$

$$C = 2r \quad (4)$$

式中, a 作为一个变量,取值范围在 2 到 0 之间,并呈递减形式,以 $a = 2 - \frac{2j}{M}$ 更新,其中 M 为最大迭代次数; r 为 $[0,1]$ 上的随机数.

2.2 发泡网攻击

为了模拟座头鲸吐气泡捕食行为,数学模型建立如下:

(a) 收缩包围:即减少式(3)中的 a 值.在这个过程中, A 的取值范围是 $[-a, a]$,一旦这个范围发生变动, a 也会相应的改变,若 A 在 $[-1, 1]$ 范围内,鲸鱼

的游动位置将确定在当前位置和猎物位置之间,这表现出座头鲸的局部搜索能力.包围过程根据式(1)进行更新.

(b) 螺旋式位置更新:鲸鱼以螺旋式的移动方式向当前最优个体靠近,将螺旋式数学模型引入,建模如下:

$$X(j+1) = D' e^{bl} \cos(2\pi l) + X^*(j) \quad (5)$$

$$D' = |X_j^* - X_j| \quad (6)$$

式中: D' 为表示的是第 i 头鲸当前位置和当前最优个体之间的距离, b 为对数螺旋形状常数; l 是 $[-1, 1]$ 上的随机数.

座头鲸在围捕猎物时不仅以螺旋形态游向猎物,还要收缩包围圈,以 50% 的概率作为阈值来更新鲸鱼位置.公式为:

$$X(j+1) = \begin{cases} X^*(j) - A \cdot D, & p < 0.5 \\ D' e^{bl} \cos(2\pi l) + X^*(j), & p \geq 0.5 \end{cases} \quad (7)$$

式中 p 为 $[0, 1]$ 上的随机数.

2.3 搜索捕食

座头鲸除选择气泡网法搜索策略之外,随机搜索猎物也是一种可行方案,更新时以系数 A 的变化为基础,若 A 超出 $[-1, 1]$ 范围,距离数据 D 随机更新,鲸鱼个体为了找寻猎物,就会偏离原本的目标猎物,使得捕食能力提高,使得 WOA 有一定的全局搜索性能.建模如下:

$$D = |C \cdot X_{\text{rand}} - X(j)| \quad (8)$$

$$X(j+1) = X_{\text{rand}} - A \cdot D \quad (9)$$

注意,这里的 X_{rand} 不是随机的位置,而是当前种群中随机一个鲸鱼个体所在的位置.

2.4 性能分析

鲸鱼优化算法寻优方式中,包围阶段和螺旋式更新位置是进行局部寻优操作,以式(1)和式(5)的形式更新当前的位置,式(1)中的参数 A 是用来对鲸鱼寻优方式进行调节的阈值,不具备调节算法收敛速度的惯性权值的功能,即收敛速度方面可以通过引入自适应权重来提升,与常用的结合自适应权重的粒子群算法^[15]不同的是,位置更新是通过最优个体的位置变动来更新此时鲸鱼个体的位置,所以是将权重与最优鲸鱼位置相结合来调整当前个体的位置.在全局搜索方面,算法依靠式(9)来扩大搜索的范围,具备一定的全局寻优能力,但对于多峰函数的寻优能力不足,说明仍存在陷入局部极值的问题.对此,引入模拟退火策略,使算法以一定概率接收较差点,具有跳出局部最优解的能力.

3 鲸鱼优化算法的改进

为了改善鲸鱼优化算法的全局寻优能力和收敛精度,本文提出了两点改进措施,一方面引入自适应权重,提升算法的局部寻优能力;另一方面结合模拟退火算法,通过接受较差点来提升全局寻优能力.

3.1 自适应惯性权重

权重因子^[16]是平衡算法全局勘探能力和局部寻优能力的关键因素。

本文借鉴文献[17]提出的自适应权重策略,并做了一定改进,由于鲸鱼优化的更新方式与粒子群不同,鲸鱼优化在更新时是以最优鲸鱼的位置为基础更新的,自适应权重加在头鲸上,而不是像粒子群那样加在步长上,所以对权重的值作了调整,越靠近最优位置时,对头鲸的扰动越小,具体操作如下:在迭代过程中,将鲸鱼个体的适应度 $f(j)$ 按升序排序,然后分成两半分别求平均适应度 f_{avg1} 和 f_{avg2} ,其中 $f_{avg1} < f_{avg2}$ 。将当前鲸鱼个体的适应度分别与 f_{avg1} 和 f_{avg2} 作比较,将惯性权重分类如下:

(1) $f(j)$ 小于等于 f_{avg1} 。

当前鲸鱼个体的适应度优于较优群体的平均适应度,说明该个体是种群中位置占优。此时使最优鲸鱼个体在较小的范围内抖动有利于探索局部最优值,故 ω 取值为(0.8, 1.2)之间的随机数。

(2) $f(j)$ 大于等于 f_{avg2} 。

当前鲸鱼个体的适应度差于较差群体的平均适应度,说明该个体位置不理想,此时为了拓宽全局寻优能力, ω 以50%的概率取为(0.3, 0.6)或(1.3, 1.6)之间的随机数,有利于该个体能够寻找其他位置的猎物,增强全局寻优能力。

(3) $f(j)$ 介于 f_{avg1} 与 f_{avg2} 之间。

当前鲸鱼个体处于群体中的一般位置,让它按原算法向最优位置靠近即可,此时 ω 取值为1。

采用这种随机惯性权重策略^[18],使权重不再取固定值,而是根据适应度的大小随机调整权值,使鲸鱼在搜寻猎物过程中更加多样化,能改善算法的求解精度和收敛速度,不易错过全局最优解。

3.2 模拟退火策略

模拟退火(Simulated Annealing, SA)算法最早的思想由Metropolis在1953年提出,1983年由Kirkpatrick等人成功引入组合优化领域^[19]。SA算法来源于对固体退火降温过程的模拟,特点是能以一定的概率接受劣质解,从而在一定程度跳出局部最优。

根据Metropolis准则,粒子在温度 T 时趋于平衡的概率为 $\exp(-\Delta E/KT)$,其中 E 是温度 T 时的内能, ΔE 是内能的改变量, K 为玻尔兹曼常数。为了方便计算,接受劣质解的概率设置为:

$$P = \exp\left(-\frac{f(X_{j+1}) - f(X_j)}{t}\right) \quad (10)$$

式中, $f(X_j)$ 为第 j 次迭代时的适应度, t 为第 j 次迭代时的温度。SA算法初始温度定义为种群初始适应度最大值与最小值的差值,当 $f(X_{j+1}) < f(X_j)$ 时,接受 X_{j+1} ,否则以概率 P 接受劣质解 X_j 。本文将模拟退火思想引入到鲸鱼

优化算法中,进一步提升鲸鱼算法的全局寻优能力。

3.3 基于模拟退火和自适应权重的鲸鱼优化算法

正如前文的分析,鲸鱼优化算法有固有的缺陷——全局寻优能力不足,模拟退火算法在全局寻优上有着天然的优势,而且自适应权重对于局部寻优也有明显的帮助。鲸鱼算法在局部寻优时,引入3.1节改进的自适应权重因子 ω ,将式(1)和式(5)改进如下:

$$X(j+1) = \omega * X^*(j) - A * D \quad (11)$$

$$X(j+1) = \omega * X^*(j) + D' e^{bl} \cos(2\pi l) \quad (12)$$

在全局寻优方面,将模拟退火算法与鲸鱼优化算法并行,在鲸鱼算法求解之后再行退火处理,退火时重新产生一个鲸鱼种群,用于对原种群位置进行调整,式(10)中的概率调整如下:

$$P = \exp\left(-\frac{f_{\text{new}}(X_j) - f(X_j)}{t}\right) \quad (13)$$

式中, $f_{\text{new}}(X_j)$ 是模拟退火阶段产生的新种群中第 j 个鲸鱼个体的适应度,通过新旧位置适应度的差值来调整对较差解的接受概率 P 。

改进算法(W-SA-WOA)的具体步骤如下:

(1)算法参数初始化。将测试函数的自变量作为鲸鱼个体的位置信息 $X = [X_1, \dots, X_n]$,在解空间内,对种群位置随机初始化,同时初始化参数,包括种群数目 N 、自适应权重因子 ω_0 、对数螺旋形状常数 b 、随机数 l 、初始迭代次数 j 、最大迭代次数 M 。

(2)计算种群的适应度,计算模拟退火算法初始温度 t_0 ,找到并记录种群中的最优个体位置 X^* 。

(3)进入迭代阶段。若 $j < M$,更新 a 、 A 、 C 、 l 和 p 。对当前种群适应度进行排序,计算平均适应度 f_{avg1} 和 f_{avg2} ,计算得到此时的自适应权重 ω 。

(4)当 $p < 0.5$ 时,若 $A < 1$,通过式(11)重新确定鲸鱼位置;若 $A \geq 1$,需要在当前群体范围内随机确定鲸鱼个体位置 X_{rand} ,通过式(9)更新当前鲸鱼位置。

(5)当 $p \geq 0.5$ 时,通过式(12)重新确定鲸鱼个体位置。

(6)进入模拟退火阶段。定义一个新的鲸群,随机化鲸鱼个体位置,计算新种群的适应度。

(7)计算更新后的原种群适应度,若新种群中鲸鱼个体的适应度优于原种群的鲸鱼,则用新种群中的鲸鱼位置替代原种群中鲸鱼的位置。

(8)若新种群中鲸鱼个体的适应度差于原种群的鲸鱼,则以式(13)中的概率 P 接受新种群的鲸鱼位置。

(9)退温操作: $t = 0.99 * t$ 。

(10)记录此时最佳鲸鱼个体的位置 X^* 及其适应度。如果 $j > M$,就到步骤11;否则, $j = j + 1$,重复步骤3~10,直到条件满足为止。

(11) 输出最优个体位置 X^* 及其适应度.

3.4 改进算法时间复杂度分析

本文算法主要由初始化种群、自适应权重、鲸鱼位置更新以及模拟退火策略组成. 当种群数目为 N , 优化问题的维度为 D 时, 本文算法时间复杂度分析如下: 初始化种群的计算复杂度为 $O(ND)$; 迭代过程, 计算自适应权重, 对各个种群的适应度进行排序, 计算复杂度为 $O(N\log N)$, 接着计算每个种群对应的权重, 复杂度为 $O(N)$; 在鲸鱼位置更新环节, 计算复杂度为 $O(ND)$; 模拟退火操作的计算复杂度为 $O(N)$. 因此每一次迭代过程中, 算法的计算复杂度为 $O(3N + N\log N + ND)$, 近似为 $O(N\log N + ND)$, 高于标准鲸鱼优化算法的复杂度 $O(ND)$. 当优化问题的维度较高时, 改进算法的计算复杂度近似为 $O(ND)$, 与标准鲸鱼优化算法一致. 表 1 中列出了遗传算法 (GA)、粒子群算法 (PSO)、标准鲸鱼算法 (WOA) 及本文算法 (W-SA-WOA) 的计算复杂度.

表 1 算法复杂度

算法	计算复杂度
GA	$O(ND + N^2)$
PSO	$O(ND + N)$
WOA	$O(ND)$
W-SA-WOA	$O(N\log N + ND)$

由表 1 可知, 算法的计算复杂度与种群数目和优化问题的维度决定, 根据大 O 的定义^[20], 当优化问题的维度较高时 ($D \gg N$), 四种算法的计算复杂度都为 $O(ND)$, 属于同一数量级.

3.5 改进算法渐进性分析

本节针对最小化寻优问题对改进算法的渐进性进行分析. 设有限非空集合 S 为解空间, 其中存在最优解集合 $\Omega = \{x^* | x^* \in S, f(x^*) < \varepsilon\}$, 其中 $f(\cdot)$ 为目标函数, ε 是可接受的适应度. 下面对算法的适应度渐进性和全局最优解渐进性进行证明.

定理 1 改进鲸鱼优化算法具有适应度渐进性. 令第 t 次迭代时算法找到的最优鲸鱼位置为 x_t^* , $\{d_t | d_t = f(x_t^*) - f(x^*), 1 \leq t \leq T\}$ 为算法生成的非负随机过程. 当 $f(x_t^*) > f(x^*)$ 时, 存在一个正常数 τ , 使得 $E(d_{t+1}) \leq E(d_t) - \tau$, 则该优化算法具有适应度渐进性.

证明 由于改进鲸鱼优化算法通过优胜劣汰的机制选出最优鲸鱼个体, 所以 $P(f(x_{t+1}^*) - f(x_t^*) > 0) = 0$ 且 $f(x_t^*) > f(x^*)$. 又因为鲸鱼种群通过三种方式更新位置, 更新过程中都有随机因素存在, 易得 $P(f(x_{t+1}^*) = f(x_t^*)) \neq 1$, 由此可知 $P(f(x_{t+1}^*) - f(x_t^*) < 0) > 0$.

令 $E[f(x_{t+1}^*) - f(x_t^*)] = -\tau_{t+1}$, 由上述分析可知 $\tau_{t+1} > 0$, 则

$$E[d_{t+1} - d_t] = E[(f(x_{t+1}^*) - f(x^*)) - (f(x_t^*) - f(x^*))]$$

$$\begin{aligned} &= E[f(x_{t+1}^*) - f(x_t^*)] \\ &= -\tau_{t+1} \end{aligned}$$

即 $E(d_{t+1}) = E(d_t) - \tau_{t+1}$.

令 $\tau = \min\{\tau_1, \tau_2, \dots, \tau_T\}$,

则有 $E(d_{t+1}) \leq E(d_t) - \tau$.

证毕.

定理 2 改进鲸鱼优化算法具有全局最优解渐进性. 令第 t 次迭代时算法找到的距离全局最优解最近的鲸鱼位置为 x'_t , $\{D_t | D_t = f(x'_t) - f(x^*), 1 \leq t \leq T\}$ 为算法生成的非负随机过程, 当 $x'_t \neq x^*$ 时, 存在一个正常数 ε , 使得 $E(D_{t+1}) = E(D_t) - \varepsilon$.

证明 第 t 次鲸鱼寻优找到的最优解集合为 S_t , 经过模拟退火后生成新的集合 S'_t , 此时最优解的搜索空间为 $S_t \cup S'_t$. 虽然在模拟退火过程中, 新产生的最优解不一定距离全局最优解更近, 但寻优时会记录之前最优位置, 即不会覆盖原始更优位置, 所以退火后新解距离全局最优解的距离不会增加即 $P(D_{t+1} - D_t > 0) = 0$. 当 $x'_t \neq x^*$ 时, 下一次迭代时, 通过模拟退火以一定概率跳变到比 x'_t 更优的位置. 因此 $P(D_{t+1} - D_t = 0) \neq 1$, $P(D_{t+1} - D_t < 0) > 0$, 令 $E(D_{t+1} - D_t) = -\varepsilon_{t+1}$, 易证 $\varepsilon_{t+1} > 0$, 即 $E(D_{t+1}) = E(D_t) - \varepsilon_{t+1}$, 令 $\varepsilon = \min\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_T\}$, 则 $E(D_{t+1}) = E(D_t) - \varepsilon$.

证毕.

4 函数测试与结果分析

4.1 参数设置

本节基于 Matlab2014a 平台仿真验证并分析 W-SA-WOA 算法对测试函数的计算性能. 所有比较算法的种群均设置为 30、迭代次数为 500, 共进行 50 次实验, 粒子群算法中, 两个学习因子都取为 1.

4.2 测试函数

为验证算法的性能, 本文选取了单峰、多峰和固定维度等共计 18 个标准测试函数^[21]进行计算, 测试函数见表 2、3、4.

表 2 单峰函数

	测试函数
f_1	Sphere Model
f_2	Schwefel's Problem 2.22
f_3	Schwefel's Problem 1.2
f_4	Schwefel's Problem 2.21
f_5	Generalized Rosenbrock's Function
f_6	Step Function
f_7	Quartic Function i. e. Niose
f_8	Generalized Schwefel's Problem 2.26

表 3 多峰函数

	测试函数
f_9	Generalized Rastrigin's Function
f_{10}	Ackley's Function
f_{11}	Generalized Griewank Function
f_{12}	Generalized Penalized Function
f_{13}	Generalized Penalized 2 Function

4.3 实验结果与分析

将 W-SA-WOA 算法与 PSO 算法、DE 算法以及 WOA 算法进行对比试验,共进行 50 次实验,并将这 50 次实验结果取平均值,并计算标准差,实验结果见表 5。

表 5 测试结果

函数	PSO		DE		WOA		W-SA-WOA	
	平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差
f_1	$3.05\text{E}-1$	$8.73\text{E}+02$	$8.2\text{E}-14$	$5.9\text{E}-14$	$1.41\text{E}-30$	$4.9\text{E}-30$	0	0
f_2	3.18136	3.78	$1.5\text{E}-09$	$9.9\text{E}-10$	$1.06\text{E}-21$	$2.39\text{E}-21$	$2.576\text{E}-121$	$6.682\text{E}-121$
f_3	$4.66\text{E}+02$	$8.23\text{E}+03$	$6.8\text{E}-11$	$7.4\text{E}-11$	$5.39\text{E}-07$	$2.93\text{E}-06$	0	0
f_4	4.5597	8.09	0	0	0.07258	0.3975	$3.569\text{E}-94$	$4.904\text{E}-94$
f_5	209.5869	70953.32	0	0	27.8656	0.7636	27.3357	0.2956
f_6	0.4934	1346.0036	0	0	3.1163	0.5324	0.0271	0.0201
f_7	0.2608	0.52	0.00463	0.0012	0.001425	0.00115	$1.17\text{E}-04$	$1.0\text{E}-04+$
f_8	-3939.61	504.84	-11080.1	574.7	-5080.76	695.797	-12447	873.3422
f_9	79.6784	20.26	69.2	38.8	0	0	0	0
f_{10}	2.2960	2.35	$9.7\text{E}-08$	$4.2\text{E}-08$	7.4043	9.8976	$3.02\text{E}-15$	$1.77\text{E}-15$
f_{11}	1.8411	10.50	0	0	0.000289	0.00159	0.0015	0.0073
f_{12}	1.4876	56820.5	$7.9\text{E}-15$	$8\text{E}-15$	0.3397	0.2149	0.0785	$2.403\text{E}-08$
f_{13}	2.1552	826407.4	$5.1\text{E}-14$	$4.8\text{E}-14$	1.8890	0.2661	0.0421	0.0289
f_{14}	3.8678	$2.51\text{E}-13$	0.99800	$3.3\text{E}-16$	2.11197	2.4986	1.0264	$1.68\text{E}-01$
f_{15}	0.0008	0.017	$4.5\text{E}-14$	0.00033	0.00057	0.000324	$4.92\text{E}-04$	$2.08\text{E}-04$
f_{16}	3	14.432	3	$2\text{E}-15$	3	$4.22\text{E}-15$	3	$1.547\text{E}-10$
f_{17}	-3.863	0.10284	N/A	N/A	-3.8562	0.0027	-3.7823	0.0380
f_{18}	-3.2663	0.3781	N/A	N/A	-2.9811	0.3767	-3.2708	0.0778

表 6 测试结果的无重复双因素分析 ($\alpha=0.05$)

F	P-value	F _{crit}
1.1227	0.3486	2.7862

表 7 Friedman 检验

算法	秩均值
W-SA-WOA	1.72
DE	1.97
WOA	2.78
PSO	3.53

同时,对计算结果进行无重复双因素分析,并通过 Friedman 非参数的统计方法对结果进行统计分析,其中显著性水平设定为 5%,实验结果见表 6 和表 7。

表 4 固定维度函数

	测试函数
f_{14}	Shekl's Foxholes Function
f_{15}	Kowalik's Function
f_{16}	Goldstein-Price Function
f_{17}	Hartman's Function
f_{18}	Hartman's Function 2

表 5 数据部分来自文献[9]、文献[10]和文献[22],N/A 表示算法对此函数不适用,无法计算出正确数值。由表 5 的计算结果可见,50 次的函数测试中,W-SA-WOA 算法在对 f_1 f_3 f_9 f_{16} 的计算中都收敛到了最小值,标准差为 0 或接近于 0,证明算法稳定性非常高。在对其他函数的计算中,虽然没有收敛到最小值,相较于 PSO 算法,除了 f_{17} 函数值略差,其余计算都明显优于 PSO 算法;相较于 DE 算法,除了 $f_4 \sim f_6$ $f_{11} \sim f_{13}$ f_{15} 的计算略差,其余函数的计算都是优于 DE 算法;相较于 WOA 算法,除了 f_{17} 的计算略差,其余均优于 WOA 算

法.改进算法在函数计算性能优异的同时,标准差也是相对其他函数更低.综合函数计算的结果来看, f_5 的计算结果误差较大,其余函数的计算结果都十分贴近于理论值,W-SA-WOA 算法在计算精度和稳定性上对比 WOA 算法有了明显的提升,证明了改进算法在全局寻优上的有效性.

根据数据统计中无重复双因素分析法,由表 6 知, $F < F_{crit}$,又因为 $P\text{-value} > 0.05$,所以四种算法在 0.05

水平上不存在显著性差异,从表 7 的 Friedman 检验结果来看,W-SA-WOA 算法的秩均值最小,性能强于其他三种算法.基于统计分析,本文改进方法性能是提升的.

为进一步验证自适应权重与模拟退火相结合带来的优势,分别进行鲸鱼优化结合自适应权重(W-WOA)和鲸鱼优化结合模拟退火(SA-WOA)的测试,共进行 50 次实验,实验结果见表 8.

表 8 测试结果

函数	WOA		W-WOA		SA-WOA		W-SA-WOA	
	平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差
f_1	1.41E-30	4.9E-30	1.56E-138	8.31E-138	0	0	0	0
f_2	1.06E-21	2.39E-21	5.08E-80	2.18E-79	0	0	2.576E-121	6.682E-121
f_3	5.39E-07	2.93E-06	1.74E-58	8.51E-58	0	0	0	0
f_4	0.07258	0.3975	5.37E-56	1.77E-55	2.14E-52	1.06E-51	3.569E-94	4.904E-94
f_5	27.8656	0.7636	27.2786	0.3685	28.1546	0.3941	27.3357	0.2956
f_6	3.1163	0.5324	0.0332	0.0174	1.0473	0.3726	0.0271	0.0201
f_7	0.001425	0.00115	4.671E-04	3.677E-04	4.06E-05	3.55E-05	1.17E-04	1.0E-04
f_8	-5080.76	695.797	-11252	1749.5	-12173	1029.9	-12447	873.3422
f_9	0	0	0	0	0	0	0	0
f_{10}	7.4043	9.8976	3.43E-15	1.63E-15	8.88E-16	0	3.02E-15	1.77E-15
f_{11}	0.000289	0.00159	0.0048	0.02	0	0	0.0015	0.0073
f_{12}	0.3397	0.2149	0.0785	1.10E-07	0.0901	0.0126	0.0785	2.403E-08
f_{13}	1.8890	0.2661	0.0737	0.0603	0.6878	0.2903	0.0421	0.0289
f_{14}	2.11197	2.4986	1.5083	0.8464	1.0601	0.2379	1.0264	1.68E-01
f_{15}	0.00057	0.000324	6.85E-04	4.64E-04	6.32E-04	2.66E-04	4.92E-04	2.08E-04
f_{16}	3	4.22E-15	3	4.87E-05	3.0018	0.0034	3	1.547E-10
f_{17}	-3.8562	0.0027	-0.3004	2.11E-04	-3.5871	0.0564	-3.7823	0.0380
f_{18}	-2.9811	0.3767	-3.2431	0.0761	-3.1869	0.0621	-3.2708	0.0778

表 9 测试结果的无重复双因素分析($\alpha=0.05$)

F	P-value	Fcrit
1.0037	0.3988	2.7862

表 10 Friedman 检验

算法	秩均值
W-SA-WOA	1.61
SA-WOA	2.36
W-WOA	2.67
WOA	3.36

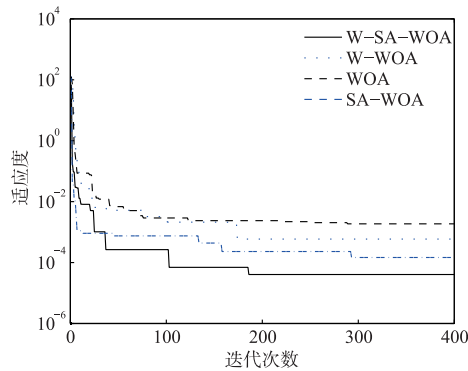
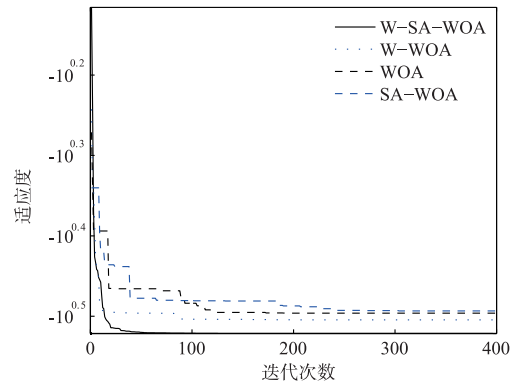
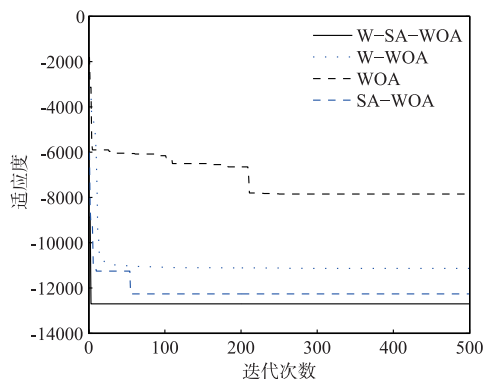
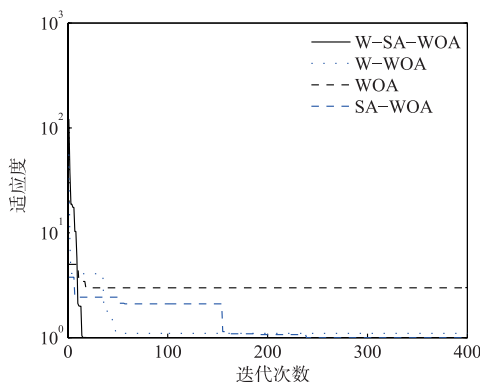
从表 8 的计算结果来看,W-WOA 算法和 SA-WOA 算法相较于 WOA 都有不同程度的性能提升,两种改进算法都有明显的优势,而将两种改进措施结合后,W-SA-WOA 算法相较于 W-WOA 算法,除了 f_{12} 计算结果相同,其余函数计算精度均明显提高;相较于 SA-WOA 算法,除了 f_2 f_7 f_{10} f_{11} 计算结果略差,其余函数计算精度明显提高.值得注意的是,在 f_4 的计算中,W-SA-WOA

算法的计算精度相对于 W-WOA 和 SA-WOA 算法有 40 个数量级的提升;在 f_8 的计算中,W-WOA 和 SA-WOA 算法计算结果误差较大,而 W-SA-WOA 算法计算结果却更加精确,接近于理论值;在 f_6 f_{13} f_{14} f_{17} 等函数的计算中,W-WOA 和 SA-WOA 算法各有优劣,而 W-SA-WOA 算法的计算精度没有折中,而是更精确,说明自适应权重与模拟退火结合到鲸鱼优化中取得了“1+1>2”的效果.

由表 9 知, $F < F_{crit}$,由于 $P\text{-value} > 0.05$,所以四种算法在 0.05 水平上不存在显著性差异,从表 10 的 Friedman 检验结果来看,W-SA-WOA 算法的秩均值最小,性能强于 W-WOA 和 SA-WOA 算法.

为了能直观展现算法在测试函数上的收敛情况,选取结果与 50 次测试平均值较为接近的一次测试,同时对比 WOA 算法、W-WOA 算法和 SA-WOA 算法,绘制收敛曲线.为方便比较,本文选取了四个测试函数计算的收敛曲线,如图 1~图 4 所示.

为了使图中的数据波动更清楚,除图 2 外,Y 轴均

图1 函数 f_7 收敛图图4 函数 f_{18} 收敛图图2 函数 f_6 收敛图图3 函数 f_{14} 收敛图

采用对数坐标轴,图2正常绘图.由图1可见,虽然W-SA-WOA算法收敛速度较慢,但能够寻优到更小值.图2~图4中,W-SA-WOA算法收敛速度较快,而且计算精度更高,说明自适应权重与模拟退火相结合后效果更佳.W-SA-WOA算法在计算初期能够快速收敛,这得益于自适应权重能够合理调整步长,后期模拟退火策略增加全局勘探范围,使计算精度更高,二者结合相得益彰.

5 结论

为解决鲸鱼优化算法容易陷入局部极值和收敛速度慢的问题,提出了一种基于自适应权重和模拟退火

的鲸鱼优化算法.通过改进的自适应权重,加快局部收敛速度,并且引入模拟退火策略,使鲸鱼算法融合了模拟退火的概率突跳性,改进算法在全局寻优能力和收敛速度方面有了明显的提升.仿真实验中对18个测试函数进行计算,结果表明,相比于粒子群算法、海豚回声定位算法和标准鲸鱼优化算法,改进算法在计算精度和收敛速度上有显著提升,同时也比较了单独结合自适应权重和模拟退火对鲸鱼优化的影响,统计分析和计算结果均证明改进的算法对这两种策略取长补短,在收敛速度加快的同时全局勘探能力也有所提升,验证了改进策略的有效性.

参考文献

- [1] Coit D. Genetic algorithms and engineering design[J]. Engineering Economist, 1998, 43(4): 379-381.
- [2] Eberhart R, Kennedy J. A new optimizer using particle swarm theory[A]. MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science[C]. IEEE, 2002. 39-43.
- [3] 王磊, 潘进, 焦李成. 免疫算法[J]. 电子学报, 2000, 28(7): 96-101.
WANG L, PAN J, JIAO L C. The immune algorithm[J]. Acta Electronica Sinica, 2000, 28(7): 96-101. (in Chinese)
- [4] Dorigo M, Maniezzo V, Colomni A. Ant system: optimization by a colony of cooperating agents. [J]. IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A, 1996, 26(1): 29.
- [5] 李晓磊. 一种新型的智能优化方法-人工鱼群算法[D]. 浙江大学, 2003.
- [6] 任子晖, 王坚, 高岳林. 马尔科夫链的粒子群优化算法全局收敛性分析[J]. 控制理论与应用, 2011, 28(4): 462-466.
REN Z H, WANG J, GAO Y L. The global convergence analysis of particle swarm optimization based on Markov chain[J]. Control Theory & Applications, 2011, 28(4):

- 462 – 466. (in Chinese)
- [7] 陈侃松,阮玉龙,戴磊,等. 区域分割的自适应变异粒子群算法[J]. 电子学报,2017,45(8):1849 – 1855.
Chen K S, Ruan Y L, Dai L, et al. Regional-segmentation self-adapting variation particle swarm optimization[J]. Acta Electronica Sinica, 2017, 45(8):1849 – 1855. (in Chinese)
- [8] 申元霞,曾传华,王喜凤,等. 并行协作骨干粒子群优化算法[J]. 电子学报,2016,44(7):1643 – 1648.
SHEN Y X, ZENG C H, WANG X F, et al. A parallel-cooperative bare-bone particle swarm optimization algorithm [J]. Acta Electronica Sinica, 2016, 44(7):1643 – 1648. (in Chinese)
- [9] Mirjalili S, Lewis A. The whale optimization algorithm [J]. Advances in Engineering Software, 2016, 95:51 – 67.
- [10] 龙文,蔡绍洪,焦建军,等. 求解大规模优化问题的改进鲸鱼优化算法[J]. 系统工程理论与实践,2017,37(11):2983 – 2994.
LONG W, CAI S H, JIAO J J, et al. Improved whale optimization algorithm for large scale optimization problems [J]. Systems Engineering-Theory & Practice, 2017, 37(11):2983 – 2994. (in Chinese)
- [11] 郭振洲,王平,马云峰,等. 基于自适应权重和柯西变异的鲸鱼优化算法[J]. 微电子学与计算机,2017,34(9):20 – 25.
GUO Z Z, WANG, MA Y F, et al. Whale optimization algorithm based on adaptive weight and Cauchy mutation [J]. Microelectronics & Computer, 2017, 34(9):20 – 25. (in Chinese)
- [12] 赵志刚,黄树运,王伟倩. 基于随机惯性权重的简化粒子群优化算法[J]. 计算机应用研究,2014,31(2):361 – 363.
ZHAO Z G, HUANG S Y, WANG W Q. Simplified particle swarm optimization algorithm based on stochastic inertia weight[J]. Application Research of Computers, 2014, 31(2):361 – 363. (in Chinese)
- [13] 杜晓昕,张剑飞,郭媛,等. 基于柯西-高斯动态消减变异的果蝇优化算法研究[J]. 计算机工程与科学,2016,38(6):1171 – 1176.
- DU X X, ZHANG J F, GU Y, et al. A fruit fly optimization algorithm with Cauchy-Gaussian dynamic reduction mutation[J]. Computer Engineering & Science, 2016, 38(6):1171 – 1176. (in Chinese)
- [14] Kaveh A, Farhoudi N. A new optimization method: Dolphin echolocation[J]. Advances in Engineering Software, 2013, 59(5):53 – 70.
- [15] Ardizzon G, Cavazzini G, Pavesi G. Adaptive acceleration coefficients for a new search diversification strategy in particle swarm optimization algorithms [J]. Information Sciences, 2015, 299(C):337 – 378.
- [16] Shi Y, Eberhart R. Modified particle swarm optimizer [A]. Proc of IEEE ICEC Conference, Anchorage [C]. IEEE, 1998. 69 – 73.
- [17] 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(03):501 – 506.
ZHAO Z G, LIN Y J, YIN Z Y. A mean particle swarm optimization algorithm based on adaptive inertia weight [J]. Computer Engineering & Science, 2016, 38(03):501 – 506. (in Chinese)
- [18] Huang X, Zhang J, Zhan Z H. Faster particle swarm optimization with random inertia weight[J]. Computer Engineering & Design, 2009, 30(3):647 – 432.
- [19] 高尚. 模拟退火算法中的退火策略研究[J]. 航空计算技术,2002,32(4):20 – 22.
GAO S. Research on annealing strategy in simulated annealing algorithm [J]. Aeronautical Computing Technique, 2002, 32(4):20 – 22. (in Chinese)
- [20] Dasgupta S, Papadimitriou C H, Vazirani U V. Algorithms [M]. China Machine Press, 2009. 15 – 18.
- [21] Fang W, Sun J, Chen H, et al. A decentralized quantum-inspired particle swarm optimization algorithm with cellular structured population[J]. Information Sciences, 2016, 330(C):19 – 48.
- [22] Mirjalili S, Mirjalili S M, Lewis A. Grey Wolf Optimizer [J]. Advances in Engineering Software, 2014, 69(3):46 – 61.

作者简介



褚鼎立 男,1993 年出生,山东烟台人. 现为国防科技大学硕士研究生,主要研究方向为智能优化算法、盲源分离算法.
E-mail:1240699100@qq.com



陈红 女,1965 年 7 月出生,安徽东至人. 现为国防科技大学电子对抗学院教授、硕士生导师. 主要研究方向为通信信号处理和调制识别.
E-mail:hplanhf@126.com