

# 基于多目标浣熊优化算法的双向 长短期记忆神经网络预测

杨凯<sup>1</sup>, 苏艳萍<sup>2</sup>, 杜强<sup>3</sup>, 马丽玲<sup>3</sup>, 杨金钰<sup>3</sup>

(1. 国网宁夏电力有限公司经济技术研究院, 银川 750000; 2. 国网宁夏电力有限公司, 银川 750000;  
3. 宁夏回族自治区电力设计院有限公司, 银川 750000)

**摘要:** 为了提高双向长短期记忆神经网络的预测性能, 针对 BiLSTM 存在的预测精度低、预测结果不稳定的问题, 提出了一种新的多目标浣熊优化算法; 在浣熊优化算法的基础上, 通过改进探索与开发算子, 结合快速非支配排序与拥挤度距离计算方法建立精英浣熊保留策略, 实现单目标到多目标的改进; 基于所提算法, 以预测均方误差及预测误差方差为目标函数对 BiLSTM 超参数进行优化, 并建立 MOCO-A-BiLSTM 预测模型, 最终实现精确稳定预测; 将所提 MOCO-A-BiLSTM 预测模型在变电工程造价数据集上进行了仿真测试, 并与其他 3 种主流算法优化后的模型进行了对比; 结果表明, 所提 MOCO-A-BiLSTM 的平均百分比误差相比与 MOSSA-BiLSTM、NSGAIII-BiLSTM、MOMVO-BiLSTM 分别降低了 69.59%、58.43%、56.67%。

**关键词:** 浣熊优化算法; 多目标优化; 双向长短期记忆神经网络; 参数优化; 预测

## BiLSM Neural Network Prediction Based on Multi-objective COA

YANG Kai<sup>1</sup>, SU Yanping<sup>2</sup>, DU Qiang<sup>3</sup>, MA Liling<sup>3</sup>, YANG Jingyu<sup>3</sup>

(1. State Grid Ningxia Electric Power Co., Ltd., Economic and Technological Research Institute, Yinchuan 750000, China; 2. State Grid Ningxia Power Co., Ltd., Yinchuan 750000, China;  
3. Ningxia Hui Autonomous Region Electric Power Design Institute Co., Ltd., Yinchuan 750000, China)

**Abstract:** To enhance the predictive performance of bidirectional long short-term memory (BiLSTM) neural networks, and address the issues of low prediction accuracy and unstable results in BiLSTM, a novel multi-objective coati optimization algorithm (COA) is proposed. Based on the COA, an elite coati retention strategy is established by improving the exploration and exploitation operators and combining with fast non-dominated sorting and crowding distance calculation method, which achieves from single-objective to multi-objective optimization. The proposed algorithm is used to optimize the hyperparameters of BiLSTM with the predictions of mean square error and error variance as the objective functions, and build the MOCO-A-BiLSTM prediction model, ultimately achieving accurate and stable predictions. Through simulation and experiment on a substation engineering cost dataset, the proposed MOCO-A-BiLSTM prediction model is made a comparison with three other mainstream algorithm-optimized models. Experimental results show that the MOCO-A-BiLSTM reduces the average percentage error by 69.59%, 58.43%, and 56.67% than the MOSSA-BiLSTM, NSGAIII-BiLSTM, and MOMVO-BiLSTM, respectively.

**Keywords:** COA; multi-objective optimization; BiLSTM neural network; parameter optimization; prediction

## 0 引言

双向长短期记忆网络 (BiLSTM, bidirectional long

short-term memory neural network) 作为一种深度学习模型, 主要用于序列数据建模和预测。BiLSTM 在语音识别、时间序列预测以及交通流量预测等领域也得到广

收稿日期:2023-12-12; 修回日期:2024-01-03。

基金项目:宁夏回族自治区自然科学基金项目(22022AAC02078)。

作者简介:杨凯(1973-),男,大学本科,高级工程师。

通讯作者:杨金钰(1994-),男,大学本科,中级工程师。

引用格式:杨凯,苏艳萍,杜强,等. 基于多目标浣熊优化算法的双向长短期记忆神经网络预测[J]. 计算机测量与控制, 2025,33(1):36-44.

泛应用。其双向建模的特性使其能够有效处理序列数据中的时序信息,使其成为处理时序数据和复杂序列关系的有力工具。郭逸婕<sup>[1]</sup>等通过 BiLSTM 网络提取序列中的信息,实现对船舶未来轨迹的预测。张加劲<sup>[2]</sup>提出了一种基于注意力机制的卷积神经网络和双向长短期网络融合模型,对航空发动机的剩余使用寿命进行预测。然而 BiLSTM 的参数需要依靠个人经验设置,所以预测的结果往往存在预测精度低,训练周期长等问题。通过引入优化算法,可以提升模型的拟合效果与预测精度。刘英<sup>[3]</sup>等对空气质量指数进行预测,通过鲸鱼优化算法帮助 Bi-LSTM 模型寻找最优网络参数,并与 BiLSTM 和 LSTM 进行比较。相较对比模型,其  $R^2$  指标分别提高了 0.26% 和 1.98%。杜睿山<sup>[4]</sup>等建立粒子群算法优化了 BiLSTM 模型,并对储层岩石脆性指数进行预测。与传统方法相比,该模型展示出更高的准确性和实用性,证明了优化算法的有效性。王帅党<sup>[5]</sup>等通过灰狼优化算对 BiLSTM 的超参数进行了优化,并通过地区短期风功率进行了仿真测试。结果表明,该优化后的模型在非线性和数据维度较高的情况下,拟合度最高,精度最好。胡春安<sup>[6]</sup>等通过贝叶斯优化 BiLSTM 的第 1、2 隐含层神经元数目、学习率和批次进行寻优,根据寻优结果对猪肉的价格建立预测模型。刘相杰<sup>[7]</sup>等使用猎食者算法对 BiLSTM 进行参数优化,其平均绝对百分比误差比 LSTM 降低了 28%。优化算法的引入可以很大程度上提升 BiLSTM 预测模型的准确性,但是已有研究中所使用的优化算法大多为单目标优化算法,且提出较早。一方面这些算法无法深度挖掘 BiLSTM 预测模型的潜力,另一方面使用这些算法优化后的 BiLSTM 模型无法保证其预测稳定性。因此需要开发一种新型高效多目标优化算法对 BiLSTM 超参数进行优化,同时提升 BiLSTM 的预测准确性及稳定性。

多目标优化算法主要分为两大类,它们各自受到不同的灵感启发。首先,一类算法汲取自然界中生物的行为策略,其中包括蛇的交配行为和瞪羚的逃避策略。Hashim 等人<sup>[8]</sup>提出了一种新的元启发式算法,模拟了蛇的交配行为,并采用竞争性伴侣选择来不断更新个体。这种算法通过模仿自然界中蛇的繁衍策略实现了多目标优化。另外,Agushaka 等人<sup>[9]</sup>以瞪羚的逃避策略为灵感,将其类比为解决优化问题的策略,并提出了相关的算法。这种方法充分利用了瞪羚在面临危险时的生存本能,将其应用到多目标优化领域。其次,另一类算法则基于经典的数学方法和技巧,以解决多目标优化问题。Ahmadianfar 等人<sup>[10]</sup>引入了数学加权平均技术,以提高算法的收敛速度,并增强探索和开发能力。这种方法通过数学权重的调整来改进算法的性能,使其更适用于多目标问题的求解。然而,“无免费午餐定理”表明

了没有一种算法适用于所有优化问题,因此不同问题可能需要不同的方法和策略。Dehghani 等人<sup>[11]</sup>则模拟了浣熊的捕猎和逃跑策略,建立了探索和开发算子,并提出了浣熊优化算法(COA, coati optimization algorithm)。该算法在仿真实验中表现出色,尤其在无约束优化问题中表现出了更好的多样性与收敛性,显示了自然界中动物行为对于算法的启发作用。现实生活中的绝大多数问题都属于多目标优化问题,而预测模型的优化就是一个无约束的多目标问题。因此将 COA 算法扩展到多目标领域,并应用在 BiLSTM 预测模型中具有重要的研究价值。

基于上述问题,提出一种基于多目标浣熊算法(MOCOA, Multi-objective coati optimization algorithm)优化的 MOCOA-BiLSTM 预测模型。通过改进更新算子,引入快速非支配排序及拥挤度距离计算方法实现了 COA 算法由单目标到多目标的改进,并利用改进后的 COA 算法优化 BiLSTM 的超参数,建立 MOCOA-BiLSTM 预测模型,弥补了现有研究中 BiLSTM 存在的预测准确度低及预测不稳定的不足。

## 1 浣熊优化算法

浣熊优化算法是 Dehghani<sup>[11]</sup>等人在 2023 年提出的一种新的动物启发式单目标优化算法。其灵感来源于生活在美国西南部及南美洲浣熊的生存行为。单目标的浣熊优化算法主要通过模拟其捕猎与摆脱捕食者猎杀的过程来实现对浣熊位置的更新,通过不断地迭代最终找到优化问题的最优解。

### 1.1 浣熊优化算法 (COA)

浣熊算法主要包含两个阶段,分别为:模仿浣熊群对鬣蜥的捕猎过程的探索阶段;模拟浣熊群逃脱捕食者追捕过程的开发阶段。

#### 1.1.1 探索阶段

在探索阶段,算子通过模拟浣熊群团体协作的狩猎模式对决策空间进行探索。当发现一只栖息在树梢的鬣蜥时,浣熊群会分为两个团体。其中一个团体爬上树梢,利用集体力量将鬣蜥逼至地面。假设鬣蜥落地位置具有随机性,另一个团体在目标落地后会迅速朝目标进行追击。模拟该过程的算子定义如下:

$$X_i^{P1} : X_{i,j}^{P1} = x_{i,j} + r \times (Iguana_j - I \times x_{i,j}),$$

$$\text{for } i = 1, 2, \dots, \left\lceil \frac{N}{2} \right\rceil, \text{ and } j = 1, 2, \dots, v \quad (1)$$

式中,  $P1$  为探索阶段,  $Iguana$  表示鬣蜥的位置信息,  $r$  为  $[0, 1]$  之间的常数,  $I$  为  $\{1, 2\}$  集合中的随机整数,  $N$  为种群规模,  $v$  为个体的决策变量维数。假设鬣蜥落地位置具有随机性,落地后另一半浣熊开始朝猎物移动,进行围捕猎杀。这一过程可以用数学公式定义为:

$$Iguana^G : Iguana_j^G = lb_j + r \times (ub_j - lb_j), j = 1, 2, \dots, v. \quad (2)$$

$$X_i^{P1} : X_{i,j}^{P1} = \begin{cases} x_{i,j} + r \times (Iguana_j^G - I \times x_{i,j}), & F_{Iguana^c} < F_i, \\ x_{i,j} + r \times (x_{i,j} - Iguana_j^G), & F_{Iguana^c} \geq F_i \end{cases}$$

for  $i = \left\lfloor \frac{N}{2} \right\rfloor + 1, \left\lfloor \frac{N}{2} \right\rfloor + 2, \dots, N$  and  $j = 1, 2, \dots, \nu$  (3)

式中, P1 为探索阶段, *Iguana* 为最佳个体位置, *r* 为 [0, 1] 之间的随机数. *I* 为小于 2 的正整数, *ub*, *lb* 分别为决策变量的上界与下界, *N* 代表种群大小,  $\nu$  为决策空间中变量的维数. *Iguana*<sup>G</sup> 代表目标落地后的位置信息.

### 1.1.2 开发阶段

探索算子对种群的更新是为了保证算法的收敛性, 而开发阶段则是为增强算法的多样性. 开发算子通过模拟浣熊遭遇捕食者时的逃脱过程丰富了种群信息, 从而保证算法多样性. 开发算子定义如下:

$$lb_j^{local} = \frac{lb_j}{t}, ub_j^{local} = \frac{ub_j}{t}, \text{ where } t = 1, 2, \dots, T \quad (4)$$

$$X_i^{P2} : x_{i,j}^{P2} = x_{i,j} + (1 - 2r) \cdot (lb_j^{local} + r \cdot (ub_j^{local} - lb_j^{local})),$$

$i = 1, 2, \dots, N, j = 1, 2, \dots, \nu.$  (5)

其中: *t* 为缩放因子, *T* 为算法最大迭代数, *P2* 代表开发阶段.

分别对 *P1* 及 *P2* 阶段的更新进行检查, 若各阶段的目标函数值  $F_i^{P1, P2}$  小于当前最优值  $F_{best_i}$ , 则保留更新, 否则保留原值. 关于浣熊优化算法详细内容请参考<sup>[11]</sup>.

$$X_i = \begin{cases} X_i^{P1, P2}, & F_i^{P1, P2} < F_{best_i} \\ X_i, & \text{else} \end{cases} \quad (6)$$

## 2 多目标浣熊优化算法

COA 的探索与开发算子可以更加高效的进行寻优, 但是其存在只能优化单一目标的局限性. MOCO 打破了单一目标的限制, 在保留 COA 高效寻优的基础上, 对多个目标同时进行优化, 并且改进后的更新算子可以更好的保持算法收敛性与多样性之间的平衡, 避免算法陷入局部最优, 与已有的主流算法相比具备一定的竞争优势.

与单目标优化不同, 多目标优化问题中的最优解不再是单个解, 而是一组最优解集, 称为帕累托集 (PS, Pareto set), 对应的目标函数称为帕累托前沿 (PF, Pareto front). 因此在单目标浣熊优化算法的基础上, 首先引入快速非支配排序与拥挤度距离计算方法<sup>[15]</sup>, 建立精英浣熊保留策略. 其次, 原有的更新算子并不适用于多目标浣熊优化算法, 对其适应性进行了修改. 鉴于所提多目标浣熊优化算法的精英浣熊保留策略可以更好的解决更新无效问题, 因此去除了原有公式 (6) 所示的更新有效性检查.

### 2.1 快速非支配排序

多目标问题中的不同目标函数之间存在相互制约的关系, 一个目标函数的减小会导致另一目标函数的增大, 因此只能尽可能从中找到一个折中处理办法. 快速非支配排序方法<sup>[12]</sup>很好的解决了这一问题, 通过比较不同个体之间的支配关系, 对种群全部个体的前沿等级进行划分. 其中支配关系为种群中某一个体  $x_i$  的所有目标函数值均优于另一个个体  $x_j$  ( $i \neq j$ ), 则称  $x_i$  支配  $x_j$ , 记作  $x_i < x_j$ . 若种群中找不到任一其他个体可以支配  $x_i$ , 则称  $x_i$  为非支配个体. 快速非支配排序的具体步骤如下:

- 1) 计算并获取种群中所有个体的目标函数值;
- 2) 通过对种群中个体逐一与其他个体进行支配关系比较, 确定第一 PF 等级下的非支配个体, 记为  $Front_1$ ;
- 3) 对去除后的剩余种群进行第二次支配比较, 并将其中的非支配个体记为  $Front_2$ ;
- 4) 以此类推, 对剩余种群进行划分, 直到确定所有个体的前沿等级.

以两目标为例, 图 1 为划分后的各等级前沿示意图. 其中 *K* 为前沿等级个数,  $f_1, f_2$  为目标函数值.

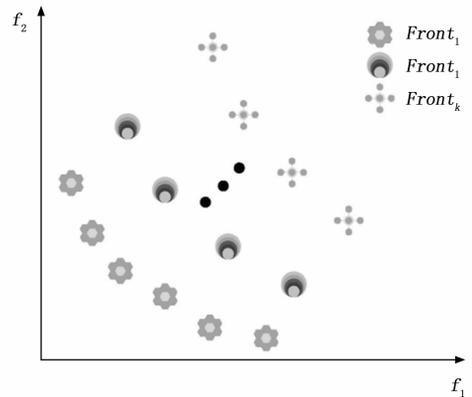


图 1 各前沿等级示意图

### 2.2 拥挤度距离计算

为了使个体分布更加均匀, 需要通过分布密度对种群进行衡量. 在快速非支配排序的基础上, 分别计算每个个体在种群中的拥挤度距离, 距离越大说明该个体所在区域的个体越少, 密度越低. 相反, 若距离较小则说明该区域的个体过于集中. 多目标拥挤度距离计算公式如下:

$$\begin{cases} D_{i,m} = \frac{F_m(i+1) - F_m(i-1)}{F_m^{max} - F_m^{min}}, m = 1, 2, \dots, M \\ D_i = \sum_{m=1}^M D_{i,m}, i = 1, 2, \dots, N \end{cases} \quad (7)$$

其中:  $D_i$  为第 *i* 个个体与该个体所在前沿等级下相邻两个个体在全部目标函数上的拥挤度距离之和, *M* 为目标函数个数.

### 2.3 精英浣熊保留策

如图 2 所示, 在多目标浣熊优化算法中需要同时对多个鬣蜥进行狩猎, 此时不同迭代数下的最优解不再是单个  $Fbest_t$ , 而是第一 PF 等级下的非支配个体集合  $Front_1$ 。在这种情况下, COA 第一阶段的更新规则不再适用, 因为鬣蜥的最佳位置不再具有唯一性, 而是从第一前沿等级个体中随机选取。通过新算子具体更新方式如下:

$$X_{i,j}^{P_i} : X_{i,j}^{P_i} = x_{i,j} + r \times (Front_{1,j} - I \times x_{i,j}),$$

$$\text{for } i = 1, 2, \dots, \left\lceil \frac{N}{2} \right\rceil, \text{ and } j = 1, 2, \dots, v. \quad (8)$$

其次, 将更新的后子代个体与更新前的父代个体进行合并, 并通过快速非支配排序确定各等级下的非支配前沿  $Front_l, l=1, 2, \dots, K$ 。最后计算各  $Front_l$  中不同个体的拥挤度距离, 从第一前沿等级开始依次选取拥挤度距离最小的个体, 直到满足停止条件。这一过程定义如下:

$$P^{G+1} = \begin{cases} P_0 \cup \sum_{l=1}^K Front_l, & \text{if } |P_0 + Front_l| \leq N \\ P_0 \cup Front_l^{|N-P_0|}, & \text{else} \end{cases} \quad (9)$$

其中:  $P^{G+1}$  为选出的新种群,  $P_0$  为被选择解的集合。当  $P_0$  中个体数为  $N$  时满足停止条件。

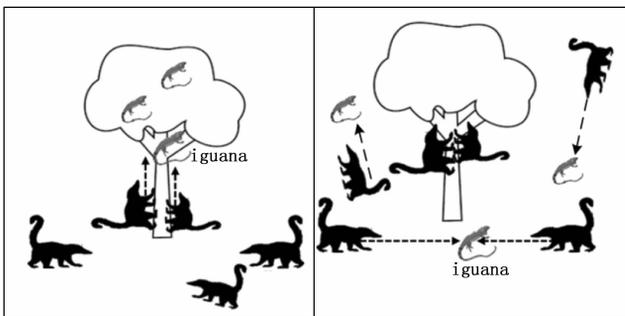


图 2 MOCOA 第一阶段模式图

### 2.4 MOCOA 优化步骤

所提 MOCOA 的具体优化步骤如下:

- 1) 设置种群大小、决策变量维度及其上下界, 迭代停止条件等参数, 并对种群进行随机初始化;
- 2) 设置目标函数, 分别并计算每个个体对应的多个目标函数适应度值;
- 3) 通过快速非支配排序将种群划分为不同的前沿等级, 并计算各前沿等级下的拥挤度距离;
- 4) 随机选取  $Front_1$  中的个体为 *Iguana*, 并根据公式 (2) ~ (8), 对种群进行更新;
- 5) 根据公式 (9), 即精英浣熊保留策略, 选出新的种群  $P^{G+1}$ ;
- 6) 判断算法是否达到迭代停止条件, 若未达到,

则转到第 4) 步, 若达到停止条件则输出  $Front_1$ 。

## 3 基于多目标浣熊优化算法的双向长短期记忆神经网络

### 3.1 LSTM 网络

LSTM 网络是 RNN 的一种改进形式, 用于处理序列数据的深度学习模型, 特别适用于具有长期依赖关系的序列数据, 如自然语言处理、语音识别和时间序列预测等任务。每个 LSTM 网络由一个或多个自连接的记忆单元和 3 个称为输入、输出和遗忘门的乘法单元组成, 它解决了 RNN 在处理长序列时产生的梯度消失和梯度爆炸问题。图 3 描述了单个 LSTM 细胞的内部结构, 遗忘门的功能是决定哪些信息应该被丢弃或保留。输入门用于更新细胞状态。输出门用于确定下一个隐藏状态的值, 隐藏状态包含先前输入的信息。在单个 LSTM 单元中, 计算公式如下:

$$f_t = \sigma(W_f \cdot [m_{t-1}, x_t] + b_f) \quad (10)$$

$$i_t = \sigma(W_i \cdot [m_{t-1}, x_t] + b_i) \quad (11)$$

$$o_t = \sigma(W_o \cdot [m_{t-1}, x_t] + b_o) \quad (12)$$

$$g_t = \tanh(W_g \cdot [m_{t-1}, x_t] + b_g) \quad (13)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot g_t \quad (14)$$

$$m_t = o_t \cdot \tanh(c_t) \quad (15)$$

$x_t$  为当前输入,  $m_{t-1}$  为隐藏层的前一状态,  $m_t$  为隐藏层当前状态,  $b_f, b_o, b_i, b_g$  为偏移量,  $W_f, W_i, W_o, W_g$  为网络参数,  $\sigma$  表示 sigmoid 函数,  $C_{t-1}$  和  $C_t$  分别是之前和当前的细胞记忆信息。

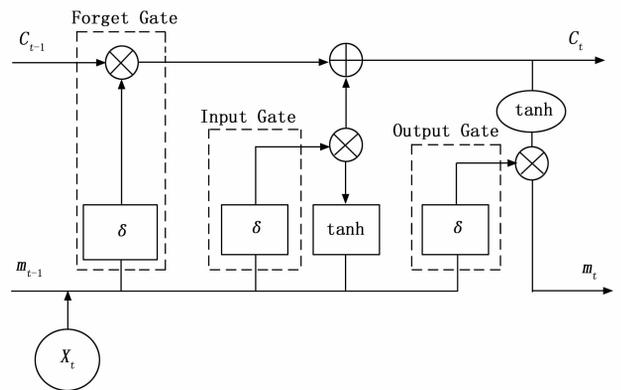


图 3 LSTM 原理示意图

### 3.2 BiLSTM 网络

单向 LSTM 在处理 and 预测时间序列中的长时间间隔和延迟方面具有显著优势, 但其只能利用过去的数据进行预测, 忽略了未来数据可能包含的重要信息。与此不同, BiLSTM 通过双向结构, 能够同时处理过去和未来的信息。具体来说, BiLSTM 在传统 LSTM 的基础上引入了一个反向 LSTM 层, 使得模型不仅能从序列的前向传播中获取信息, 还能从反向传播 (即未来时刻的输入) 中获得更多上下文信息, 从而更加全面地理解

当前时刻的状态<sup>[14]</sup>，BiLSTM的原理如图 4 所示。

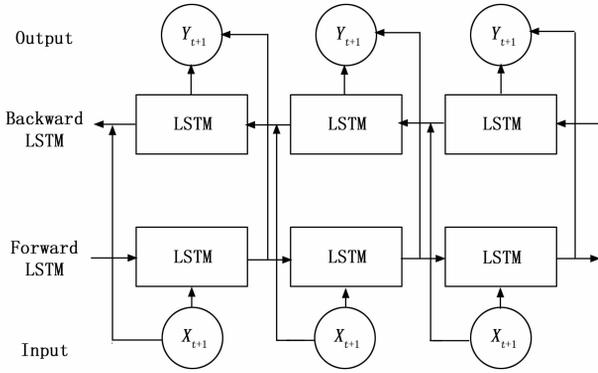


图 4 BiLSTM 原理示意图

将图 4 中的前向 LSTM 层计算向量记为  $\vec{h}$ ，后向 LSTM 层计算向量记为  $\overleftarrow{h}$ ，可得输出  $y$  的最终计算结果为：

$$h_t = \vec{a}h + \overleftarrow{\beta}h \quad (16)$$

$$y = \sigma(h_t) \quad (17)$$

其中： $\alpha$ 、 $\beta$  为常数，且二者之和为 1。

### 3.3 MOCOA-BiLSTM 预测模型

利用多目标浣熊优化算法对 BiLSTM 预测模型的多个超参数进行优化，包括学习率、迭代次数、隐藏层节点数、全连接层节点。在上述超参数作为 MOCOA 输入的同时，为了同时提升 BiLSTM 模型的预测准确性与稳定性，分别选取均方误差 (MSE, mean square error) 及误差方差为两目标函数，定义如下：

$$Fun. \begin{cases} f_1 = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2 \\ f_2 = std(|\hat{y}_i - y_i|) \end{cases} \quad (18)$$

其中： $N$  为样本数， $\hat{y}_i$  为预测值， $y_i$  为真实值， $std$  为方差函数。

MOCOA-BiLSTM 预测模型的流程图如图 5 所示。首先，确定 MOCOA 优化超参数的上下界区间，并设置 MOCOA 参数对种群进行初始化。其次，将初始化后种群的个体信息传递给 BiLSTM 模型对其进行训练。然后，根据模型训练后的指标计算 MOCOA 的种群目标函数值，并对种群进行更新。最后，检查算法是否满足终止条件。满足条件则输出最优超参数，建立 MOCOA-BiLSTM 最优参数模型，不满足则继续对种群进行迭代更新，直到满足停止条件。

## 4 结果与分析

### 4.1 MOCOA 性能测试

为了评估新算法的性能，我们首先将其与 NSGA-III<sup>[16]</sup>、MOSSA<sup>[17]</sup> 及 MOMVO<sup>[18]</sup> 等主流优化算法进行比较。测试问题选取了多目标优化基准测试问题 MOP<sup>[19]</sup> 以及 ZDT<sup>[20]</sup> 子集，包括 MOP1 至 MOP4 以及 ZDT1 至 ZDT4 这两类两目标问题，如表 1 所示。现有研究中种群规模设置为 100 已成为普遍共识<sup>[21-22]</sup>，考虑到过小的

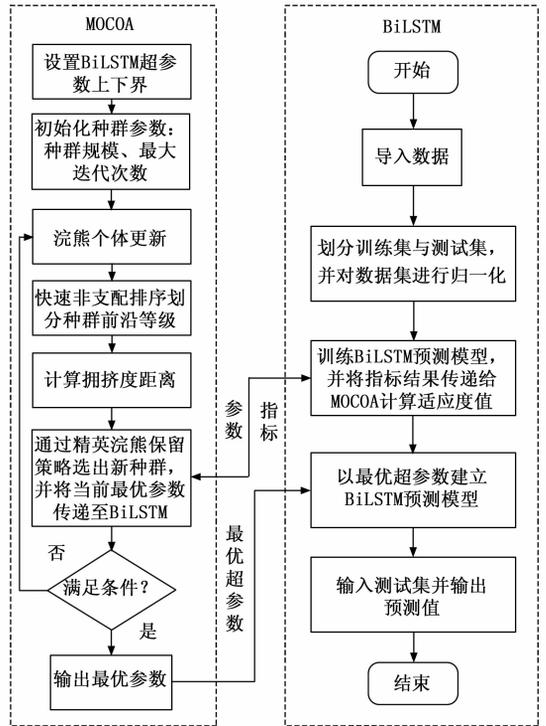


图 5 MOCOA-BiLSTM 预测模型流程图

种群规模无法满足对种群多样性的衡量，而过大的种群会导致计算复杂度过大的问题，因此本研究所有 4 种算法的种群规模均采用经典设置 100，既满足了种群多样性需求的同时，又保持了较低的计算复杂度。算法最大评估次数设置为 10000 次，当总评估次数达到最大次数是算法停止。其余参数按照原始文献中的设定进行配置。为了确保实验的公平性，所有算法均独立运行 30 次，并通过倒世代距离 (IGD, Inverse Generation Distance) 指标对各算法的表现进行评估。分别记录最佳值、平均值、中间值、最差值及方差，其中最佳值与平均值可以很好的衡量算法的探索能力，方差则反应了算法寻优的稳定性。IGD 指标计算结果为无量纲值。实验结果表明，MOCOA 在 8 个测试问题上均表现出最佳性能。表 2 和表 3 中列出了 IGD 值，其中以加粗的方式标出了最优 IGD 值。数据清晰地显示，除了 MOP1 问题上提升较小，其余 7 个问题中 MOCOA 的数值均比对比算法提升了一个数量级。且 MOCOA 的最佳值、中间值、平均值几乎一致，结合方差数值来看，所提算法同时具有良好探索能力与稳定性。在 MOP4 问题上，MOCOA 算法的优化效果最为突出，与对比算法相比，最佳值分别提升了 85.34%、72.58% 及 64.95%，图 6 也清晰地显示了这一优势。MOCOA 所生成的近似前沿与真实前沿更加吻合，同时解的分布也更加均匀，展现了优异的性能。图 6 中的  $f_1$ 、 $f_2$  分别为无量纲的测试函数值。

### 4.2 MOCOA-BiLSTM 预测模型仿真测试

变电工程造价影响因素分析有助于识别和理解造成

表 1 测试函数

<p>MOP1:</p> $Func \begin{cases} \min f_1(x) = (1 + G(x))x_1 \\ \min f_2(x) = (1 + G(x))(1 - \sqrt{x_1}) \end{cases}$ $G(x) = 2\sin(\pi x_1) \sum_{i=2}^n (-0.9t_i^2 +  t_i ^{0.5})$ $t_i = x_i - \sin(0.5\pi x_1)$	<p>MOP2:</p> $Func \begin{cases} \min f_1(x) = (1 + G(x))x_1 \\ \min f_2(x) = (1 + G(x))(1 - x_1^2) \end{cases}$ $G(x) = 10\sin(\pi x_1) \sum_{i=2}^n \left( \frac{ t_i }{1 + e^5  t_i } \right)$ $t_i = x_i - \sin(0.5\pi x_1)$
<p>MOP3:</p> $Func \begin{cases} \min f_1(x) = (1 + G(x))\cos(\frac{\pi x_1}{2}) \\ \min f_2(x) = (1 + G(x))\sin(\frac{\pi x_1}{2}) \end{cases}$ $G(x) = 10\sin(\pi x_1) \sum_{i=2}^n \left( \frac{ t_i }{1 + e^5  t_i } \right)$ $t_i = x_i - \sin(0.5\pi x_1)$	<p>MOP4:</p> $Func \begin{cases} \min f_1(x) = (1 + G(x))x_1 \\ \min f_2(x) = (1 + G(x))(1 - \sqrt{x_1}\cos^2(2\pi x_1)) \end{cases}$ $G(x) = 10\sin(\pi x_1) \sum_{i=2}^n \left( \frac{ t_i }{1 + e^5  t_i } \right)$ $t_i = x_i - \sin(0.5\pi x_1)$
<p>ZDT1:</p> $Func \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = G(x) \sqrt{1 - \frac{x_1}{G(x)}} \end{cases}$ $G(x) = 1 + \frac{9(\sum_{i=2}^n x_i)}{(n-1)}$	<p>ZDT2:</p> $Func \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = G(x) \left[ 1 - \left( \frac{x_1}{G(x)} \right)^2 \right] \end{cases}$ $G(x) = 1 + \frac{9(\sum_{i=2}^n x_i)}{(n-1)}$
<p>ZDT3:</p> $Func \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = G(x) \sqrt{1 - \frac{x_1}{G(x)}} - x_1 \sin(10\pi x_1) \end{cases}$ $G(x) = 1 + \frac{9(\sum_{i=2}^n x_i)}{(n-1)}$	<p>ZDT4:</p> $Func \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = G(x) \sqrt{1 - \frac{x_1}{G(x)}} \end{cases}$ $G(x) = 1 + 10(n-1) + \sum_{i=2}^n [x_i^2 - 10\cos(4\pi x_i)]$
<p>DTLZ2:</p> $Func \begin{cases} \min f_1(x) = (1 + G(x))\cos(\frac{\pi x_1}{2})\cos(\frac{\pi x_2}{2}) \\ \min f_2(x) = (1 + G(x))\cos(\frac{\pi x_1}{2})\sin(\frac{\pi x_2}{2}) \\ \min f_3(x) = (1 + G(x))\sin(\frac{\pi x_1}{2}) \end{cases}$ $G(x) = \sum_{i=3}^n (x_i - 0.5)^2$	<p>DTLZ4:</p> $Func \begin{cases} \min f_1(x) = (1 + G(x))\cos(\frac{\pi x_1^{100}}{2})\cos(\frac{\pi x_2^{100}}{2}) \\ \min f_2(x) = (1 + G(x))\cos(\frac{\pi x_1^{100}}{2})\sin(\frac{\pi x_2^{100}}{2}) \\ \min f_3(x) = (1 + G(x))\sin(\frac{\pi x_1^{100}}{2}) \end{cases}$ $G(x) = \sum_{i=3}^m (x_i - 0.5)^2$

表 2 所提算法与对比算法在 MOP1-MOP4 问题上的 IGD 值

算法	MOP1 算法					算法	MOP2 算法				
	最佳	平均	中间	方差	最差		最佳	平均	中间	方差	最差
NSGAIII	0.386 7	0.661 6	0.831 3	0.244 2	0.832 6	NSGAIII	0.012 2	0.021 0	0.021 8	0.008 5	0.034 2
MOSSA	0.765 2	0.788 7	0.785 1	0.013 2	0.797 6	MOSSA	0.016 5	0.022 4	0.024 0	0.006 0	0.030 3
MOMVO	0.780 9	0.796 0	0.787 0	0.013 8	0.812 1	MOMVO	0.020 3	0.022 7	0.023 9	0.001 4	0.023 8
MOCOA	0.356 9	0.360 1	0.360 7	0.004 2	0.363 0	MOCOA	0.004 9	0.005 5	0.005 5	0.000 5	0.005 7
算法	MOP3 算法					算法	MOP4 算法				
	最佳	平均	中间	方差	最差		最佳	平均	中间	方差	最差
NSGAIII	0.012 6	0.023 6	0.026 0	0.010 2	0.037 0	NSGAIII	0.023 2	0.048 9	0.057 9	0.022 7	0.077 3
MOSSA	0.018 3	0.023 5	0.021 9	0.005 3	0.031 7	MOSSA	0.012 4	0.017 7	0.018 9	0.004 7	0.024 1
MOMVO	0.017 8	0.024 6	0.024 3	0.004 6	0.029 7	MOMVO	0.009 7	0.012 1	0.012 4	0.001 4	0.013 5
MOCOA	0.005 1	0.005 6	0.005 7	0.000 3	0.006 0	MOCOA	0.003 4	0.003 5	0.003 5	0.000 1	0.003 7

表 3 所提算法与对比算法在 ZDT1-ZDT4 问题上的 IGD 值

算法	算法 ZDT1					算法 ZDT2	MOP2 算法				
	最佳	平均	中间	方差	最差		最佳	平均	中间	方差	最差
NSGAIH	0.017 3	0.033 4	0.031 6	0.014 9	0.058 5	NSGAIH	0.014 9	0.045 1	0.033 6	0.036 2	0.108 0
MOSSA	0.022 0	0.038 7	0.039 9	0.011 0	0.051 5	MOSSA	0.032 8	0.060 3	0.055 2	0.021 0	0.087 0
MOMVO	0.017 7	0.022 9	0.023 1	0.003 6	0.027 1	MOMVO	0.025 2	0.052 6	0.034 8	0.040 4	0.123 1
MOCOA	0.003 8	0.003 9	0.004 1	0.000 1	0.004 5	MOCOA	0.004 1	0.004 1	0.003 9	0.000 1	0.004 8
算法	算法 ZDT3					算法 ZDT2	MOP4 算法				
	最佳	平均	中间	方差	最差		最佳	平均	中间	方差	最差
NSGAIH	0.083 3	0.165 8	0.169 1	0.052 3	0.226 5	NSGAIH	0.083 5	0.068 5	0.169 3	0.033 1	0.227 1
MOSSA	0.046 3	0.074 2	0.073 2	0.019 3	0.099 6	MOSSA	0.143 2	0.395 6	0.482 4	0.201 7	0.578 2
MOMVO	0.021 0	0.026 3	0.024 3	0.005 5	0.036 2	MOMVO	2.016 3	3.384 3	3.627 7	0.819 6	4.079 8
MOCOA	0.004 7	0.005 1	0.004 4	0.000 2	0.005 3	MOCOA	0.004 2	0.003 7	0.004 3	0.000 1	0.004 6

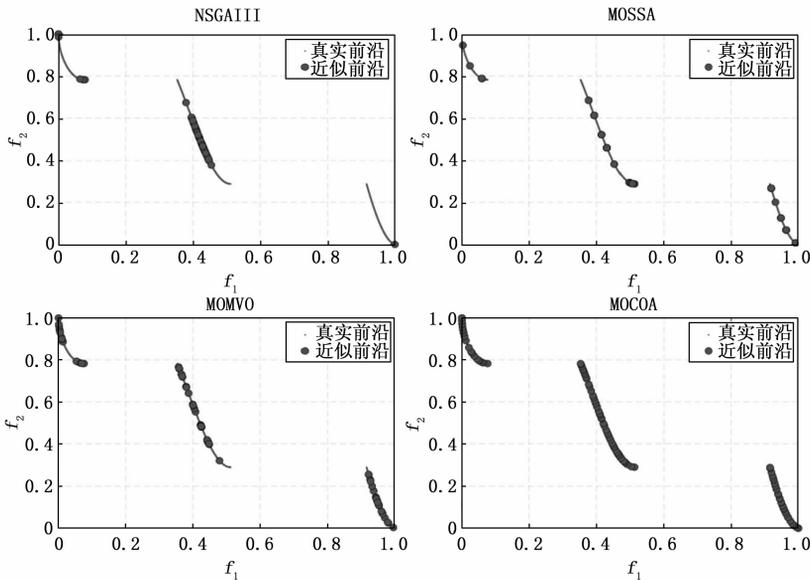


图 6 4 种算法在 MOP4 问题上的测试结果

造价波动的根本原因，为工程决策提供科学依据。首要因素之一是技术层面，不同的技术路径和设计方案将直接影响所需的资源投入，从而直接关系到总体造价。本文以宁夏某电网公司已完工的 20 项变电工程历史数据为样本，并收集其中 9 项指标作为 MOCOA-BiLSTM 预测模型的特征输入，通过历史数据驱动的方法，对变电工程造价进行预测。输入特征因素包括：主控楼面积、本期台数、单台容量、高压侧出线回数、低压侧出线回数、高压侧断路器单价、低压侧断路器台数、低压电容器数量、场地平整费用。部分变电工程数据样本如表 4 所示。对数据集进行划分，其中 80% 为训练集，20% 为测试集，并对数据进行最大最小归一化处理。

仿真测试中 MOCOA 的种群规模  $N$  设置为 100，最大迭代次数设置为 100。学习率的上下界为别设置为 0.01 与 0.001，训练迭代次数上下界分别为 20、100，BiLSTM 隐含层的节点数上界设置为 30，下界设置为

5，全连接层的节点数的区间上下界分别为 1 和 100。

为了更好地衡量所提算法对 BiLSTM 预测准确性的提升，分别通过对比算法 NSGAIH、MOSSA、MOMVO 对 BiLSTM 进行超参数优化，并通过均方根误差 (RMSE)、均方误差 (MSE)、平均绝对百分比误差 (MAPE) 对预测结果进行评价。3 种评价指标定义如下：

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (19)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (20)$$

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \frac{|\hat{y}_i - y_i|}{y_i} \quad (21)$$

其中： $n$  为样本数量， $y_i$  为真实值， $\hat{y}_i$  为预测值。

如图 7 所示，MOCOA 优化后的前沿分布近似为一条曲线，选取其中拐点为最优预测模型，并计算评价指

表 4 变电工程数据部分样本

样本编号	主控楼面积	本期台数	单台容量	高压侧出现回数	低压侧出线回数	高压侧断路器单价	低压侧断路器台数	低压电容器容量	场地平整费用
1	108	2	50	2	16	10	28	4	15
2	305	2	40	2	6	10	18	4	29
3	414	1	32	2	4	10	7	0	3
4	292	1	50	2	7	58	11	0	8
5	617	2	50	3	16	64	16	4	4
6	2882	2	50	2	12	57	21	4	8
7	300	1	40	2	7	90	11	2	5
8	2456	2	63	2	24	66	34	4	27
9	2535	2	63	2	24	62	34	4	42
10	257	2	40	4	8	70	20	4	40
.....	.....	.....	.....	.....	.....	.....	.....	.....	.....
66	515	1	50	3	7	0	10	1	5
67	1246	1	40	2	11	54	11	4	7
68	253	1	40	1	5	60	11	2	11
69	1470	2	50	2	20	60	20	4	87
70	2546	2	63	2	24	44	34	4	323

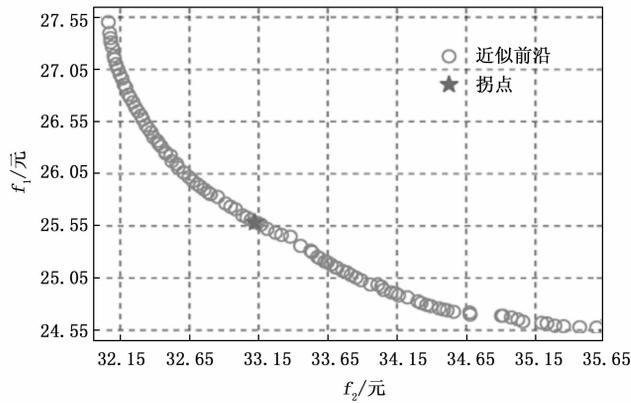


图 7 近似帕累托前沿

标与其他模型进行对比。对比结果如表 5 所示。所提 MOCO A-BiLSTM 预测模型在  $MSE$ 、 $RMSE$ 、 $MAPE$  3 个指标上均优于其他算法的优化模型, 进一步证明了所提算法具有良好的性能, 并且证明了其对于 BiLSTM 预测精度提升的有效性, 其中  $MSE$  与  $RMSE$  单位为元,  $MAPE$  为无量纲指标。与 MOSSA-BiLSTM、NSGAI II-BiLSTM、MOMVO-BiLSTM 相比, MOCO A-BiLSTM 的  $MSE$  分别降低了 69.47%、61.39%、58.92%,  $RMSE$  分别降低 68.28%、58.43%、55.92%,  $MAPE$  分别降低了 69.59%、58.43%、56.67%。

表 5 模型误差评价对比

模型/指标	$MSE$ /元	$RMSE$ /元	$MAPE$
MOCO A-BiLSTM	33.11	131.74	3.92%
MOSSA-BiLSTM	108.45	415.34	12.89%
NSGAI II-BiLSTM	85.76	316.92	9.43%
MOMVO-BiLSTM	80.59	298.89	9.05%

图 8 所示为几种模型的预测结果对比图。从图中可以看出, MOCO A-BiLSTM 与真实值始终保持相同趋势, 且差值较小, 更加清晰的显示了 MOCO A-BiLSTM 在预测精度及预测稳定性上的良好性能。

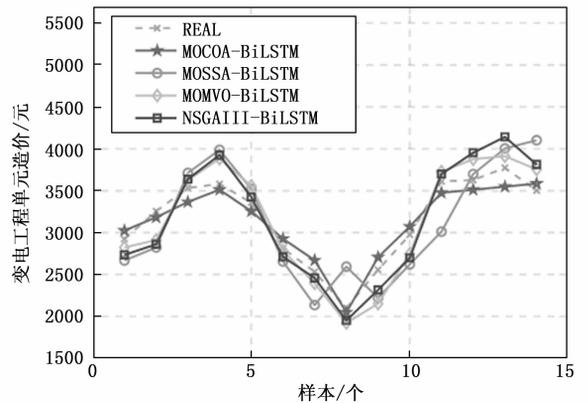


图 8 MOCO A-BiLSTM 与对比算法优化后的模型预测结果对比图

### 5 结束语

为了提升 BiLSTM 预测模型的预测准确性及预测稳定性, 首先通过对浣熊优化算法的更新算子进行改进, 使更新规则适用于多目标问题。其次引入快速非支配排序与拥挤度距离计算方法, 提出了一种新的多目标浣熊优化算法, 实现浣熊优化算法由单目标到多目标的转变。最后通过所提算法对 BiLSTM 的超参数进行优化, 提出了一种新的 MOCO A-BiLSTM 预测模型, 通过以均方误差及误差方差为目标函数, 实现了 BiLSTM 预测性能的提升。测试问题仿真结果表明所提算法在多样性及收敛性方面具有良好的性能。变电工程造价预测仿真测试结果表明, 使用 MOCO A 对 BiLSTM 的优化

效果优于 MOSSA、NSGAIII 及 MOMVO 算法，进一步证明了所提算法的有效性。同时 MOCOBA-BiLSTM 在变电工程造价问题上的预测准确性也表明了其在实际工程问题中的具有良好的应用价值。

参考文献:

[1] 郭逸婕, 张君毅, 王 鹏. 基于注意力机制的 TCN-BiLSTM 船舶轨迹预测 [J]. 计算机测量与控制, 2024, 32 (1): 30-36.

[2] 张加劲. 基于注意力机制和 CNN-BiLSTM 模型的航空发动机剩余寿命预测 [J]. 电子测量与仪器学报, 2022, 36 (8): 231-237.

[3] 刘 英, 裴莉莉, 郝雪丽. 基于 WOA-BiLSTM 模型空气质量指数预测 [J]. 计算机系统用, 2022, 31 (10): 389-396.

[4] 杜睿山, 李宏杰, 孟令东, 等. 基于 PSO-BiLSTM 的储层岩石脆性指数预测 [J]. 海南大学学报 (自然科学版), 2023, 41 (3): 260-267.

[5] 王帅党, 程 鹏. 基于 GWO 优化 BiLSTM 模型的地区短期风功率的预测 [J]. 黑龙江科学, 2023, 14 (16): 28-31.

[6] 胡春安, 江 维. 基于 VMD-BO-BiLSTM 的猪肉价格预测模型 [J]. 应用科学学报, 2023, 41 (4): 692-704.

[7] 刘相杰, 刘小生, 张龙威. 基于 VMD-HPO-BiLSTM 的大坝变形预测 [J]. 大地测量与地球动力学, 2023, 43 (8): 851-855.

[8] HASHIM F A, HUSSIEN A G. Snake Optimizer: A novel meta-heuristic optimization algorithm [J]. Knowledge-Based Systems, 2022, 242: 1-34.

[9] AGUSHAKA J O, EZUGWU A E, ABUALIGAH L. Gazelle optimization algorithm; a novel nature-inspired meta-heuristic optimizer [J]. Neural Computing and Applications, 2023, 35 (5): 4099-4131.

[10] AHMADIANFAR I, HEIDARI A A, NOSHADIAN S, et al. INFO: An efficient optimization algorithm based on weighted mean of vectors [J]. Expert Systems with Applications, 2022, 195: 1-26.

[11] DEHGHANI M, MONTAZERI Z, TROJOVSKÁ E, et al. Coati optimization algorithm; a new bio-inspired meta-heuristic algorithm for solving optimization problems [J].

[16] KOONGA C S, SHIHB C, HSIUNG C P A, et al. Automatic testing environment for multi-Core embedded software ATEMES. The Journal of Systems and Software, 2012, 85 (1): 43-60.

[17] DUENAS J C, CUADRADO F, GARCIA B, et al. System Virtualization Tools for Software [Z].

[18] DEVELOPMENT [J]. Internet Computing, IEEE, 2009, 13 (5): 52-59.

Knowledge-Based Systems, 2023, 259: 1-43.

[12] DEB K, PRATAP A, AGARWAL S, et al. A fast and elitist multiobjective genetic algorithm; NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2002, 6 (2): 182-197.

[13] 张雅雯, 张玉臣, 吴 越, 等. 面向网络流量数据增强的生成对抗网络改进研究 [J]. 计算机工程与应用, 2024, 60 (18): 275-284.

[14] 吴永洪, 张智斌. 基于麻雀搜索优化的 Attention-BiLSTM 短期电力负荷预测 [J]. 自动化仪表, 2023, 44 (8): 91-95.

[15] 李宏玉, 彭 康, 宋来鑫, 等. 基于 EMD-BiLSTM-ANFIS 的负荷区间预测 [J]. 吉林大学学报 (信息科学版): 2024, 42 (11): 176-185.

[16] DEB K, JAIN H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints [J]. IEEE Transactions on Evolutionary Computation, 2013, 18 (4): 577-601.

[17] MIRJALILI S, GANDOMI A H, MIRJALILI S Z, et al. Salp Swarm Algorithm; A bio-inspired optimizer for engineering design problems [J]. Advances in Engineering Software, 2017, 114: 163-191.

[18] MIRJALILI S, JANGIR P, MIRJALILI S Z, et al. Optimization of problems with multiple objectives using the multi-verse optimization algorithm [J]. Knowledge-Based Systems, 2017, 134: 50-71.

[19] ZITZLER E, DEB K, THIELE L. Comparison of multiobjective evolutionary algorithms; Empirical results [J]. Evolutionary Computation, 2000, 8 (2): 173-195.

[20] LIU H L, GU F, ZHANG Q. Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems [J]. IEEE Transactions on Evolutionary Computation, 2013, 18 (3): 450-455.

[21] DHIMAN G, SINGH K K, SONI M, et al. MOSOA: A new multi-objective seagull optimization algorithm [J]. Expert Systems with Applications, 2021, 167: 1-22.

[22] KAVEH A, LAKNEJADI K. A new multi-swarm multi-objective optimization method for structural design [J]. Advances in Engineering Software, 2013, 58: 54-69.

[19] ZHANG Y F, ZHANG G, LIU Y, et al. Research on services encapsulation and virtualization access model of machine for cloud manufacturing [J]. Journal of Intelligent Manufacturing, 2017, 28 (5): 1109-1123.

[20] MWAFFAQ ABU-ALHAIJA, NIDAL M. Turab, AbdelRahman Hamza. Extensive Study of Cloud Computing Technologies, Threats and Solutions Prospective [J]. 计算机系统科学与工程, 2022, 41 (1): 225-240.