

# 基于 VMD-MSFOA-LSTM 短期电力负荷预测

杨松叶, 刘 微

(沈阳理工大学 信息科学与工程学院, 沈阳 110159)

**摘要:** 为提高电力系统中电力负荷预测的精度, 本文提出一种基于多策略改进的海星优化算法(multi strategy improved starfish optimization algorithm, MSFOA), 优化变分模态分解(VMD)与长短期记忆(LSTM)网络相结合的电力负荷预测模型(VMD-MSFOA-LSTM)。首先, MSFOA 算法结合了佳点集初始化策略、动态对立学习机制、线性递减系数和精英增强解质量策略; 其次, 基于 MSFOA 算法优化 LSTM 网络的超参数, 搭建 VMD-MSFOA-LSTM 负荷预测模型; 最后, 选取澳大利亚电力负荷数据集对模型进行评估。实验结果表明, VMD-MSFOA-LSTM 模型较原始 LSTM 模型均方根误差、平均绝对误差和平均绝对百分比误差分别降低了 86.53%、86.71% 和 87.59%, 决定系数提高了 16.94%, 验证了本文所提模型在电力负荷预测中的优越性能。

**关键词:** 电力负荷预测; 海星优化算法; 变分模态分解; 长短期记忆网络

中图分类号: TM715

文献标志码: A DOI:10.3969/j.issn.1003-1251.2026.04.006

## Short-term Power Load Forecasting Based on VMD-MSFOA-LSTM

YANG Songye, LIU Wei

(Shenyang Ligong University, Shenyang 110159, China)

**Abstract:** To enhance the accuracy of power load forecasting in power systems, a power load forecasting model(VMD-MSFOA-LSTM) was proposed, that combined variational mode decomposition (VMD) with a long short-term memory (LSTM) network, optimized by a multi-strategy improved starfish optimization algorithm(MSFOA). Firstly, MSFOA integrated the Halton sequence initialization strategy, dynamic opposition-based learning mechanism, linearly decreasing coefficient, and elite enhancement solution quality strategy. Secondly, the hyperparameters of the LSTM network were optimized based on MSFOA to construct the VMD-MSFOA-LSTM load forecasting model. Finally, the Australian power load dataset was selected to evaluate the model. The experimental results show that compared with the original LSTM model, the VMD-MSFOA-LSTM model reduces the root mean square error, mean absolute error, and mean absolute percentage error by 86.53%, 86.71%, and 87.59%, respectively, and increases the coefficient of determination by 16.94%, verifying the superior performance of the proposed model in power load forecasting.

**Key words:** power load forecasting; starfish optimization algorithm; variational mode decomposition; long short-term memory network

风力发电的核心力量是自然风,但其固有的强随机性、间歇性和高波动性特征,易给电力系统的稳定运行造成不利影响。因此,开展高精度电力负荷预测不仅能够合理安排机组调度,保证电网安全运行,还能提高发电厂电能利用率。

在短期负荷预测中,传统统计方法适应性较弱,对于复杂非线性负荷特性的拟合能力有限,难以充分捕捉负荷变化的时序依赖与周期规律。目前,部分群智能优化算法具备较强的学习能力并在电力负荷预测领域具有广泛应用,比如经典的鲸鱼优化算法(WOA)<sup>[1]</sup>、哈里斯鹰优化算法(HHO)<sup>[2]</sup>等。WOA对参数的敏感性高,严重依赖探索因子和概率因子,而HHO缺失快速调整搜索方向的能力,灵活性不足。群智能优化算法虽各有优点,却普遍存在收敛速度慢、参数依赖性强并容易陷入局部最优等问题。

长短期记忆(LSTM)是短期电力负荷预测中常用的模型,但单一LSTM容易受数据噪声干扰,超参数需要凭借经验进行设置,从而导致预测精度较低。因此,可通过群智能优化算法寻找LSTM的最优超参数,从而提高模型的性能,达到更高的预测精度。文献[1]提出了EEMD-WOA-LSTM模型,用于短期电力负荷预测,该组合预测模型的预测误差更小,预测精度更高;文献[2]提出了Cd-HHO-LSTM模型用于短期电力负荷预测,该模型与LSTM相比能始终实现更低的均方误差;文献[3]采用改进的蜣螂优化算法(HDBO)优化LSTM的超参数,构建了HDBO-LSTM短期电力负荷预测模型,该模型的指标MAPE、RMSE、MAE和COV较DBO-LSTM分别降低了10.48%、15.64%、10.50%和15.68%;文献[4]采用VMD技术和LSTM网络构成VMD-LSTM预测模型,该模型与LSTM模型相比MAE、MAPE和RMSE分别降低了39.7%、37.5%和30.9%,决定系数( $R^2$ )提高了20.6%。

因为负荷预测模型的精度与输入数据的质量有很大关系,对数据进行进一步处理可以提升预测模型的学习效果。VMD能够将复杂序列分解为多个振幅和频率可调的信号<sup>[5]</sup>,不仅能够实现信号的精准分解,运算效率也较高<sup>[6]</sup>。为进一步提升模型预测精度,本文采用更加优秀的海星优化算法(SFOA)并对其进行改进,通过引入佳点集初始化策略与动态对立学习机制,以增强种群多样性和全局搜索能力;同时引入线性递减系数与精英增强解质量策略,实现算法探索与开发能力

的有效平衡。基于上述改进,本文提出一种多策略改进的海星优化算法(MSFOA),并采用文献[7]中的前12个测试函数验证改进算法的有效性。本文先采用VMD将原始的复杂时间序列分解成若干个平稳子序列,降低非平稳性对预测的干扰;其后通过MSFOA算法搜索LSTM网络的最优超参数,从而解决手动调参的盲目性;最后对每个子序列独立的预测结果进行叠加重构,输出最终结果。

## 1 相关理论和方法

### 1.1 变分模态分解

变分模态分解(VMD)是将原始信号分解为若干个具有明确物理意义的平稳模态分量(IMF),每个分量都具有窄带性的特征并且不同分量的中心频率是互不重叠的。相比经验模态分解等传统方法,VMD具有更强的稳定性<sup>[8]</sup>。

在VMD分解过程中, $K$ 值与惩罚因子 $\alpha$ 的选取对分解效果影响显著。 $K$ 值过小会导致分解不足,产生模态混叠; $K$ 值过大则会导致过分解,产生无意义的虚假分量。惩罚因子 $\alpha$ 则影响各IMF分量的带宽, $\alpha$ 值越小,带宽越大。本文通过中心频率观察法与重构误差分析相结合的方法确定VMD参数,最终设定 $K=6$ 、 $\alpha=2\ 000$ ,确保各IMF分量中心频率互不重叠且具有明确物理意义,同时重构误差最小。

### 1.2 LSTM网络结构

LSTM通过门控机制解决传统循环神经网络在处理长序列数据时的梯度消失和梯度爆炸问题,能够有效捕捉时间序列中的长期依赖关系<sup>[9]</sup>。LSTM通过输入门、遗忘门和输出门实现信息的筛选、存储和输出。

遗忘门的作用是筛选历史信息,决定是否保留记忆细胞中存储的历史状态。输入门用于筛选当前信息,决定是否将当前时刻的新信息放入记忆细胞。输出门起到筛选输出信息的作用,能够决定记忆细胞中的状态是如何映射为当前时刻的隐状态的。

### 1.3 海星优化算法

海星优化(SFOA)算法是一种用于全局优化的生物元启发算法,模拟海星的探索、捕食和再生等行为<sup>[10]</sup>。

#### 1.3.1 随机初始化种群

在初始化阶段,种群构建为 $N \times d$ 的位置矩阵。其中 $N$ 为种群规模, $d$ 为目标函数维度。每

一个个体位置更新表达式为

$$\mathbf{X}_i = \mathbf{L} + r(\mathbf{U} - \mathbf{L}), i = 1, 2, \dots, N \quad (1)$$

式中:  $\mathbf{X}_i$  为种群中第  $i$  只海星的位置;  $\mathbf{U}$  和  $\mathbf{L}$  分别表示搜索空间的上界与下界;  $r$  为  $[0, 1]$  区间的随机值。

完成种群初始化之后, 每次迭代过程中进入探索阶段还是开发阶段由  $[0, 1]$  区间的随机值  $rand$  和固定探索概率系数共同决定。

### 1.3.2 探索阶段

SFOA 算法在探索阶段采用 5 维和 1 维相结合的混合搜索模式。

若目标函数维度大于 5, 搜索范围大, 需要海星五条手臂进行搜索, 数学表达式为

$$\begin{cases} \mathbf{Y}_{i,p}^T = \mathbf{X}_{i,p}^T + a_1 (\mathbf{X}_{best,p}^T - \mathbf{X}_{i,p}^T) \cos\theta, r \leq 0.5 \\ \mathbf{Y}_{i,p}^T = \mathbf{X}_{i,p}^T - a_1 (\mathbf{X}_{best,p}^T - \mathbf{X}_{i,p}^T) \sin\theta, r > 0.5 \end{cases} \quad (2)$$

式中:  $\mathbf{Y}_{i,p}^T$  和  $\mathbf{X}_{i,p}^T$  分别为第  $p$  维海星的更新位置与当前位置;  $\mathbf{X}_{best,p}^T$  为当前最优位置;  $a_1$  为移动步长;  $\theta$  为  $[0, \pi/2]$  区间的搜索摆动角,  $\cos\theta$  和  $\sin\theta$  分别表示海星手臂以相同概率向左或向右进行探索并靠近当前最优解。

若目标函数维度不大于 5, 仅需海星一个手臂进行移动搜索, 数学表达式为

$$\mathbf{Y}_{i,p}^T = E_T \mathbf{X}_{i,p}^T + A_1 (\mathbf{X}_{k_1,p}^T - \mathbf{X}_{i,p}^T) + A_2 (\mathbf{X}_{k_2,p}^T - \mathbf{X}_{i,p}^T) \quad (3)$$

式中:  $\mathbf{X}_{k_1,p}^T$  和  $\mathbf{X}_{k_2,p}^T$  分别为第  $p$  维两个随机选择海星  $k_1, k_2$  的位置;  $A_1$  和  $A_2$  为  $(-1, 1)$  区间内的随机数;  $E_T$  为海星的能量。  $E_T$  数学表达式为

$$E_T = \frac{T_{\max} - T}{T_{\max}} \cos\theta \quad (4)$$

式中:  $T_{\max}$  为最大迭代次数;  $T$  为当前迭代次数。

### 1.3.3 开发阶段

SFOA 算法在开发阶段采用并行双向搜索策略, 结合种群全局最优位置与随机海星位置更新个体位置; 同时因海星捕食时移动缓慢易受威胁, 可能会断臂求生, 故仅对种群最后一个个体执行再生策略, 其位置更新数学表达式为

$$\begin{cases} \mathbf{Y}_i^T = \mathbf{X}_i^T + r_1 \mathbf{d}_{m1} + r_2 \mathbf{d}_{m2}, & i \neq N \\ \mathbf{Y}_i^T = \exp(-T \times N / T_{\max}) \mathbf{X}_i^T, & i = N \end{cases} \quad (5)$$

式中:  $\mathbf{Y}_i^T$  和  $\mathbf{X}_i^T$  为第  $i$  只海星的更新位置与当前位置;  $r_1$  和  $r_2$  为  $(0, 1)$  内的随机数;  $\mathbf{d}_{m1}$  和  $\mathbf{d}_{m2}$  分别为全局最优位置与随机选择的两个海星之间的距离。

SFOA 算法虽能在迭代中平衡全局搜索与局部收敛以求解最优值, 但仍存在初始化种群分布不均、收敛速度慢、易陷入局部最优的缺陷<sup>[11]</sup>。

## 2 SFOA 算法的改进及性能测试

### 2.1 改进策略设计

#### 2.1.1 佳点集初始化

佳点集是一类在单位超立方体  $[0, 1]^d$  上构造的均匀点集, 具有较好的均匀覆盖性。为解决 SFOA 算法初始化种群分布不均、多样性不足的问题, 引入佳点集初始化, 能够确保生成的初始解更均匀地覆盖整个搜索空间, 有效增强种群多样性, 从而降低算法陷入局部最优解的可能性<sup>[12]</sup>。在空间中构造包含  $N$  个点的佳点集, 数学表达式为

$$\mathbf{P}_N = \{\mathbf{P}(k) = (\{kg_1\}, \dots, \{kg_d\}) \mid k = 1, 2, \dots, N\} \quad (6)$$

式中:  $\mathbf{P}_N$  为佳点集;  $\mathbf{P}(k)$  为第  $k$  个佳点;  $g_j$  为第  $j$  维生成系数。  $g_j$  数学表达式为

$$g_j = 2 \cos(2j\pi/p) \quad (7)$$

式中:  $1 \leq j \leq d$ ;  $p$  为满足  $(p-3)/2 \geq d$  条件的最小素数。

#### 2.1.2 动态对立学习

SFOA 算法在探索阶段搜索模式是依赖海星手臂扭转运动完成的, 虽实现了一定的空间遍历, 但是当优化问题的维度比较高或者搜索空间不对称时, 仅靠最优解引导的局部搜索容易陷入局部最优。为增强算法的全局探索能力与搜索灵活性, 引入动态对立学习, 该机制通过生成对立解拓展搜索空间, 避免一些潜在优质区域被忽略<sup>[13]</sup>。动态对立学习能加强随机缩放与非对称扰动, 能够贴合当前迭代搜索状态。动态对立解数学表达式为

$$\mathbf{X}_{D0} = \mathbf{X} + E_T \times rand_1 \times (rand_2 \times \mathbf{X}_0 - \mathbf{X}) \quad (8)$$

式中:  $\mathbf{X}_{D0}$  为生成的动态对立解;  $\mathbf{X}$  为当前解;  $rand_1$  和  $rand_2$  均为  $[0, 1]$  范围内的独立随机数;  $\mathbf{X}_0$  为当前解生成的对立解。

#### 2.1.3 线性递减系数

SFOA 算法采用的是固定探索概率系数控制探索与开发阶段的切换, 而固定探索概率系数有明显缺陷。在迭代前期, 算法需要通过大范围探索充分覆盖搜索空间, 但固定概率会限制探索强度, 易陷入局部最优的陷阱; 而到了迭代后期, 算法应聚焦局部开发, 固定概率仍保留大量不必要的探索行为, 既造成计算资源的冗余消耗, 也会拖慢算法的收敛速度。两种模式的搜索逻辑缺乏针对性适配, 为此加入线性递减系数 ( $GP$ ), 通过动态调整  $GP$  以匹配迭代进程的需求, 实现前期高效探索、后期精准开发。其数学表达式为

$$GP = 0.5 - (0.5 - 0.1)(T/T_{max}) \quad (9)$$

开始时设  $GP = 0.5$ , 为探索阶段提供比较高的触发概率, 广域遍历搜索空间; 随着迭代次数的增加线性递减至 0.1, 逐步降低探索频率, 提升开发占比, 从而更好平衡探索与开发。

### 2.1.4 精英增强解质量

在 SFOA 算法迭代后期虽然能通过再生机制更新部分个体, 但一方面对于适应度更优的个体的引导作用只能通过当前的最优解和随机个体的距离向量来体现, 信息传递效率较低, 无法快速向更优区域收敛; 另一方面, 解的更新过程缺少精细化扰动, 导致收敛精度受限。通过加入精英增强解质量策略, 构建动态精英池汇聚优质个体信息, 结合精英均值和自适应扰动, 充分挖掘精英个体中蕴含的潜在最优解特征<sup>[14]</sup>。

构建动态精英池需要先对随机筛选出的三个精英个体计算均值  $X_{avg}$ , 与当前全局最优解  $X_{best}$  线性混合得到基准候选解  $X_{new1}$ , 采用双阶段自适应扰动实现精英增强解, 其数学表达式为

$$\begin{cases} X_{new} = X_{new1} + r_1 w_1 |(X_{new1} - X_{avg}) + randn|, & w_1 < 1 \\ X_{new} = X_{new1} + r_1 w_1 |(u \cdot X_{new1} - X_{avg}) + randn|, & w_1 \geq 1 \end{cases} \quad (10)$$

式中:  $X_{new}$  为经自适应扰动后生成的精英增强解;  $r_1$  随机取值为  $-1, 0, 1$ ;  $u$  为  $[0, 2]$  区间随机缩放向量;  $randn$  为标准正态分布随机向量;  $w_1$  为扰动权重。  $w_1$  数学表达式为

$$w_1 = 2r \cdot \exp(-cT/T_{max}) \quad (11)$$

式中  $c$  为  $[0, 5]$  区间随机系数。

将以上策略加入 SFOA 得到 MSFOA 算法, 其具体流程如图 1 所示。

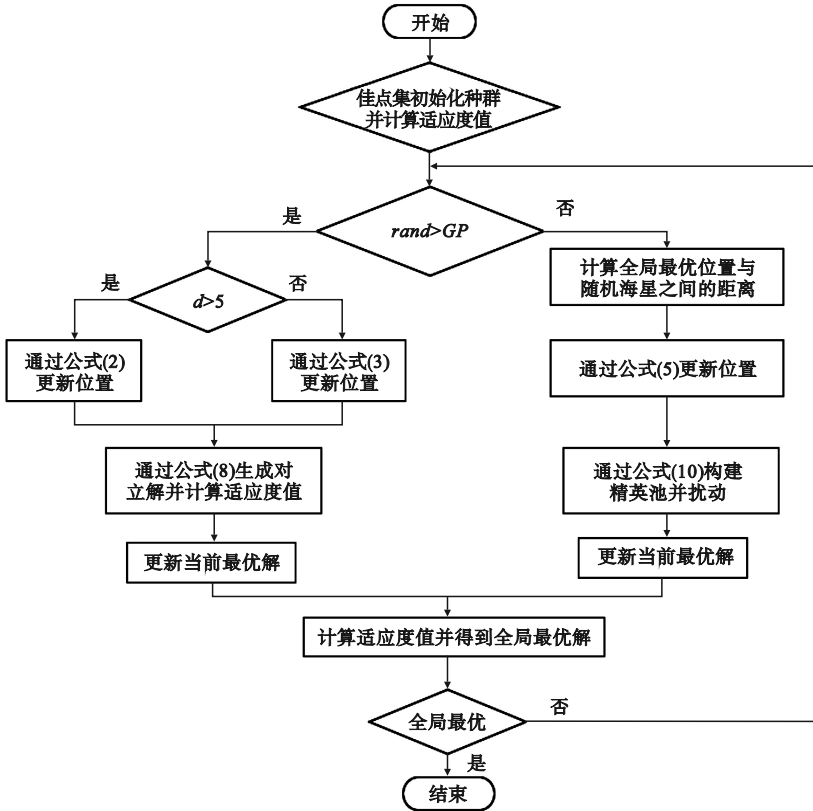


图 1 MSFOA 算法流程图

Fig. 1 Flowchart of MSFOA algorithm

## 2.2 时间复杂度分析

MSFOA 算法时间复杂度主要包含种群初始化、适应度评估和迭代更新三个关键部分。影响时间复杂度的主要参数是最大迭代次数  $T_{max}$ 、问题维度  $d$ 、种群规模  $N$  和评估一次适应度函数的成本  $C_f$ 。佳点集种群初始化和适应度评估时间复杂度分别为  $O(N \times d)$  和  $O(T \times N \times C_f)$ 。在一次迭代中, 算法执行探索或开发、对部分个体的动态

对立学习和精英增强解质量。个体位置更新、动态对立解生成、精英池构建与扰动和边界处理复杂度均为  $O(N \times d)$ 。因此, 单次迭代复杂度为  $O(N \times d)$ 。故 MSFOA 算法时间复杂度为  $O(T \times N \times (d + C_f))$ 。

HHO、WOA 等算法的全部时间复杂度也为  $O(T \times N \times (d + C_f))$ 。说明 MSFOA 算法并没有因为策略的加入增加时间复杂度。

### 2.3 MSFOA 的性能测试

实验将 MSFOA 与 WOA<sup>[15]</sup>、AOA<sup>[16]</sup>、HHO<sup>[2]</sup>、HO<sup>[17]</sup> 和包含 SFOA 在内的五种算法在 CEC2017 数据集上进行性能比较。由于 CEC2017 测试集比较庞大,选取 12 个测试函数进行验证。所有算法的参数设置保持与参考文献一致,种群数量为 30,最大迭代次数为 500,且每个算法在每个测试函数上独立运行 30 次,其统计结果如表 1 所示。

从函数均值 (Mean) 与标准差 (Std) 的统计结

果来看,MSFOA 算法在全局搜索和稳定性上具有显著的优势:除测试函数 F10 的性能略逊于 HHO 外,该算法在其余所有测试函数上均优于对比算法。另外 MSFOA 算法在多数函数上标准差更小,体现出其在迭代过程中解的波动更小,鲁棒性更强。图 2 为六种算法在迭代过程中的收敛曲线。结合收敛规律分析可知,MSFOA 算法收敛速度快于其他对比算法,进一步验证所提改进算法在全局搜索、收敛效率及稳定性上的优越性。

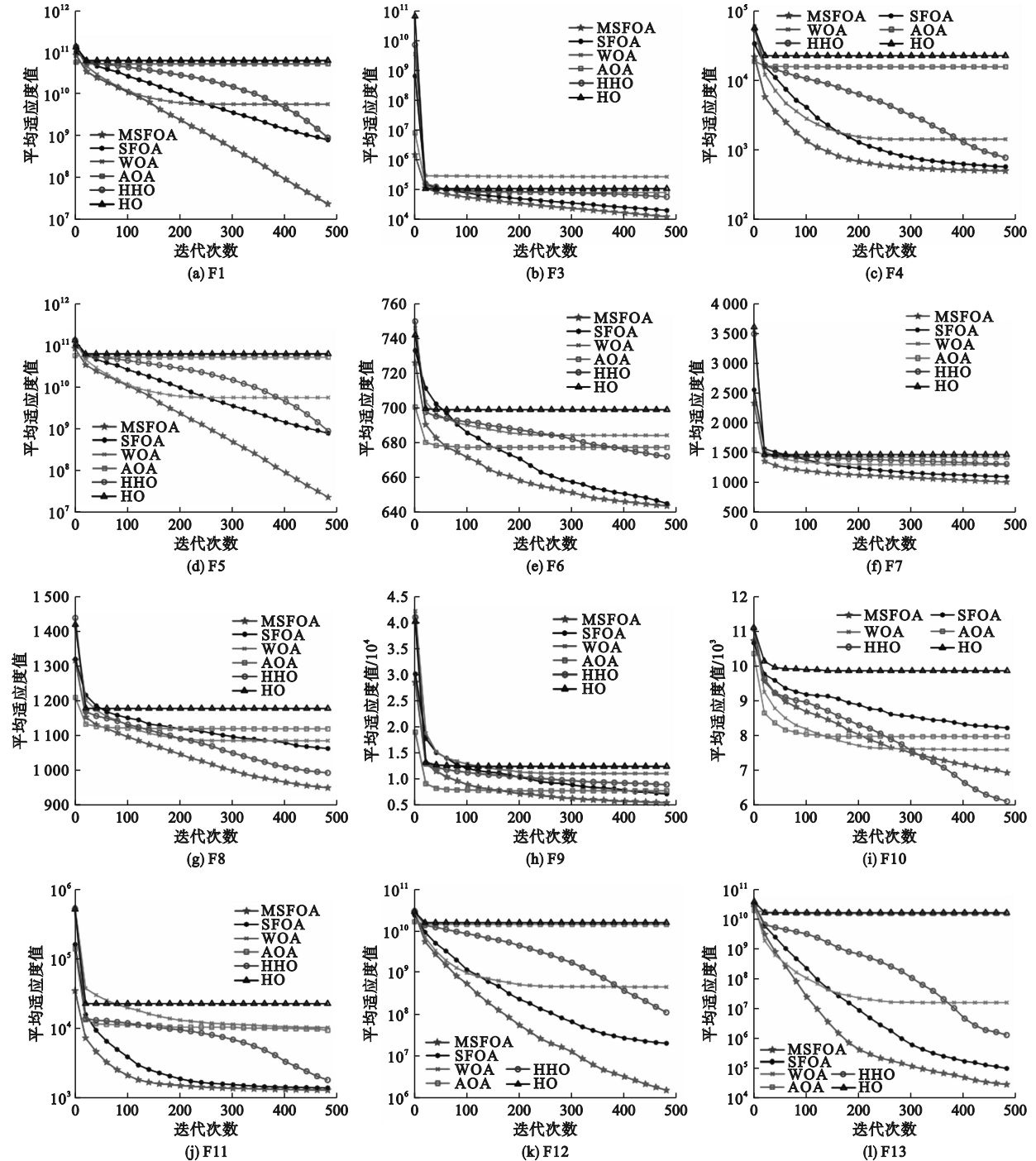


图 2 六种算法在不同测试函数上的收敛曲线

Fig. 2 Convergence curves of six algorithms on different test functions

表 1 六种算法在不同测试函数上的实验数据(维度为 30)

Table 1 Experimental data of six algorithms on different test functions ( $d = 30$ )

函数		MSFOA	SFOA	WOA	AOA	HHO	HO
F1	Mean	$2.09 \times 10^7$	$7.20 \times 10^8$	$4.66 \times 10^9$	$5.40 \times 10^{10}$	$4.64 \times 10^8$	$6.41 \times 10^{10}$
	Std	$8.71 \times 10^6$	$6.29 \times 10^8$	$1.39 \times 10^9$	$1.09 \times 10^{10}$	$6.63 \times 10^8$	$9.23 \times 10^9$
F3	Mean	$1.26 \times 10^4$	$2.02 \times 10^4$	$2.62 \times 10^5$	$8.15 \times 10^4$	$5.67 \times 10^4$	$1.16 \times 10^5$
	Std	$3.82 \times 10^3$	$7.12 \times 10^3$	$7.13 \times 10^4$	$1.12 \times 10^4$	$6.40 \times 10^3$	$2.99 \times 10^4$
F4	Mean	$5.14 \times 10^2$	$5.90 \times 10^2$	$1.37 \times 10^3$	$1.59 \times 10^4$	$7.02 \times 10^2$	$2.10 \times 10^4$
	Std	$2.12 \times 10^1$	$8.52 \times 10^1$	$4.67 \times 10^2$	$4.68 \times 10^3$	$8.88 \times 10^1$	$5.31 \times 10^3$
F5	Mean	$6.74 \times 10^2$	$7.83 \times 10^2$	$8.62 \times 10^2$	$8.94 \times 10^2$	$7.67 \times 10^2$	$9.47 \times 10^2$
	Std	$3.78 \times 10^1$	$4.64 \times 10^1$	$6.25 \times 10^1$	$4.18 \times 10^1$	$4.77 \times 10^1$	$3.59 \times 10^1$
F6	Mean	$6.40 \times 10^2$	$6.48 \times 10^2$	$6.79 \times 10^2$	$6.82 \times 10^2$	$6.67 \times 10^2$	$6.95 \times 10^2$
	Std	$1.07 \times 10^1$	$1.44 \times 10^1$	$1.20 \times 10^1$	7.40	5.79	$1.02 \times 10^1$
F7	Mean	$1.00 \times 10^3$	$1.09 \times 10^3$	$1.30 \times 10^3$	$1.40 \times 10^3$	$1.32 \times 10^3$	$1.46 \times 10^3$
	Std	$7.99 \times 10^1$	$4.63 \times 10^1$	$8.17 \times 10^1$	$4.89 \times 10^1$	$6.02 \times 10^1$	$6.46 \times 10^1$
F8	Mean	$9.47 \times 10^2$	$1.06 \times 10^3$	$1.08 \times 10^3$	$1.12 \times 10^3$	$1.00 \times 10^3$	$1.17 \times 10^3$
	Std	$3.45 \times 10^1$	$4.27 \times 10^1$	$4.72 \times 10^1$	$3.21 \times 10^1$	$2.74 \times 10^1$	$2.28 \times 10^1$
F9	Mean	$5.85 \times 10^3$	$7.90 \times 10^3$	$1.25 \times 10^4$	$8.44 \times 10^3$	$8.84 \times 10^3$	$1.28 \times 10^4$
	Std	$2.23 \times 10^3$	$2.69 \times 10^3$	$4.49 \times 10^3$	$1.20 \times 10^3$	$1.07 \times 10^3$	$1.70 \times 10^3$
F10	Mean	$6.80 \times 10^3$	$8.26 \times 10^3$	$7.74 \times 10^3$	$7.85 \times 10^3$	$6.05 \times 10^3$	$9.82 \times 10^3$
	Std	$9.91 \times 10^2$	$7.79 \times 10^2$	$7.84 \times 10^2$	$5.30 \times 10^2$	$6.67 \times 10^2$	$4.77 \times 10^2$
F11	Mean	$1.30 \times 10^3$	$1.42 \times 10^3$	$1.01 \times 10^4$	$1.03 \times 10^4$	$1.59 \times 10^3$	$2.33 \times 10^4$
	Std	$4.75 \times 10^1$	$6.94 \times 10^1$	$4.77 \times 10^3$	$3.36 \times 10^3$	$2.25 \times 10^2$	$1.16 \times 10^4$
F12	Mean	$1.30 \times 10^6$	$5.42 \times 10^6$	$4.64 \times 10^8$	$1.32 \times 10^{10}$	$7.62 \times 10^7$	$1.58 \times 10^{10}$
	Std	$2.28 \times 10^6$	$5.31 \times 10^6$	$3.31 \times 10^8$	$3.85 \times 10^9$	$7.08 \times 10^7$	$3.26 \times 10^9$
F13	Mean	$1.70 \times 10^4$	$8.92 \times 10^4$	$2.02 \times 10^7$	$1.36 \times 10^{10}$	$1.04 \times 10^6$	$1.66 \times 10^{10}$
	Std	$1.40 \times 10^4$	$7.09 \times 10^4$	$3.33 \times 10^7$	$5.52 \times 10^9$	$4.96 \times 10^5$	$7.96 \times 10^9$

Wilcoxon 秩和检验是一种非参数统计方法,当检验结果小于 0.05 时,算法间存在统计学层面的显著性能差异;而当检验结果大于 0.05,表明性能未呈现显著区分。为进一步验证 MSFOA 算法与

其他算法在性能上的显著差异,以五种算法在测试函数上独立运行 30 次的适应度值为样本,显著性水平设为 0.05 进行验证,检验结果如表 2 所示。

表 2 五种算法在 CEC2017 上的 Wilcoxon 秩和检验

Table 2 Wilcoxon rank-sum test of five algorithms on CEC2017

函数	SFOA	WOA	AOA	HHO	HO
F1	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$
F3	$6.74 \times 10^{-6}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$
F4	$5.97 \times 10^{-9}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.34 \times 10^{-11}$	$3.02 \times 10^{-11}$
F5	$1.96 \times 10^{-10}$	$4.50 \times 10^{-11}$	$3.02 \times 10^{-11}$	$2.44 \times 10^{-9}$	$3.02 \times 10^{-11}$
F6	$2.81 \times 10^{-2}$	$6.07 \times 10^{-11}$	$5.49 \times 10^{-11}$	$4.62 \times 10^{-10}$	$3.02 \times 10^{-11}$
F7	$4.69 \times 10^{-8}$	$3.34 \times 10^{-11}$	$3.02 \times 10^{-11}$	$4.50 \times 10^{-11}$	$3.02 \times 10^{-11}$
F8	$2.87 \times 10^{-10}$	$6.70 \times 10^{-11}$	$3.02 \times 10^{-11}$	$7.70 \times 10^{-8}$	$3.02 \times 10^{-11}$
F9	$2.27 \times 10^{-3}$	$7.12 \times 10^{-9}$	$1.02 \times 10^{-5}$	$6.01 \times 10^{-8}$	$3.69 \times 10^{-11}$
F10	$6.53 \times 10^{-6}$	$1.70 \times 10^{-3}$	$1.53 \times 10^{-4}$	$6.67 \times 10^{-2}$	$3.34 \times 10^{-11}$
F11	$9.26 \times 10^{-9}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$1.33 \times 10^{-10}$	$3.02 \times 10^{-11}$
F12	$1.69 \times 10^{-9}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$4.08 \times 10^{-11}$	$3.02 \times 10^{-11}$
F13	$9.06 \times 10^{-8}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$	$3.02 \times 10^{-11}$

从检验结果来看,在几乎所有场景下的检验结果均小于 0.05,仅在 F6、F10 中展现出一定的竞争性能。从该统计学角度可证实,MSFOA 算法的性能提升并不是因为偶然性所造成的,而是其加入的改进策略发挥的作用,验证了改进策略的合理性与有效性。

### 3 VMD-MSFOA-LSTM 短期电力负荷预测模型性能评估

#### 3.1 模型构建逻辑与流程

由于原始负荷数据具有随机性和非平稳性,

利用 VMD 数据分解技术使时序数据更加平稳化,从而构建 VMD-MSFOA-LSTM 模型的短期电力负荷预测方法,实现过程如图 3 所示。具体实现步骤如下:

步骤 1 收集原始负荷数据并对数据预处理;

步骤 2 利用 VMD 将数据分解成一系列 IMF 分量;

步骤 3 利用 MSFOA 算法优化后的 LSTM 网络预测所有子序列,MSFOA 算法优化的是 LSTM 中隐藏单元数目、最大训练周期和初始学习率;

步骤 4 将各分量的预测结果汇总,得到原始负荷数据的最终预测值,即 VMD-MSFOA-LSTM 的电力负荷预测结果。

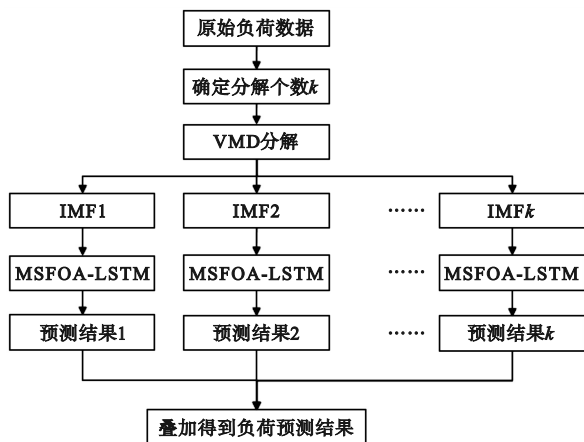


图 3 VMD-MSFOA-LSTM 模型预测流程图

Fig. 3 Prediction flowchart of VMD-MSFOA-LSTM model

### 3.2 数据集选取及模型评价指标

数据集选取的是澳大利亚某地区 2006 年 1 月 1 日至 31 日一个月的数据,以采样间隔为 30 min 的数据作为输入模型样本,即每天包含 48 条数据,总计 1 488 条,能精准捕捉日内负荷波动。并且该数据集数据完整性高,仅存在 3 处缺失值且没有极端异常值,经周期性填补与归一化处理,满足模型训练需求<sup>[18]</sup>。正因为该数据集负荷波动贴近实际应用场景并且数据规模适中,保证模型充分学习的同时又能避免过拟合。将该数据集前 70% 作为训练数据,后 30% 作为预测数据进行训练。

模型评价指标采用均方根误差 (RMSE)、平均绝对误差 (MAE)、平均绝对百分比误差 (MAPE) 和  $R^2$  来评价模型的预测误差。

### 3.3 实验结果分析

为解决 LSTM 模型超参数手动设置的盲目

性,本文选取对其性能影响最为显著的三个核心超参数进行协同优化。首先隐藏单元数目直接决定 LSTM 网络的模型容量与特征捕捉能力。其次训练周期直接影响模型的收敛状态,找到最佳停止点才能平衡学习效果与泛化能力。最后初始学习率作为控制神经网络权重更新步长的关键参数,其取值会直接影响优化过程的收敛效果与稳定性。上述三个参数分别从模型结构、训练过程和优化动力学三个不同维度共同决定 LSTM 的最终性能,因此找到使其进行协同优化的最优参数至关重要。

#### 3.3.1 超参数优化结果

通过 MSFOA 算法在迭代中不断搜索寻找一组能够使适应度函数值最小的最优超参数组合,其优化后的超参数结果如表 3 所示。

表 3 超参数取值结果

Table 3 Hyperparameter value results

超参数	数值
隐藏单元数目	40
训练周期	200
初始学习率	0.023 23

#### 3.3.2 预测结果对比

为验证提出模型的有效性,选取 LSTM、VMD-LSTM、VMD-SFOA-LSTM 和 VMD-MSFOA-LSTM 四个模型在同一数据集上进行实验,得到的预测结果如图 4 和表 4 所示。

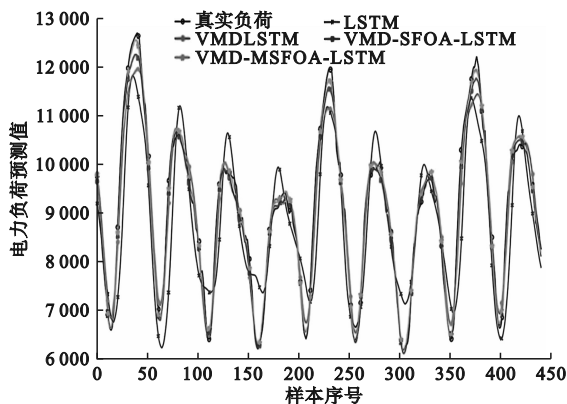


图 4 预测结果对比图

Fig. 4 Comparison chart of prediction results

从预测结果对比以及评估指标可以看出,经过数据分解后的模型预测效果显著优于未进行数据分解的模型。与 LSTM 模型相比,VMD-LSTM 模型的 RMSE、MAE 和 MAPE 分别降低

57.98%、63.97% 和 66.77%,  $R^2$  提高 14.46%, 说明 LSTM 模型的预测精度与输入数据质量密切相关。与 VMD-LSTM 模型相比, VMD-SFOA-LSTM 模型的 MAE、RMSE 和 MAPE 分别降低 43.51%、33.18% 和 31.18%,  $R^2$  提高 2.20%, 进一步表明采用 SFOA 算法优化 LSTM 超参数, 能够有效解决参数手动设置不当对预测效果产生的影响。与 VMD-SFOA-LSTM 模型相比, VMD-MSFOA-LSTM 模型的 MAE、RMSE 和 MAPE 分别降低 43.27%、44.80% 和 45.73%,  $R^2$  提高 0.7%。验证了策略加入的有效性, 使超参数更加精确, 能够提高短期电力负荷预测精度。

表 4 负荷预测评估结果

Table 4 Results of load forecasting evaluation metrics

模型	RMSE	MAE	MAPE/%	$R^2$
LSTM	670.191	534.675	6.049	0.828
VMD-LSTM	281.548	192.615	2.010	0.967 9
VMD-SFOA-LSTM	159.040	128.704	1.383	0.990
VMD-MSFOA-LSTM	90.228	71.047	0.751	0.997

为进一步验证 VMD-MSFOA-LSTM 模型预测性能提升的统计显著性, 采用 Wilcoxon 符号秩和检验对模型在测试集上的预测误差分布进行对比分析。选取 VMD-MSFOA-LSTM 作为基准, 分别与 LSTM、VMD-LSTM 和 VMD-SFOA-LSTM 模型的预测绝对误差序列进行两两比较。显著性水平设为 0.05。检验结果如表 5 所示, VMD-MSFOA-LSTM 与其余三个对比模型之间的检验结果均小于 0.05, 低于显著性水平。该统计结果证明了本文所提 VMD-MSFOA-LSTM 模型与其余三个模型之间性能差异具有统计学意义, 其预测精度显著优于对比模型。

表 5 Wilcoxon 秩和检验结果

Table 5 Wilcoxon rank-sum test results

对比模型	检验结果
LSTM	$1.53 \times 10^{-5}$
VMD-LSTM	$6.67 \times 10^{-3}$
VMD-SFOA-LSTM	$4.50 \times 10^{-2}$

## 4 结论

为提高短期电力负荷预测的精度, 本文提出

一种新的电力负荷预测模型 VMD-MSFOA-LSTM, 并通过与不同模型对比验证了该模型对于电力负荷预测的优越性与实用性。其中 VMD 对采集到的原始数据进行分解预处理, 然后针对 SFOA 算法的缺陷加入佳点集初始化策略、动态对立学习机制、线性递减系数和精英增强解质量策略, 实验证明 MSFOA 算法对提升 LSTM 超参数优化的精度起到了更好的效果。

尽管本文构建的模型在短期负荷预测中表现优异, 但是在向中长期预测领域迁移时, 明显存在特征体系适配性不足、长时序依赖捕捉能力有限等一系列问题, 难以满足中长期预测对周期性等特征挖掘的需求。未来重点优化特征维度与时序结构, 推动模型向中长期负荷预测领域迁移, 为能源系统中长期规划与跨周期调度提供更好的支撑。此外, 本研究中的 VMD 分解参数是根据经验观察设定。尽管当前设置已取得良好效果, 但并非全局最优解。VMD 参数的选取直接影响子序列的平稳性和可预测性, 未来的研究方向将构建 VMD-MSFOA-LSTM 的完全自适应预测框架, 即利用优化算法同步寻找 VMD 与 LSTM 的最优参数组合, 以期进一步提升模型的自动化水平和预测精度。

## 参考文献 (References):

- [1] Shao L, Guo Q J, Li C, et al. Short-term load forecasting based on EEMD-WOA-LSTM combination model [J]. Applied Bionics and Biomechanics, 2022, 2022: 2166082.
- [2] VASUDEVAN S, JOTHINATHAN K. Short-term electrical load forecasting through optimally configured long short-term memory [J]. Majlesi Journal of Electrical Engineering, 2024, 18(4): 1-10.
- [3] 蔡春雷, 刘微, 任腾腾. 基于 HDBO-LSTM 的短期电力负荷预测方法 [J]. 沈阳理工大学学报, 2025, 44(5): 21-28.  
Cai C L, Liu W, Ren T T. Short-term power load forecasting method based on HDBO-LSTM [J]. Journal of Shenyang Ligong University, 2025, 44(5): 21-28. (in Chinese)
- [4] Fang M K, Zhang F F, Zhu D, et al. The influence of variational mode decomposition on LSTM prediction accuracy—a case study with wind turbine power signals [J]. Renewable Energy, 2025, 245: 122863.
- [5] Dragomiretskiy K, Zosso D. Variational mode decomposition [J]. IEEE Transactions on Signal Processing, 2014, 62(3): 531-544.
- [6] 盛四清, 金航, 刘长荣. 基于 VMD-WSGRU 的风电场发电功率中短期及短期预测 [J]. 电网技术, 2022, 46(3): 897-904.  
Sheng S Q, Jin H, Liu C R. Short-term and mid-short-term wind power forecasting based on VMD-WSGRU [J]. Power System Technology, 2022, 46(3): 897-904. (in Chinese)

(下转第 58 页)

- [15] Zhang L Y, Wu X, Liu Z H, et al. ESD-YOLOv8: an efficient solar cell fault detection model based on YOLOv8 [J]. IEEE Access, 2024, 12: 138801 – 138815.
- [16] Liu Z G, Sun B S, Bi K Y. Optimization of YOLOv7 based on PConv, SE attention and wise-IoU [J]. International Journal of Computational Intelligence and Applications, 2024, 23: 2350033.
- [17] Hughes D P, Salathé M. An open access repository of images on plant health to enable the development of mobile disease diagnostics [PP/OL]. arXiv (2016 – 04 – 12) [2025 – 08 – 15]. <https://doi.org/10.48550/arXiv.1511.08060>.
- [18] Universitas Atma Jaya. Tomato leaf disease dataset [DS/OL]. Roboflow Universe (2023 – 04 – 01) [2025 – 08 – 15]. <https://universe.roboflow.com/universitas-atma-jaya/tomato-leaf-disease-rxcft>.
- [19] Zhu X K, Lyu S C, Wang X, et al. TPH-YOLOv5: improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios [C]//2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). Montreal, QC, Canada; IEEE, 2021: 2778 – 2788.
- [20] Hussain M. YOLOv1 to v8: unveiling each variant; a comprehensive review of YOLO [J]. IEEE Access, 2024, 12: 42816 – 42833.
- [21] Murat A A, Kiran M S. A comprehensive review on YOLO versions for object detection [J]. Engineering Science and Technology, an International Journal, 2025, 70: 102161.
- [22] Ali M L, Zhang Z. The YOLO framework: a comprehensive review of evolution, applications, and benchmarks in object detection [J]. Computers, 2024, 13 (12): 336.
- [23] Sapkota R, Flores-Calero M, Qureshi R, et al. YOLO advances to its genesis: a decadal and comprehensive review of the You Only Look Once (YOLO) series [J]. Artificial Intelligence Review, 2025, 58 (9): 274.

(责任编辑:宋颖韬)

## (上接第41页)

- [16] Rao D Y, Xu T Y, Wu X J. TGFuse: An infrared and visible image fusion approach based on transformer and generative adversarial network [J]. IEEE Transactions on Image Processing, 2023, PP (99): 1.
- [17] Zhao Z X, Xu S, Zhang C X, et al. DIDFuse: deep image decomposition for infrared and visible image fusion [C]//2020 29th International Joint Conference on Artificial Intelligence (IJCAI). Yokohama, Japan; IJCAI Organization, 2020: 970 –

976.

- [18] Li H, Wu X J. DenseFuse: a fusion approach to infrared and visible images [J]. IEEE Transactions on Image Processing, 2019, 28 (5): 2614 – 2623.
- [19] Li H, Wu X J, Durrani T. NestFuse: an infrared and visible image fusion architecture based on nest connection and spatial/channel attention models [J]. IEEE Transactions on Instrumentation and Measurement, 2020, 69 (12): 9645 – 9656.

(责任编辑:和晓军)

## (上接第49页)

- [7] Awad N H, Ali M Z, Suganthan P N. Ensemble sinusoidal differential covariance matrix adaptation with Euclidean neighborhood for solving CEC2017 benchmark problems [C]//2017 IEEE Congress on Evolutionary Computation (CEC). Donostia, Spain; IEEE, 2017: 372 – 379.
- [8] Zhang G, Liu H C, Zhang J B, et al. Wind power prediction based on variational mode decomposition multi-frequency combinations [J]. Journal of Modern Power Systems and Clean Energy, 2019, 7 (2): 281 – 288.
- [9] Van Houdt G, Mosquera C, Nápoles G. A review on the long short-term memory model [J]. Artificial Intelligence Review, 2020, 53 (8): 5929 – 5955.
- [10] Zhong C T, Li G, Meng Z, et al. Starfish optimization algorithm (SFOA): a bio-inspired metaheuristic algorithm for global optimization compared with 100 optimizers [J]. Neural Computing and Applications, 2025, 37 (5): 3641 – 3683.
- [11] Oueslati R, Ouertani M W, Amdouni A, et al. MS-FSOA-LightGBM: multi-strategy starfish optimization algorithm with LightGBM for software defect prediction [J]. Procedia Computer Science, 2025, 270: 4054 – 4063.
- [12] Liu S L, Jin Z K, Lin H T, et al. An improve crested porcupine algorithm for UAV delivery path planning in challenging environments [J]. Scientific Reports, 2024, 14: 20445.

- [13] Abdollahpour A, Rouhi A, Pira E. An improved gazelle optimization algorithm using dynamic opposition-based learning and chaotic mapping combination for solving optimization problems [J]. The Journal of Supercomputing, 2024, 80 (9): 12813 – 12843.
- [14] 尚凯凯. 融合扰动策略的自适应哈里斯鹰优化算法 [J]. 计算机与数字工程, 2025, 53 (2): 338 – 346.
- Shang K K. Adaptive Harris Hawks optimization algorithm based on disturbance strategies [J]. Computer and Digital Engineering, 2025, 53 (2): 338 – 346. (in Chinese)
- [15] Mirjalili S, Lewis A. The whale optimization algorithm [J]. Advances in Engineering Software, 2016, 95: 51 – 67.
- [16] Hashim F A, Hussain K, Houssein E H, et al. Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems [J]. Applied Intelligence, 2021, 51 (3): 1531 – 1551.
- [17] Amiri M H, Mehrabi Hashjin N, Montazeri M, et al. Hippopotamus optimization algorithm: a novel nature-inspired optimization algorithm [J]. Scientific Reports, 2024, 14: 5032.
- [18] Guan Y M, Shi Y L, Wang G, et al. Electricity behaviors anomaly detection based on multi-feature fusion and contrastive learning [J]. Information Systems, 2025, 127: 102457.

(责任编辑:和晓军)