

分类号: TP394

学校代码: 10363

密 级: 公开

学 号: 2220320134



安徽工程大学

Anhui Polytechnic University

硕士学位论文

题目 基于二次分解和IQPSO-BILSTM的

风速组合预测模型

论 文 作 者 侯浩天

校 内 导 师 孙驷洲 副教授

校 外 导 师 柳士江

专业学位类别 电子信息

研究方向(领域) 控制理论与控制工程

论文提交日期: 2025年 5月 18日

分类号：TP394

单位代码：10363

密类级：公开

学位代号：2220320134

题 目： 基于二次分解和 IQPSO-BILSTM 的
风速组合预测模型

题 目： A HYBRID MODEL BASED ON TWO-LAYER
DECOMPOSITION AND IQPSO-BILSTM FOR
WIND SPEED PREDICTION

学 生 姓 名： 侯浩天

校 内 导 师： 孙驷洲副教授

校 外 导 师： 柳士江

专 业 学 位 类 别： 电子信息

研 究 方 向 (领 域)： 控制理论与控制工程

论文答辩日期： 2025 年 5 月 10 日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名： 侯浩天

日期：2025年5月18日

安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 ☐，在 ____ 年解密后适用本版权书。

本学位论文属于

不保密 ☒。

学位论文作者签名：

侯浩天

指导教师签名：

孙翊洲

日期：2025 年 5 月 18 日

日期：2025 年 5 月 18 日

基于二次分解和 IQPSO-BILSTM 的风速组合预测模型

摘 要

随着社会对能源需求的不断增长,传统化石能源日益枯竭且对环境严重污染,为此,我国正积极推进清洁能源的发展。风力发电作为一种清洁、可再生的能源形式,已成为全球能源开发的核心领域之一。然而,由于风速具有波动性大和随机性强的特性,风力存在的不确定性导致并网过程中会对电力系统的运行、调度和成本等方面产生显著负面影响。大量研究表明,准确风速预测对于风能的开发与利用,以及电力系统的安全稳定运行等具有至关重要的作用。因此,本文构建了基于二次分解和 IQPSO-BILSTM 的新型风速组合预测模型,以提高风速预测精度。主要工作和研究容如下:

(1)针对传统量子粒子群(QPSO)算法的初始种群遍历差、收缩-膨胀系数无法促进种群多样性和迭代后期种群聚集强引起局部的最优问题,对传统 QPSO 算法进行改进,获得改进型 QPSO,即 IQPSO 算法。首先,采用 Tent 映射函数初始化种群,使初始粒子更均匀地遍历搜索空间,提升可求解的遍历性。其次,利用基于目标函数动态变化的收缩-膨胀系数调整方法,使系数能够根据问题特性自适应地调整,增强算法的适应性。最后,引入柯西变异操作,在迭代过程中对粒子位置进行扰动,使粒子跳出局部最优解,进入其它可能解的空间,以提升算法全局搜索能力。

(2)针对自适应噪声完备集成经验模态(CEEMDAN)分解方法在

分解过程中产生较大排列熵的模态分量，本文采用变分模态分解(VMD)对排列熵较大的模态分量进行二次分解，降低较大排列熵模态分量的复杂程度。VMD 能够有效分离信号中的噪声和有效信息，从而降低模态分量的排列熵，提升风速分解效果。通过二次分解，消除风速中噪声分量，能有效地提高预测模型的预测性能和稳定性。

(3)为了提高双向长短期记忆网络(BILSTM)捕捉风速局部特征能力和对重要风速特征的关注度，采用一维卷积神经网络(1D-CNN)对风速数据的局部特征进行提取；此外，提高模型对重要特征的关注度，利用注意力机制(Attention)对 BILSTM 输出值进行加权处理，自动关注对风速预测影响较大的特征，以提升模型的预测精度。

(4)为了验证本文新建的风速组合预测模型，利用内蒙古某风电场的实测风速数据对预测模型进行测试，并和其它典型的风速预测模型进行对比。实验结果显示，基于二次分解和 IQPSO-BILSTM 的风速组合预测模型，不论单步还是多步预测均展现出较高的预测精度和稳定性。

关键词：风速预测，二次分解，改进量子粒子群算法，深度学习，组合模型

A HYBRID MODEL BASED ON TWO-LAYER DECOMPOSITION AND IQPSO-BILSTM FOR WIND SPEED PREDICTION

ABSTRACT

With the continuous growth of society's demand for energy, traditional fossil energy is increasingly depleted and causes severe environmental pollution. Therefore, China is actively promoting the development of clean and renewable energy. Wind power generation, as a clean, renewable and economically resource, has become one of the key areas of global energy development. However, due to the highly fluctuating and random nature of wind speed, the uncertainty of wind power can have significant negative impacts on the operation, dispatching and cost of the power system during the grid connection process. A large number of studies have shown that accurate wind speed prediction is of crucial importance for the development and utilization of wind energy and the safe and stable operation of the power system. Therefore, this paper constructs a new wind speed combined prediction model based on secondary decomposition and IQPSO-BILSTM. The main work and research contents are shown as follows:

(1) In order to solve the problems of poor initial population traversal and fixed shrinkage-expansion coefficient of the traditional Quantum Particle Swarm (QPSO) algorithm that cannot promote population diversity and the local optimal problem of strong population aggregation in the late iteration stage, the traditional QPSO algorithm was improved to obtain the improved QPSO. Firstly, the Tent function is used to map the initial population, so that the initial particles traverse the search space more uniformly, and the candidate solution is improved. Secondly, the shrinkage-expansion coefficient adjustment method based on the dynamic change of the objective function is used to make the coefficient adaptively adjust according to the characteristics of the problem, so as to enhance the adaptability of the algorithm. Finally, the Cauchy mutation operation is introduced to perturb the position of the particles in the iterative process, so that the particles jump out of the local optimal solution and enter the space of other candidate solutions, which can significantly improve the global search ability of the algorithm.

(2) In view of the modal components with large permutation entropy generated by the Adaptive Noise Complete Ensemble Empirical Mode Decomposition (CEEMDAN) method in the decomposition process, variational mode decomposition (VMD) is used to decompose the modal components with large permutation entropy to reduce the complexity of the modal components with large permutation entropy. VMD

can effectively separate the noise and effective information in the signal, thereby reducing the entropy of the arrangement of the modal components and improving the effect of wind speed decomposition. Through secondary decomposition, the noise component in the wind speed can be eliminated, which can effectively improve the prediction performance and stability of the prediction model.

(3) In order to improve the ability of Bidirectional Long Short-Term Memory Network (BILSTM) to capture the local features of wind speed and pay attention to important wind speed features, the one-dimensional convolutional neural network (1D-CNN) was used to extract the local features of wind speed data. In addition, the attention of the model to important features is improved, and the BILSTM output value is weighted by the attention mechanism, and the features that have a greater impact on the wind speed prediction are automatically paid attention to, so as to improve the prediction accuracy of the model.

(4) In order to verify the newly built wind speed combination prediction model in this paper, the measured wind speed data of wind farm in Inner Mongolia province were used to test the prediction model and compare it with other typical wind speed prediction models. The experimental results show that the combined wind speed prediction model based on two-layer decomposition and IQPSO-BILSTM shows high prediction accuracy and stability regardless of single-step or multi-step prediction.

KEYWORDS: Wind speed forecasting, Secondary decomposition, Improved Quantum Particle Swarm Optimization, Deep learning model, Hybrid models.

目 录

摘 要.....	I
ABSTRACT.....	III
第 1 章 绪论.....	1
1.1 研究背景及意义.....	1
1.2.1 风速预测研究背景.....	1
1.2.1 风速预测研究意义.....	3
1.2 国内外研究现状.....	3
1.3 研究现状分析.....	10
1.4 研究内容及章节安排.....	10
第 2 章 改进量子粒子群算法.....	13
2.1 粒子群算法工作原理.....	13
2.2 量子粒子群算法工作原理.....	15
2.3 改进量子粒子群算法.....	17
2.4 IQPSO 算法性能测试.....	20
2.1 本章小结.....	23
第 3 章 构建基于二次分解和 IQPSO-BILSTM 的风速组合预测模型.....	24
3.1 风速分解相关理论.....	24
3.1.1 经验模态分解.....	24
3.1.2 自适应噪声完备集成经验模态分解.....	25
3.1.3 变分模态分解.....	26
3.1.4 排列熵.....	27
3.2 卷积神经网络.....	28
3.3 双向长短期记忆网络.....	32
3.3.1 循环神经网络原理.....	32
3.3.2 双向长短期记忆网络.....	33
3.4 注意力机制原理.....	35

3.5 构建基于二次分解和 IQPSO-BILSTM 的风速组合预测模型	36
3.6 本章小结.....	41
第 4 章 实验结果分析	42
4.1 风速数据预处理.....	42
4.1.1 实验数据选择.....	42
4.1.2 数据分解.....	44
4.2 滑动预测机理.....	49
4.3 实验 I：模型精确性验证.....	50
4.3.1 基于二次分解预测模型准确性验证.....	51
4.3.2 本文模型准确性验证.....	54
4.4 实验 II：模型稳定性验证.....	57
4.5 本章小结.....	58
第 5 章 结论与展望	59
5.1 研究结论.....	59
5.2 研工作展望.....	60
参考文献.....	61
攻读硕士期间取得的学术成果.....	65
致谢.....	66

第 1 章 绪论

1.1 研究背景及意义

1.2.1 风速预测研究背景

社会的进步离开不了能源的消耗，2023 年中国各类能源消耗占比如图 1-1 所示。目前我国主要的能源来源于传统化石燃料(煤，石油，天然气)^[1]，这类能源的消耗占据我国能源消耗的 82%。然而，化石能源的产生需要上万年时间的积累，对于人类来说这些化石能源是不可再生的能源。改革开放以来，我国经济经历快速增长，对于能源的需求也日益增长。因此，我国政府大力发展可再生能源的开发和利用^[2]。

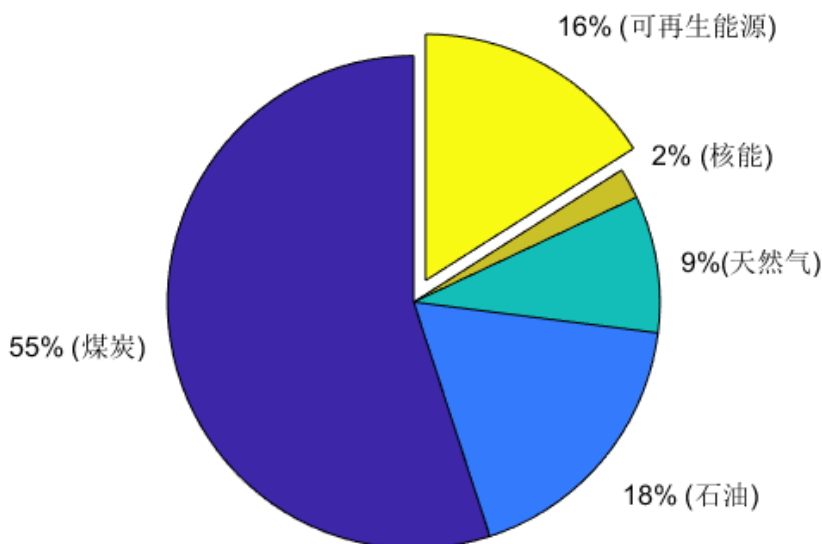


图 1-1 2023 年中国各类能源消耗占比

我国近年来能源消费总量如图 1-2 所示，从 2004 年的 18.8 亿吨标准煤，逐步增长到 2023 年已达 57.2 亿吨标准煤。该能源消耗增长体现了我国经济的快速发展，其中大部分为化石能源，大量消耗化石燃料会对环境产生大的污染和温室效应^[3]。而风能作为一种无污染的清洁可再生能源，是代替化石能源的最好选择之一。

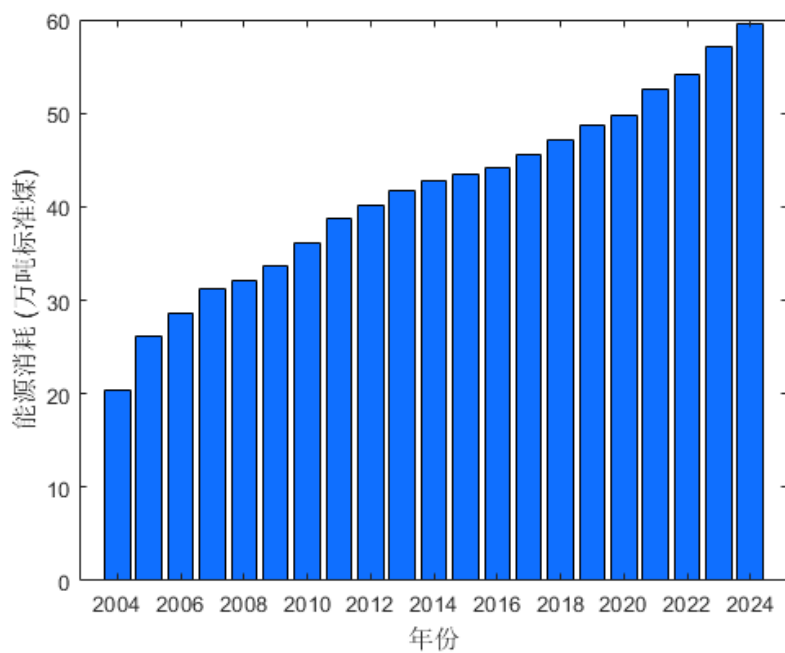


图 1-2 2004 年至 2023 中国能源消耗总量(GW)

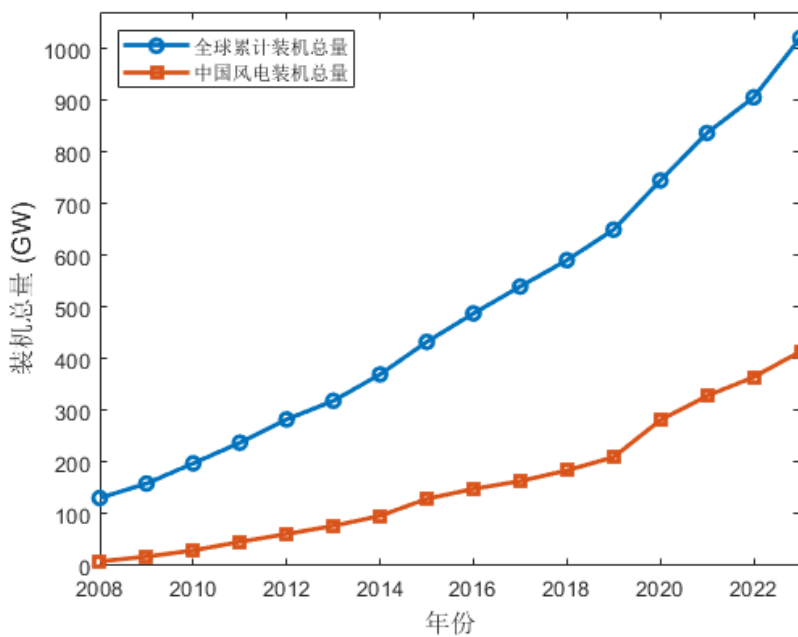


图 1-3 2008 年至 2023 年风电累计装机总量

近年来,我国风电产业的发展取得了显著成效,如图 1-3 所示,2008 年全世界风电装机总量为 131GW,而中国风电装机总量仅为 8.39GW。2023 年,全球风电累计装机总量已经达到 1021GW,而中国风电装机容量则从 8.39GW 增长至 414.34GW^[4]。由此可以看出,全世界对于风电项目的投入越来越大。其中,我

国对于风电的发展尤为重视，目前，我国的风电装机总量已占全球市场的重要份额^[5]。随着“十四五”规划以及“碳达峰、碳中和”战略目标的提出，中国将要大力提升风电发电规模^[6]。

1.2.1 风速预测研究意义

风力发电不仅能为社会的发展提供清洁能源，还能减轻过度消耗传统化石能源带来的环境污染。然而，风速具有随机性，波动性和间歇性等特点，这些特点使得风电并网时电力系统的安全稳定大大降低^[7]。因此，精确的风速预测可以帮助风电场提前调整发电计划，优化电力调度，从而提高风电的利用率和电网的稳定性^[8]。

不同于火力发电，风力发电的动力来源是风能。通过对风速进行预测，管理人员能够更好地了解未来一段时间内不同地区的风能资源分布情况，选择最合适的建设地点建设风电场。同时，通过准确的风速预测，可以帮助风电场的维修人员对风电机组维护制定合理的计划。若是在未来某段时间内风速较高，维修人员可以在风速较低的时段，则可以对风力发电机进行检修和维护，减少发电损失。可见高精度的风速预测对风电的发展具有重要意义。

1.2 国内外研究现状

为了有效应对风速的随机性，波动性和间歇性等特点对风力发电的影响，国内外许多研究人员提出了许多有关风速预测的方法。通过这些方法可以准确的进行风速预测，提高风电场的运作效率。风速预测大致可以从预测的时间尺度、预测范围、预测方法及建模机理等方面进行分类^[9]，如图 1-4 所示。



图 1-4 风速预测分类

一、按照风速预测尺度分

风速预测时间尺度还没有统一的定义，根据文献[10] 可以将其分为超短期预测、短期预测、中期预测和长期预测。超短期预测通常指几秒至十几分钟的时间范围，通过超短期预测可以及时调整风力发电机的工作状态，提高发电效率以及保障电网的稳定性。短期预测的时间范围为十几分钟至 6 小时，通过短期预测，可以提前掌握未来一段时间内的风速变化，为电网调度提供参考，优化能源利用，避免电力过剩或不足的情况发生。中期预测范围为 6 小时至 1 天，通过中期预测，可以进行电力系统管理和对能源交易的策略进行调整，从而实现资源合理高效的配置。长期预测通常指 1 天至 1 周或更长时间的预测，通过

对风速进行长期预测，可以提前安排维护工作，避免设备的突发故障。

二、按照风速预测范围分

根据预测范围的不同，风速预测可以分为单风机预测、风电场预测和风电区域预测^[11]。单风机预测能够精确统计每台风机实时输出功率，根据风场中的风能分布情况，帮助调度部门调整风机出力。风场预测的预测对象是风电场内的众多风机，根据该风电场风机整体的出力情况，帮助电力调度部门精确部署出力计划^[12]。风电区域预测覆盖更大范围，通过某个地区中几个风电场的整体发电情况，评估该地区用电情况、并网电能质量、电力经济效益等各项指标。

三、按照风速预测方法分

根据建模方法的不同，风速预测可以分为物理模型、统计模型、机器学习模型和深度学习模型四种类型^[13]。

(1)物理模型

基于物理模型的风速预测方法，主要是指数值天气预报(NWP)^[14]和天气研究与预报模式(WRF)^[15]。该模型建立在动力气象学的物理方程之上，通过气象站、雷达和气象卫星等设备完成资料搜集，再与现有的预报资料统结合。目前大多数的国家采用该模型进行天气预报。文献[[16] 提出计及 NWP 风速横纵向误差的循环确定性策略梯度(RDPG)风速修正模型，旨在解决现有方法忽视 NWP 风速序列与历史风速序列间超前或延迟效应导致的横向误差问题。通过确定 NWP 风速待修正点的超前/滞后时序区间(LLTI)，利用 LLTI 内的统计特征替代原始气象特征，并采用沙普利可加性解释法(SHAP)评估各特征对模型输出的贡献^[18]。最后，建立基于 RDPG 的 NWP 风速修正模型。文献[18]不仅考虑了 NWP 的风速、风向、温度、湿度、压强这 5 类典型特征，还考虑多个位置的 NWP 信息，并采用了最大相关-最小冗余原则提取输入变量。结果表明,该模型方法能有效的提取风速信息,有利于提高预测精度。文献[19]提出基于 WRF 数值模式的深度置信网络(DBN)风速预测模型研究，以 WRF 模式预报输出的多种气象数据为基础，通过 DBN 挖掘气象数据和风速数据之间的关系^[20]。结果表明，该模型能很好地预测风速并且得到了较高的预测精度。虽然物理模型能够提供较为准确的长期趋势预测，但该方法对于短期的风速预测不够准确。

(2)统计模型

统计模型主要通过学习历史风速信息对未来的风速进行预测，主要采用时间序列分析方法和空间相关性法^[21]。文献[22]针对风速的非线性非平稳特性，提出了一种结合经验模态分解(EMD)和自回归滑动平均模型(ARMA)的预测方法。引入高频谐波克服 EMD 分解中模态混叠，并对风速经行分解得到较为平稳的模态分量(IMF)，然后利用 ARMA 模型对各 IMF 进行建模预测，最后将各子序列的预测结果相加，得到风速的最终预测值。文献[23] 提出局域均值分解(LMD)和 ARMA 组合风速预测方法，通过 LMD 分解将原始风速信号分解为不同频率的 IMF，分别建立 ARMA 时间序列模型进行预测，最后累加各个分量的预测值得到预测结果。将 LMD-ARMA 与 EMD-ARMA、ARMA 模型的预测结果进行误差分析，结果表明，局域均值分解与时间序列相结合建立的模型预测误差相对较低。文献[24]提出基于修正后 ARIMA-GARCH 模型的超短期风速预测方法，利用模糊理论、风速概率分布特性及相关性分析，对 GARCH 模型中的条件方差进行加权混合修正，以更精确进行点预测与区间预测，通过中国甘肃气象站的风速数据验证，证明了该方法的可行性和有效性。

(3)机器学习模型

在机器学习模型中，首先从预处理的风速数据中提取特征，随后利用历史数据训练和优化预测模型，以提高预测结果的准确性。最终，通过训练好的模型对未来风速进行预测并输出结果。典型的机器学习模型包括支持向量机(SVM)^[25]、最小二乘支持向量机(LSSVM)^[26]、XGBoost^[27]等。文献[28] 采用 ARIMA 算法来预测每个分解后的子序列。并计算预测序列和真实值之间的误差得到误差序列，使用 SVM 对该误差序列进行预测。将误差预测值和子序列预测值相加得到最终的预测值。为验证模型的泛化性，采用三个数据集经行验证，实验结果表明，该模型具有更好的预测能力。文献[29]将风速序列通过小波分解(WD)和变分模态分解(VMD)得到一系列风速子序列。对这些子序列分别建立 LSSVM 的预测模型，并累加这些子序列的预测值。文献[30] 首先使用 PSO - LSSVM 模型对小波阈值去噪后的风速序列进行预测。其次计算预测值和真实值之间的误差，并利用 XGBoost 对误差进行修正。最后，将修正误差和原始预测数据相结合得到最终预测值。与物理模型和统计模型相比，机器学习模型的预测精度有所提高，但还是会存在过拟合的问题。

(4)深度学习模型

深度学习是通过建立多层神经网络模型，从大数据中提取特征并进行学习与预测^[31]。常见的深度学习模型包括 BP 神经网络、卷积神经网络(CNN)、循环神经网络(RNN)、长短期记忆网络(LSTM)以及前后长短期记忆网络(BILSTM)等。文献[32]首先使用快速经验模态分解(FEEMD) 将原始风速分解，通过各个子序列的熵值，将熵值相似的子序列合并。然后利用麻雀搜索算法(SSA)对 BP 神经网络的学习率进行迭代寻优。最后将训练好的 BP 神经网络对合并后的子序列进行预测，得到最终的风速预测结果。文献[33] 通过 CNN 对气象数据的局部特征进行提取，利用双向门控循环单元(BIGRU)提取局部特征中的时序性。此外，引入 注意力机制(Attention)，为 BIGRU 隐式状态赋予不同的权重，以完成超短期风速预测^[34]。最后，利用中国西北某风电场的实际数据对模型的可行性和有效性进行验证。文献[35]首先使用孤立森林(Isolated Forest) 算法对原始风速数据进行异常值检测。然后，采用同步压缩小波变换 (SWT)方法对原始风速数据进行去噪，最后，通过 LSTM 建立风速预测模型对处理后的风速数据进行预测。文献[36] 提出了一种 CNN、BILSTM 和基于时变滤波器的经验模态分解(TVFEMD) 的深度学习模型。通过 TVFEMD 将原始风速数据分解为规律性较强的本征模态函数 (IMF)，使用随机森林 (RF) 分析每个 IMF，并将 IMF 合并减少训练时常。采用 CNN-BILSTM 对合并就得风速子序列进行预测。最后，将各子模型的预测结果进行组合，得到最终的预测结果。在本研究中，将所提模型应用于四个月风速数据集，并建立了不同的比较模型。实验结果表明，CNN-BILSTM 深度学习模型可以更有效地进行预测。

四、按风速预测机理分

根据预测机理的不同，风速预测模型可以分为单一预测模型以及组合预测模型。

(1)单一预测模型

单一预测模型是一种直接利用风速数据进行预测的方法，这种方法通常依赖于单一的算法或模型结构，如统计模型，物理模型以人工智能算法等。Jiang 等人^[37] 为了提高预测精度，主要从两个方面进行，一方面提出一种采用 Boosting 算法提高模型的预测能力，另一方面计算所提方法的现有误差边界。

以中国江苏省东海岸一年的真实数据进行测试, 试验结果表明, 所提方法实现了更准确的预测。Wang 等人^[38]提出了一种用于 NWP 风速的序列传递校正算法 (STCA), 除了时间 $t + 1$ 的 NWP 风速外, 在时间 t 测得的风速也作为输入变量引入风速校正模型, 并确定了校正输入和输出之间的映射关系。通过应用 STCA, 建立了 5 个常用的模型来纠正 NWP 风速误差。该研究以华北和华南两个风电场的实际运行数据为例, 所提出的 NWP 风速校正算法在短期风速预测上具有一定的精度。然而, 风速受多种复杂因素影响, 单一模型预测精度有限。为了提高预测效果, 研究人员通常会结合多种算法和改进措施以提高模型的预测精度。

(2)组合预测模型

目前组合预测模型可以分为三类: 数据预处理策略组合模型、参数优化策略组合模型、模型改进策略组合模型。

数据预处理策略组合模型是通过数据分解技术将原始风速时间序列分解为多个具有不同特征的子序列, 从而更好地掌握风速变化的内在规律, 降低风速预测的难度, 提高风速预测的准确性。在风速预测中, 常用的数据分解算法包括经验模态分解(EMD)、变分模态分解(VMD)和小波变换(WT)等。文献[39] 将 EMD 和埃尔曼神经网络 (ENN) 的结合用于预测风速。首先, 原始风速数据集被 EMD 分解为本征模态函数 (IMF) 的集合和一个残差。其次, 使用相应的 ENN 预测每个子序列。最后, 计算各个子序列预测值之和得到最终的预测结果。目前, EMD 已经行了多次改进, 例多元经验模态分解^[40](MEMD), 集成经验模态分解^[41](EEMD), 快速多元经验模态分解^[42](FMEMD), 自适应噪声完备集成经验模态分解^[43](CEEMDAN)。文献[44] 提出 EEMD 和改进的和谐搜索算法 (IHarmony)。首先, 通过 EEMD 将初始风速数据集分解为几个集成经验模态函数, 然后利用快速傅里叶变换对每个子模态 IMF 进行特征提取, 结合 IHarmony 算法确定决策树数量和分裂特征数量。最后, 利用随机森林模型预测每个 IMF 进行预测。通过辽宁省朝阳区 2015 年 4 月 27 日至 2015 年 5 月 22 日风速数据进行验证, EEMD-RF-IHS 模型能够有效预测短期风速。文献[45] 建立了 CEEMDAN-SE-Improved PIO-GRNN 风速预测模型。首先, 利用 CEEMDAN 将原始风速序列分解为几个不同样本熵的子序列。然后, 通过样本熵 (SE) 将相似

的子序列组合成一个新的序列。之后,使用改进的鸽子启发算法(PIO)优化广义回归神经网络(GRNN)模型来预测每个新序列。结果表明,所提模型在短期风速预测的精度方面取得了很大的进步。

参数优化策略组合模型是借助优化算法对预测模型的超参数进行寻优,提高风速预测的准确性。常见的优化算法有遗传算法^[46]、鲸鱼优化算法(WOA)^[47]、粒子群算法(PSO)^[48]、模拟退火算法^[49]、最大期望(EM)算法^[50]。文献[51]采用模糊 C-means (FCM)对数值天气预报(NWP)聚类归一后,采用极限学习机(ELM)对新的数据进行预测,为了提高 ELM 的预测精度,采用 WOA 对预测模型的输入权重和隐藏层阈值进行优化。同时利用高斯混合模型(GMM)计算不同时间尺度下的风速预测置信区间。文献[52]使用模糊 C-means (FCM)将历史气象数据聚类到,并采用变分模态分解(VMD)将原始信号划分为多频分量,鲸鱼优化算法(WOA)对 VMD 的参数进行外部优化,以确定 VMD 分解的最佳层数。在改进的麻雀搜索算法(ISSA)的帮助下,提高 LSSVM 预测精度,来自中国西北地区的实际数据进行了测试,验证了模型的有效性。文献[53]将头脑风暴优化(BSO)和布谷鸟搜索(CS)算法相结合,对互相关(CC)分析和支持向量回归(SVR)模型的参数进行寻优,并采用交叉验证技术避免模型过拟合。以中国风电场收集的风速数据进短期风速预测,实验结果表明,该算法可以获得较高的精度和较强的稳定性。

模型改进策略组合模型主要在模型的结构上进行改造。文献[54]将一种新的自注意力时间卷积网络(SATCN)与长短期记忆(LSTM)相结合,以预测未来风速。采用 SATCN 以更多地关注贡献更大的特征。在 SATCN 之后使用 LSTM 来进一步构建特征和输出之间的连接,以预测未来的风速。文献[55]通过使用改进的损耗收缩 Adam 优化器(LsAdam)加速长短期记忆(LSTM)网络。使用过去的损失变化信息迭代调整学习率,使得 LSTM 能够以更少的训练周期获得更高的预测精度。文献[56]结合了雾凇优化算法(RIME)、变分模态分解(VMD)、多头自注意力(MSA)机制和长短期记忆(LSTM)进行风速预测。首先利用 RIME 优化 VMD 模态数和惩罚参数。然后构建基于 MSA 机制构建 LSTM 短期风速预测模型,使用 ReLU 函数作为神经网络的激活函数。结果表明,所提出的模型误差更小,预测准确性更高。文献[57]提出了基于遗忘门(EFFG)的

超短期风电预测误差跟随 LSTM 模型，所提出的模型利用预测值与实际值的差值更新遗忘门的输出，从而减少前一时刻的预测误差对此时风速预测精度的影响，提高风速预测精度。

与单一模型相比，这种组合预测的方法能够更好地捕捉风速变化的复杂性和不确定性，为风速预测提供了更为精确的结果

1.3 研究现状分析

通过国内外众多学者们在风速预测领域上的不懈研究，风速预测的预测方法也越来越多。例如物理模型，统计模型，机器学习模型以及深度学习模型。现阶段，风速预测的主流方法大多采用深度学习模型，例如 CNN、RNN、LSTM 以及 GRU 等神经网络应用于风速预测的研究中。这些新颖预测模型的出现为风速领域研究的不断深入提供了理论依据，专家学者们以这些神经网络模型为基础，不断地进行优化和改进，来提高风速预测的精度和稳定性。

本文以深度学习算法为主，采用组合模型“分解-优化-预测-重构”的策略对风速进行预测。目前，对于风速的分解，大都采用 VMD、EMD、EEMD 等分解算法对风速进行一次分解。该分解方法虽然能将风随机性较强的风速数据分解为较为平缓的子序列，但是，某些子序列仍然包含着复杂的信息，这些子序列会降低风速预测的精度。因此，将采用二次分解的策略，对复杂的子序列进行二次分解，提高预测精度。本文采用的 BILSTM 神经网络作为预测模型。然而，传统的 BILSTM 神经网络模型对于局部特征的提取和重点特征的关注略有不足。因此，采用 CNN 对局部特征提取，attention 机制关注重点特征，以此来提高 BILSTM 预测模型的精度和稳定性。深度学习模型的参数包括学习率、丢弃率、隐含层神经元个数等，现阶段对于这些参数的选取大多依据经验设置。本文将采用 QPSO，对这些参数进行迭代寻优。然而，本文构建的预测模型较为复杂，传统的 QPSO 容易陷入局部最优。因此，本文将会对 QPSO 进行改进，避免 QPSO 陷入局部最优，提高风速预测模型精度。

1.4 研究内容及章节安排

本文主要围绕风速预测建模进行研究，构建基于二次分解和 IQPSO-BILSTM 的风速组合预测模型，旨在构建一种新型短期风速组合预测模型，以

提高风速预测精度。本文主要研究内容如下：

(1)风速二次分解策略：CEEMDAN 方法在对风速数据进行分解时，会产生样本熵较大的模态分量。针对这一问题，本文采用二次分解策略进行风速分解。对 CEEMDAN 分解后产生的排列熵大的模态分量，采用 VMD 进行二次分解，进一步细化模态分量，从而得到更加准确的特征，确保风速数据的特征提取更加有效。

(2)基于 BILSTM 模型的风速预测：首先，采用 CNN 从分解的子序列中提取局部特征，并降维构成新的时间序列。然后，通过 BILSTM 同时处理正向和反向的时间序列，对数据的时序特征进行捕捉。最后，采用 Attention 机制，对不同的时序特征赋予不同的权重，提升模型对于重点信息的关注。

(3)QPSO 的改进：QPSO 在求解复杂的多峰问题时，搜索精度较低，容易陷入局部最优。为了解决这些问题，首先利用 Tent 映射对初始种群进行初始化，以提高种群的遍历性；然后，根据个体适应度值，自适应调整收缩-膨胀系数，进一步提升算法的全局搜索能力和局部收敛能力；最后，采用柯西变异机制帮助种群跳出局部最优，增强了粒子搜索范围，提高全局搜索能力。

本文具体章节安排如下：

第一章 绪论：本章节详细介绍了风速预测研究的背景与意义，分析国内外有关风速预测的研究现状。对研究现状进行分析，构建基于二次分解和 IQPSO-BILSTM 的风速组合预测模型，并阐述全文的结构和主要内容。

第二章 改进量子粒子群算法：本章节对 QPSO 进行改进。首先，采用了 Tent 映射对初始种群进行优化。然后，根据目标函数的适应值来更新收缩-膨胀系数。最后，本文引入了柯西变异操作，对最优粒子进行扰动。采用 Sphere、Rastrigin、Ackley 和 Schwefel 四个测试函数对 IQPSO 进行验证，并以粒子群算法和量子粒子群算法作为对比。

第三章 基于二次分解和 IQPSO-BILSTM 的风速组合预测模型：本章构建了一种基于二次分解和 IQPSO-BILSTM 的风速组合预测模型。采用 CEEMDAN-VMD 分解方法对原始风速数据进行二次分解。将分解后的数据划分为训练集，验证集以及测试集。训练集用于训练构建的 CNN-BILSTM-Attention 预测模型。IQPSO 通过验证集的目标函数对 CNN-BILSTM-Attention 预测模型的学习率、卷

积核个数、BILSTM 隐含层神经元数以及丢失率进行寻优。最后通过验证集对 CNN-BILSTM-Attention 预测模型进行验证，输出最终的预测结果。

第四章 实验结果分析：为了验证新建的风速预测模型准确性和稳定性，采用内蒙古某风电场历史风速数据，对基于二次分解和 IQPSO-BILSTM 的风速组合预测模型进行测试，并与 IQPSO-CNN、IQPSO-LSTM、IQPSO-BILSTM 等预测模型进行比较。采用统计误差统计 R^2 、RMSE、MAPE 以及方差作为评价指标，对模型的准确性和稳定性进行评估。

第五章 总结与展望。总结全文的研究工作，分析创新点并指出新建风速预测组合模型存在不足，并展望今后研究的方向。

第 2 章 改进量子粒子群算法

量子行为粒子群优化算法, 通常称为量子粒子群算法(QPSO)受粒子群算法(PSO)算法中粒子寻优轨迹与量子力学理论启发, 是 PSO 的变种优化算法。与传统 PSO 算法相比, QPSO 具有控制参数少、没有粒子速度变量、结构简单且全局搜索能力更强等特点。但 QPSO 算法和其它优化算法一样存在“早熟性收敛”的问题, 尤其求解复杂问题时, 由于 QPSO 收敛速度快, 在迭代后期算法种群的多样性骤减, 使得粒子早熟性收敛, 从而获得局部最优解。因此, 为了提高 QPSO 优化性能和降低算法“早熟性收敛”可能性, 对 QPSO 算法进行改进。首先, 采用了 Tent 映射函数对初始种群初始化, 以提升 QPSO 算法初始种群在可求解空间的遍历性; 其次, 采用基于目标函数的适应值动态调整收缩-膨胀系数, 使动态变化收缩-膨胀系数更好地适应不同问题求解; 最后, 为了确保当前位置为最优解, 引入了柯西变异操作对种群的粒子扰动, 使其跳出局部最优解, 进入其它可求解空间, 增强算法的全局搜索能力。为了验证改进后的 QPSO 的有效性, 采用 Sphere、Rastrigin、Ackley 和 Schwefel 四个测试函数对 PSO、QPSO 和 IQPSO 进行验证。

2.1 粒子群算法工作原理

粒子群算法(PSO)由 Kennedy 和 Eberhart 于 1995 年提出^[58]。其灵感来源于鸟群、鱼群等生物群体, 这些群体会通过团队合作从而寻找到食物。其中, PSO 算法中每一个粒子代表着一只鸟或一条鱼, 寻找到数量最多的食物为寻优的最终目标。

粒子群算法中粒子运动轨迹如图 2-1 所示。每个粒子具有位置和速度, 通过调整位置和速度寻找最优解的位置。在这个过程中, 受到个体认知和社会认知的影响。个体认知是指粒子自生找到的最好位置(个体最优)。社会认知代表整个群体找到的最好位置(全局最优)。通过个体最优和全局最优, 帮助粒子群不断调整自身的速度和位置, 从而寻找到最优解。

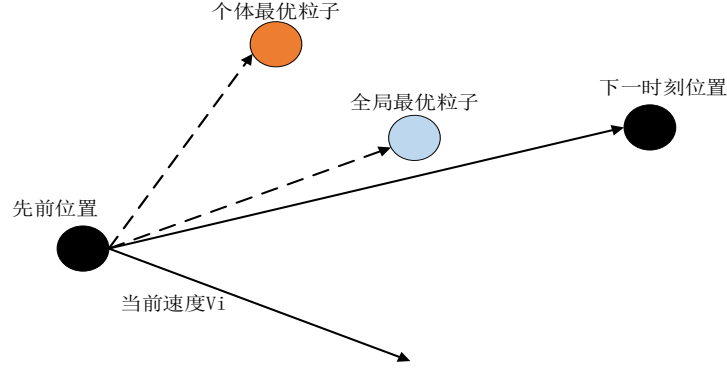


图 2-1 粒子运动轨迹示意图

在 N 维空间中有一群粒子，粒子数为 m ，在第 i 个粒子，其在 D 维空间中的位置为 $x_i=(x_{i1}, x_{i2} \cdots x_{iD})$ ，其飞行速度为 $v_i=(v_{i1}, v_{i2} \cdots v_{iD})$ 。粒子的个体最优 P_{best} 和全局最优 g_{best} 。粒子的速度更新和位置新如式(2-1)和(2-2)所示：

$$v_i(t+1) = v_i(t) + C_1 U_1 (p_{best} - x_i(t)) + C_2 U_2 (g_{best} - x_i(t)) \quad (2-1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2-2)$$

式中， C_1 为个体认知能力， C_2 为社会认知能力， U_1 ， U_2 为在区间 $[0, 1]$ 内的随机数。

多峰问题是指具有多个局部最优解的优化问题，传统 PSO 算法在处理复杂的多峰问题容易陷入局部最优，无法找到正的最优解。为了解决这个问题，Shi 和 Eberhart^[59]对传统 PSO 的速度引入了惯性因子 ω ，对粒子在空间中速度进行调整。加入惯性因子后 PSO 粒子的速度和位置更新如式(2-3)和(2-4)所示。

$$v_i(t+1) = \omega \times v_i(t) + C_1 U_1 (p_{best} - x_i(t)) + C_2 U_2 (g_{best} - x_i(t)) \quad (2-3)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2-4)$$

惯性因子使得粒子在更新速度时不仅仅依赖个体最优和全局最优，还取决于历史速度。这种改进增加了 PSO 算法全局搜索能力，避免陷入局部最优，从而改善了传统 PSO 算法在处理多峰问题的不足。加入惯性因子的 PSO 算法被称为标准 PSO 算法。标准粒子群算法的流程图如图 3-2 所示。具体寻优步骤如下：

(1)初始化 PSO 算法参数。包括学习因子 C_1 和 C_2 ，惯性权重 ω 以及粒子群的种群数目。(2)计算目标函数的适应度值。(3)将每个粒子的适应度值和其个体最优 P_{best} 和全局最优 g_{best} 进行比较，如果当前适应度值要优于 P_{best} 或 g_{best} ，则

更新其个体最优和全局最优。(4)判断是否满足终止条件，如果不满足终止条件。则更新粒子的速度和位置；如果满足终止条件，则输出优化结果。

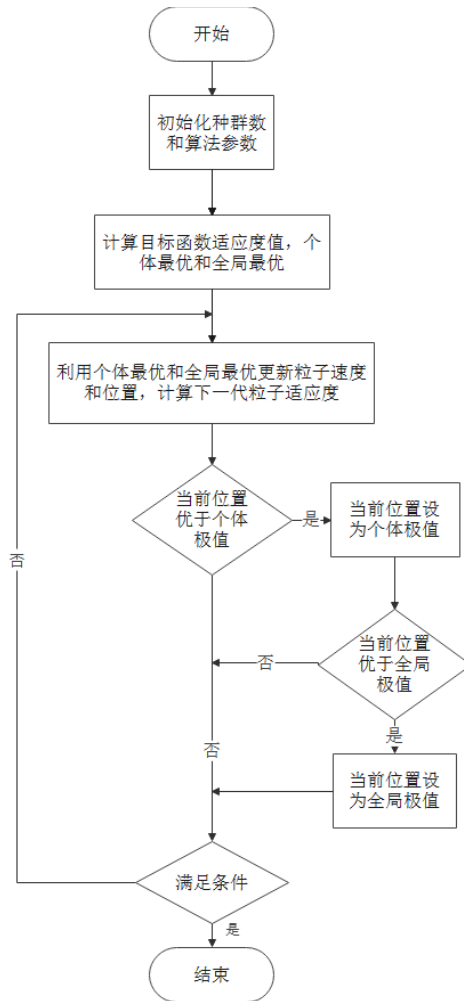


图 2-2 标准粒子群算法流程图

2.2 量子粒子群算法工作原理

在 PSO 算法中，粒子的状态是由位置和速度的改变而改变。粒子的速度并不是无限大的，随着时间的推移，速度会降低。这就意味着粒子的寻优范围会逐步降低，很容易陷入局部最优，无法保证在 1 的概率下找到最优解^[60]。为了改善这一缺陷，孙俊等人在 PSO 算法上引入量子特性提出了量子粒子群(QPSO)算法。在量子空间中，当粒子运动时，其中心位置会受到一个势场的吸引。通过这种吸引，会约束粒子的运动轨迹，确保粒子能够收敛。同时被吸引的粒子会以一定的概率逃离约束，出现在空间中的任意位置，提高了粒子的随机性，从而增加了算法全局收敛能力^[61]。

在量子空间中，粒子的位置无法确定，采用波函数 $\psi(X, t)$ 表示粒子的状态，

其中 X 表示粒子在空间中的位置，而粒子在量子空间某位置出现的概率如公式(2-5)所示：

$$|\Psi|^2 dx dy dz = Q dx dy dz \quad (2-5)$$

式中 Q 表示概率密度函数，且函数满足归一化条件，如公式(2-6)所示：

$$\int_{-\infty}^{+\infty} |\Psi|^2 dx dy dz = \int_{-\infty}^{+\infty} Q dx dy dz = 1 \quad (2-6)$$

粒子的变化状态由薛定谔方程确定，如公式(2-7)所示：

$$i\hbar \frac{\partial}{\partial t} \Psi(X, t) = \hat{H} \Psi(X, t) \quad (2-7)$$

式中 i 为虚数单位， $\hbar$ 为普朗克常量， $\hat{H}$ 为哈密顿算子，如公式(2-8)所示：

$$\hat{H} = -\frac{\hbar^2}{2m} \nabla^2 + V(x) \quad (2-8)$$

式中 m 为粒子质量， ∇^2 为拉普拉斯算符， $V(x)$ 是粒子的势场。

假设在某一粒子群体中，一定会存在某一吸引点 p ，那么该粒子在一维 Delta 势阱中的量子束缚态如公式(2-9)所示：

$$\Psi(Y) = \frac{1}{\sqrt{L}} e^{-|Y|/L} \quad (2-9)$$

式中 L 是 Delta 势阱的本征长度， $Y=X-p$ ，相应概率密度的函数 Q 如公式(2-10)所示：

$$Q(X) = |\Psi(X)|^2 = \frac{1}{L} e^{-2|Y|/L} \quad (2-10)$$

相应的概率密度分布的函数如公式(2-11)所示：

$$u = e^{-2|Y|/L} \quad (2-11)$$

式中 u 为(0, 1)上的随机数。则粒子的位置如公式(2-12)所示：

$$X = p \pm \frac{L}{2} \ln\left(\frac{1}{u}\right) \quad (2-12)$$

在多维空间中，粒子 i 在 j 维上的位置如公式(2-13)所示：

$$X_{i,j}(T+1) = p_{i,j}(T) \pm \frac{L_{i,j}(T)}{2} \ln\left(\frac{1}{u}\right) \quad (2-13)$$

式(2-13)中， L 值过大会导致粒子群发散，过小又会陷入局部最优。因此需要引入了平均最好位置 M_{best} ，如公式(2-14)所示。

$$M_{best} = \frac{1}{M} \sum_{i=1}^M P_{i-best}(t) \quad (2-14)$$

式中 P_{i-best} 为粒子 i 最优位置，粒子 i 的吸引子如公式(2-15)所示：

$$P_{ij} = \lambda \cdot P_{i-best}(t) + (1 - \lambda)g_{best}(t) \quad (2-15)$$

式中 g_{best} 为全局最优粒子。

L 的取值如公式(2-16)所示：

$$L = 2\alpha \cdot |M_{best} - x(t)| \quad (2-16)$$

相应的粒子位置如公式(2-17)所示：

$$x_{ij}(t+1) = P_{ij} + \alpha |M_{best} - x_{ij}(t)| \ln\left(\frac{1}{u}\right) \quad (2-17)$$

式中 α 为收缩-扩张因子，一般的 $\alpha < 1.781$ ，来保证算法收敛。

2.3 改进量子粒子群算法

QPSO 在处理复杂多峰问题时具有较强的全局搜索能力，但其也存在一些不足。首先，QPSO 算法采用随机初始化策略初始种群分布。这种方法虽然简单高效，但存在一定弊端。粒子随机分布如图 2-3 所示，在一个 $[0, 1]$ 的二维平面上，100 个粒子被随机初始化，观察这些粒子分布位置可以看出，一些粒子会过度集中在某些区域，而在部分区域中，一些粒子的分布则较为稀少甚至没有。这样的粒子分布不仅增加了算法的寻优时间，还降低了算法的寻优能力。

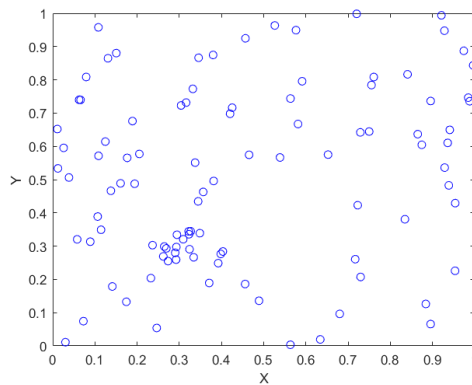


图 2-3 粒子随机分布图

为了避免 QPSO 算法在粒子初始化时会局限于某个区域，本文采用 Tent 映射来初始化 QPSO 的种群分布。Tent 映射公式(2-18)所示：

$$x_{n+1} = \begin{cases} \frac{x_n}{\tau} & 0 < x_n < \tau \\ \frac{1-x_n}{1-\tau} & \tau < x_n < 1 \end{cases} \quad (2-18)$$

式中, x_n 是序列的当前值, τ 为一个介于 0 和 1 之间的常数。

使用 Tent 映射进行粒子群初始化后, 粒子的分布如图 2-4 所示。相对于随机初始化, 采用 Tent 映射后的粒子群的在空间上的分布更为均匀。这种初始化降低了算法的寻优时间, 还提升了算法的寻优能力。

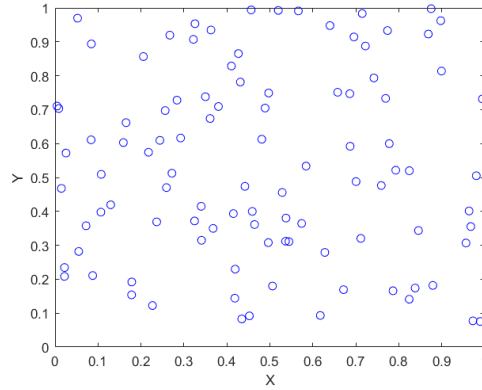


图 2-4 粒子 tent 映射分布图

其次 QPSO 中用线性减少策略来更新收缩膨胀系数, 该系数是算法中唯一需要人为设置的参数。更新方法如公式(2-19)所示:

$$\alpha = \alpha_{\min} + \frac{(\alpha_{\max} - \alpha_{\min}) \times (i_{\max} - i_n)}{i_{\max}} \quad (2-19)$$

式中 $i_{\max}$ 为最大迭代次数, i_n 为当前迭代次数。 $\alpha_{\max}$ 为收缩-膨胀系数初始值, $\alpha_{\min}$ 为收缩-膨胀系数最终值。

由公式(2-19)可知, 迭代初期 α 较大, 具有较高的全局搜索能力。在迭代末期 α 较小, 具有较强的局部搜索能力。然而, 在实际运用中, 粒子的位置是随机的, 也许粒子在迭代后期, 寻找到的最优解为局部最优, 此时的 α 值较小, 无法经行全局搜索, 找到全局最优。因此 α 的取值也需要根据粒子位置和全局最优位置来设定, 本文提出基于目标函数的适应值动态变化收缩-膨胀系数, 新的收缩-膨胀系数变化如公式(2-20)所示:

$$\alpha = \alpha_{\min} + e^{\left(1 - \frac{f(t)}{gbest(t)}\right)} (\alpha_{\max} - \alpha_{\min}) \quad (2-20)$$

式中 $f(t)$ 表示目标函数的适应度值。

由公式(2-20)可知,若粒子距离最优解较近,收缩-膨胀系数较大,扩大搜索范围,避免陷入局部最优解。若粒子距离最优解较远,收缩-膨胀系数较小,确保种群能够收敛。通过这种方式,提高了算法全局搜索能力和局部搜索能力。

最后,在迭代后期,粒子会聚集在某个最优解区域,为了进一步确认当前最优解为全局最优,通过柯西变异来产生远离原点的随机位置。柯西分布的峰值较小,但其尾部较长,这表明其能够产生更大的扰动,从而帮助算法跳出局部最优^[62]。柯西变异如公式(2-21)所示:

$$x_{i+1} = x_{best} + \text{cauchy}(0,1) \quad (2-21)$$

式中 $\text{cauchy}(0,1)$ 为标准柯西分布。

通过上述改进方法对 QPSO 算法进行改进,提高了算法的全局搜索能力。改进后的量子粒子群(IQPSO)算法的流程图如图 2-5 所示。

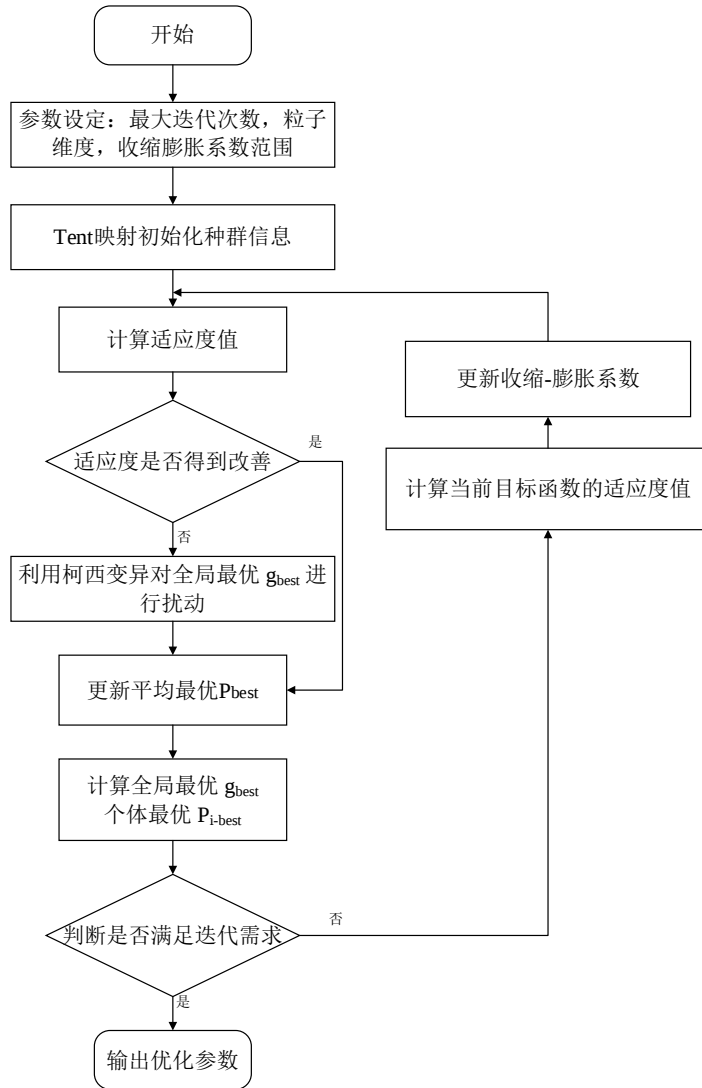


图 2-5 改进量子粒子群算法流程图

2.4 IQPSO 算法性能测试

为了验证 IQPSO 算法的寻优性能, 本文选取了 Sphere、Rastrigin、Ackley 和 Schwefel 四个典型的测试函数进行实验验证。尤其是 Rastrigin 和 Schwefel 复杂的多峰结构, 更能测试算法的搜索能力。选取 PSO 和 QPSO 作为对比算法。

表 2-1 测试函数

名称	函数表达式	Lb	Ub	fmin
Sphere	$f_1(x) = \sum_{i=1}^n x_i^2$	-100	100	0
Rastrigin	$f_2(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)] + 10n$	-5.12	5.12	0
Ackley	$f_3(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + 20 + e$	-5	5	0
Schwefel	$f_4(x) = 418.9829n - \sum_{i=1}^n x_i \sin(\sqrt{ x_i })$	-500	500	0

测试函数信息如表 2-1 所示, Lb 和 Ub 分别表示决策变量的下界和上界, $fmin$ 表示该函数的最小值。如表进行算法测试时, 在参数设置方面, 三种算法的最大迭代次数、种群规模和空间维度都保持一致。本章节在检验算法的性能时, 采用 windows10(64 位)操作系统, 测试平台采用 MATLAB2023a 软件, 为了对算法测试结果进行数据统计分析, 对每个基准测试函数独立运行 10 次, 最后统计其结果。为了准确分析 IQPSO 算法的性能, 采用以下四个指标:

平均值($mean$)计算如公式 (2-22) 所示:

$$mean = \frac{1}{m} \sum_{i=1}^m best_i \quad (2-22)$$

标准差(std)计算如公式 (2-23) 所示:

$$std = \sqrt{\frac{1}{m-1} \sum_{i=1}^m (best_i - mean)^2} \quad (2-23)$$

最小值(min)计算如公式 (2-24) 所示:

$$\min = \min\{best_1, best_2, \dots, best_n\} \quad (2-24)$$

最大值(max)计算如公式 (2-25) 所示:

$$\max = \max\{best_1, best_2, \dots, best_n\} \quad (2-25)$$

三种算法对 Sphere、Rastrigin、Ackley 以及 Schwefel 的四个测试函数的仿真实验统计结果如表 2-2 所示。表中为三种算法各运行 10 次所得到的平均值

(*mean*)、标准差(*std*)、最小值(*min*)和最大值(*max*)

表 2-2 三种算法的寻优性能对比结果

测试函数	评价指标	PSO	QPSO	IQPSO
Sphere	标准差	4.9813E+02	6.3355E+01	9.1958E+00
	平均值	1.7138E+03	3.7704E+02	5.0710E+01
	最大值	9.1577E+03	3.5695E+03	4.4187E+02
	最小值	5.3023E-08	1.1806E-20	1.5245E-35
Rastrigin	标准差	3.7085E-01	2.6698E-01	8.7972E-02
	平均值	1.5432E+00	6.1666E-01	3.9599E-01
	最大值	9.5264E+00	2.6586E+00	3.1496E+00
	最小值	5.0200E-07	1.0200E-10	8.1700E-14
Ackley	标准差	1.7051E+00	1.2262E+00	2.0115E-01
	平均值	3.5700E+00	3.4987E+00	9.6339E-01
	最大值	1.4808E+01	1.4636E+01	6.7074E+00
	最小值	1.2782E-04	5.7500E-10	4.4400E-16
Schwefel	标准差	3.27E+00	2.58E+00	2.25E+00
	平均值	5.96E+00	5.60E+00	5.02E+00
	最大值	4.60E+01	4.13E+01	2.74E+01
	最小值	3.22E-02	1.14E-02	2.55E-05

为了更加直观地对比 IQPSO 算法和 PSO 以及 QPSO 的收敛速度和计算精度。图 2-6、图 2-7、图 2-8 和图 2-9 为三种算法应用上述 4 种不同测试函数所得结果的函数值对比收敛曲线图。

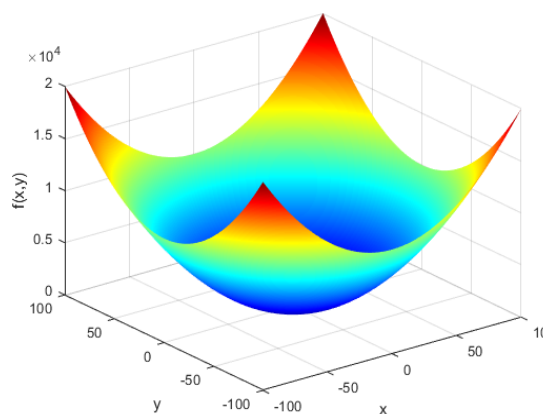
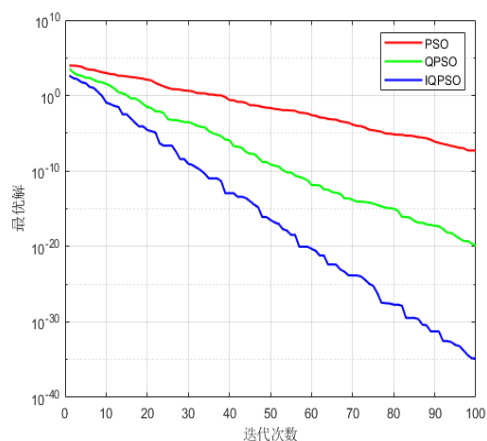


图 2-6 (a) Sphere 三维图



(b) Sphere 收敛曲线图

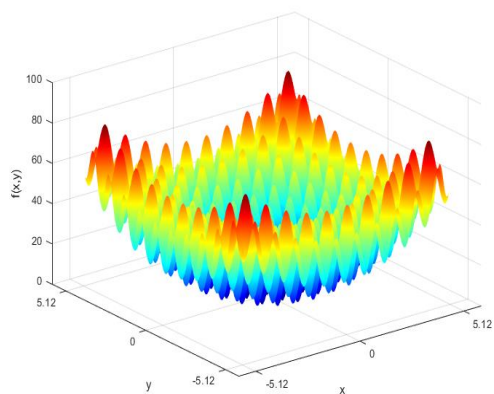
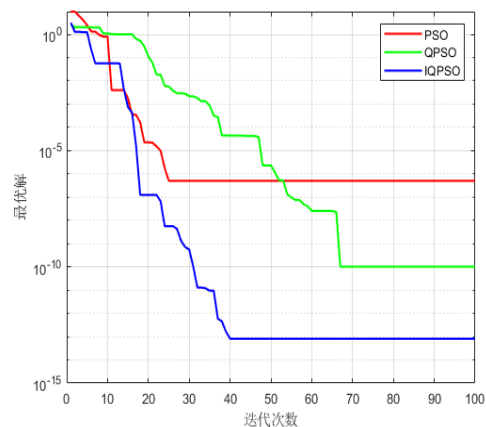


图 2-7(a) Rastrigin 三维图



(b) Rastrigin 收敛曲线图

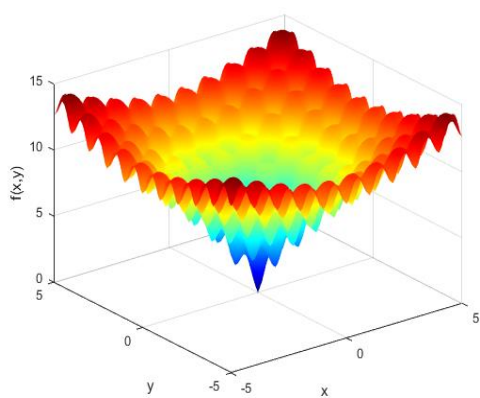
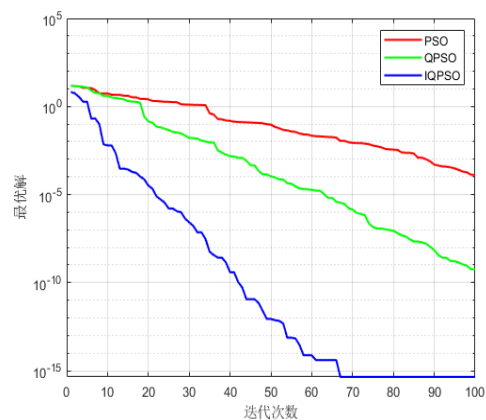


图 2-8 (a) Ackley 三维图



(b) Ackley 收敛曲线图

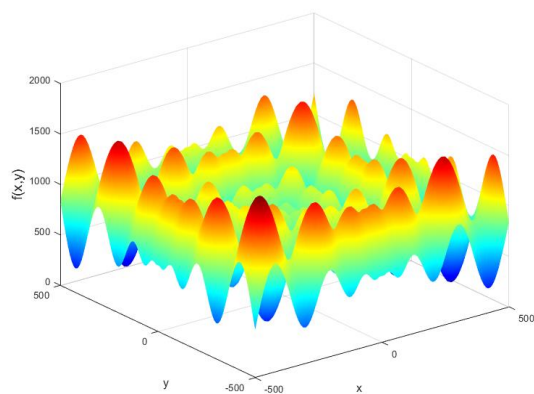
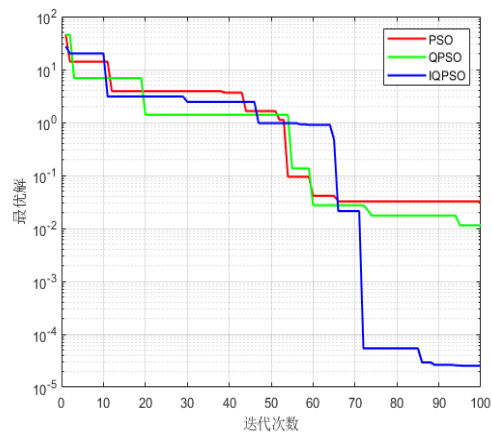


图 2-9 (a) Schwefel 三维图



(b) Schwefel 收敛曲线

如图 2-6 (a)所示, Sphere 函数仅仅只有一个全局最小值, 函数图像是一个简单抛物面, 适合简单验证算法的收敛精度。IQPSO 在 Sphere 函数寻优得到的最小值仅为 $1.5245\text{E}-35$, 远低于 PSO 的最小值 $5.3023\text{E}-08$ 和 QPSO 的最小值 $1.1806\text{E}-20$ 。而 Rastrigin 测试函数在空间中具有多个局部极小值, 且每个局部极

小值逐步接近最小值。如图 2-7 (a)所示, 该函数的图像类似于蜂窝, 具有大量的波峰和波谷。如图 2-8 (a)所示, Ackley 函数具有一个全局最小值和多个局部极小值, 图像类似于碗状。Schwefel 函数和 Rastrigin、Ackley 类似, 具有一个全局最小和多个局部最小, 且该函数的全局最小位于搜索空间边缘。如图 2-9 (a)所示, Schwefel 函数图像呈现波浪结构。这些具有多峰结构的测试函数可以较好的考验优化算法的全局收敛能力。例如 IQPSO 算法在 Schwefel、Rastrigin 以及 Ackley 测试函数中的最小值分别为 $8.17\text{E-}14$ 、 $4.44\text{E-}16$ 、 $2.55\text{E-}05$ 。远低于 QPSO 算法和 PSO 算法在同样测试函数中的最小值。这体现了 IQPSO 在处理复杂问题时具有较高的全局搜索能力和搜索精度。

2.1 本章小结

本章首先介绍了 PSO 算法以及 QPSO 算法工作原理; 其次, 针对 QPSO 早熟性收敛的问题, 对 QPSO 进行改进: (1)采用 Tent 映射函数初始化 QPSO 初始种群, tent 映射函数使初始种群能够在搜索空间内更加均匀分布和遍历性, 避免了粒子过度聚集的可能性; (2)采用基于目标函数的适应值动态变化收缩-膨胀系数, 使得 QPSO 既能在全局搜索时具有较强的探索能力, 又能在局部搜索时具有更高的精度和收敛速度; (3)通过柯西变异策略来扰动当前粒子位置, 使其跳出局部最优解, 进入其它可求解空间, 增强算法的全局搜索能力。最后, 通过 Sphere、Rastrigin、Ackley 和 Schwefel 四个测试函数对 PSO、QPSO 以及 IQPSO 进行测试。实验结果表明, 与传统 PSO 和 QPSO 相比, IQPSO 具有更优的全局搜索能力和更快的收敛速度。

第 3 章 构建基于二次分解和 IQPSO-BILSTM 的风速组合预测模型

本章构建了一种基于二次分解和 IQPSO-BILSTM 的风速组合预测模型，旨在提高风速预测的准确性和稳定性。首先，采用 CEEMDAN 对原始风速数据进行分解，对排列熵值最大的分量进行 VMD 二次分解，降低数据的复杂度。其次，在 BILSTM 模型中引入了 CNN 和 Attention，构建 CNN-BILSTM-Attention 预测模型，提升 BILSTM 对于局部特征的提取能力以及对重要特征的关注度。最后，采用 IQPSO 对 CNN-BILSTM-Attention 模型的超参数进行寻优，从而提高模型的准确性和稳定性。

3.1 风速分解相关理论

3.1.1 经验模态分解

EMD 是黄镌^[63]等人于 1998 年提出的信号分析方法。EMD 分解可以将一系列复杂的风速数据自动分解为一定数量平稳的子序列，这些子序列可以体现风速数据变化规律。通过求取每一次分解数据的极值点以及包络线来确定子序列数量，避免了传统方法手动确定分解层数^[64]。EMD 分解流程如图 3-1 所示，计算步骤如下：

步骤 1：识别信号 $x(t)$ 的极值大值点 $h_{max}(t)$ 和极小值点 $h_{min}(t)$ ，并通过插值方法分别构造上包络线 $m_{max}(t)$ 和下包络线 $m_{min}(t)$ 。

步骤 2：计算上下包络线的平均值 $m(t)$ 如公式(3-1)所示：

$$m(t) = \frac{(m_{max}(t) + m_{min}(t))}{2} \quad (3-1)$$

步骤 3：将信号 $x(t)$ 减去这一平均值 $m(t)$ ，从而得到一个初步的振荡分量 $h(t)$ 。如若平均值 $m(t)=0$ ，则输出 $h(t)$ ；如果平均值 $m(t)$ 不为 0，则循环上述步骤。如公式(3-2)所示：

$$h(t) = x(t) - m(t) \quad (3-2)$$

步骤 4：经过多次循环，信号被分解为若干模态分量 (IMF) 和一个残余分量 $r(t)$ 。如公式(3-3)所示：

$$r(t) = x(t) - h(t) \quad (3-3)$$

步骤 5: 判断残余分量 $r(t)$ 是否为常量或者为单调函数, 如果满足条件, 则输出该残余分量 $r(t)$, 如果不满足条件, 则残余分量作为新的分解信号 $x(t)$, 再次进行分解。

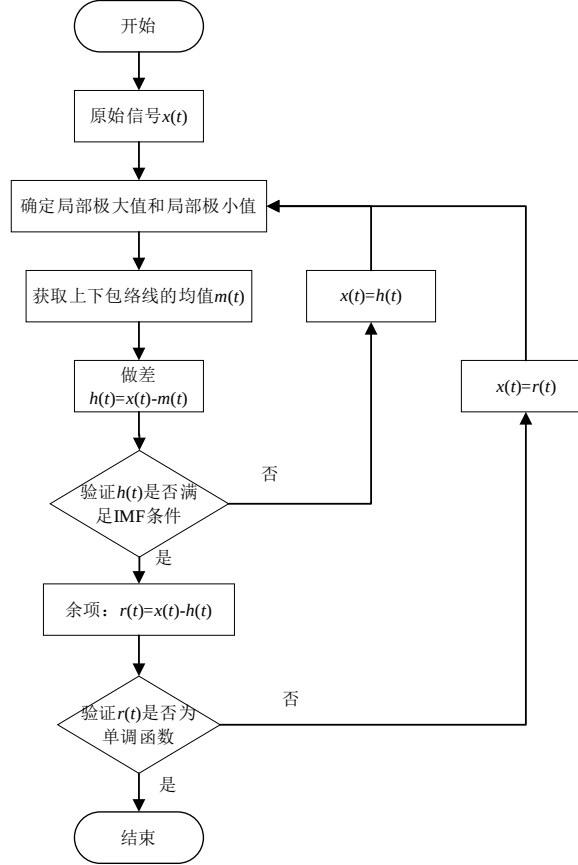


图 3-1 EMD 分解流程图

然而, 在 EMD 分解过程中, 某些 IMF 的具有较宽的频率带, 可能包含来自多个频率的信号。为了克服 EMD 分解产生的模态混叠现象, 研究者提出了多种 EMD 的改进算法, 其中 CEEMDAN 是一种较为常用且有效的方法^[65]。

3.1.2 自适应噪声完备集成经验模态分解

CEEMDAN 分解是一种基于 EMD 分解发展而来的方法, 通过正负高斯分布减轻解决 EMD 分解方法存在的模态混叠问题^[66]。同时 CEEMDAN 在分解过程中添加 EMD 分解后的 IMF 分量, 以提高分解的准确性和稳定性。已知待分解的信号为 x , 首先引入高斯白噪声 $w(t)$ 并按照公式(3-4)进行分解。

$$x^{(i)} = x + eI_1(w^{(i)}) \quad (3-4)$$

式中 $E_I(\cdot)$ 为 EMD 产生的第一阶模态分量, e_I 为第一阶时加入的信噪比, $w(i)$ 为白噪声。计算第一阶段的残差 r_1 如公式(3-5)所示:

$$r_1 = \frac{1}{n} \sum_{i=1}^n M\{x + e_1 \cdot E_1[w^{(i)}]\} \quad (3-5)$$

式中 $M\{\cdot\}$ 为所用信号局部均值的操作符。

将原信号减去第一阶段的残差，得到第一阶段的模态分量，如公式(3-6)所示：

$$imf_1 = x - r_1 \quad (3-6)$$

在残差 r_1 中添加噪声，得到第二阶段的残差和模态分量，如公式(3-7)所示：

$$r_2 = \frac{1}{n} \sum_{i=1}^n M\{r_1 + e_2 \cdot E_2[w^{(i)}]\} \quad (3-7)$$

$$imf_2 = r_2 - r_1 \quad (3-8)$$

以此类推，求得第 k 个阶段的残差和模态分量，如公式(3-9)和(3-10)所示：

$$r_k = \frac{1}{n} \sum_{i=1}^n M\{r_{k-1} + e_k \cdot E_k[w^{(i)}]\} \quad (3-9)$$

$$imf_k = r_{k-1} - r_k \quad (3-10)$$

CEEMDAN 分解后的某些模态成分可能呈现出较大的排列熵值，这意味着这些模态成分的复杂度较高，导致模型的预测精度较低。为了解决这一问题，本文采取 VMD 对 CEEMDAN 分解后产生的具有较大排列熵的模态分量进行进一步的分解。

3.1.3 变分模态分解

变分模态分解(VMD)在 2014 年被 Dragomiretskiy 提出的一种完全非递归的数据分解方法^[67]。将一组信号分解为 K 个模态分量(IMF)，每个 IMF 都具有中心频率和带宽。而 VMD 分解便是通过迭代寻优的方法来确定每个 IMF 的中心频率，使 K 个 IMF 的带宽和最小。若 K 值较小，意味着每个 IMF 的带宽较大，会丢失一些重要信息。若 K 值较大，多个 IMF 的中心频率的大小相近，会产生更多噪声。因此，需要根据中心频率来确定 K 值大小。VMD 分解信号的如公式(3-11)所示：

$$\begin{aligned} \min_{\{u_k\}, \{\omega_k\}} & \left\{ \sum_k \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\} \\ \text{s.t.} & \sum_{k=1}^K u_k = f \end{aligned} \quad (3-11)$$

式中 f 为分解信号， k 为分解个数， $\{u_k\}$ 为模态分量， $\{\omega_k\}$ 为中心频率。为

求模型最优解，引入二次惩罚项以及拉格朗日乘数，如公式(3-12)所示：

$$L(\{u_k\}, \{\omega_k\}, \lambda) = \alpha \sum_k \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 + \left\| f(t) - \sum_k u_k(t) \right\|_2^2 + \left\langle \lambda(t), f(t) - \sum_k u_k(t) \right\rangle \quad (3-12)$$

式中 λ 为拉格朗日乘子， α 为惩罚参数，利用交替方向乘子法(AMMD)对公式(3-12)进行迭代求解，得到更新公式(3-13)：

$$\begin{cases} \hat{u}_k^{n+1}(\omega) = \frac{\hat{f}(\omega) - \sum_{i \neq k} \hat{u}_i(\omega) + \hat{\lambda}(\omega) / 2}{1 + 2\alpha(\omega - \omega_k)^2} \\ \omega_k^{n+1} = \frac{\int_0^\infty \omega |\hat{u}_k^{n+1}(\omega)|^2 d\omega}{\int_0^\infty |\hat{u}_k^{n+1}(\omega)|^2 d\omega} \\ \hat{\lambda}^{n+1}(\omega) = \hat{\lambda}^n(\omega) + \gamma \left(\hat{f}(\omega) - \sum_k \hat{u}_k^{n+1}(\omega) \right) \end{cases} \quad (3-13)$$

式中 n 为迭代次数， ω 为频率值， γ 为噪声容忍度。当满足公式(3-14)时，迭代终止。

$$\frac{\|\hat{u}_k^{n+1} - \hat{u}_k^n\|_2^2}{\|\hat{u}_k^n\|_2^2} < \varepsilon \quad (3-14)$$

式中 ε 为收敛容差。

对于 CEEMDAN 分解后的排列熵较大的子序列，采用 VMD 进行二次分解，能有效的降低信号的复杂性，提高最终的预测精度。

3.1.4 排列熵

排列熵(PE)是一种用于衡量时间序列复杂度的非线性方法。排列熵的值通常介于 0 和 1 之间，值越接近 0，表示时间序列的规律性越强，有利于提高风速预测模型的精度；而值越接近 1，表示时间序列的模式越复杂，会降低模型预测的准确度^[68]。排列熵算法的具体步骤如下。

步骤 1 分解后的模态分量经过相空间重构可得到新的矩阵 Y ，如公式(3-15)所示：

$$Y = \begin{bmatrix} x(1) & x(1+t) & \cdots & x[1+(m-1)t] \\ x(2) & x(2+t) & \cdots & x[2+(m-1)t] \\ x(j) & x(j+t) & \cdots & x[j+(m-1)t] \\ \vdots & \vdots & \cdots & \vdots \\ x(K) & x(K+t) & \cdots & x[K+(m-1)t] \end{bmatrix} \quad (3-15)$$

式中 t 为延迟时间, m 为嵌入维数, $K=n-(m-1)t$ 。

步骤 2 将矩阵 Y 中每一个重构分量按照升序重新排列,得到符号序列如公式(3-16)所示:

$$r(g)=\{j_1, j_2, \cdots j_m\} \quad (3-16)$$

式中 $g=1, 2, 3 \dots n$, 且 $n < m!$, 所有排列模式共有 $m!$ 种可能

步骤 3 统计所有符号序列的概率, 如公式(3-17)所示:

$$P(j)=\{P_1, P_2, \cdots P_k\} \quad (3-17)$$

式中 P 为每种符号出现的概率。

步骤 4 计算排列熵, 如公式(3-18)所示:

$$H_{pe} = -\sum_{j=1}^k P_j \ln P_j \quad (3-18)$$

步骤 5 为了使排列熵与嵌入维度无关, 可以进行标准化处理如公式(3-19)所示:

$$h_{pe} = \frac{H_{pe}}{\ln m!} \quad (3-19)$$

3.2 卷积神经网络

卷积神经网络(CNN)是一种用于特征提取和图像处理的深度学习模型, 它的设计灵感来源于生物视觉系统, 尤其是视觉中的神经元结构^[69]。如图 3-2 所示, CNN 的主要结构包括输入层、卷积层、池化层、全连接层以及输出层, 这些层通过协同工作, 能够有效地提取输入数据中的特征, 并最终完成分类、检测等任务。

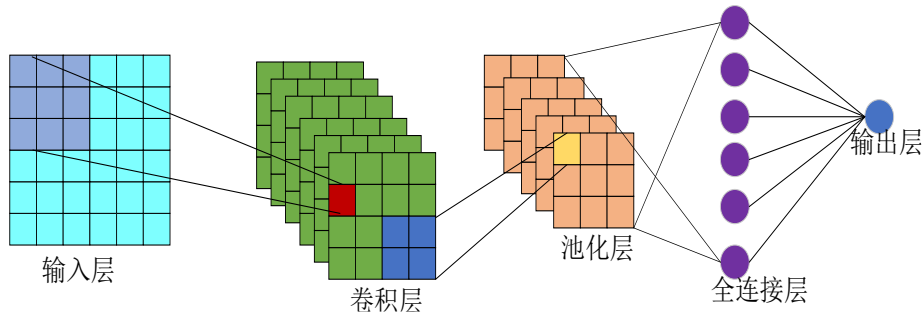


图 3-2 卷积神经网络

在 CNN 结构中第一层就是输入层, 用于输入需要处理地数据。该网络不仅能够接收一维的数据, 还可以接收二维乃至多维的数组。输入层本身并不进行

任何计算，输入层的作用是将这些数据传递给后续的卷积层进行处理。

卷积层是 CNN 的核心部分，它通过卷积核对输入层接收的数据进行局部感知和特征提取。风速数据为一维时间序列，其卷积层工作原理如图 3-3 所示

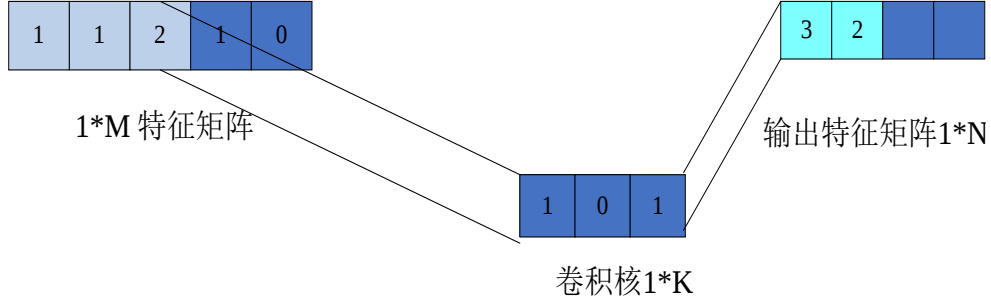


图 3-3 1D-CNN 工作原理

一维卷积核会从原始数据最左边开始滑动，每次滑动按照一定的步幅，计算每个位置上的卷积结果，这些结果为输入序列某个位置上的局部特征。局部特征提取后会通过激活函数对其进行非线性变化。卷积层特征提取如公式(3-20)所示：

$$feature = \varphi(w^n \otimes x + b^n) \quad (3-20)$$

式中， $\varphi(\cdot)$ 是激活函数， w^n 、 x 和 b^n 分别表示内核中加权因子的向量、核数、输入序列的向量和偏差向量， $\otimes$ 是卷积运算。

激活函数是用来提高 CNN 卷积层非线性特性，使其能够提取复杂的特征。如果缺少激活函数，输入与输出的关系将会是线性相关。常见的激活函数有 Sigmoid 函数、双曲正切函数(Tanh)以及线性整流函数(Relu)。

(1) Sigmoid 函数

Sigmoid 激活函数的数学表达式如公式(3-21)所示，导数表达式如公式(3-22)所示，原函数图像和导数图像如图 3-4 所示。

$$f(x) = \text{Sigmoid}(x) = \frac{1}{1+e^x} \quad (3-21)$$

$$f'(x) = \left(\frac{1}{1+e^x}\right)' = f(x)[1-f(x)] \quad (3-22)$$

式中 x 为 Sigmoid 为激活函数的输入。

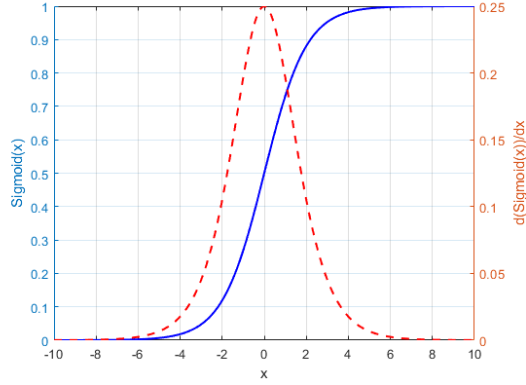


图 3-4 Sigmoid 函数及其导数示意图

Sigmoid 将输入值 x 映射到 $[0,1]$ 的范围内。Sigmoid 原函数图如图 3-4 蓝色曲线所示，是一条从 0 到 1 的平滑 S 形曲线。当 x 趋近于正无穷时， $f(x)$ 趋近于 1； x 趋近于负无穷时， $f(x)$ 趋近于 0。Sigmoid 导数图像如图 3-4 红色曲线所示，是一条中间突起的曲线。当 x 接近 0 时， $f(x)$ 导数接近 0.25，当 x 远离 0 时， $f(x)$ 导数趋近于 0。

(2) Tanh 函数

Tanh 激活函数的数学表达式如公式(3-23)所示，导数表达式如公式(3-24)所示，原函数图像和导数图像如图 3-5 所示。

$$g(x) = \text{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3-23)$$

$$g'(x) = 1 - (g(x))^2 \quad (3-24)$$

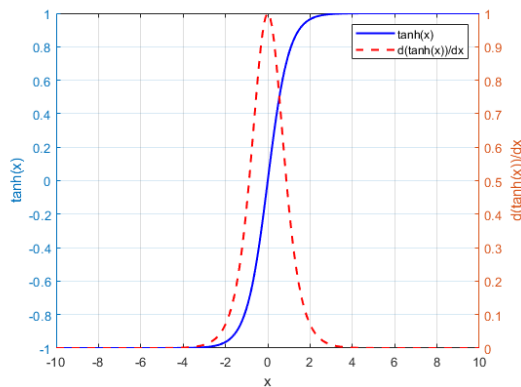


图 3-5 Tanh 函数及其导数示意图

Tanh 函数将 Sigmoid 映射范围的 $[0,1]$ 改成了 $[-1,1]$ 。Tanh 原函数图如图 3-4 蓝色曲线所示，是一条从 -1 到 1 的平滑 S 形曲线。当 x 趋近于正无穷时， $g(x)$ 趋近于 1； x 趋近于负无穷时， $g(x)$ 趋近于 -1。Tanh 导数图像如图 3-4 红色曲线所

示，是一条中间突起的曲线。当 x 接近 0 时， $g(x)$ 导数接近 1，当 x 远离 0 时， $g(x)$ 导数趋近于 0。

(3) Relu 函数

Relu 激活函数的数学表达式如公式(3-26)所示，导数表达式如公式(3-27)所示，原函数图像和导数图像如图 3-6 所示。

$$h(x) = \text{Relu}(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (3-25)$$

$$h'(x) = \begin{cases} 1 & x > 0 \\ \text{undefined} & x = 0 \\ 0 & x < 0 \end{cases} \quad (3-26)$$

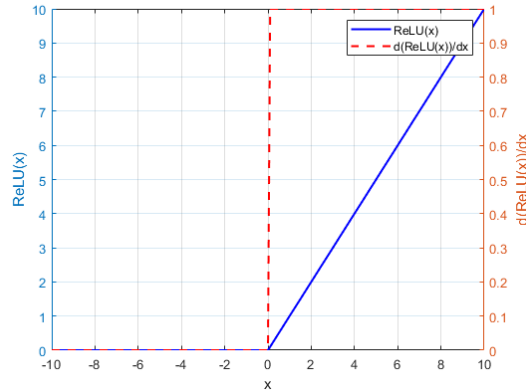


图 3-6 ReLU 函数及其导数示意图

ReLU 原函数图如图 3-6 蓝色曲线所示，是一条在负半轴为 0、正半轴为斜率为 1 的直线。当 x 趋近于正无穷时， $h(x)$ 趋近于正无穷； x 趋近于负无穷时， $h(x)$ 一直为 0。ReLU 导数图像如图 3-6 红色曲线所示，是一条在负半轴为 0、正半轴为 1 的水平线。当 x 趋近于正无穷时， $h(x)$ 导数一直为 1； x 趋近于负无穷时， $h(x)$ 导数一直为 0。

Sigmoid 函数的输出范围是 0 到 1，十分适用于概率问题。但 Sigmoid 函数在 x 过大或过小时，梯度会变得非常小甚至会消失，使得神经网络的训练变得非常缓慢。与 Sigmoid 函数相比，Tanh 函数在梯度消失的问题上有所缓解。但 Tanh 函数的计算复杂度较高，导致神经网络训练较为缓慢。ReLU 在 x 大于 0 时，梯度恒为 1，这有效地缓解了梯度消失问题，加快了神经网络的训练速度。目前，ReLU 在深度学习中非常流行。

池化层主要用于数据降维和特征筛选。在风速预测中，池化操作可以有效地减少特征图的维度，保留最重要的特征，同时降低计算复杂度，防止过拟合。

常用的池化操作最大池化和平均池化^[70]。

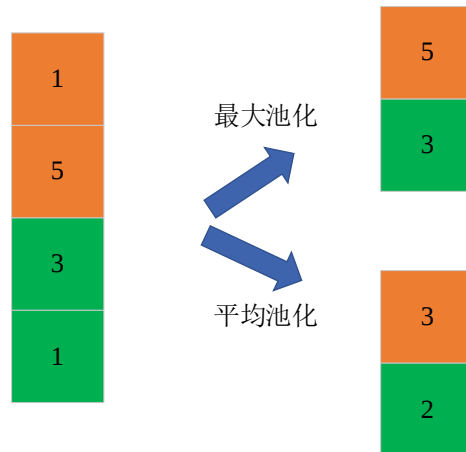


图 3-7 最大池化和平均池化运算过程

最大池化和平均池化的运算过程如图 3-7 所示。最大池化通过保留局部窗口内的最大值，增强了模型对显著特征的捕捉能力，通过最大池化操作，保留的数据分别为 5 和 3；而平均池化则计算窗口内的平均值，减少噪声影响，通过平均操作，得到的数据分别为 3 和 2。通过池化层的处理，可以从数据中逐步提取抽象且具有代表性的特征，忽略风速数据中的微小波动，从而更关注整体的变化趋势，有助于提高风速预测的精度。

经过卷积和池化操作后，会进入全连接层。全连接层主要作用是将从卷积和池化层中提取的特征映射到最终的输出空间。在这一层中，使用激活函数来引入非线性，使得神经网络能够拟合更复杂的函数。同时会使用一些方法(如 Dropout)来防止过拟合。在处理风速数据时，全连接层可以将提取的风速特征与风速预测值进行关联。输出层生成最终的预测结果。在风速预测任务，输出层通常直接输出一个连续值，表示预测的风速。

3.3 双向长短期记忆网络

3.3.1 循环神经网络原理

循环神经网络(RNN)是一类具有循环结构的人工神经网络，用于处理序列数据。与传统的前馈神经网络不同，RNN 当前时刻隐藏层的输出不仅仅取决于当前时刻输入层的输入的数据，还和前一时刻隐藏层输出的数据有关^[70]。RNN 的工作原理如图 3-8 所示， x_t 为当前时刻输入层， s_t 为当前时刻隐藏层， o_t 为当前时刻输出层， s_{t-1} 为前一时刻隐藏层。

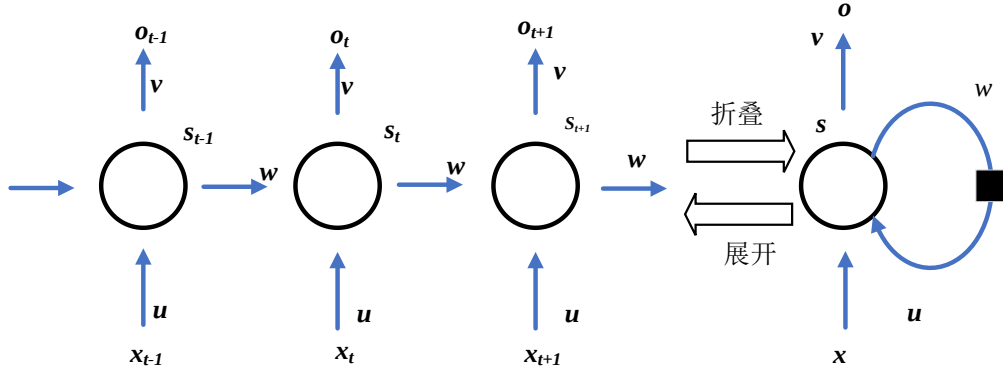


图 3-8 RNN 工作原理

RNN 在时刻 t 隐藏层的值如公式(3-27)所示：

$$s_t = f(ws_{t-1} + ux_t + b) \quad (3-27)$$

式中 s_t 是时刻 t 的隐藏状态， x 是时刻 t 的输入， f 是激活函数， u 和 w 是权重矩阵， b 为偏置项；输出值 o_t 如公式(3-28)所示：

$$o_t = g(v \cdot s_t) \quad (3-28)$$

式中 g 是激活函数， v 为权重矩阵。

3.3.2 双向长短期记忆网络

RNN 在处理较长的数据时，可能会出现梯度消失或者梯度爆炸问题。梯度消失是指 RNN 网络会遗忘靠前的数据信息，而梯度爆炸则是太专注于头部信息，两种问题均会降低模型的预测精度。为了解决这一问题，Hochreiter 在 1997 年提出长短期记忆网络(LSTM)应用于长序列问题的研究^[71]。LSTM 的工作原理如图 3-9 所示，LSTM 通过输入门、遗忘门和输出门控制了时序信息的流动，解决 RNN 神经网络的梯度消失和梯度爆炸。

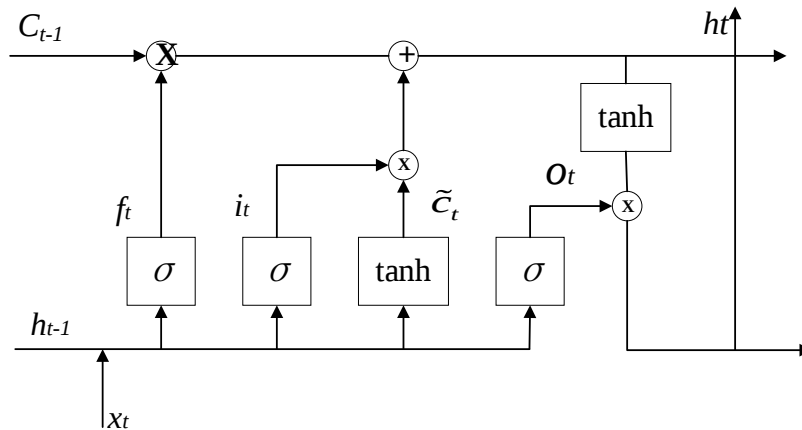


图 3-9 LSTM 结构

遗忘门的作用是决定哪些旧的信息应该从记忆单元中丢弃。遗忘门如公式(3-29)所示:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (3-29)$$

式中 σ 是 ReLU 激活函数, W_f 是遗忘门的权重矩阵, h_{t-1} 是上一个时间步的隐藏状态, x_t 是当前时间步的输入。

输入门负责决定哪些新的信息将被加入到记忆单元中。输入门如公式(3-30)和公式(3-31)所示:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3-30)$$

$$\tilde{c}_t = \varphi(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3-31)$$

式中 $\tilde{c}_t$ 记忆单元的输入状态, 表示新信息的保留程度。 i_t 是输入门的激活值, 表示当前时间步的新信息。

记忆单元更新如公式(3-32)所示:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c} \quad (3-32)$$

式中 c_t 为记忆单元状态。

输出门则负责决定哪些信息将从细胞状态中输出, 并作为当前时刻的隐藏状态。输出门如公式(3-33)和公式(3-34)所示:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (3-33)$$

$$h_t = o_t \odot \varphi(c_t) \quad (3-34)$$

式中 h_t 是当前时间步的隐藏状态。

风速的变化不仅和过去时刻的风速有关, 还和后续的风速有关。LSTM 模型可以有效地提取前一时刻的数据特征, 却无法提取后续数据的特征。因此在 LSTM 模型的基础上提出了一种双向长短期记忆(BILSTM)模型, 该模型包含输入层、正向 LSTM 层、反向 LSTM 以及输出层^[72]。如图 3-10 所示, 在 BILSTM 中, 其中前向 LSTM 的输出值为 $\overrightarrow{ht}$, 后向 LSTM 的输出值为 $\overleftarrow{ht}$, 输出结果 $Ht = [\overrightarrow{ht}, \overleftarrow{ht}]$ 。

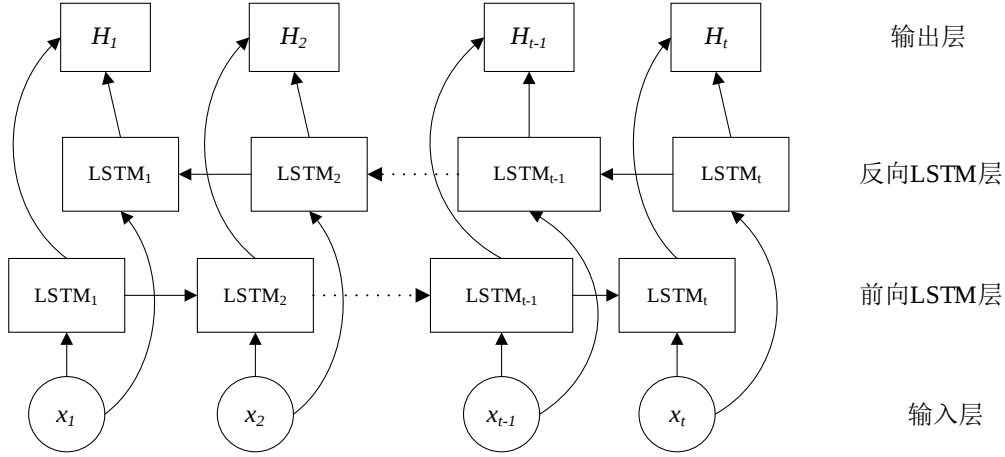


图 3-10 BiLSTM 结构

3.4 注意力机制原理

注意力机制(Attention)是源于人类大脑对于重点信息的提取。当人类面对大量信息时，会重点关注重要部分，对于无关部分会适当忽略，以此来提高信息处理效率和准确性^[73]。在 BiLSTM 中，可以通过 Attention 调整不同特征权重，对于重要的特征，会赋予较大的权重，一般的特征赋予较小的权重，丢弃无关特征的权重。Attention 结构如图 3-11 所示，Attention 具体运用如下所示。

首先通过 $\tanh$ 函数计算各个输出向量的相关性 S_n ，如公式(3-35)所示：

$$S_n = V \tanh(WH_t + UH_t + b) \quad (3-35)$$

式中 H_t 为隐藏层的输出值， V ， W ， U ， b 为模型需参数。

其次，对 S_n 进行 softmax 归一化得到 a_n ，如公式(3-36)所示：

$$a_n = \frac{\exp(S_n)}{\sum_{k=1}^n \exp(S_k)} \quad (3-36)$$

利用注意力权重 a_n 对 BiLSTM 的输出向量进行加权求和，得到输出结果，如公式(3-37)所示：

$$F = \sum_{i=1}^t a_i \times H_i \quad (3-37)$$

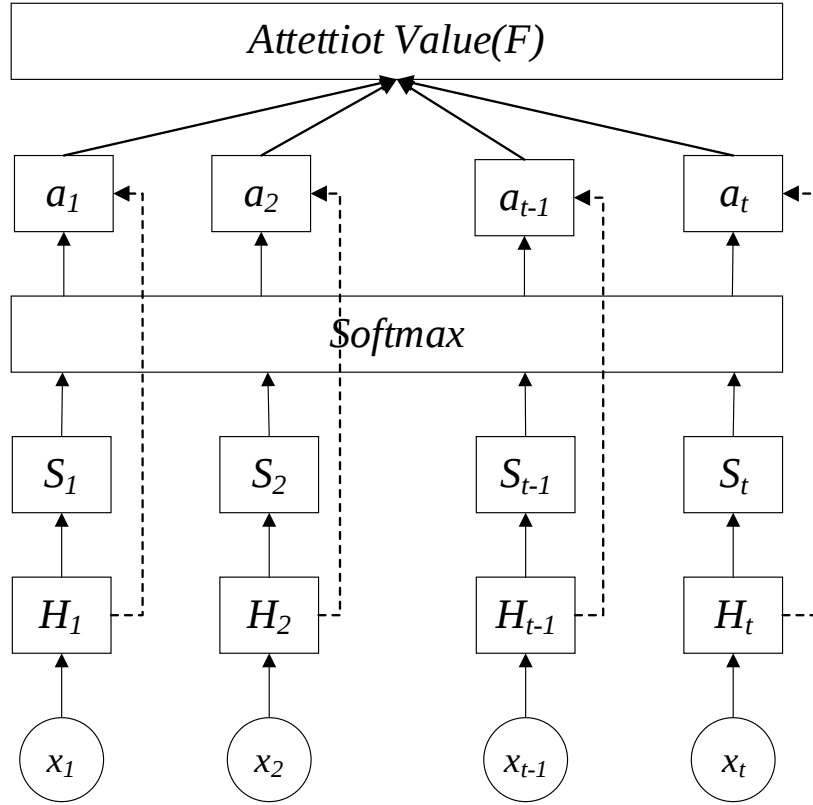


图 3-11 Attention 结构

3.5 构建基于二次分解和 IQPSO-BILSTM 的风速组合预测模型

本文构建的基于二次分解和 IQPSO-BILSTM 的风速组合预测模型，具体流程如图 3-12 和图 3-13 所示，步骤如下：

步骤 1：数据预处理。原始风速数据中可能存在异常值或缺失。为了确保数据的质量，需要对原始数据进行清洗，清洗后的数据采用平均值法来补全缺失数据。对于某个时间点的缺失风速数据，可以计算该时间点前后一段时间内的风速平均值，并将其作为缺失值的估计。数据集的划分是数据预处理中的关键步骤。通常需要将数据集划分为训练集、验证集和测试集。训练集用于模型的训练，验证集用于调整模型的超参数，测试集则用于最终评估模型的泛化能力。本文采用 85% 的数据作为训练集，10% 的数据作为验证集，5% 的数据作为测试集。这种划分方式能够确保模型有足够的数据进行训练，同时也有一定比例的数据用于验证和测试。

步骤 2：对风速数据经行二次分解。CEEMDAN 是一种改进的 EMD 方法，能够有效克服传统 EMD 中的模态混叠问题。通过 CEEMDAN 分解，原始风速

数据被分解为多个子序列，每个子序列代表了不同时间尺度的风速变化特征。接下来，计算各个子序列的排列熵。排列熵是一种衡量时间序列复杂度的指标，熵值越大表示序列的复杂度越高，越难以预测。通过计算每个子序列的排列熵，可以识别出哪些子序列对模型预测精度的负面影响最大。通常，排列熵最高的子序列具有较高的随机性和不规则性，这些子序列会降低模型的预测精度。为了进一步降低数据的复杂度，对排列熵高的子序列采用 VMD 进行二次分解。VMD 是一种基于变分原理的信号分解方法，能够将复杂的信号分解为多个具有不同中心频率的子模态。通过 VMD 分解，排列熵最高的子序列被进一步分解为多个更简单的子模态，从而降低了数据的复杂度。

步骤 3：构建 CNN-BILSTM-Attention 预测模型。CNN-BILSTM-Attention 结构如图 3-12 所示。第一层为输入层。本文输入的数据为经过 CEEMDAN-VMD 分解后子序列。第二层为 CNN 层。通过 1×3 卷积核对输入层接收的数据进行局部感知和特征提取，并采用 ReLU 激活函数提高 CNN 卷积层非线性特性。通过最大池化处理，从数据中逐步提取抽象且具有代表性的特征。第三层为 BILSTM 层。该层接输入的数据来源于 CNN 层提取的局部特征，通过输入门、遗忘门和输出门来控制信息的流动，缓解梯度爆炸和梯度消失问题。通过 BILSTM 的双向结构对数据的时序性进行捕捉。第四层为丢弃层，随机丢弃一些神经元，防止模型过拟合。第五层为 Attention 层。该层的输入为 BILSTM 隐藏层的输出，通过 Attention 提取出重要信息，忽略无关信息，提高预测模型的精度和稳定性。第六层为全连接层。通过对 Attention 层数据进行映射得到最终的预测结果。第七层为输出层，输出预测结果。

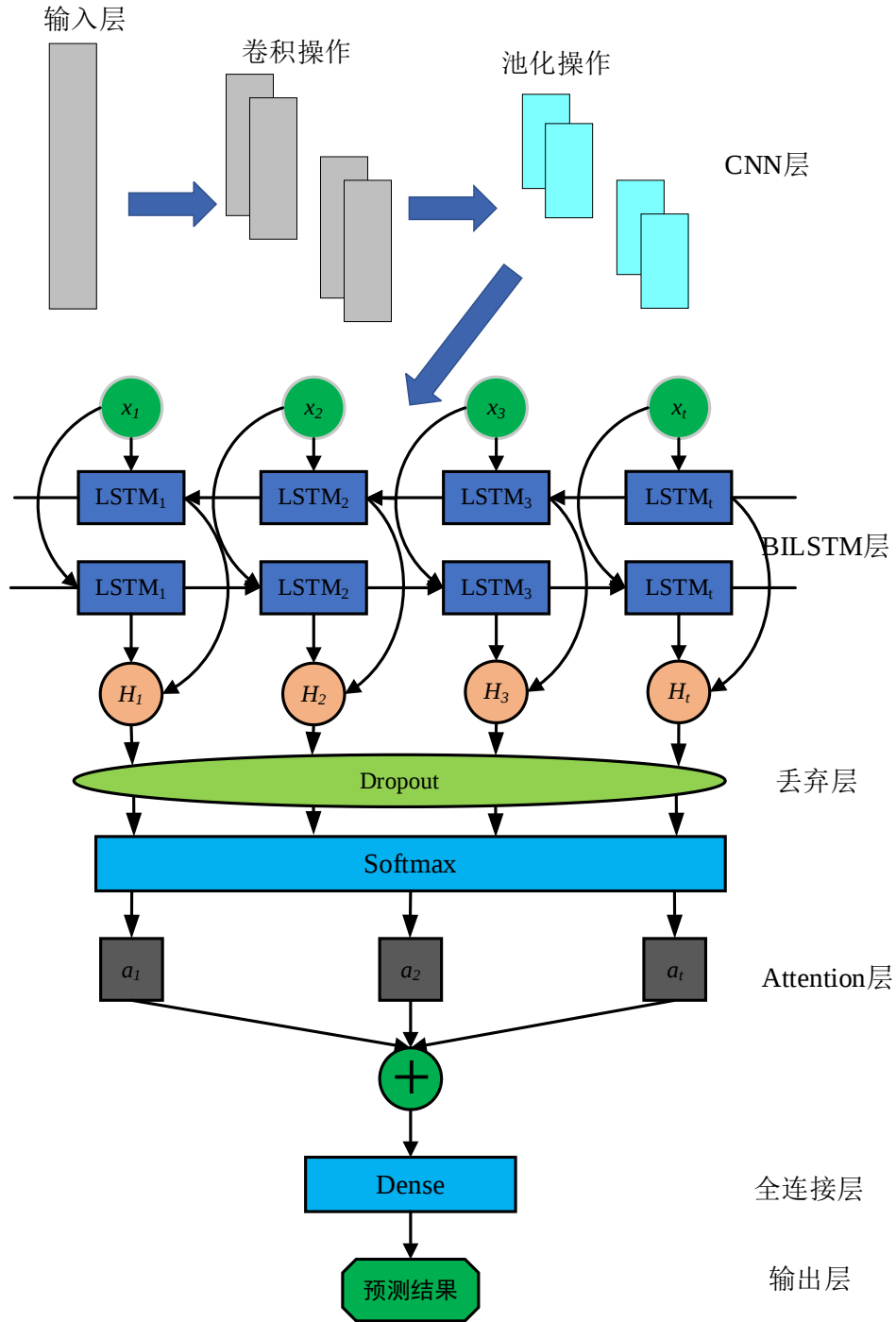


图 3-12 CNN-BILSTM-Attention 结构图

步骤 4：采用 IQPSO 算法对 CNN-BILSTM-Attention 预测模型参数进行优化。IQPSO 的优化目标是最小化验证集的均方误差(MSE)，如公式(3-35)所示。模型超参数及其搜索范围如表 4-1 所示。

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - y_h)^2 \quad (3-35)$$

式中 N 为风速样本容量， y_i 为风速实际值， y_h 为风速预测值。

表 4-1 模型超参数搜索范围

超参数名称	搜索范围
学习率	[0.0001,0.01]
卷积核个数	[0,64]
BILSTM 隐含层神经元数	[1,256]
丢失率	[0.1,0.5]

学习率是控制每次权重更新步幅的参数，决定了模型在训练过程中收敛的速度和稳定性^[74]。卷积核个数决定了卷积神经网络对风速局部特征提取的能力^[75]。BILSTM 隐含层神经元数决定了神经网络对于风速时序性提取的能力^[76]。丢失率是防止神经网络过拟合的一种正则化方法，通常在训练过程中随机丢弃一部分神经元^[77]。通过在这些超参数范围内进行调优，可以找到最佳的组合，以提高模型的准确性和稳定性。

通过不断迭代和搜索，IQPSO 能够找到一组最优的超参数，使得 CNN-BILSTM-Attention 模型在验证集上的表现达到最佳，对不同的测试集运用相同的训练方法，每个子序列会得到不同的参数组合。保留最优的参数组合，用于测试集的预测任务。

步骤 5：将利用训练好的 CNN-BILSTM-Attention 预测模型对相应子序列的测试集进行预测，并将各预测结果累加，得到最终的风速预测结果

步骤 6：对预测模型的精确性和稳定性检验。首先使用多种评估指标，如均方误差(RMSE)、平均绝对误差(MAE)以及决定系数(R^2)对模型的准确性进行检验。RMSE 对大误差较为敏感，可以直观反映模型的预测误差范围，如公式(3-36)所示。MAE 直接反映了预测误差的平均水平，如公式(3-37)所示。 R^2 用于评估模型的解释能力，越接近 1，说明模型对数据的解释能力越强，如公式(3-38)所示：

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - y_h)^2} \quad (3-36)$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - y_h| \quad (3-37)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - y_h)^2}{\sum_{i=1}^n (y_i - y_k)^2} \quad (3-38)$$

式中 N 为风速样本容量, y_i 为风速实际值, y_h 为风速预测值, y_k 为风速序列平均值。

其次。利用方差(σ^2)对模型的稳定性进行评估。 σ^2 越小数据的离散程度或分散程度越低, 模型更稳定, σ^2 如公式 (3-39)所示:

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n (y_k - y_b)^2 \quad (3-39)$$

式中 n 为风速样本容量, y_k 为预测值, y_b 为预测的平均数。

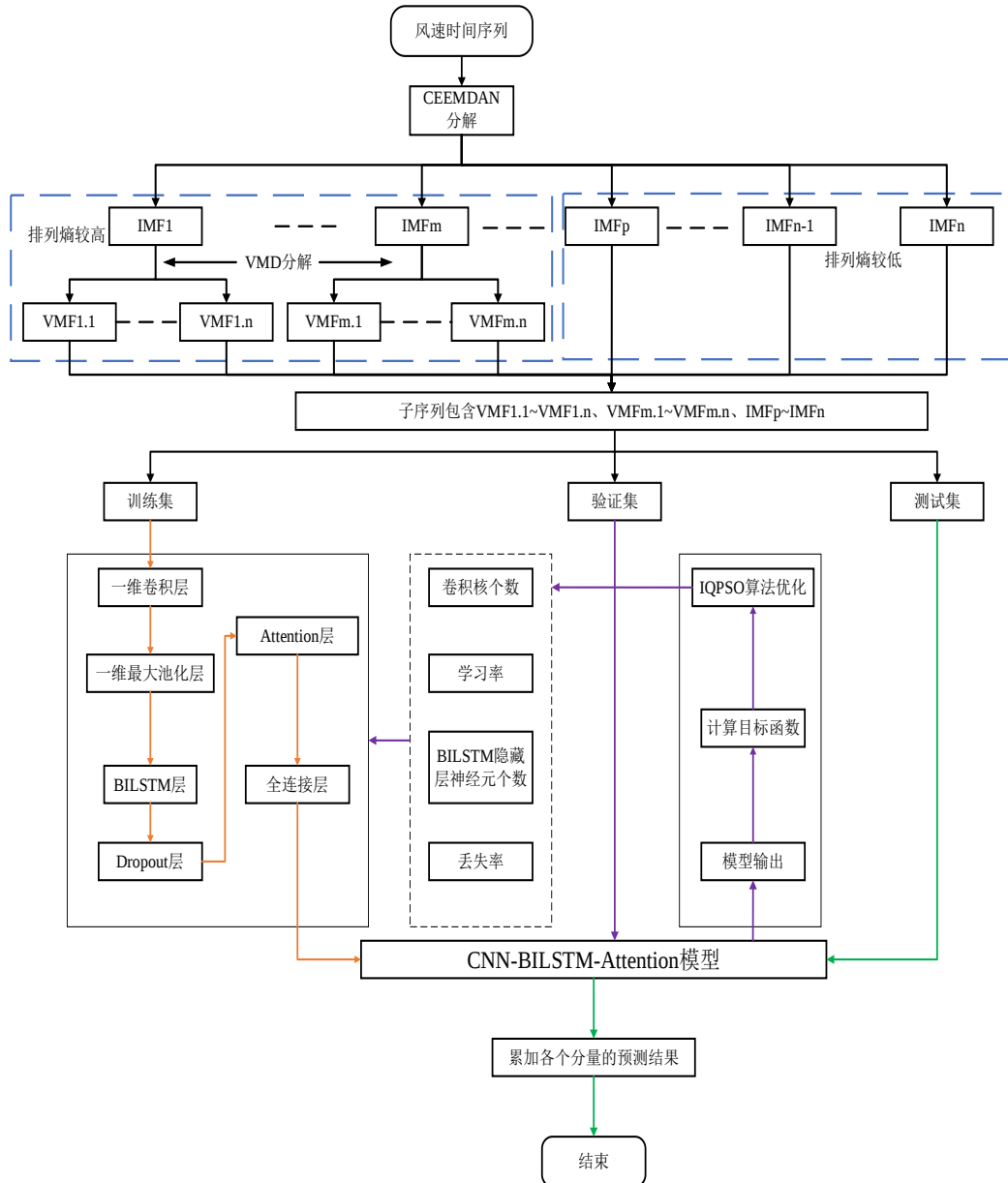


图 3-13 二次分解和 IQPSO-BILSTM 的流程图

3.6 本章小结

本章首先阐述二次分解的相关理论，包括 CEEMDAN、VMD 和排列熵等，以及风速特征选择和风速预测方法，包括 CNN、LSTM、BILSTM 和 Attention 注意力机制的结构和相关理论；然后，利用 CEEMDAN、VMDCNN、LSTM、BILSTM 和 Attention 构建了 CEEMDAN-VMD-IQPSO-CNN-BILSTM-Attention 风速组合预测模型。该模型首先利用 CEEMDAN、VMD 和排列熵理论对风速数据进行预处理，然后，CNN 通过卷积技术提取数据的局部特征，BILSTM 通过前向 LSTM 和后向 LSTM 对数据的时序性进行捕捉，Attention 提取出特征中的重要信息，忽略无关信息；利用 IQPSO 对模型参数进行优化，进一步提高预测模型泛化能力。

第4章 实验结果分析

为了验证新建风速预测模型的有效性,利用内蒙古某风电场的实测历史风速数据进行测试新建风速预测模型和其它典型的风速预测模型。首先,对选取两组的风速数据进行 CEEMDAN-VMD 二次分解,数据被分解为不同的固有模态分量,旨在有效地去除噪声,提高后续分析的准确性。其次,采用滑动窗口的方式对分解后的数据进行构建,设置输入和输出。最后,设置两组实验,分别验证 CEEMDAN-VMD-IQPSO-CNN-BILSTM-Attention 在单步预测,二步预测以及三步的预测效果。采用 R^2 、RMSE、MAPE 以及 Var 作为评价指标,对模型的准确性和稳定性进行验证。实验I首先对未使用分解算法的模型如 IQPSO-CNN、IQPSO-LSTM、IQPSO-BILSTM,使用一次分解算法模型如 CEEMDAN-IQPSO-BILSTM、EMD-IQPSO-BILSTM,以及使用二次分解算法模型 CEEMDAN-VMD-IQPSO-BILSTM 准确性进行验证,其次验证本文构建的预测模型的准确性。实验II验证本文模型的在一步、二步乃至三步的预测任务上稳定性,并以前文的预测模型作为对照组。

4.1 风速数据预处理

4.1.1 实验数据选择

内蒙古作为中国一个地理面积广阔但人口相对稀少的省份,拥有得天独厚的风能资源。广袤的草原和独特的气候条件,使得该地区成为风能开发的理想场所。在风电产业方面,内蒙古的风电装机容量居全国前列,已成为我国风能发电的重要基地。本章选取了国家气象中心所记录的内蒙古某风电场的风速数据进行研究。这些数据具有较高的可信度和代表性,能够较好地反映当地的风力资源状况。通过对这些数据的深入分析,可以提取出风速变化的规律,建立适合当地风电场特点的风速预测模型。

本章选取了内蒙古某风电场 2018 年 7 月份和 12 月份的风速数据作为测试样本,数据采样时间间隔为 15 分钟,每组数据共计 2976 份数据。其中,85%的数据作为训练集,用于模型的训练。10%的数据作为验证集,用于模型的调参,5%的数据作为测试集。样本 A 的风速数据如图 4-1 所示,样本 B 的风速数据如

图 4-2 所示，统计指标如表 4-1 所示。在样本 A 风速的最大值、最小值、均值、标准差分别为 18.6440(m/s)、0.0010(m/s)、6.2563(m/s)、2.8839(m/s)。样本 B 风速的最大值、最小值、均值、标准差分别为 12.4660(m/s)、0.0270(m/s)、4.4103(m/s)、2.3005(m/s)。由表 4-1、图 4-1 和图 4-2 可知，风速样本波动大和非线性强。

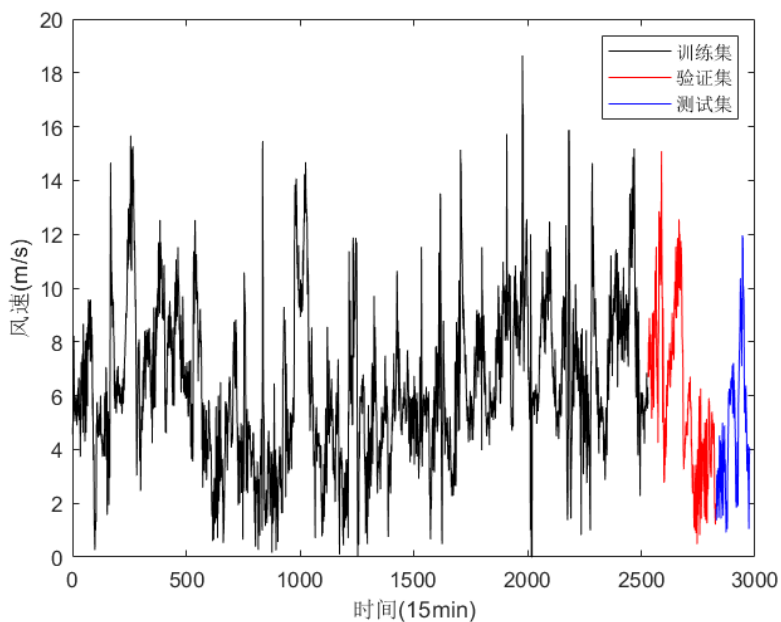


图 4-1 样本 A: 2018 年 7 月份风速数据

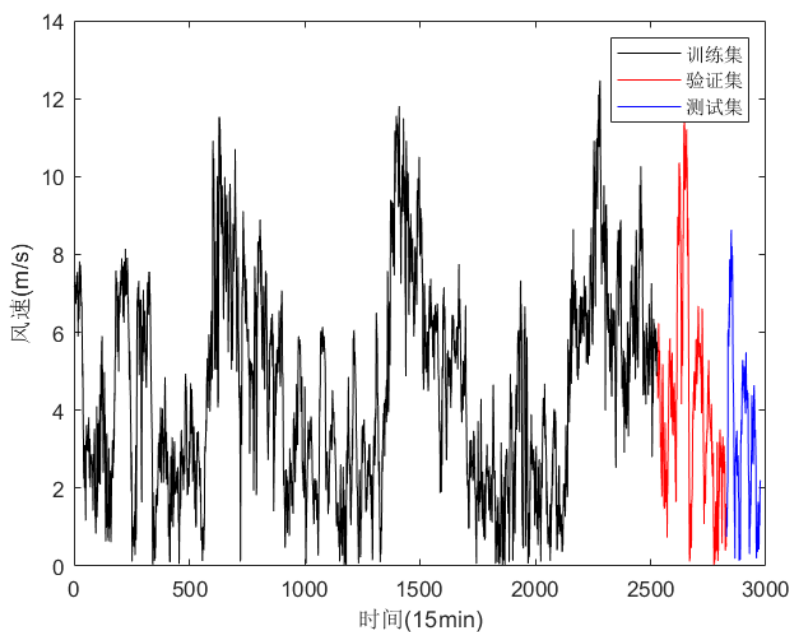


图 4-2 样本 B: 2018 年 12 月份风速数据

表 4-1 风速数据的统计指标(m/s)

数据集	样本	最大值	最小值	均值	标准差
样本 A	所有样本	18.6440	0.0010	6.2563	2.8839
	训练集	18.6440	0.0010	6.3610	2.8611
	验证集	15.0880	0.4870	6.0883	3.0629
	测试集	11.9600	0.9190	4.8038	2.4839
样本 B	所有样本	12.4660	0.0270	4.4103	2.3005
	训练集	12.4660	0.0270	4.4950	2.3096
	验证集	11.6120	0.0300	4.2072	2.3380
	测试集	8.6300	0.1330	3.3720	1.8854

4.1.2 数据分解

由图 4-1 和图 4-2，以及表 4-1 可知，样本 A 和样本 B 风速数据随机性和不确定性较强。因此应用 CEEMDAN 分解方法对样本 A 和样本 B 进行分解，得到的分解结果分别如图 4-3 和图 4-4 所示，以降低原始风速数据的随机性。样本 A 得到分解子序列记为 IMFA，样本 B 分解得到的子序列记为 IMFB。

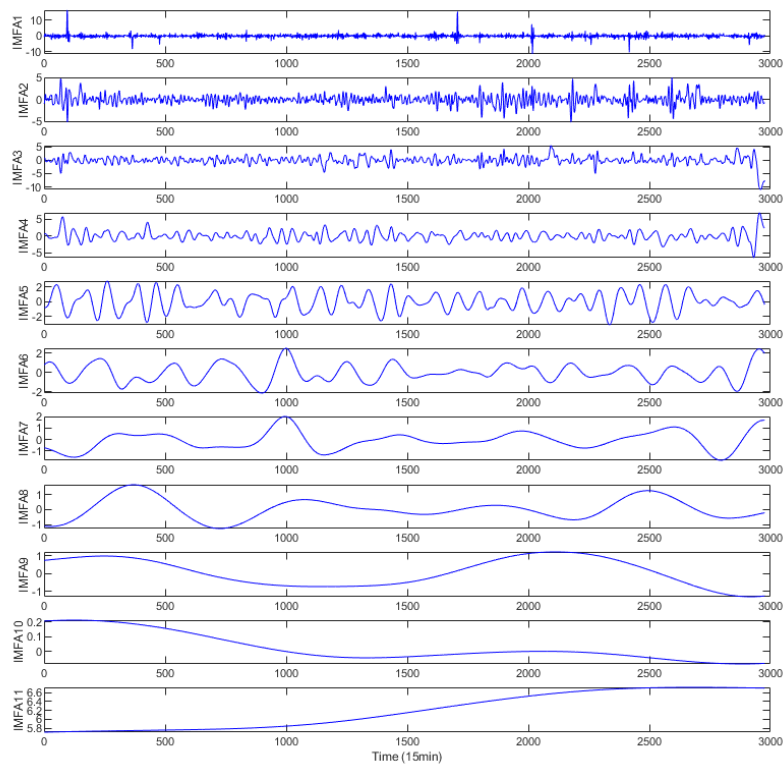


图 4-3 样本 A 的 CEEMDAN 分解图

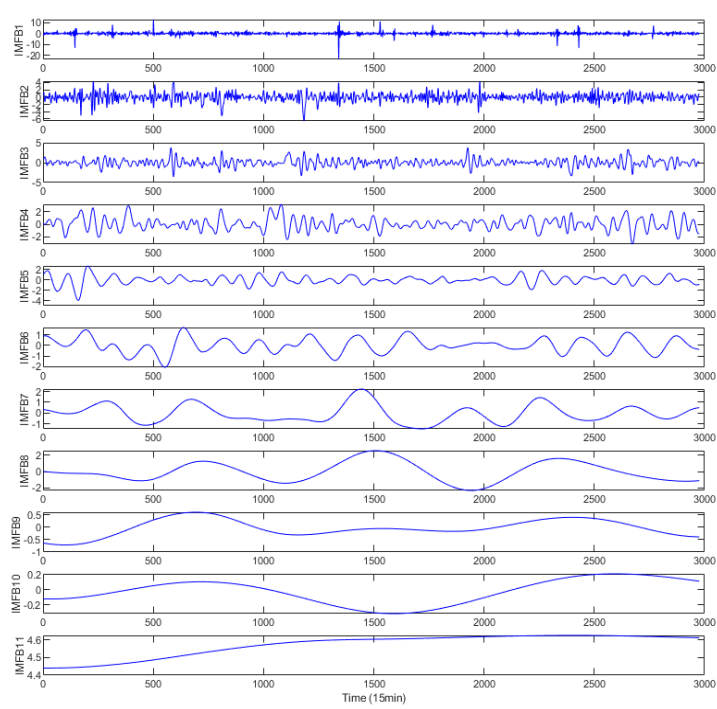


图 4-4 样本 B 的 CEEMDAN 分解图

样本 A 和样本 B 在完成 CEEMDAN 分解后分别生成 11 个子序列(IMF)，降低原始风速数据的不规律性。通过计算每个 IMF 的排列熵，可以观察每个 IMF 的非线性。如图 4-5 所示，样本 A 的 IMFA1 和 IMFA2 的排列熵分别为 0.9939 和 0.8332，其他模态分量排列熵均远小于 IMFA1 和 IMFA2，可以认为 IMFA1 和 IMFA2 具有较多的噪声，需要采用 VMD 进行二次分解，降低 IMFA1 和 IMFA2 的复杂度，提高预测模型的精度和稳定性。

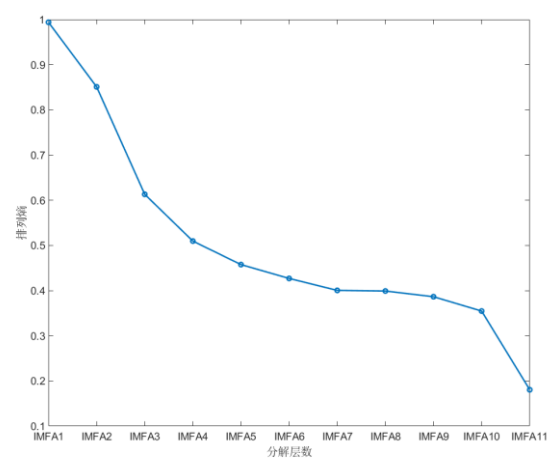


图 4-5 样本 A :CEEMDAN 分解各模态分量排列熵

样本 B 经过 CEEMDAN 分解后, 各个子序列的模态分量如图 4-6 所示。和样本 A 类似, IMFB1 和 IMFB2 的排列熵分别为 0.9759 和 0.8664, 均远大于其余模态分量的排列熵, 需要对该子序列采用 VMD 进行二次分解。

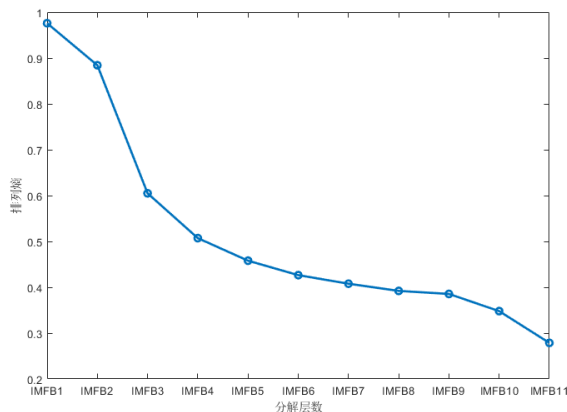


图 4-6 样本 B :CEEMDAN 分解各模态分量排列熵

对样本 A 的 IMFA1 和 IMFA2 进行 VMD 分解, 得到的子序列分别为 VMFA1 和 VMFA2。VMFA1 中心频率变化如表 4-2 所示, 当分解层数为 7 时, 中心频率变化趋于平稳, 此时会被认为过度分解, 因此 VMD 分解层数为 6。VMFA2 中心频率变化如表 4-2 所示, 当分解层数为 6 时, 中心频率变化趋于平稳, 因此分解层数为 5。IMFA1 分解的结果如图 4-7 所示。IMFA2 分解的结果如图 4-8 所示。VMFA1.1~ VMFA1.7, VMFA2.1~ VMFA2.5 为 VMD 分解后的子序列, Res 为分解后的残差信号。

表 4-2 IMFA1 不同分解层数下的中心频率

K	VMF A1.1	VMF A1.2	VMF A1.3	VMF A1.4	VMF A1.5	VMF A1.6	VMF A1.7
1	0.1277						
2	0.0470	0.2175					
3	0.0613	0.1507	0.3457				
4	0.0508	0.1451	0.2355	0.4065			
5	0.0346	0.1221	0.1972	0.3164	0.4614		
6	0.0256	0.0916	0.1592	0.2288	0.3418	0.4620	
7	0.0296	0.0898	0.1453	0.2033	0.2875	0.3927	0.4008

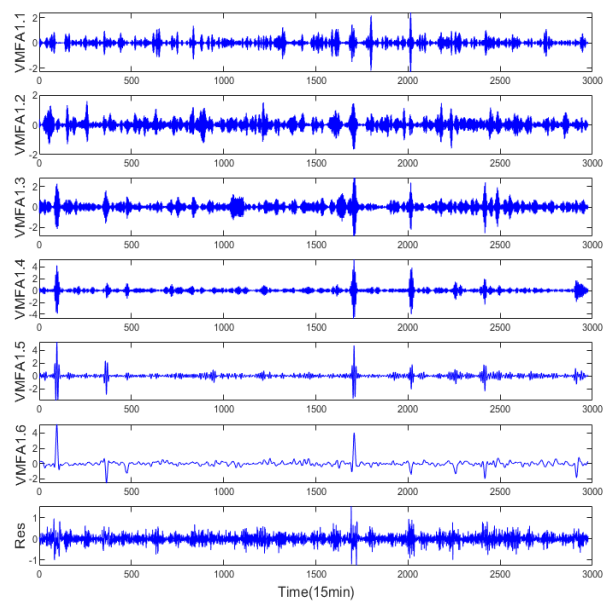


图 4-7 IMFA1 分量 VMD 分解结果

表 4-3IMFA2 不同分解层数下的中心频率

K	VMFA2.1	VMFA2.2	VMFA2.3	VMFA2.4	VMFA2.5
1	0.0654				
2	0.0516	0.1025			
3	0.0271	0.0497	0.1131		
4	0.0281	0.0532	0.0846	0.1292	
5	0.0197	0.0351	0.0749	0.1392	0.1434

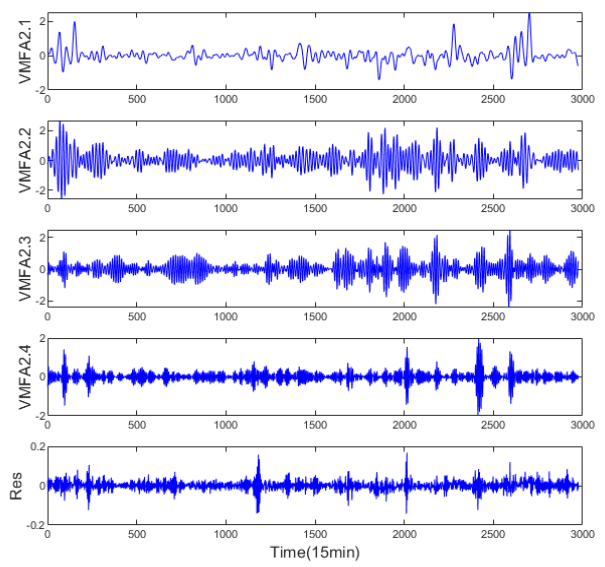


图 4-8 IMFA2 分量 VMD 分解结果

对样本 B 排列熵较高的 IMFB1 和 IMFB2 进行 VMD 分解，得到子序列分别为 VMFB1 和 VMFB2。VMFB1 中心频率变化如表 4-4 所示，当分解层数为 7 时，中心频率变化趋于平稳，此时会被认为过度分解，因此 VMD 分解层数为 6。VMFB2 中心频率变化如表 4-5 所示，当分解层数为 6 时，中心频率变化趋于平稳，因此分解层数为 5。IMFB1 分解的结果如图 4-9 所示。IMFB2 分解的结果如图 4-10 所示。

表 4-4 IMFB1 不同分解层数下的中心频率

K	VMF B1.1	VMF B1.2	VMF B1.3	VMF B1.4	VMF B1.5	VMF B1.6
1	0.1780					
2	0.1436	0.2547				
3	0.0367	0.1302	0.2921			
4	0.0347	0.1139	0.2367	0.3483		
5	0.0686	0.1101	0.1964	0.2686	0.3848	
6	0.0228	0.0819	0.1380	0.2233	0.3089	0.3143

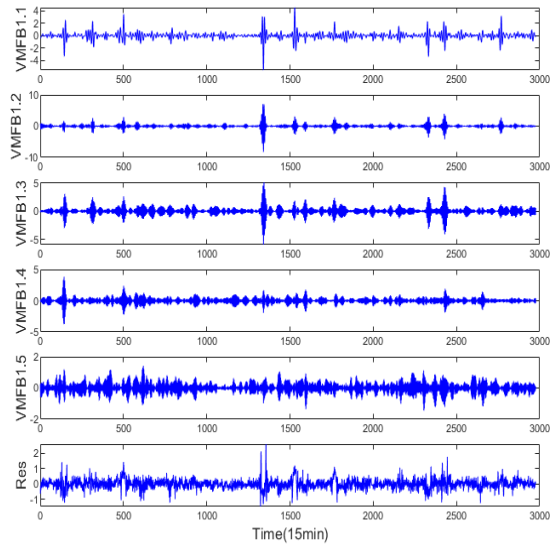


图 4-9 IMFB1 分量 VMD 分解结果

表 4-5 IMFB2 不同分解层数下的中心频率

K	VMF B2.1	VMF B2.2	VMF B2.3	VMF B2.4	VMF B2.5	VMF B2.6
1	0.0786					
2	0.0570	0.1152				
3	0.0338	0.0622	0.1165			
4	0.0227	0.0359	0.7837	0.1224		
5	0.0205	0.0301	0.0647	0.0987	0.1344	
6	0.0178	0.0319	0.0680	0.0833	0.1289	0.1373

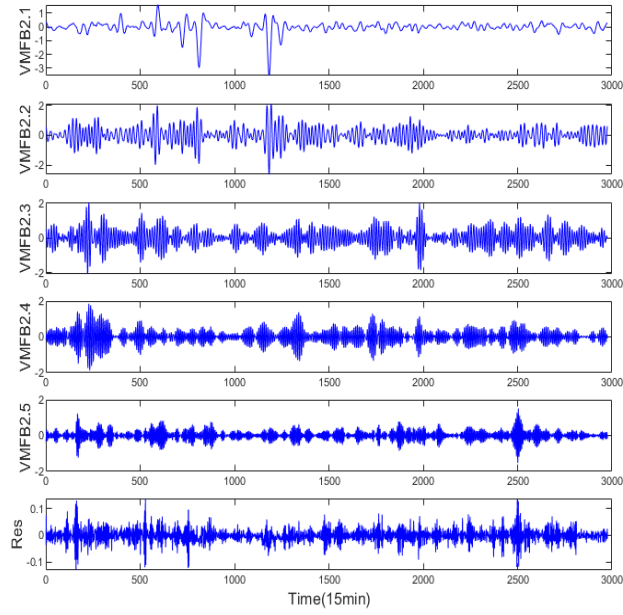


图 4-10 IMFB2 分量 VMD 分解结果

通过对风速样本 A 和样本 B 进行 CEEMDAN-VMD 二次分解，能够有效减少风速数据中的噪声，提高风速预测模型的准确性。

4.2 滑动预测机理

滑动窗口机制(SWM)是风速预测中处理风速数据的常见手段。如图 4-11 所示，SWM 通过设定固定的窗口大小在风速数据上经行滑动，提取输入向量。窗口后部分数据作为输出向量。现有一列风速数据 $x=\{x_1, x_2, x_3, \dots, x_n\}$ ，分别进行单步预测和三步预测，构建预测模型的输入和输出向量。单步预测的输入向量、输出向量如公式(4-1)所示：

$$\begin{aligned} (x_1, x_2, \dots, x_h) &\Rightarrow (x_{h+1}) \quad t=1 \\ (x_2, x_3, \dots, x_{h+1}) &\Rightarrow (x_{h+2}) \quad t=2 \\ (x_k, x_{k+2}, \dots, x_{k+h}) &\Rightarrow (x_{h+k+1}) \quad t=k \end{aligned} \quad (4-1)$$

式中 h 为窗口大小，在 $t=k$ 时刻，模型的输入 向量为 $\{x_k, x_{k+2}, \dots, x_{k+h}\}$ ，模型的输出向量为 $\{x_{k+h+1}\}$ 。

三步预测的输入向量、输出向量如公式(4-2)所示：

$$\begin{aligned}
(x_1, x_2, \dots, x_h) &\Rightarrow (x_{h+1}, x_{h+2}, x_{h+3}) \quad t=1 \\
(x_2, x_3, \dots, x_{h+1}) &\Rightarrow (x_{h+2}, x_{h+3}, x_{h+4}) \quad t=2 \\
(x_k, x_{k+2}, \dots, x_{k+h}) &\Rightarrow (x_{h+k+1}, x_{h+k+2}, x_{h+k+3}) \quad t=k
\end{aligned} \tag{4-2}$$

式中 h 为窗口大小，在 $t=k$ 时刻，模型的输入向量为 $\{x_k, x_{k+2}, \dots, x_{k+h}\}$ ，模型的输出向量为 $\{x_{k+h+3}\}$ 。

本文进行风速预测时，假设选用窗口大小为 24。以前 24 组数据作为模型的输入，以后几组数据作为模型的输出。

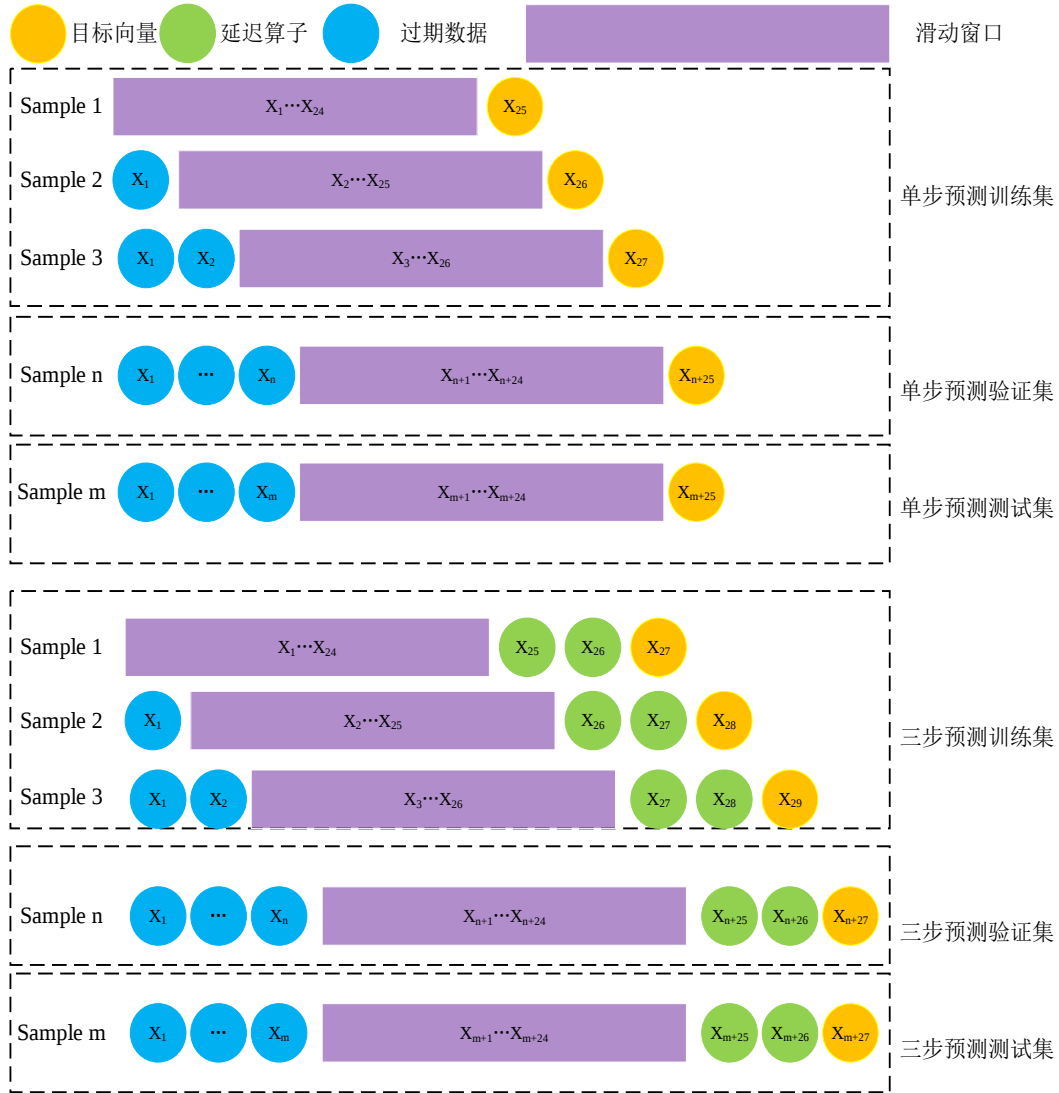


图 4-11 滑动预测机理示意图

4.3 实验I：模型精确性验证

为验证本文构建的基于二次分解和 IQPSO-BILSTM 的预测模型能够有效的提高风速预测的精度，本章节通过样本 A 和样本 B 进行行预测。在风速预测中，

RMSE、 R^2 和MAPE各自反映了模型预测性能不同方面预测准确性。RMSE是一种衡量预测值与真实值之间差异的指标，RMSE的值越小，表示模型的预测越准确。 R^2 值则表示模型拟合数据的程度，越接近1说明模型的预测能力越强。MAPE衡量的是预测误差占实际值的比例，给出了误差的百分比形式。

4.3.1 基于二次分解预测模型准确性验证

为了验证二次分解能够提高预测模型的准确性，本文采用未使用分解算法的模型如IQPSO-CNN、IQPSO-LSTM、IQPSO-BILSTM。使用一次分解算法模型如CEEMDAN-IQPSO-BILSTM、EMD-IQPSO-BILSTM。以及使用二次分解算法模型CEEMDAN-VMD-IQPSO-BILSTM二次分解预测模型进行比较。采用 R^2 、RMSE、MAPE作为评价指标。

表 4-6 单步预测各模型评价指标

预测模型	样本 A			样本 B		
	R^2	RMSE (m/s)	MAPE (%)	R^2	RMSE (m/s)	MAPE (%)
IQPSO-CNN	0.8741	0.8784	22.4919	0.8307	0.8551	21.1472
IQPSO-LSTM	0.9047	0.7642	18.6919	0.8614	0.7738	20.0412
IQPSO-BILSTM	0.9193	0.7031	16.7301	0.8987	0.6616	18.8356
EMD-IQPSO-BILSTM	0.9551	0.4626	11.675	0.9452	0.44	11.9885
CEEMDAN-IQPSO-BILSTM	0.9622	0.2843	6.9261	0.9592	0.3125	6.1425
CEEMDAN-VMD-IQPSO-BILSTM	0.9798	0.195	4.9379	0.9755	0.2028	5.0529

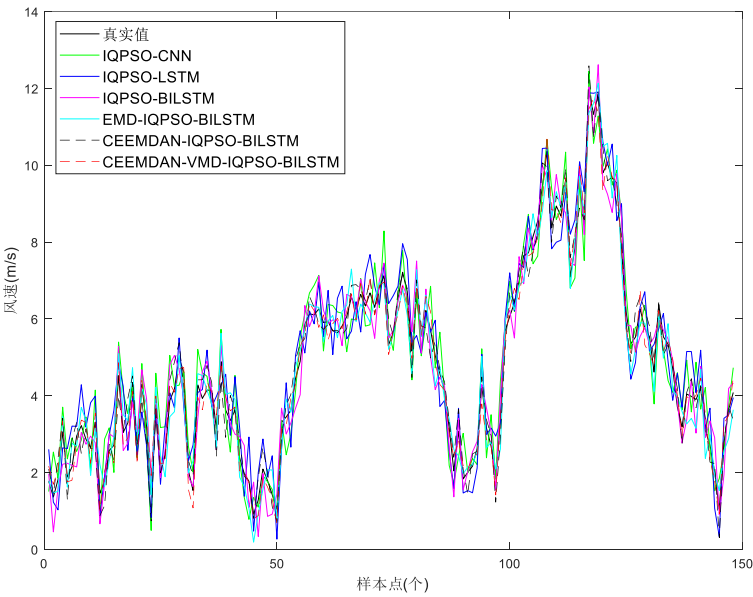


图 4-12 样本 A 单步预测

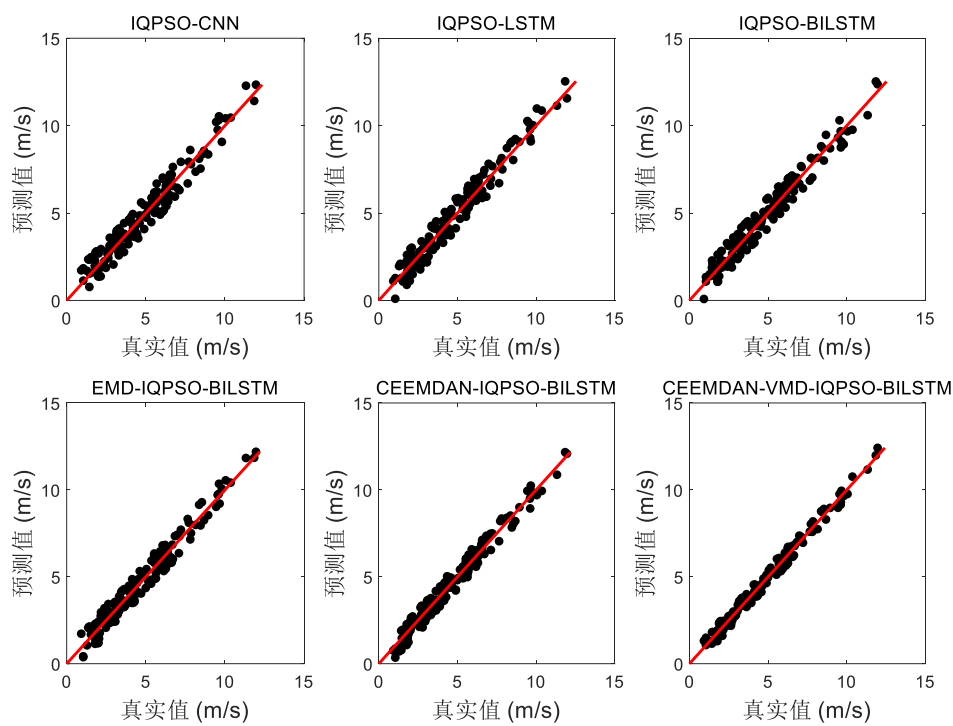
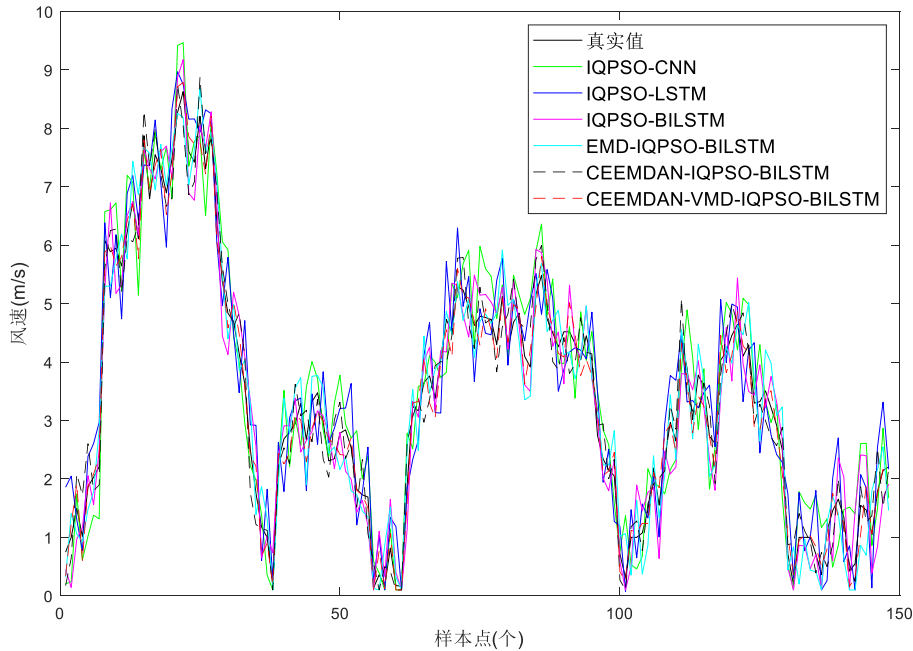


图 4-13 样本 A 模型预测散点图



4-14 样本 B 单步预测

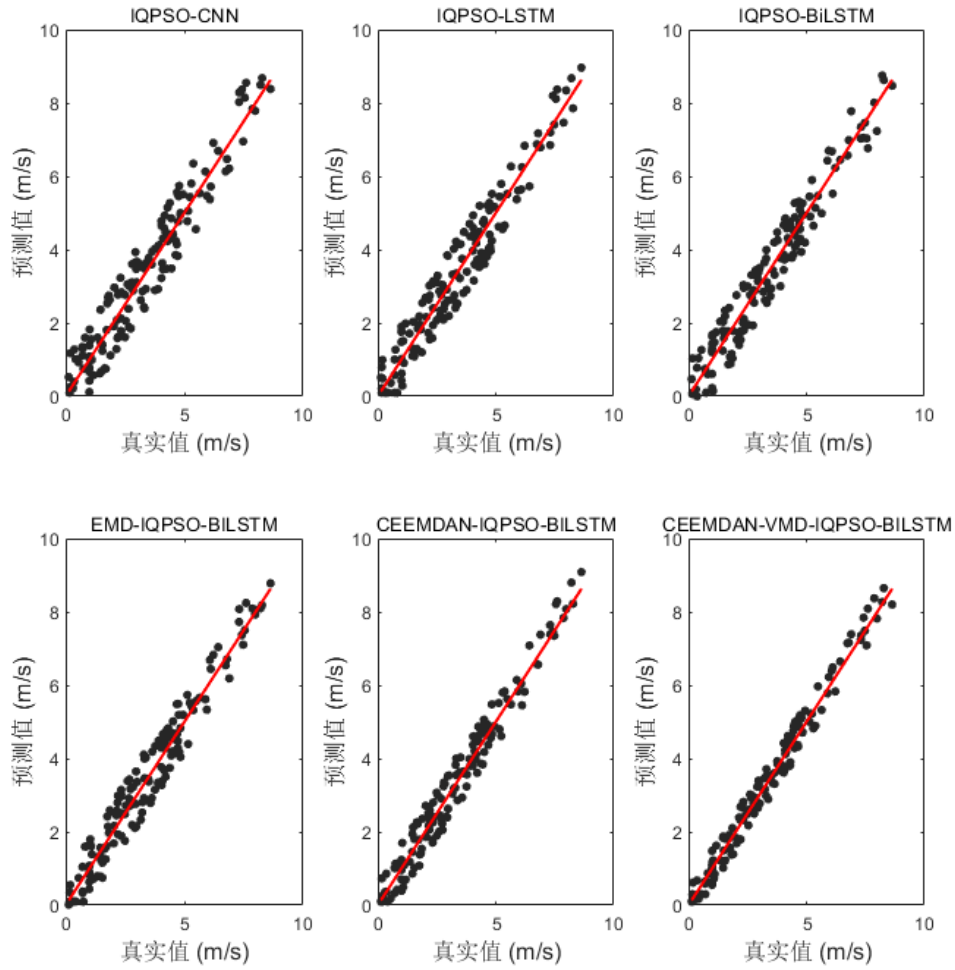


图 4-15 样本 B 模型预测散点图

如图 4-12 和图 4-14 所示，各个模型在样本 A 和样本 B 的预测任务中，均能较为准确的预测出未来的风速数据，通过图中的预测曲线，可以大致看出，IQPSO-BiLSTM 模型的预测效果要优于 IQPSO-LSTM 和 IQPSO-CNN。同时未使用分解算法的预测模型，预测效果要逊于使用分解算法的预测模型，其中 CEEMDAN-VMD 分解算法对预测模型的精度提升要优于 EMD、CEEMDAN。为了更加直观的体现出 CEEMDAN-VMD 在预测算法中的优势，本章节计算各个模型在样本 A 和样本 B 中的 R^2 、RMSE、MAPE。评价指标如表 4-6 所示。在未使用分解算法处理风速数据时，模型的预测效果较差。在样本 A 中 IQPSO-CNN、IQPSO-LSTM、IQPSO-BiLSTM 的 R^2 分别为 0.8741、0.9047 和 0.9193，可以看出 BiLSTM 预测精度要优于 CNN 和 LSTM。在引入 EMD、CEEMDAN 和 CEEMDAN-VMD 分解算法后，BiLSTM 模型的预测精度普遍得到了提升。

在样本 A 中 EMD-IQPSO-BILSTM 的 R^2 、RMSE 以及 MAPE 分别为 0.9193、0.7031m/s、16.7301%。CEEMDAN-IQPSO-BILSTM 的 R^2 、RMSE 以及 MAPE 分别为 0.9622、0.2843m/s、6.9261%。在样本 A 中 CEEMDAN-VMD-IQPSO-BILSTM 的 R^2 、RMSE 以及 MAPE 分别为 0.9798、0.195m/s、4.9379%。样本 B 的 R^2 、RMSE 以及 MAPE 分别为 0.9755、0.2028m/s、5.0529%。如图 4-13 和图 4-15 所示，相对于未采用分解策略的预测模型，采用分解策略的预测模型的预测值更接近真实值，而 CEEMDAN-VMD-IQPSO-BILSTM 在样本 A 和样本 B 的预测值更靠近真实值。由此可见采用二次分解策略的预测模型预测精度最高。

4.3.2 本文模型准确性验证

为进一步验证本文构建的预测模型的准确度要优于 CEEMDAN-VMD-IQPSO-BILSTM。对样本 A 和样本 B 进行一步、二步和三步预测。

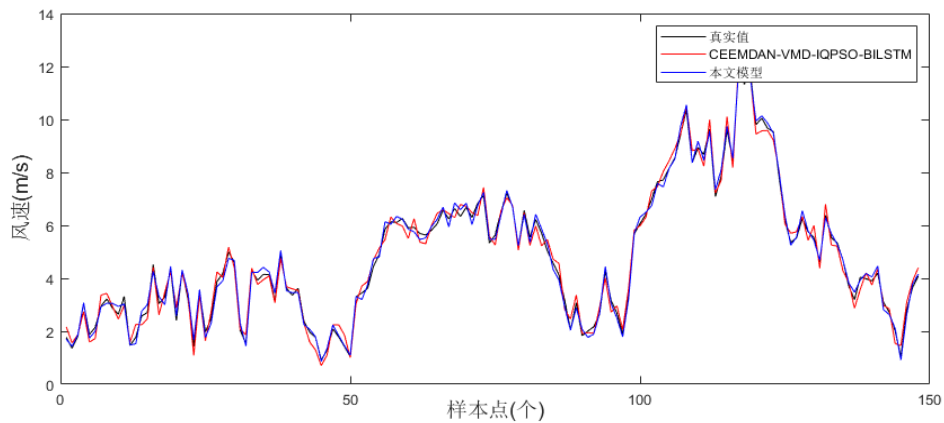


图 4-16 样本 A 单步预测

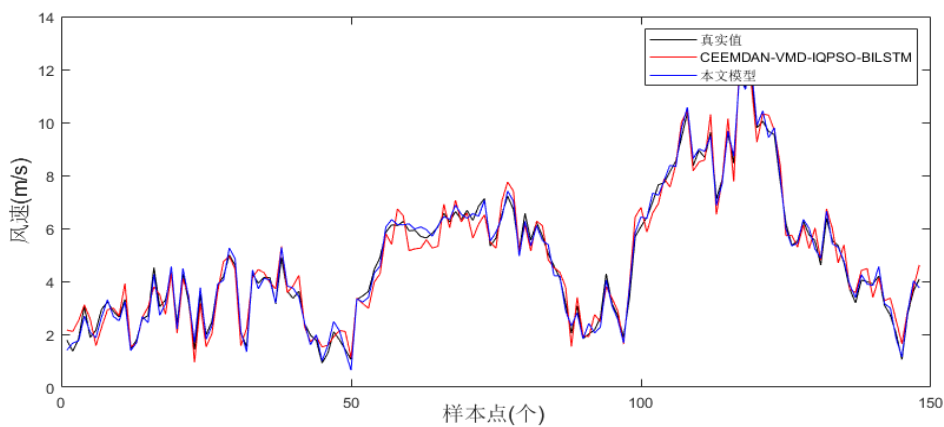


图 4-17 样本 A 二步预测

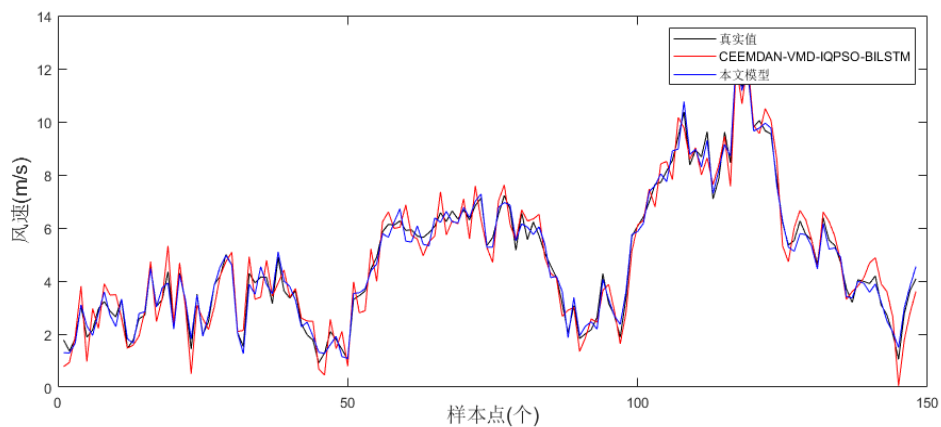


图 4-18 样本 A 三步预测

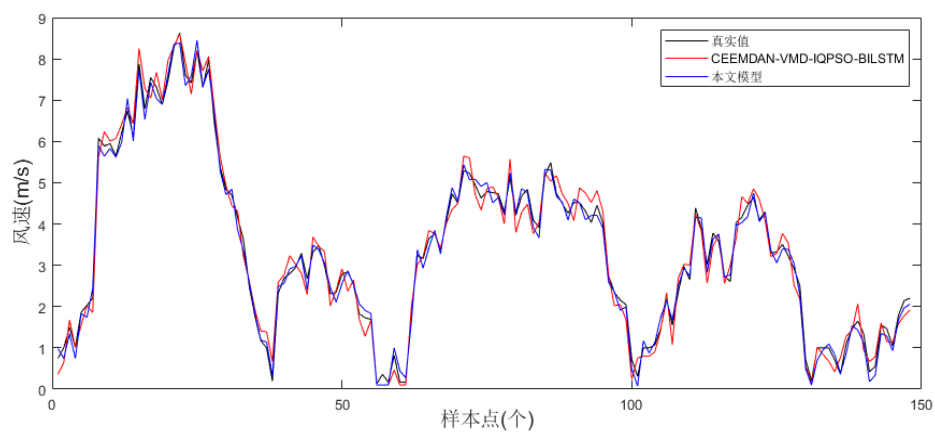


图 4-19 样本 B 单步预测

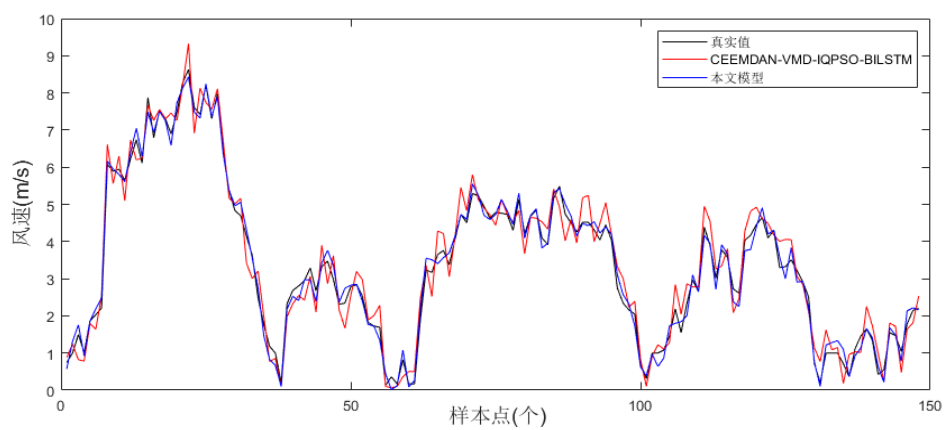


图 4-20 样本 B 二步预测

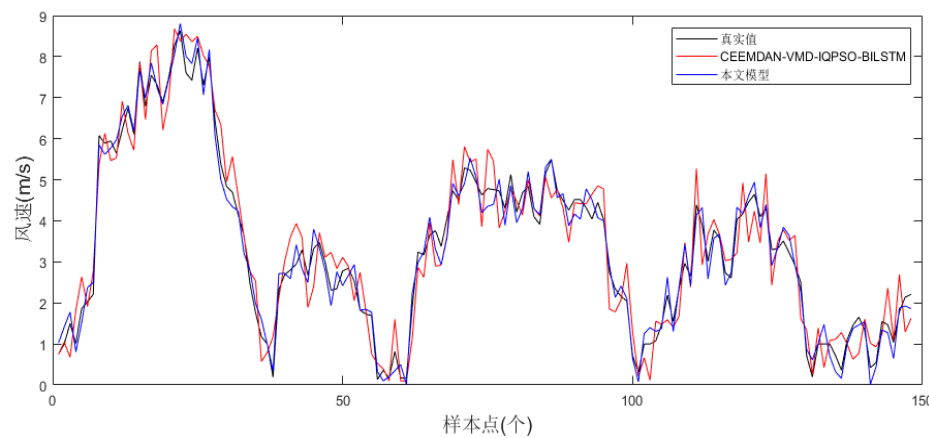


图 4-21 样本 B 三步预测

表 4-6 样本 A 多步预测各模型评价指标

预测步数	CEEMDAN-VMD-IQPSO-BILSTM			本文模型		
	R ²	RMSE(m/s)	MAPE(%)	R ²	RMSE(m/s)	MAPE(%)
一步	0.9798	0.1950	4.9379	0.9846	0.1813	3.6667
二步	0.9502	0.5272	10.8206	0.9721	0.3806	5.5346
三步	0.9175	0.8707	14.2714	0.9564	0.5920	7.4213

表 4-7 样本 B 多步预测各模型评价指标

预测步数	CEEMDAN-VMD-IQPSO-BILSTM			本文模型		
	R ²	RMSE(m/s)	MAPE(%)	R ²	RMSE(m/s)	MAPE(%)
一步	0.9755	0.2028	5.0529	0.9830	0.1735	4.8042
二步	0.9487	0.6705	12.9133	0.9772	0.3951	8.7252
三步	0.9085	0.9957	27.7750	0.9422	0.6076	10.6125

如图 4-16 至图 4-21，表 4-6 和表 4-7 所示。CEEMDAN-VMD-IQPSO-BILSTM 预测模型和本文构建的模型随着预测步数的增加，预测模型的精度均开始下降。在样本 A 中，CEEMDAN-VMD-IQPSO-BILSTM 单步预测的 R^2 为 0.9798，RMSE 为 0.1950m/s，MAPE 为 4.9379%。在三步预测中 R^2 降低了 0.0623，RMSE 提高了 0.6757m/s，MAPE 提高了 9.338%。本文构建的模型在样本 A 的单步预测中 R^2 ，RMSE，MAPE 分别为 0.9846、0.1813m/s、3.6667%。在样本 A 的三步预测中 R^2 ，RMSE，MAPE 分别为 0.9564、0.5920m/s、7.4213%。在样本 B 的三步预测中 R^2 ，RMSE，MAPE 分别为 0.9422、0.6076m/s、10.6125%。可见，本文构建的基于二次分解和 IQPSO-BILSTM 的风速组合预测模型预测精度均高于 CEEMDAN-VMD-IQPSO-BILSTM。因此，而本文构建的预测模型在不论是单步还是多步预测中均保持较高的预测精度。

4.4 实验II：模型稳定性验证

为验证本文构建的基于二次分解和 IQPSO-BILSTM 的风速组合预测模型具有更高的稳定性，本章节通过计算 IQPSO-CNN、IQPSO-LSTM、IQPSO-BILSTM、EMD-IQPSO-BILSTM、CEEMDAN-VMD-IQPSO-BILSTM 等对比模型在样本 A 和样本 B 单步预测，二步预测以及三步预测上的方差，结果表 4-8 所示。

方差是统计学中衡量数据的分布离散程度。在风速预测中方差大小直接反映了预测模型的稳定性。当样本方差较大时，表明数据越分散，模型的稳定性较差；反之，当样本方差较小时，数据越集中，表明模型的预测结果较为稳定，可靠性较高。

表 4-8 各模型预测方差((m/s)²)

预测模型	单步预测		二步预测		三步预测	
	样本 A	样本 B	样本 A	样本 B	样本 A	样本 B
IQPSO-CNN	0.6984	0.5993	0.8865	0.8342	0.9008	0.9118
IQPSO-LSTM	0.5204	0.5586	0.7855	0.7961	0.8762	0.9042
IQPSO-BILSTM	0.4593	0.4654	0.5428	0.5396	0.7618	0.8755
EMD-IQPSO-BILSTM	0.2067	0.2047	0.4975	0.4623	0.7242	0.7514
CEEMDAN-IQPSO-BILSTM	0.0816	0.0830	0.2031	0.2128	0.4135	0.5543
CEEMDAN-VMD-IQPSO-BILSTM	0.0402	0.0398	0.1196	0.1514	0.2804	0.3241
本文模型	0.0314	0.0269	0.0885	0.0761	0.1561	0.1486

从表格中的数据可以看出，本文模型在单步预测，二步预测以及三步预测中均体现了较高的稳定性。单步预测中，本文模型在样本 A 和样本 B 的预测方差分别为 $0.0314(\text{m/s})^2$ 和 $0.0269(\text{m/s})^2$ ，远低于其他模型，如 IQPSO-CNN 的 $0.6984(\text{m/s})^2$ 和 $0.5993(\text{m/s})^2$ ，以及 EMD-IQPSO-BILSTM 的 $0.2067(\text{m/s})^2$ 和 $0.2047(\text{m/s})^2$ 。这表明基于二次分解和 IQPSO-BILSTM 的风速组合预测模型在单步预测中更稳定。在二步和三步预测中，随着预测步数的增加，各种预测模型的稳定性都开始下降。例如 CEEMDAN-VMD-IQPSO-BILSTM 在样本 A 的二步和三步预测中，方差分别提升至 $0.1196(\text{m/s})^2$ 和 $0.2804(\text{m/s})^2$ 。CEEMDAN-IQPSO-BILSTM 在样本 B 的二步和三步预测中，方差分别提升至 $0.2128(\text{m/s})^2$ 和 $0.5543(\text{m/s})^2$ 。而本文的构建的模型在样本 A 和样本 B 二步和三步预测中，方差分别为 $0.0885(\text{m/s})^2$ 、 $0.0761(\text{m/s})^2$ 、 $0.1561(\text{m/s})^2$ 和 $0.1486(\text{m/s})^2$ 。这一结果表明，

随着预测步长的增加，基于二次分解和 IQPSO-BILSTM 的风速组合预测模型仍然具有较高的稳定性。

4.5 本章小结

本章采用内蒙古某风电场 2018 年 7 月和 12 月的实测风速数据对新建风速预测模型的准确性验证和稳定性进行验证。首先，使用 CEEMDAN 对原始风速数据进行了初步的分解，并计算各个模态分量的排列熵。对于排列熵较大的模态分量，采用 VMD 进行进一步的二次分解，通过中心频率确定分解层数。接着，采用滑动窗口方法构建输入向量和输出向量。在对比实验中，采用了多种模型作为对照组，包括 IQPSO-CNN、IQPSO-LSTM、IQPSO-BILSTM、EMD-IQPSO-BILSTM、CEEMDAN-IQPSO-BILSTM 以及 CEEMDAN-VMD-IQPSO-BILSTM。通过 R^2 、RMSE、MAPE 以及方差对各模型的准确性和稳定性进行评估。结果表明，基于二次分解和 IQPSO-BILSTM 的风速组合预测模型具有更高的预测精度和稳定性，有利于促进风电行业的发展。

第 5 章 结论与展望

5.1 研究结论

风能发电不仅能缓解消耗传统化石能源带来的温室效应，还可以促进社会的健康发展。由于风速数据的随机性和不确定性，简单的预测模型并不能准确的进行风速预测。因此，本文以二次分解、深度学习模型以及优化算法为理论基础，构建基于二次分解和 IQPSO-BILSTM 的风速组合预测模型。通过二次分解对风速进行降噪分解，BILSTM 结合 CNN 和 Attention 提取风速特征，IQPSO 算法通过迭代寻优，设置预测模型的超参数。主要工作及结论如下。

(1)采用 CEEMDAN-VMD 对原始风速数据进行分解。CEEMDAN 分解能够有效的解决 EMD 分解产生的模态混叠现象，可以将风速数据分解为多个有规律的子序列，从而降低数据的复杂性。接着，采用 VMD 分解算法对 CEEMDAN 分解后的排列熵较高的子序列进行进一步分解，降低排列熵较高的子序列的复杂度。通过 CEEMDAN 和 VMD 的双重分解，可以产生更有规律的子序列，有利于提高模型的预测精度。

(2)采用CNN、BILSTM以及Attention相结合的预测模型。通过CNN提取风速变化的局部特征。利用BILSTM的输入门、遗忘门和输出门来控制信息的流动，依靠双向结构对风速数据的时序性进行捕捉。通过Attention提取出重要信息，忽略无关信息，提高预测模型的精度和稳定性。

(3) 采用IQPSO算法对CNN-BILSTM-Attention预测模型参数进行优化。首先，采用Tent映射函数初始化QPSO初始种群，使其能够在搜索空间内更加均匀分布和遍历性，避免了粒子过度聚集的可能性。采用基于目标函数的适应值动态变化收缩-膨胀系数，使得QPSO既能在全局搜索时具有较强的探索能力，又能在局部搜索时具有更高的精度和收敛速度。通过柯西变异策略来扰动当前粒子位置，使其跳出局部最优解，进入其它可求解空间，增强算法的全局搜索能力。最后，通过Sphere、Rastrigin、Ackley和Schwefel四个典型函数对IQPSO的全局搜索能力和收敛效率进行测试。通过IQPSO对预测模型的超参数进行优化，CNN-BILSTM-Attention预测性能得到了进一步提升。

(4)利用内蒙古某风电场 2018 年 7 月和 12 月的风速数据进行单步和多步的预测。使用 RMSE、MAE、 R^2 以及方差对模型的准确性和稳定性进行评估。并

以IQPSO-CNN、IQPSO-LSTM、IQPSO-BILSTM等预测模型作为对照组。实验结果表明，本文构建的基于二次分解和IQPSO-BILSTM在具有更高的预测精度和稳定性。

5.2 工作展望

本文构建了基于二次分解和 IQPSO-BILSTM 的风速组合预测模型，通过不同数据集验证了模型的稳定性和准确性。但仍然存在需要完善部分。今后的研究方向和重点为：

(1)CNN-BILSTM-Attention 模型在选择激活函数时，选用的是模型默认的激活函数，然而激活函数的选择会影响模型最终效果。因此在后续的研究中会考虑更多的激活函数。

(2)风速受到除了和历史风速的影响，还受到风向、温度、湿度等其它因素影响，未来需要考虑更多的影响风速的因素。

(3)本文测试风速数据来自内蒙古某风电场，然而我国海上风力资源仍然十分丰富。未来将会对海上风速进行更多的研究。

参考文献

- [1] 田崇翼, 王学睿, 王瑞琪. 基于 CEEMD-SVM 的风速混合预测模型研究 [J]. 计算机时代, 2023, (07): 24-28.
- [2] 王东风, 张博洋, 李青博, 等. 基于 OU 过程和 Vine-Copula 的多风电场短期风速预测[J]. 太阳能学报, 2025, 46(02): 529-538.
- [3] 潘超, 孙惠, 王超, 等. 基于机群空间解耦信息聚合的风速超短期预测[J/OL]. 电网技术, 1-12[2025-03-13].
- [4] 李延满, 张军, 胡杨, 等. VMD 和混合深度学习框架融合的短期风速预测 [J]. 计算机系统应用, 2023, 32 (09): 169-176.
- [5] 伍耘, 葛佳敏, 王文烨, 等. 贝叶斯优化超参数的空时融合压缩残差网络在风速区间预测中的研究[J]. 电力系统保护与控制, 2025, 53(01): 13-23.
- [6] 尹锐, 郑海涛, 李旭涛, 等. 基于异混合分布的分布预测方法及其在风速分布预测中的应用[J]. 自然灾害学报, 2024, 33(06): 143-150.
- [7] 马同旭, 刘帅, 刘卫亮, 等. 基于两阶段多维数据生成与实时健康指数的风机齿轮箱故障预警[J]. 仪器仪表学报, 2024, 45(11): 266-276.
- [8] 傅质馨, 王宝池, 刘皓明, 等. 基于风速-功率组合分解重构的海上风电功率预测[J]. 太阳能学报, 2024, 45(11): 418-426.
- [9] 王胜研. 基于组合算法的风电场风速及风电功率预测研究[D]. 大连交通大学, 2024.
- [10] 张驰. 风电场短期风速预测若干问题研究[D]. 东南大学, 2017.
- [11] 张子晗. 基于风速预测与健康状态评估的风电机组发电量预测研究[D]. 华中科技大学, 2020.
- [12] 刘思凡, 秦华旺. 基于 DSS-UNet 算法的网格化风速预测 [J]. 科学技术与工程, 2024, 24 (26): 11074-11080.
- [13] 张子晗. 基于风速预测与健康状态评估的风电机组发电量预测研究[D]. 华中科技大学, 2020.
- [14] Liu C, Zhang X, Mei S, et al. Numerical weather prediction enhanced wind power forecasting: Rank ensemble and probabilistic fluctuation awareness[J]. Applied Energy, 2022, 313: 118769.
- [15] Kang M, Ko K, Kim M. Verification of the Reliability of Offshore Wind Resource Prediction Using an Atmosphere–Ocean Coupled Model[J]. Energies, 2020, 13(1): 254.
- [16] 黄南天, 唐立鹏, 戴千斌. 计及 NWP 风速横纵向误差的循环确定性策略梯度风速修正模型 [J]. 电力系统保护与控制, 2024, 52 (23): 10-21.
- [17] 刘瑞雪, 李佳轩, 李云. 水库消落带植物多样性空间格局预测模型及环境解释——基于 XGBoost-SHAP 模型框架 [J]. 生态学报, 2024, 44 (21): 9652-9669.
- [18] 宋家康, 彭勇刚, 蔡宏达, 等. 考虑多位置 NWP 和非典型特征的短期风电功率预测研究 [J]. 电网技术, 2018, 42 (10): 3234-3242.
- [19] 马婉贞, 钱育蓉, 范迎迎. 基于 WRF 数值模式的 DBN 风速预测模型研究 [J]. 计算机仿真, 2018, 35 (03): 332-336.
- [20] Tuerxun W, Xu C, Guo H, et al. Fault classification in wind turbine based on deep belief network optimized by modified tuna swarm optimization algorithm[J]. Journal of Renewable and Sustainable Energy, 2022, 14(3).
- [21] 沈永梅, 王琼, 何哲飞. 马氏链点值预测方法和时序分析预测方法的统计试验研究 [J].

- 统计与决策, 2012, (04): 18-21.
- [22] 喻敏, 常毓婵, 袁浩, 等. 基于 HP-EMD 和 ARMA 的短期风速预测 [J]. 中国科技论文, 2016, 11 (05): 566-570.
- [23] 岳勇, 陈雯婷, 聂伟. LMD 和 ARMA 组合风速预测方法 [J]. 中国测试, 2020, 46 (08): 126-130.
- [24] 丁藤, 冯冬涵, 林晓凡, 等. 基于修正后 ARIMA-GARCH 模型的超短期风速预测 [J]. 电网技术, 2017, 41 (06): 1808-1814.
- [25] Mi X, Liu H, Li Y. Wind speed prediction model using singular spectrum analysis, empirical mode decomposition and convolutional support vector machine[J]. Energy conversion and management, 2019, 180: 196-205.
- [26] Zhang Y, Wang P, Ni T, et al. Wind power prediction based on LS-SVM model with error correction[J]. Advances in Electrical and Computer Engineering, 2017, 17(1): 3-8.
- [27] Tao T, Liu Y, Qiao Y, et al. Wind turbine blade icing diagnosis using hybrid features and Stacked-XGBoost algorithm[J]. Renewable Energy, 2021, 180: 1004-1013.
- [28] Chen N, Sun H, Zhang Q, et al. A short-term wind speed forecasting model based on EMD/CEEMD and ARIMA-SVM algorithms[J]. Applied Sciences, 2022, 12(12): 6085.
- [29] Qin B, Huang X, Wang X, et al. Ultra-short-term wind power prediction based on double decomposition and LSSVM[J]. Transactions of the Institute of Measurement and Control, 2023, 45(14): 2627-2636.
- [30] Shi Y. Short-term wind speed forecasting based on a hybrid model that integrates PSO-LSSVM and XGBoost[J]. International Journal of Low-Carbon Technologies, 2024, 19: 1138-1143.
- [31] Fukuoka R, Suzuki H, Kitajima T, et al. Wind speed prediction model using LSTM and 1D-CNN[J]. Journal of Signal Processing, 2018, 22(4): 207-210.
- [32] Zhu T, Wang W, Yu M. Short-term wind speed prediction based on FEEMD-PE-SSA-BP[J]. Environmental Science and Pollution Research, 2022, 29(52): 79288-79305.
- [33] MENG Y Y, CHANG C, HUO J Y, et al. Research on ultra-short term prediction model of wind power based on attention mechanism and CNN-BIGRU combined [J]. Frontiers in Energy Research, 2022.
- [34] 樊雅洁, 王聪, 张宏立, 等. 组合聚类 and 深度学习模型的风电场群风速预测 [J]. 哈尔滨工业大学学报, 2024, 56 (12): 71-80.
- [35] Li Z, Luo X, Liu M, et al. Short-term prediction of the power of a new wind turbine based on IAO-LSTM[J]. Energy Reports, 2022, 8: 9025-9037.
- [36] Tuerxun W, Xu C, Guo H, et al. A wind power forecasting model using LSTM optimized by the modified bald eagle search algorithm[J]. Energies, 2022, 15(6): 2031.
- [37] Jiang Y, Chen X, Yu K, et al. Short-term wind power forecasting using hybrid method based on enhanced boosting algorithm[J/OL]. Journal of Modern Power Systems and Clean Energy, 2017, 5(1): 126-133.
- [38] Wang H, Han S, Liu Y, et al. Sequence transfer correction algorithm for numerical weather prediction wind speed and its application in a wind power forecasting system[J/OL]. Applied Energy, 2019, 237: 1-10.
- [39] Wang J, Zhang W, Li Y, et al. Forecasting wind speed using empirical mode decomposition and Elman neural network[J]. Applied soft computing, 2014, 23: 452-459.
- [40] 鲁铁定, 李祯, 贺小星. 考虑噪声影响的 MEMD-XGBoost 方法在 GNSS 高程时间序列

- 建模和预测中的应用 [J]. 国防科技大学学报, 2024, 46 (06): 149-158.
- [41] 石文娟, 凌凡, 张俊权, 等. 基于 ISSA 优化 EEMD 的用户侧电力负荷预测方法 [J]. 信息技术, 2025, (02): 97-103.
- [42] 郑潜, 乔丹, 郎恂, 等. 基于噪声辅助快速多维经验模式分解的运动想象脑电信号分类方法 [J]. 智能科学与技术学报, 2020, 2 (03): 240-250.
- [43] 马玉坤, 王泳, 梁世锋, 等. 考虑储能的 CEEMDAN-LSTM 的工业园区短期负荷预测 [J]. 电气自动化, 2025, 47 (01): 58-60+63.
- [44] Yu M. Short-term wind speed forecasting based on random forest model combining ensemble empirical mode decomposition and improved harmony search algorithm[J]. International Journal of Green Energy, 2020, 17(5): 332-348.
- [45] Ding J, Chen G, Huang Y, et al. Short-term wind speed prediction based on CEEMDAN-SE-improved PIO-GRNN model[J]. Measurement and Control, 2021, 54(1-2): 73-87.
- [46] 魏来, 谢义超, 喻敏. 基于 IVMD-PSO-LSTM 模型的短期风速预测 [J]. 计算机与数字工程, 2024, 52 (06): 1708-1713.
- [47] 张伟, 李军霞, 陈维望. 基于蝙蝠算法优化 VMD 参数的滚动轴承复合故障分离方法 [J]. 振动与冲击, 2022, 41 (20): 133-141.
- [48] 颜宏文, 邹丹. 基于关联规则的 PSO-Elman 短期风速预测 [J]. 计算机工程与应用, 2017, 53 (23): 261-266.
- [49] 李红, 徐晨燕, 刘恒宇. 基于深度强化学习+模拟退火的地磁感知导航方法研究 [J/OL]. 北京航空航天大学学报, 1-13[2025-01-13].
- [50] 孙波, 董浩, 王璐, 等. 计及电动汽车需求响应的风储混合系统日前联合优化调度模型 [J]. 太阳能学报, 2021, 42 (04): 107-114.
- [51] Gu B, Hu H, Zhao J, et al. Short-term wind power forecasting and uncertainty analysis based on FCM-WOA-ELM-GMM[J]. Energy Reports, 2023, 9: 807-819.
- [52] Cai Y, Hao S, Wen X, et al. Short-term power prediction by using least square support vector machine with variational mode decomposition in a photovoltaic system[J]. IEEE Access, 2023, 11: 143486-143500.
- [53] Jiang P, Qin S, Wu J, et al. Time series analysis and forecasting for wind speeds using support vector regression coupled with artificial intelligent algorithms[J]. Mathematical Problems in Engineering, 2015, 2015(1): 939305.
- [54] Xiang Ling, Liu J, Yang X, et al. Ultra-short term wind power prediction applying a novel model named SATCN-LSTM[J]. Energy Conversion and Management, 2022, 252: 115036.
- [55] Huang J, Niu G, Guan H, et al. Ultra-short-term wind power prediction based on LSTM with loss shrinkage adam[J]. Energies, 2023, 16(9): 3789.
- [56] Wang H, Zhang Y M, Mao J X, et al. A probabilistic approach for short-term prediction of wind gust speed using ensemble learning[J/OL]. Journal of Wind Engineering and Industrial Aerodynamics, 2020, 202: 104198.
- [57] Zhang P, Li C, Peng C, et al. Ultra-short-term prediction of wind power based on error following forget gate-based long short-term memory[J]. Energies, 2020, 13(20): 5400.
- [58] Zhang J, Yang F, Nie G, et al. Optimal Trajectory Planning for Minimizing Base Disturbance of a Redundant Space Robot with IQPSO[J]. Journal of Electrical and Computer Engineering, 2022, 2022(1): 3398810.
- [59] Shi Y, Eberhart R C. Empirical study of particle swarm optimization[C]//Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406). IEEE, 1999, 3:

- 1945-1950.
- [60] Wang L H, Yan Z H, Yang Y T, et al. A classification and prediction hybrid model construction with the IQPSO-SVM algorithm for atrial fibrillation arrhythmia[J]. *Sensors*, 2021, 21(15): 5222.
 - [61] Sun J, Li C, Song Y, et al. Network Security Situation Prediction Based on TCAN-BIGRU Optimized by SSA and IQPSO[J]. *Computer Systems Science & Engineering*, 2023, 47(1).
 - [62] Gai D, Shen X, Chen H, et al. Medical image fusion using the PCNN based on IQPSO in NSST domain[J]. *IET image processing*, 2020, 14(9): 1870-1880.
 - [63] Huang N E, Shen Z, Long S R, et al. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis[J]. *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences*, 1998, 454(1971): 903-995.
 - [64] Liu H, Tian H, Li Y. An EMD-recursive ARIMA method to predict wind speed for railway strong wind warning system[J]. *Journal of Wind Engineering and Industrial Aerodynamics*, 2015, 141: 27-38.
 - [65] Cao J, Li Z, Li J. Financial time series forecasting model based on CEEMDAN and LSTM[J]. *Physica A: Statistical mechanics and its applications*, 2019, 519: 127-139.
 - [66] Karijadi I, Chou S Y. A hybrid RF-LSTM based on CEEMDAN for improving the accuracy of building energy consumption prediction[J]. *Energy and Buildings*, 2022, 259: 11908.
 - [67] Dragomiretskiy K, Zosso D. Variational mode decomposition[J]. *IEEE transactions on signal processing*, 2013, 62(3): 531-544.
 - [68] 王旭光, 张可, 李潇, 等. 基于变量相关注意力机制的短期风速预测 [J]. *太阳能学报*, 2023, 44 (08): 467-476.
 - [69] Kattenborn T, Leitloff J, Schiefer F, et al. Review on Convolutional Neural Networks (CNN) in vegetation remote sensing[J]. *ISPRS journal of photogrammetry and remote sensing*, 2021, 173: 24-49.
 - [70] 张越, 臧海祥, 韩海腾, 等. 基于 FSN-MCCN-SA-BILSTM 的短期风速预测[J]. *太阳能学报*, 2024, 45(08): 529-536.
 - [71] Song D, Yu M, Wang Z, et al. Wind and wave energy prediction using an AT-BILSTM model[J]. *Ocean Engineering*, 2023, 281: 115008.
 - [72] Bashir T, Wang H, Tahir M, et al. Wind and solar power forecasting based on hybrid CNN-ABILSTM, CNN-transformer-MLP models[J]. *Renewable Energy*, 2025, 239: 122055.
 - [73] 刘擎龙, 张宏立, 王聪, 等. 基于序列到序列和注意力机制的超短期风速预测[J]. *太阳能学报*, 2021, 42(09): 286-294.
 - [74] 王岩, 陈耀然, 韩兆龙, 等. 基于互信息理论与递归神经网络的短期风速预测模型[J]. *上海交通大学学报*, 2021, 55(09): 1080-1086.
 - [75] Liang T, Chai C, Sun H, et al. Wind speed prediction based on multi-variable Capsnet-BILSTM-MOHHO for WPCCC[J]. *Energy*, 2022, 250: 123761.
 - [76] Zhang Y M, Wang H. Multi-head attention-based probabilistic CNN-BILSTM for day-ahead wind speed forecasting[J]. *Energy*, 2023, 278: 127865.
 - [77] Zhang D, Chen B, Zhu H, et al. Short-term wind power prediction based on two-layer decomposition and BITCN-BILSTM-attention model[J]. *Energy*, 2023, 285: 128762.

攻读硕士期间取得的学术成果

论文：

EI 论文一作一篇

EI 论文三作一篇

项目：

风力发电场的风能预测与监控研究(主持)

新型电力系统源网荷储协同规划设计技术研发(参与)

竞赛：

安徽省统计建模二等奖

亚太地区数学建模全国二等奖

致谢

行文至此，毕业论文已经基本完成，也意味着我快毕业了。回顾十几年的求学生涯，我遇到很多人，他们给我很多帮助，让我走到这里。

首先，要感谢的是我的父母。我的父母从来不会在学习上给我压力，从来不会干预我的选择，一直默默支持我。

其次，我要感谢我的老师和同学。最先要感谢的是我的导师——孙驷洲老师，感谢他在三年里对我的学习上与生活上的亲切关怀。在学业上一遍又一遍的帮我检查论文，指正不当之处。在生活上，给我介绍实习工作，让我有宝贵的实习经历，在校招中能够帮助我找到心仪的工作。成为孙老师的学生，是我人生道路上重要的财富。我要感谢同窗王胜鹏对我的重要帮助。日常进行学习与研究的时候，时刻给予我科研上的指导和意见。

最后，我要感谢我的女朋友。感谢她能够理解我和关心我。

感谢对本论文进行评审的各位老师，感谢你们在百忙之中抽出时间审阅论文。祝愿你们身体健康，工作顺心！

尚德敏学 唯实惟新

