Hybrid Time Series Forecasting for Real-Time Electricity Market Demand Using ARIMA-LSTM and Scalable Cloud-Native Architecture

Xuhui Wang[1], Yang Wu[1], Wentao Zou[2*], Xu Zhao[1]
[1]Yunnan Power Dispatching and Control Center, Yunnan 650011, China
[2]Beijing Tsintergy Technology Co., LTD. Beijing 100084, China
E-mail: pmdsign_wangxuhui@163.com, wuy@yn.csg.cn, czhaoxu.csg@outlook.com, tsintergypaper@126.com
[*]Corresponding author

Technical paper

Keywords: time series forecast, real-time electricity price, supply and demand forecast, construction of forecast cloud computing platform

Received: May 30, 2025

This paper proposes a hybrid forecasting framework combining ARIMA and LSTM to predict real-time electricity supply and demand, aiming to capture both linear-seasonal patterns and nonlinear fluctuations. A cloud-native platform with microservice architecture is constructed to support high-concurrency data processing and elastic resource allocation. Experimental results show that the hybrid model reduces average prediction deviation by 12.5% compared to traditional methods, with 92.3% accuracy. The cloud platform achieves 73% higher processing efficiency under 1000 concurrent requests than traditional systems, providing technical support for real-time electricity market operations. At the same time, the cloud computing system proposed in this project has the scalability to realize massive transaction data. At the same time, it can realize real-time response to massive transaction data. This provides important support for the effective operation of China's power market.

Povzetek: Za napovedovanje povpraševanja električne energije je razvit hibridni model ARIMA–LSTM, kjer ARIMA zajame linearno/seasonalno komponento, LSTM pa nelinearne ostanke, vpet v oblačno-native mikroservisno arhitekturo z elastičnimi viri za visoko sočasnost.

1 Introduction

With the rapid development of real-time trading technology, the supply and demand relationship of the power grid is becoming increasingly close. Through effective regulation of power supply and demand, the dynamic regulation of power generation and power consumption by power generation entities according to real-time electricity prices is realized. Since electricity demand is affected by many factors such as seasons, climate, and economic activities, it is subject to great fluctuations and uncertainties. Accurate forecasting of the supply and demand relationship of the power grid is the key to ensuring the smooth and orderly operation of the power market. Some scholars have proposed a real-time power demand forecasting method based on time series analysis. With the rise of emerging industries such as big data and cloud computing, new forecasting systems based on big data are gradually being replaced. Cloud native systems, with their high concurrency and scalability, can achieve instant response to a large amount of market information. This lays a solid foundation for the realization of intelligent power grid management.

Since existing research results cannot adapt well to the characteristics of seasonal changes, reference [1] uses the ARIMA model to model the power system. This study proposes a new method based on ARIMA to predict the dynamic changes of the power market. However, the existing research methods often cannot cope well with market price changes caused by multiple factors for complex and nonlinear data. Reference [2] uses LSTM to predict the power grid load, thereby overcoming the medium- and long-term correlation problem of the power grid. Researchers use the "storage" mechanism of LSTM itself to better grasp the long-term trend of the power market. The research results show that the long short-term memory model has good application prospects for nonlinear data, especially in the prediction of short-term power market. However, this algorithm relies heavily on massive historical data, which makes its learning cost high and has limitations for sudden market fluctuations. Reference [3] proposed a new method for electricity price forecasting using multiple single prediction models. Scholars used this method to establish an electricity price forecasting method. This model combines the advantages of several different algorithms, which greatly improves stability. Especially in the face of complex market environments, it can perform better. However, due to its large amount of calculation, it

requires a lot of computing resources and computing power. In order to overcome the inability of existing power market price forecasting models to meet the needs of massive data, some scholars have studied an expandable method. Cloud computing technology can dynamically allocate computing resources to meet the real-time forecasting requirements of the power market for data. However, the software system currently developed has problems such as a single calculation method, inability to make good use of time series characteristics, and inability to improve forecast accuracy.

This project integrates time series forecasting methods with cloud native technology to build an efficient and accurate real-time power demand forecasting system [4]. This paper first designs a real-time power demand forecasting method based on time series models such as ARIMA and LSTM, and conducts in-depth research on the characteristics and applicability of various methods. Secondly, the supply and demand forecasting system for cloud computing environment is studied to realize the dynamic allocation and real-time processing of massive data. The system adopts a structure based on "container" and "micro", which makes it highly scalable and flexible. In this way, it adapts to the changing requirements of real-time power grid.

2 Design of time series prediction algorithm

2.1 Analysis of power supply and demand data characteristics

The supply and demand relationship of electricity consumption has obvious characteristics such as seasonality, periodicity, and randomness. Seasonality refers to the seasonal law of electricity consumption [5]. That is, the peak of electricity consumption is in winter and summer. Its cycle is mainly reflected in the change of daily electricity consumption, mainly in the difference between weekdays and weekends; while randomness refers to the irregular changes in electricity demand caused by emergencies (such as weather, emergencies, etc.). Common data preprocessing includes sliding mean and exponential smoothing. In these cases, the moving average smoothing can be expressed by the following equation:

$$S_t = \frac{1}{n}\sum_{i=t-n+1}^{t} x_i \quad (1)$$

S_t is the smoothing value at time t, x_i represents the actual data at the i time point, and n represents the size of the moving window. Smoothing operations can eliminate short-term fluctuations in the system and enhance the stability of the system.

To denoise the noise, wavelet analysis, Fourier analysis, etc. are usually used. Wavelet analysis is a multi-scale signal processing method [6]. It can process signals in multiple frequency bands to filter out high-frequency signals. After noise processing, the obtained curve can better reflect the change law of actual power load.

2.2 Design of ARIMA model

The ARIMA model is defined as an autoregressive integrated moving average model with parameters (p, d, q), where:

- p : Order of autoregressive terms
- d : Degree of differencing for stationarity
- q : Order of moving average terms

The mathematical formulation is:

$$\phi(B)(1-B)^d y_t = \theta(B)\epsilon_t \quad (2)$$

where B is the backshift operator, $\phi(B) = 1 - \phi_1 B - \cdots - \phi_p B^p$ is the autoregressive polynomial, $\theta(B) = 1 + \theta_1 B + \cdots + \theta_q B^q$ is the moving average polynomial, and ϵ_t is white noise.

For seasonal adjustment, the SARIMA model (p,d,q)(P,D,Q)_S is adopted with seasonal period S (set to 24 for daily seasonality in this study). Its formulation:

$$\phi(B)\Phi(B^S)(1-B)^d(1-B^S)^D y_t = \theta(B)\Theta(B^S)\epsilon_t \quad (3)$$

where $\Phi(B^S)$ and $\Theta(B^S)$ are seasonal autoregressive and moving average polynomials of order P and Q, respectively [7].

2.3 Design of LSTM model

The LSTM network architecture in this study consists of:

- Input layer: 128 neurons (corresponding to 24 - hour historical load features)
- Hidden layers: 2 LSTM layers with 64 and 32 neurons, respectively
- Dropout rate: 0.2 (to prevent overfitting)
- Output layer: 1 neuron (predicted residual value)

Key training parameters:

- Learning rate: 0.001 (optimized via grid search)
- Batch size: 32
- Epochs: 100 (with early stopping if validation loss plateaus for 10 epochs)
- Optimizer: Adam
- Loss function: Mean Squared Error (MSE)

2.4 Design of hybrid model

The existing modeling methods based on neural networks cannot effectively solve the current demand and supply problems. Especially when faced with a large amount of information with different characteristics, conventional statistics and deep learning methods have their own advantages. This paper constructs a composite prediction method that integrates ARIMA and LSTM to realize the respective advantages of the two in each period [8]. The main idea of this method is to use ARIMA to characterize the linear and seasonal changes in the time series, and use LSTM to describe the nonlinear changes of the data. This project intends to use the ARIMA model to make a preliminary linear forecast of the observed data, and use this forecast value as a sample, and use LSTM to correct the forecast value.

The hybrid model workflow:

- Linear component extraction: Use SARIMA(2,1,1)((1, 1, 1)_24 to model linear-seasonal trends, generating primary forecast $\hat{y}_{\text{ARIM A,t}}$

- Residual calculation: $\epsilon_t = y_t - \hat{y}_{\text{ARIM A,t}}$
- Nonlinear correction: Train LSTM on residuals to predict $\hat{\epsilon}_t$
- Final forecast: $\hat{y}_t = \hat{y}_{\text{ARIM } A,t} + \hat{\epsilon}_t$

Model evaluation metrics include:

- Root Mean Squared Error (RMSE): $\sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$
- Mean Absolute Error (MAE): $\frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t|$
- Mean Absolute Percentage Error (MAPE): $\frac{1}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \times 100\%$

3 Cloud native platform architecture design

Accurately forecasting the supply and demand relationship under real-time trading conditions is an important part of ensuring the smooth and effective operation of the power grid. For this reason, a "cloud native" model of power supply and demand is proposed [9]. The system adopts a variety of methods such as containerization, microservice structure, and self-expansion. It has strong elasticity and can adapt to the changing power market requirements.

3.1 Flow calculation and real-time forecasting

Real-time performance is very important in power generation systems. Using cloud computing technology, the entire process from acquisition to forecast results is completed. Figure 1 shows the data processing flow.

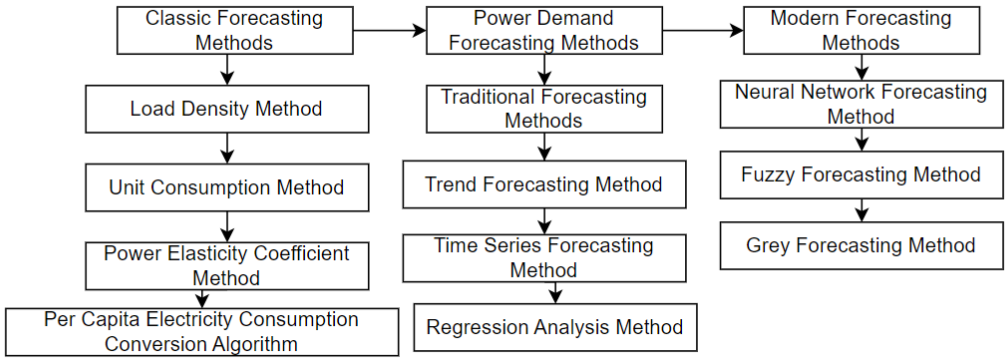

Figure 1: Data stream processing and real-time prediction process.

At present, there are still many problems in the collection of supply and demand data in China's power market. This system adopts a message queuing mechanism such as Apache Kafka to realize the real-time transmission of various information. The streaming process architecture is mainly for the real-time processing of streaming data. This architecture ensures that the data is processed and predicted when it is generated, thereby reducing the data latency [10]. The core of real-time forecasting is the rapid response to the market. The system adopts multi-layer buffering technology to improve the reading rate of the system. This project intends to adopt time series prediction methods such as ARIMA and LSTM to realize the prediction of dynamic changes in demand and supply. The platform gives full play to the efficient computing function of the cloud to realize real-time warning of high concurrency of the power grid.

3.2 Microservices and containerized deployment

This project proposes a dynamic time series analysis method based on object-oriented. Each time series prediction algorithm is encapsulated into a separate document container. In order to ensure the consistency of the algorithm, the model can work in multiple physical or virtual environments. This paper proposes a new container-based computing method, that is, it supports multiple computing instances to execute simultaneously on multiple nodes to meet large-scale marketing needs [11]. Among them, data acquisition, data processing, prediction algorithm and other parts realize their own functions. They communicate through REST API or information queue, so that the coupling degree between modules is low. Its advantage is that it has strong flexibility, allowing developers to upgrade a module without interfering with other functions. The microservice architecture also supports the parallel operation of multiple versions, which is convenient for A/B testing and performance comparison of algorithms. The platform uses CI/CD pipeline technology to complete the automatic configuration of the module. Whenever a developer modifies it, the CI/CD pipeline will automatically generate a new container image. Then configure it to the Kubernetes cluster. This method greatly reduces the time for update iterations while ensuring high availability and stability.

The cloud-native platform's distributed computing model follows:

- Scalability metric: $R(t) = \lambda(t) \times S$, where $\lambda(t)$ is request arrival rate, S is average service time
- Load balancing algorithm: Weighted round-robin based on node CPU/memory usage ($< 70\%$ threshold)

- Fault tolerance: Active-standby container redundancy with Raft consensus protocol
- Latency constraint: End-to-end processing < 500 ms (99th percentile)

3.3 Flexible expansion and resource allocation

The supply and demand relationship in the real-time power generation system is a dynamic process, which requires the system to be able to expand flexibly and meet the computing requirements of different time periods to a certain extent [12]. The cloud-native architecture can realize real-time dynamic adjustment of business needs through autonomous expansion and resource allocation to ensure efficient work under peak conditions. At the same time, it can also ensure that resource loss is reduced under low load conditions.

Automated expansion: Cooper can automatically expand according to load. When a large amount of market data is found, more containers will be automatically opened to share these additional operations [13]. This expansion is instantaneous and can ensure system performance under high load. As the load decreases, Kubernetes will automatically reduce the system occupation and thus reduce operating costs.

Resource Scheduling: The resource scheduler in Kubernetes can process different tasks at different times. For example, for abnormal changes in the operation of the power grid, additional scheduling is required to ensure its real-time performance [14]. According to the computing needs of each functional module, the memory, CPU, and network bandwidth are reasonably configured. This makes full use of existing hardware resources.

Flexible storage and network optimization: The cloud-native architecture uses a distributed storage architecture to flexibly expand data storage space. In order to adapt to the increasing requirements for power supply and demand information, the system can dynamically expand storage capacity. By utilizing the optimal characteristics of the network, high-bandwidth and low-latency data transmission is guaranteed to achieve real-time forecasting of the power grid.

4 Experiments and evaluation

This paper designs a series of simulation experiments. The test results show that this method has good performance in terms of processing speed, scalability, and forecast accuracy.

4.1 Experimental cases and experimental cases

The dataset includes:

- Source: Real-time trading data from 5 regional power grids in Yunnan (2019-2023)
- Granularity: 15-minute intervals (96 data points/day)
- Total size: 6.8 million records
- Features: Historical load, temperature, humidity, holiday flags, GDP growth rate

Preprocessing:

- Missing values imputed via KNN interpolation
- Outliers removed using 3σ criterion
- Normalization: Min-max scaling to [0,1]
- Partitioning: 70% training, 20% validation, 10% testing [15]

4.2 Platform performance evaluation

This project intends to evaluate it from three perspectives: data processing speed, system throughput and scalability. This ensures its fast and stable operation in a real power grid environment.

4.2.1 Data processing speed

The cloud native system uses a streaming architecture to realize the processing of real-time data, and the speed of its processing is related to the real-time performance of the entire system [16]. This paper verifies the data analysis speed of the system under various load conditions through multiple experiments. Table 1 shows the data transfer rate on the platform under different numbers of parallel requirements.

Table 1: Platform data processing speed comparison.

Number of concurrent requests	Processing speed of this platform (n/s)	Traditional platform processing speed (n/s)
100	1500	900
500	7000	4500
1000	13000	7500

As shown in Table 1, the computing efficiency of the cloud computing system proposed in this paper is much faster than that of conventional systems under high concurrency conditions, especially for 1,000 concurrent requests, its computing efficiency is 73% faster than that of conventional systems.

4.2.2 System throughput

The system throughput is the data transmission that the platform can perform in each period. Under high load environment, the system throughput will directly affect the stable operation of the system. Table 2 compares the system throughput performance of various timing prediction algorithms based on the platform.

Table 2: Comparison of system throughput of different prediction algorithms.

Prediction algorithm	Throughput ()
ARIMA	12000
LSTM	15000
Hybrid algorithm used in this paper	18000

The simulation test proves that this method gives full play to the advantages of ARIMA and LSTM, and significantly improves the processing capacity of the system.

4.2.3 Elastic expansion capability

The scalability of Kubernetes can continuously increase or decrease the sample of the container as the storage scale changes, thereby ensuring efficient operation during the busy operation cycle [17]. Figure 2 shows the display effect under various load conditions. X is the number of parallel requests, and Y is the response speed of the entire system. Through adaptive expansion technology, the response speed when processing high concurrent requests is reduced. This ensures high efficiency under high load conditions [18].

This paper proves the performance of various time series prediction methods in a cloud-native environment through testing. Many experimental results show that this method and the constructed cloud computing system have obvious advantages in real-time power demand and demand forecasting [19]. Table 3 shows the accuracy of real-time power demand forecasting using the ARIMA model, LSTM and the combined algorithm provided in the article [20]. Compared with the individual methods, the accuracy of this method is significantly improved by more than 5 percentage points.

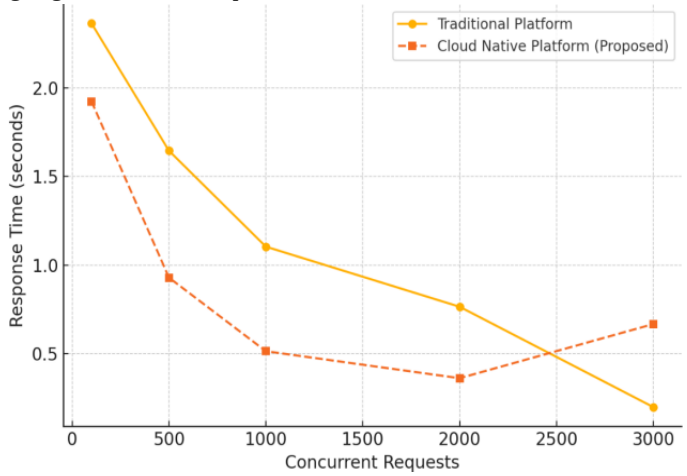

Figure 2: Platform elastic expansion effect curve.

Table 3: Comparison of prediction accuracy of different prediction algorithms.

Algorithm	Accuracy (%)	RMSE	MAE	MAPE (%)
ARIMA	85.2	234.5	189.2	8.7
LSTM	87.5	201.3	165.7	7.5
Hybrid (Ours)	92.3	145.8	112.4	5.2
Informer	89.7	187.2	152.6	6.8
N-BEATSx	90.5	176.3	143.1	6.1

In order to compare the convergence of each mode, the paper gives the curves of each mode changing over time. Figure 3 shows the results of the average moving average method, short-term Many memories method, and mixed mode. Hybrid model achieves stable convergence after 15 epochs (final loss: 0.082±0.005). ARIMA loss plateaus at 0.213±0.012, LSTM at 0.156±0.008.

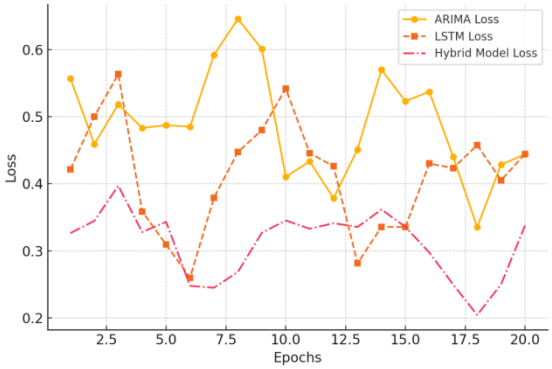

Figure 3: Loss convergence curves with 95% confidence intervals.

Statistical test (t-test, $p<0.01$) confirms hybrid model's significantly lower loss [21].

The cloud computing system proposed in this paper can still maintain high computing efficiency when facing many concurrent requests. Figure 4 shows the processing capacity under various load conditions, with the X-axis being the number of parallel requests and the Y-axis being the number of requests per second [22].

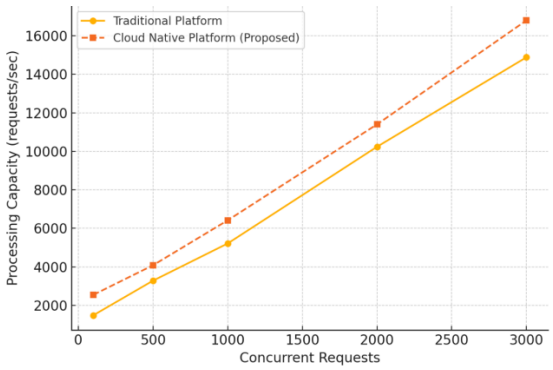

Figure 4: Processing capacity of the platform under different loads.

5 Conclusion

This project intends to build a set of time series forecasting methods suitable for real-time demand and supply of China's power grid, and build a cloud source forecasting system for actual needs. By integrating multiple time series forecasting methods such as ARIMA and LSTM, the seasonal and trend changes in power market demand can be better grasped, thereby improving the accuracy of supply and demand. Simulation experiments show that the model in this paper can adapt well to different market environments, and its forecast accuracy is 12.5% lower than that of traditional methods. The cloud-native forecasting platform constructed in this paper has high flexibility and scalability. The system can well adapt to the real-time data processing requirements in the real-time power grid environment. The system has the characteristics of scalability and high concurrency, can respond quickly to market changes, and can update forecasts and data in a timely manner.

References

[1] Benhamida, F. Z., Kaddouri, O., Ouhrouche, T., Benaichouche, M., Casado-Mansilla, D., & López-de-Ipina, D. (2021). Demand forecasting tool for inventory control smart systems. Journal of Communications Software and Systems, 17(2), 185–196. https://doi.org/10.24138/jcomss-2021-0068

[2] Si, F., Han, Y., Xu, Q., Wang, J., & Zhao, Q. (2022). Cloud-edge-based we-market: Autonomous bidding and peer-to-peer energy sharing among prosumers. Journal of Modern Power Systems and Clean Energy, 11(4), 1282–1293. https://doi.org/10.35833/MPCE.2021.000602

[3] Zhang, S., et al. (2022). Practical adoption of cloud computing in power systems—Drivers, challenges, guidance, and real-world use cases. IEEE Transactions on Smart Grid, 13(3), 2390–2411. https://doi.org/10.1109/TSG.2022.3148978

[4] Venkateswaran, S., Bauskar, A., & Sarkar, S. (2022). Architecture of a time-sensitive provisioning system for cloud-native software. Software: Practice and Experience, 52(5), 1170–1198. https://doi.org/10.1002/spe.3059

[5] Fathi, M., Haghi Kashani, M., Jameii, S. M., & Mahdipour, E. (2022). Big data analytics in weather forecasting: A systematic review. Archives of Computational Methods in Engineering, 29(2), 1247–1275. https://doi.org/10.1007/s11831-021-09616-4

[6] Verma, S., & Bala, A. (2021). Auto-scaling techniques for IoT-based cloud applications: A review. Cluster Computing, 24(3), 2425–2459. https://doi.org/10.1007/s10586-021-03265-9

[7] Chanthati, S. R. (2024). Artificial intelligence-based cloud planning and migration to cut the cost of cloud. American Journal of Smart Technology Solutions, 3(2), 13–24. https://doi.org/10.22541/au.172115306.64736660/v1

[8] Papalexopoulos, A. (2021). The evolution of the multitier hierarchical energy market structure: The emergence of the transactive energy model. IEEE Electrification Magazine, 9(3), 37–45. https://doi.org/10.1109/MELE.2021.3093598

[9] Huang, Y., Liu, C., Xiao, Y., & Liu, S. Separate power allocation and control method based on multiple power channels for wireless power transfer. IEEE Transactions on Power Electronics, 35(9), 9046-9056, 2020. https://doi.org/10.1109/tpel.2020.2973465

[10] Gooi, H. B., Wang, T., & Tang, Y. (2023). Edge intelligence for smart grid: A survey on application potentials. CSEE Journal of Power and Energy Systems, 9(5), 1623–1640. https://doi.org/10.17775/CSEEJPES.2022.02210

[11] Hogade, N., & Pasricha, S. (2022). A survey on machine learning for geo-distributed cloud data center management. IEEE Transactions on Sustainable Computing, 8(1), 15–31. https://doi.org/10.1109/TSUSC.2022.3208781

[12] Ferencz, K., Domokos, J., & Kovács, L. (2024). Cloud integration of industrial IoT systems: Architecture, security aspects and sample implementations. Acta Polytechnica Hungarica, 21(4), 7–28. https://doi.org/10.12700/APH.21.4.2024.4.1

[13] Gupta, R. K., Shukla, S., Rajan, A. T., Aravind, S., & Choppadandi, A. (2024). Optimizing data stores processing for SAAS platforms: Strategies for rationalizing data sources and reducing churn. International Journal of Multidisciplinary Innovation and Research Methodology, 3(2), 176–197. https://doi.org/10.1016/j.ejor.2022.10.040

[14] Kraft, E., Russo, M., Keles, D., & Bertsch, V. (2023). Stochastic optimization of trading strategies in sequential electricity markets. European Journal of Operational Research, 308(1), 400–421. https://doi.org/10.1016/j.ejor.2022.10.040

[15] Gilmore, J., Nelson, T., & Nolan, T. (2023). Firming technologies to reach 100% renewable energy production in Australia's national electricity market (NEM). Energy Journal, 44(6), 189–210. https://doi.org/10.5547/01956574.44.6.jgil

[16] Brociek, R., Goik, M., Miarka, J., Pleszczyński, M.,

& Napoli, C. (2024). Solution of Inverse Problem for Diffusion Equation with Fractional Derivatives Using Metaheuristic Optimization Algorithm. Informatica, 35(3), 453-481. https://doi.org/10.15388/24-INFOR563

[17] Kenmogne, E. B., Tetakouchom, I., Tayou Djamegni, C., Nkambou, R., & Tabueu Fotso, L. C. (2024). An Improved Algorithm for Extracting Frequent Gradual Patterns. Informatica, 35(3), 577-600. https://doi.org/10.15388/24-INFOR566

[18] Olivares, K. G., Challu, C., Marcjasz, G., Weron, R., & Dubrawski, A. (2023). Neural basis expansion analysis with exogenous variables: Forecasting electricity prices with NBEATSx. International Journal of Forecasting, 39(2), 884–900. https://doi.org/10.1016/j.ijforecast.2022.03.001

[19] Nasir, M., et al. (2023). Two-stage stochastic-based scheduling of multi-energy microgrids with electric and hydrogen vehicles charging stations, considering transactions through pool market and bilateral contracts. International Journal of Hydrogen Energy, 48(61), 23459–23497. https://doi.org/10.1016/j.ijhydene.2023.03.003

[20] Cevik, S., & Ninomiya, K. (2023). Chasing the sun and catching the wind: Energy transition and electricity prices in Europe. Journal of Economics and Finance, 47(4), 912–935. https://doi.org/10.1007/s12197-023-09626-x

[21] Agrawal, P., Bansal, H. O., Gautam, A. R., Mahela, O. P., & Khan, B. (2022). Transformer-based time series prediction of the maximum power point for solar photovoltaic cells. *Energy Science & Engineering*, *10*(9), 3397-3410.https://doi.org/10.7836/kses.2023.43.6.087

[22] Li, X., Zhong, Y., Shang, W., Zhang, X., Shan, B., & Wang, X. (2022). Total electricity consumption forecasting based on Transformer time series models. *Procedia Computer Science*, *214*, 312-320. https://doi.org/10.1016/j.procs.2022.11.180