

Large Language Model Assisted Interpretability in Graph Convolution Network for AGC Dispatch

Xiaoshun Zhang, *Member, IEEE*, Kun Zhang, Zhengxun Guo, and Penggen Wang

Abstract—Automatic generation control (AGC) dispatch is the key task of secondary frequency regulation for interconnected grids. To generate a high-quality dispatch solution, numerous machine learning techniques, such as reinforcement learning and graph convolutional networks (GCNs), have been developed for AGC dispatch. However, they are challenging to apply to a real-world power grid due to their weak interpretability. Hence, this work proposes a novel approach to large language model (LLM)-assisted interpretability in GCN for AGC dispatch. Firstly, the impact of input features (e.g., the total regulation command and the regulation capacities of various resources) on dispatch solutions is assessed quantitatively using Shapley additive explanations (SHAP) for global interpretability. Then, local interpretability for GCN is achieved using an LLM-assisted, model-agnostic local interpretable model-agnostic explanations (LIME), which can provide actionable insights into the model's decision logic. SHAP shows that the top eight features drive decisions, while the rest average just 15.8% of the leading feature's contribution. Unit outputs correlate positively with their own history and negatively with others. Swapping LIME's linear model for a decision tree boosts multiple metrics by over 50%. Experimental results further confirm that this method not only clearly uncovers the relationships between input features and AGC dispatch outputs, but also faithfully reconstructs the GCN's decision logic across different dispatch scenarios.

Index Terms—AGC dispatch, graph convolution network, interpretability, large language model.

I. INTRODUCTION

AUTOMARIC generation control (AGC) is an essential tool in power systems, designed to regulate grid frequency and manage tie-line power flows through real-time adjustments of generation and other resources [1]. However, the evolving composition of the power grid has introduced several complexities. The integration of diverse frequency regulation resources, including renewable energy, electric vehicles, and energy storage systems [2], has significantly increased the dimensionality of AGC operations, making coordination more challenging. Renewable energy sources, such as wind [3]

and solar [4], exhibit inherent intermittency and randomness, leading to frequent and unpredictable fluctuations in supply. Meanwhile, the variability of electric vehicle charging behavior [5] and the operational constraints of energy storage systems further contribute to uncertainty. Together, these factors result in a highly stochastic and dynamic system that poses substantial challenges for AGC dispatch. The need to address these interconnected issues has made the development of advanced, robust AGC strategies a critical focus for modern power systems.

Given these challenges, researchers and engineers have explored a range of strategies to enhance AGC dispatch performance. These strategies can be broadly categorized into three types: traditional engineering-based methods, meta-heuristic algorithms, and machine learning (ML) algorithms. Each method offers unique advantages and addresses different aspects of the dynamic and stochastic demands of modern power grids. In addition, Table I summarizes the metrics of these three types of methods.

Traditional engineering-based methods, centered on proportional allocation, dispatch priority, and command decomposition, have long been adopted thanks to their simplicity and speed [2]. While these methods establish a foundational framework for AGC dispatch, they struggle with the complexity and high dimensionality introduced by renewable energy integration and energy storage systems. In such settings, meta-heuristic algorithms have emerged as powerful alternatives [1], effectively addressing optimization challenges beyond the reach of conventional methods. However, despite their strong performance in high-dimensional and stochastic environments, many meta-heuristic algorithms require extensive computations and iterative operations, resulting in longer solution times that make them less suitable for real-time AGC applications [6]. To effectively enhance the efficiency of AGC dispatch, machine learning algorithms have recently been distinguished. Nevertheless, their “black-box” nature raises significant interpretability challenges [7].

In complex power grid scenarios, multi-dimensional inputs—such as renewable energy output, load fluctuations, total power regulation commands, and historical data across different dimensions—collectively influence dispatch results. Current methods often fail to quantify the contribution of individual variables, complicating strategic adjustments and anomaly handling [8]. Furthermore, most deep learning and deep reinforcement learning approaches employ intricate neural network architectures, thereby obscuring the internal computational logic and decision-making processes [9], thus un-

Manuscript received March 27, 2025; revised June 3, 2025; Accepted July 18, 2025. Date of online publication March 3, 2026; date of current version March 9, 2026. This work was supported by the National Natural Science Foundation of China (No. 52577087) and the Guangdong Basic and Applied Basic Research Foundation (No. 2024A1515030012).

X. S. Zhang, K. Zhang, Z. X. Guo (corresponding author, email: 2310276@stu.neu.edu.cn), and P. G. Wang are with the Foshan Graduate School of Innovation, Northeastern University, Foshan 528311, China and College of Information Science and Engineering, Northeastern University, Shenyang 110819, China.

DOI: 10.17775/CSEEJPES.2025.02260

TABLE I
SUMMARY OF THE THREE TYPES OF METHODS

Method type	Specific algorithm	Advantages	Disadvantages	Interpretability
Engineering	Command decomposition method [10], [11] Capacity-based method [12] Sensitivity-based approach [13]	• Simplicity • High speed • Low computational cost	• Poor adaptability • Fixed strategies • Limited scalability	High
Meta-heuristic	NSGA II [14], PSO [15], GA [16]	• Global search capability • High-dimensional data processing	• High computation time • Population dependency	Moderate
Machine learning	DRL [17]–[20], CLAC [21], Q-Learning [22], [23], DDNN [24], EDL [25], HCEQ [26]	• Fast online inference • High-dimensional data processing	• Black-box nature • Offline training dependency	Low

dermining trust among power dispatch personnel.

To address this issue, researchers have begun exploring the model-agnostic explanation methods. Among these, Shapley additive explanations (SHAP) based on game-theoretic Shapley values [27] and local interpretable model-agnostic explanations (LIME) [28] driven by local surrogate models have emerged as focal points in preliminary investigations due to their methodological universality. SHAP enables global quantification of feature contributions [29], revealing the aggregate weight of input variables on dispatch decisions. Meanwhile, LIME employs locally linear decode-control command-generation logic under specific operational scenarios. The synergistic complementarity between these two approaches provides a methodological foundation for tackling the challenge of real-time decision transparency in AGC applications.

Nevertheless, existing explanation tools demand dual expertise in both power system operations and ML model mechanics to decipher their outputs. Furthermore, without prior engineering expertise, dispatchers may struggle to connect these explanations to actionable grid-control insights. Recent advances in large-scale pre-trained foundation models (e.g., ChatGPT, DeepSeek) have demonstrated unprecedented capabilities in contextual reasoning and domain-specific knowledge integration, enabled by multi-billion parameter architectures trained on cross-disciplinary corpora [30].

This study proposes a decision-interpretation framework that integrates explainable artificial intelligence (XAI) techniques with an LLM. It achieves both global and local analytical dimensions using SHAP and LIME, respectively. The underlying model is a graph convolutional network (GCN), which excels at capturing spatial-temporal dependencies among interconnected generation units in AGC dispatch. Due to its strong performance but limited transparency, GCN is chosen as the target for interpretability enhancement using hybrid XAI methods. The principal innovations are listed as follows:

- 1) Hybrid XAI-driven dual-perspective explanatory framework: A novel global-local interpretability architecture is proposed, including SHAP-based global feature contribution analysis and LIME-driven local decision pathway decoding.
- 2) LLM-enhanced semantic bridging mechanism: GPT is leveraged to transform raw XAI outputs into actionable grid operation insights for constructing a domain-aware knowledge translation system. This system effectively bridges the semantic gap between statistical XAI outputs and domain-specific decision logic, enabling dispatchers to directly derive actionable insights from model interpretability.

This paper is organized as follows: Section II introduces a GCN-based framework for AGC dispatch, outlining its architecture and operational mechanisms. Section III explains the LLM-assisted interpretability methodology for GCN in AGC dispatch, emphasizing its innovative interpretability enhancements. Section IV analyzes GCN input contributions, inter-variable correlations, and node-level decision-making logic. Section V concludes with contributions and future research directions.

II. GCN FOR AGC DISPATCH

A. Application Framework

AGC plays a critical role in ensuring accurate frequency regulation and maintaining system stability. As shown in Fig. 1, the framework integrates key components, including the independent system operator (ISO), proportional-integral (PI) controllers, and diverse energy resources.

In practical operation, the ISO collects real-time frequency deviation (Δf) and tie-line power deviation (ΔP_T) of the control area through sensors. These parameters, influenced by generator dispatch errors, load fluctuations, system failures, and system topology changes, serve as critical inputs for the PI controller. Then, the PI controller generates a total regulation command to correct imbalances. Furthermore, the total regulatory command is allocated to power generation units by GCN, such as coal-fired, storage, hydrogen, wind, and solar units. Its response process can be characterized as the relationship between the regulatory power input and the resulting time-domain regulatory power output, incorporating inherent system dynamics such as response time delay and energy transfer characteristics [1]. The actual output of an AGC unit can be computed through a combination of frequency-domain analysis and inverse Laplace transformation, which are commonly used methods in dynamic system modeling. These methods are described in detail in the literature [31].

B. Optimization Model of AGC Dispatch

The objective of the AGC dispatch problem is to optimize the dynamic response performance of the control area by minimizing the total power deviation between the AGC command and the actual power output of all participating generation units [24]. This ensures system stability while considering operational constraints.

i) Objective function: The optimization objective can be formulated as:

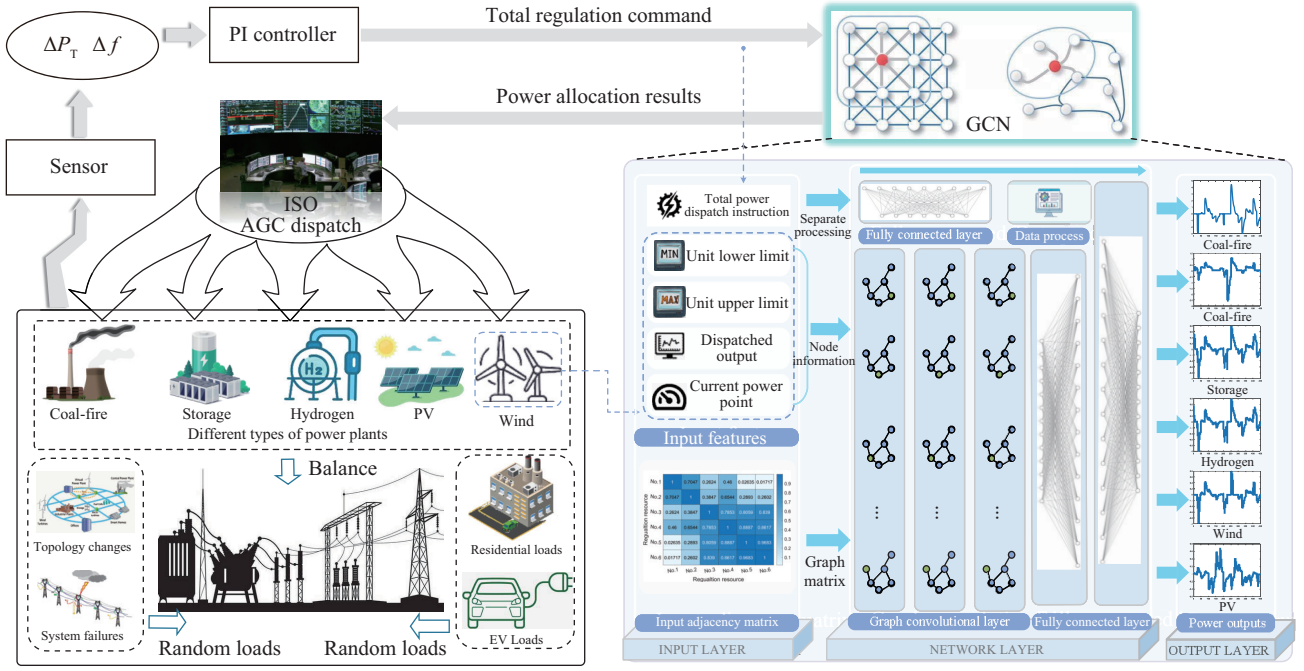


Fig. 1. Control framework of AGC dispatch with high-penetration renewable energy.

$$\text{Minimize } f = \sum_{k=1}^N \left| \Delta P^C(k) - \sum_{i=1}^n \Delta P_{\text{out}}^i(k) \right| \quad (1)$$

where $\Delta P^C(k)$ represents the total regulation power command at time k ; $P_{\text{out}}^i(k)$ denotes the actual power output of the i -th AGC unit at time k ; N is the total number of control intervals in the dispatch horizon; n is the number of AGC units.

ii) *Constraints:* To ensure feasible and reliable operation, the following constraints are incorporated.

The power input from all AGC units must satisfy the total regulation power command at each time interval:

$$\sum_{i=1}^n \Delta P_{\text{in}}^i(k) = \Delta P^C(k), \quad \forall k \quad (2)$$

Each AGC unit's power output must operate within its designated upper and lower limits:

$$\Delta P_{\text{min}}^i \leq \Delta P_{\text{out}}^i(k) \leq \Delta P_{\text{max}}^i, \quad \forall i, k \quad (3)$$

where ΔP_{min}^i and ΔP_{max}^i denote the minimum and maximum allowable power outputs for the i -th unit.

The rate of change of power output for each unit must adhere to its ramp rate limitations:

$$-R^i \leq \frac{\Delta P_{\text{out}}^i(k) - \Delta P_{\text{out}}^i(k-1)}{\Delta T} \leq R^i, \quad \forall i, k \quad (4)$$

where R^i denotes the maximum ramp rate of the i -th unit.

The direction of each unit's regulation must align with the sign of the total AGC command:

$$\Delta P_{\text{in}}^i(k) \cdot \Delta P^C(k) \geq 0, \quad \forall i, k \quad (5)$$

C. Graph Convolution Network for AGC Dispatch

The GCN-based AGC dispatch framework is designed to enhance the precision and efficiency of AGC in power systems with high renewable energy penetration [31]. As shown in Fig. 1, the procedure begins by extracting input features, including the unit's upper and lower limits, the dispatched output from the previous timestep, and the current power point. In total, the input feature vector comprises 125 dimensions, including the global dispatch command and its four historical values (5 features), unit-level constraints (upper/lower limits and their four-step histories for six units, totaling 60 features), historical outputs for each unit over five steps (30 features), and cumulative outputs (integral values) over five steps for each unit (30 features). These inputs are preprocessed alongside an adjacency matrix representing the dynamic coupling relationships among generating units. The GCN outputs optimized dispatch commands, which are subsequently corrected to ensure feasible outputs that align with physical and operational constraints.

The GCN architecture models the interaction between generating units as a graph. Each node in the graph corresponds to a generating unit, and edges represent their dynamic relationships. The adjacency matrix and feature matrix jointly drive the GCN layers, where non-linear transformations (e.g., ReLU activations) are applied to extract meaningful latent representations of system states. These layers iteratively process node-level information, enabling the model to learn the spatial dependencies of power system units. The final layer aggregates these features to provide a total power dispatch command that minimizes system errors and improves stability.

The propagation rule for the GCN is given by

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} + b^{(l)}) \quad (6)$$

where $H^{(l)}$ represents the node features at the layer l ; $W^{(l)}$ and $b^{(l)}$ are trainable parameters; and σ is activation function.

III. LLM ASSISTED INTERPRETABILITY IN GCN FOR AGC DISPATCH

A. Interpretable Method Based on SHAP and LLM

SHAP has emerged as a powerful tool for enhancing the interpretability of GCN-based AGC dispatch frameworks [32]. As shown in Fig. 2, the main steps of SHAP explanation are as follows: First, the original data and background data are fed into the model for black-box prediction. The model processes the input data and provides a prediction, while the specific internal computations remain opaque.

Next, the base value, i.e., the average prediction value of the model in the absence of specific feature influences, can be calculated based on the background data, as follows:

$$y_{\text{base}} = \frac{1}{M} \sum_{m=1}^M GCN(\mathbf{X}_m) \quad (7)$$

where y_{base} denotes the baseline prediction value of GCN; M represents the total number of reference samples selected from the background dataset; $\mathbf{X}_m$ corresponds to the feature matrix of the m -th background sample with n feature dimensions, and $GCN(\cdot)$ symbolizes GCN that maps input graph data to the model's output space.

The computation of the base value serves as the starting point of the SHAP explanation, representing the model's default output when specific features are not considered.

In the SHAP computation (explainer) stage, the Shapley value formula is used to measure each feature's contribution.

Specifically, the marginal contribution of each feature in different subsets is calculated as:

$$\phi_i(GCN) = \sum_{S \subseteq F \setminus \{i\}} w_i [GCN(S \cup \{i\}) - GCN(S)] \quad (8)$$

where S represents a feature subset; $GCN(S)$ denotes the prediction result using only the feature subset S ; i indicates the index of the feature whose contribution is being evaluated; and w_i is computed by:

$$w_i = \frac{|S|!(|F| - |S| - 1)!}{|F|!} \quad (9)$$

This process determines each feature's impact on the final prediction result by computing its marginal contribution across different combinations.

Once the Shapley values are computed, the specific contribution of each feature can be obtained for model interpretation. SHAP possesses several excellent properties, including efficiency, symmetry, linearity, and consistency. These properties ensure the fairness and reliability of the SHAP method, providing strong theoretical support for interpreting complex models.

Based on these Shapley values, the framework uses the PandasAI library to query SHAP data through natural language. This avoids overloading the LLM with large or high-dimensional data, which could cause hallucinations or exceed its context limits. Natural language queries also allow dispatchers to flexibly access the specific interpretability information they need without predefined code. With this data, the LLM can produce visualization code to help dispatchers better understand and analyze the GCN's scheduling process.

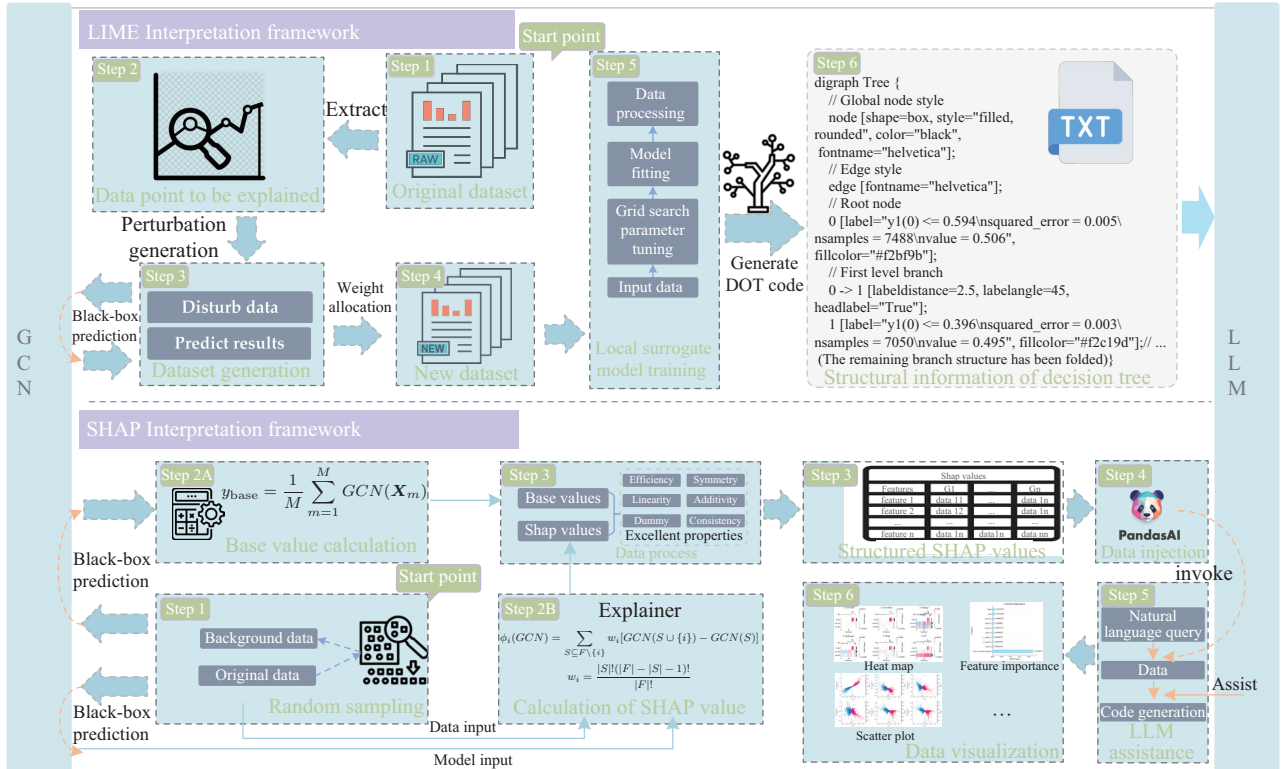


Fig. 2. The framework of LLM assisted hybrid XAI-driven dual-perspective explanatory.

formalized as:

$$T = \arg \min_{T \in \mathcal{T}} \sum_{i=1}^N \pi_x(x'_i) [f(x'_i) - T(x'_i)]^2 + \lambda \cdot \Omega(T) \quad (10)$$

here, $L(f, T, \pi_x)$ denotes the objective function to be minimized, as follows:

$$L(f, T, \pi_x) = \sum_{i=1}^N \pi_x(x'_i) [f(x'_i) - T(x'_i)]^2 + \lambda \cdot \Omega(T) \quad (11)$$

The weighting kernel function $\pi_x(x'_i)$ is defined as:

$$\pi_x(x'_i) = \exp \left(-\frac{D(x, x'_i)^2}{\sigma^2} \right) \quad (12)$$

where T denotes the trained decision tree model; $\mathcal{T}$ represents the set of all possible decision trees; $\pi_x(x'_i)$ is the weighting kernel function, which emphasizes perturbation samples that are closer to the sample x ; λ is the regularization parameter, used to control the complexity of the tree (e.g., maximum depth, minimum samples for split).

iv) *LLM verification and logical consistency testing*: To verify the accuracy of the explanations generated by the LLM, two levels of validation are conducted, as illustrated in Fig. 3:

Logical consistency testing: We design structured consistency checks to assess whether the LLM accurately understands and interprets the decision tree structure. Specifically, the decision-tree model is exported in graph format (GV), transformed into structured text, and then processed by the LLM. The LLM's interpretation is evaluated through predefined test questions based on decision paths, splitting conditions, and leaf outputs. Only when all consistency checks are passed do we proceed with further semantic summarization.

Validation against electrical engineering constraints: Beyond structural consistency, we introduce a dedicated validation module to ensure the LLM's explanations respect core power-system operational constraints. The validation process comprises two steps:

- **Automated rule-based matching**: LLM-extracted rules are cross-referenced against a curated constraint rule base that includes hard constraints (e.g., ramp rate limits, output bounds, AGC command sign constraints) and expert knowledge (e.g., use of thermal units for large disturbances).
- **Manual expert verification**: Key rule matches are reviewed by power system engineers to confirm alignment between LLM explanations and domain-specific practice of the tree structure.

C. Integration of SHAP and LIME

This work integrates SHAP and LIME to achieve complementary interpretability for GCN-based AGC dispatch.

- **SHAP** evaluates the average contribution of each input feature across the entire dataset, offering a global view of feature importance. This helps identify which factors consistently influence the model's dispatch strategy.
- **LIME**, in contrast, focuses on individual samples, generating interpretable surrogate models to approximate the original model's behavior in a local neighborhood. This

allows us to trace the reasoning behind specific dispatch decisions, such as why a certain unit was prioritized under a given operating condition.

By combining both methods, we obtain a dual-perspective interpretability framework: SHAP provides overall feature attribution, while LIME reveals fine-grained decision logic at the instance level. This design ensures both strategic insight and operational transparency, enhancing the trustworthiness and usability of GCN in real-world AGC systems.

D. The Application of Interpretable Mechanism in the Dispatching System

In practical AGC dispatching systems, as shown in Fig. 4, real-time feature data—such as total regulation commands, resource capacities, and historical load profiles—are collected and fed into the pretrained GCN model for generating dispatch decisions. However, due to the model's black-box nature, dispatch operators often lack visibility into how these decisions are made, which poses risks to trust and operational safety.

To address this issue, a hybrid interpretability framework combining SHAP, LIME, and a LLM is deployed at the control center. As illustrated in Fig. 4, the permutation explainer estimates feature contributions by randomly permuting the inputs, allowing dispatchers to quickly identify key factors; the exact explainer computes Shapley values via full-subset evaluation, enabling precise optimization of the GCN model. SHAP provides global feature importance to reveal which inputs most influence the model's decisions, while LIME generates local decision paths for specific scenarios. The LLM further transforms these outputs into natural language explanations and interactive visualizations, enabling dispatchers to intuitively understand, verify, and interact with the model's reasoning.

Beyond interpretability, this mechanism also supports model correction and optimization. When abnormal feature attributions or illogical decision paths are identified—such as non-critical inputs being assigned high importance or slow-response units being activated in low-demand conditions—these insights can be fed back into the model for refinement. Redundant features can be removed to improve computational efficiency, and targeted data augmentation or retraining can be applied to problematic regions. This creates a human-in-the-loop feedback cycle that enables continuous model improvement and safer deployment in real-world power systems.

IV. CASE STUDIES

In this study, a two-area load frequency control (LFC) model consisting of six units is employed for performance evaluation and interpretable analysis. Reference [31] elaborates on the settings of time delays, ramp rates, and transfer function parameters for each unit, and presents specific parameters of the adjacency matrix. The configuration parameters of the GCN model are shown in Table II. To evaluate the fitting performance and decision-making logic of the GCN model, two types of test signals are introduced, i.e., a random signal and a step signal. For each signal type, the reliability of the GCN's decision-making process is examined from both global

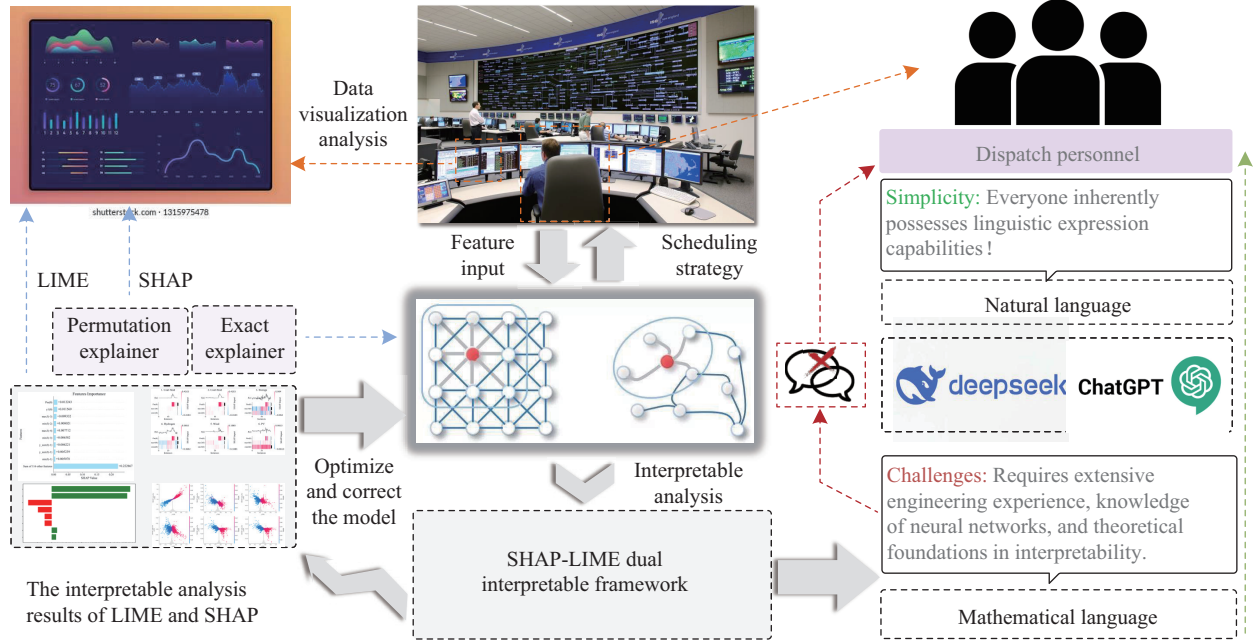


Fig. 4. The application of interpretable mechanism in the actual dispatching system.

TABLE II
THE BASIC PARAMETERS OF GCN TRAINING

Parameters	Values
Neurons of graph convolution layer	[16], [32], [32]
Neurons of fully connected layers	layer ₁ = 32; layer ₂ = 1120
Mini-batch size	64
Learning rate	0.001
Number of epochs	300

and local perspectives. The explainable results will reveal the GCN model's underlying logic in AGC dispatch, thereby enhancing dispatchers' trust in the GCN model.

A. Off-Line Network Training

To comprehensively evaluate the performance of the proposed method, this study adds a comparison with the traditional decision tree method. The results in Table III show that GCN achieves the best performance in both absolute mean error and maximum error across the ACE, Δf , and ΔP metrics. In particular, GCN outperforms decision tree and PROP in error control and robustness, demonstrating its effectiveness and application value in power system state estimation.

GCN trained on a GA-optimized LFC model converged after 300 epochs. To capture the GA's control characteristics, we applied random disturbances every 100 seconds over a 24-hour simulation, generating over 20,000 samples. Each sample includes 125 input features representing the system state Fig. 5 shows test performance under disturbances, comparing GCN and GA via power dispatch and ACE metrics. The results show that the mean absolute error (MAE) is 3.5838, the root mean square error (RMSE) is 5.3695, the coefficient of determination (R^2) reaches 0.9651, and the mean absolute

TABLE III
COMPARISON OF DIFFERENT TYPES OF FITTING ALGORITHMS

Metrics	Algorithm	AbsMean	Max	Min	AbsMax
ACE	PROP	22.063	94.621	-78.016	94.621
	Decision tree	21.622	96.279	-78.929	96.279
	GCN	19.174	85.457	-71.175	85.457
Δf	PROP	0.1059	0.5000	-0.4163	0.5000
	Decision tree	0.1038	0.5118	-0.4178	0.5118
	GCN	0.0924	0.4607	-0.3886	0.4607
ΔP	PROP	9.7331	44.0770	-36.3870	44.0770
	Decision tree	9.5383	44.6720	-36.9120	44.6720
	GCN	8.4930	39.4980	-32.8200	39.4980

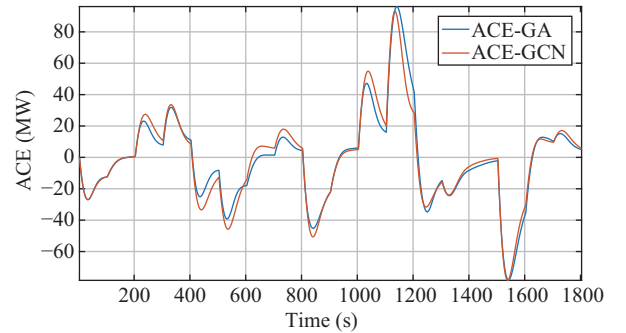


Fig. 5. Simulation results for the AGC system under random signal disturbances.

percentage error (MAPE) is 6.75%. These results quantitatively confirm that GCN effectively approximates the behavior of the GA-based control strategy. While numerical alignment exists, interpretability analyses are required to confirm GCN's decision-making logic aligns with GA principles.

B. Interpretable Analysis Under Step Disturbance

Figure 6(a) illustrates the output of different generation units over a 0–560 s period under varying step signal amplitudes. It can be observed that all units see a rapid initial increase in output following the step signal trigger, gradually stabilizing to a steady-state condition. To compare the response of conventional and renewable units, the coal-fired generator (G_1) and wind generator (G_5) are examined. The coal-fired unit, with a larger capacity per unit, shows a significant increase in peak output as the disturbance amplitude rises, ranging from 22 MW to 82 MW. In contrast, the wind unit's response to the same step disturbance (2 MW to 82 MW) is considerably smaller, with a maximum output of only around 0.2 MW. When the disturbance amplitude is large (e.g., AM=82 MW), the wind unit's output remains near zero due to limited available wind energy, indicating its inability to support

such a substantial sudden load increase.

Figure 6(b) shows the impact of the total dispatch instruction P_m on the unit output distribution over a time span of 100 control cycles, with a step amplitude of 50 MW. In the figure, red represents a positive contribution, and blue represents a negative contribution. The darker the color, the stronger the positive or negative impact. The topmost curve represents the synthesis of the SHAP values of the selected three features, that is, the total contribution. The three selected features are the total dispatching instructions, the upper capacity limit and the lower capacity limit corresponding to each unit, respectively.

By comparing the output response curves of the six units, we can clearly observe the influence of the unit's capacity constraints on the model output. Specifically, the output contributions of each unit show significant differences. For example, two coal-fired units exhibit a noticeable increase

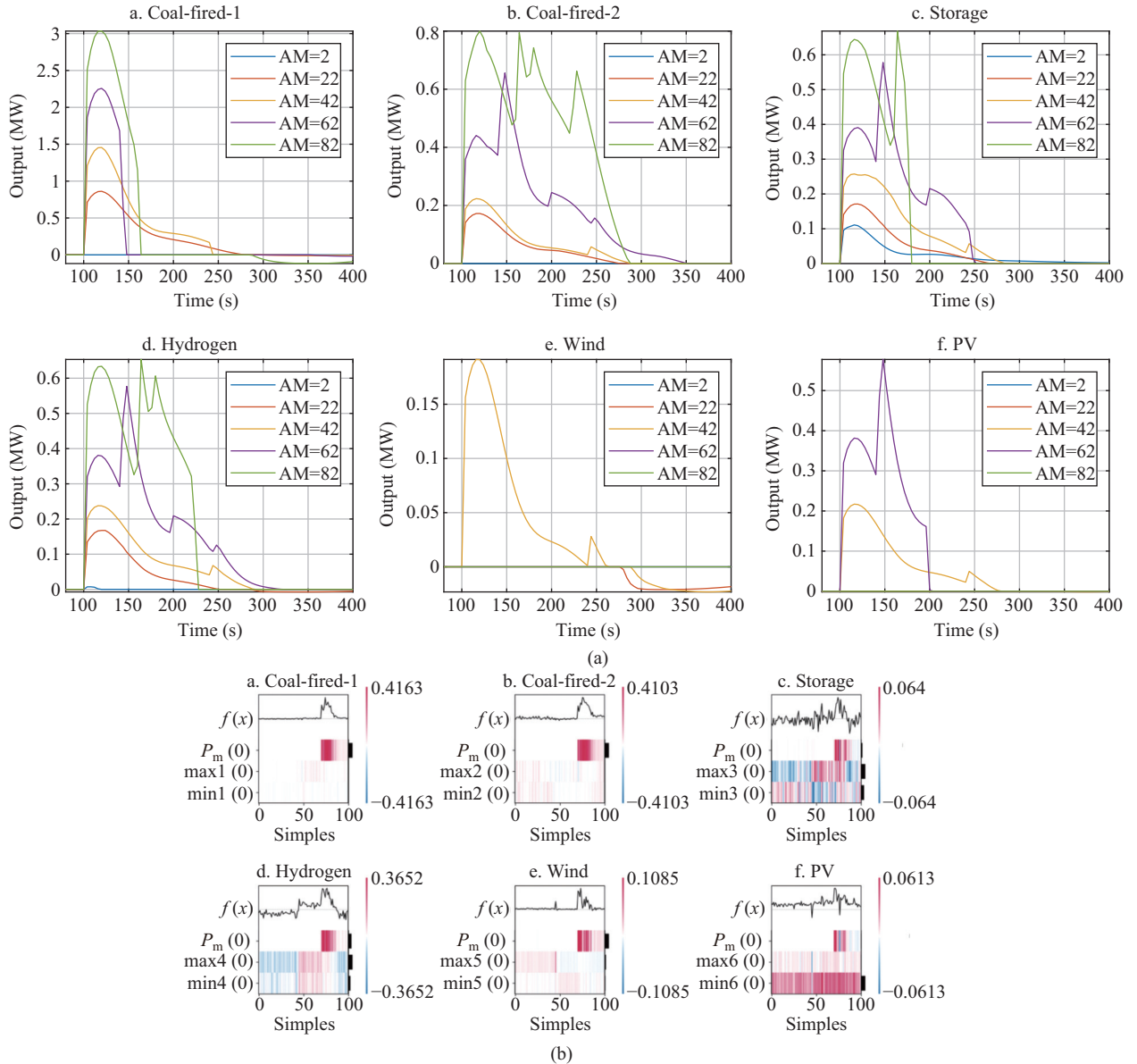


Fig. 6. The interpretable analysis of unit output curves under step response. (a) Output of six units for 80 control cycles at different step amplitudes. (b) Heatmap of input features for unit output under step response.

and fluctuation around the 70th control cycle. The AGC total dispatch command has the most significant supporting effect on the output of these coal-fired units, with minimal influence from the capacity constraints. In contrast, the hydrogen units show varying degrees of influence from the three inputs between control cycles 45 and 100. This suggests that the GCN dispatching strategy first calls on hydrogen units, which have faster response speeds but lower capacity, to balance system fluctuations. Later, coal-fired units, which have slower response speeds but larger adjustable capacities, are called to further support system stability. Unlike the coal-fired and hydrogen units, the output of the energy storage, wind, and photovoltaic units is very small, especially for the energy storage units. Its maximum output is 0.064, less than 1/6 of the maximum output of the coal-fired units. From the figure, we can observe that the capacity constraints of the energy storage unit have a significant impact on its output. This is related to the fast response speed of energy storage units, but their limited capacity. Their role is to balance small system fluctuations in real time.

The heatmap distribution indicates that the GCN is highly dependent on the total dispatch instruction when allocating the output of various units. Based on the characteristics of the units and their capacity constraints, it performs differentiated dispatch scheduling for thermal power, energy storage, hydrogen, wind, and photovoltaic power at different time periods. This strategy allows the strengths and weaknesses of different units to complement each other. It not only aligns with engineering experience and intuition but also coincides with the “response speed-weighted allocation” idea found in some traditional methods, such as command decomposition.

To evaluate the prediction performance of different surrogate models within the LIME framework, we conducted 5-fold cross-validation on three representative models: ridge, lasso, and decision tree. Table IV summarizes their MAE, root RMSE, MAPE, and coefficient of R^2 . The results demonstrate that, compared to traditional linear models (lasso and ridge), the decision tree surrogate significantly outperforms in all metrics, especially MAE and R^2 . This indicates its greater suitability as a local surrogate model in AGC dispatch interpretability tasks. Notably, the decision tree model achieves an average R^2 of 0.73, substantially higher than -0.004 for Lasso and 0.27 for Ridge, highlighting its superior capability in capturing local nonlinear patterns.

TABLE IV
COMPARISON OF DIFFERENT LIME LOCAL LINEAR MODELS

Model type	MAE ($\times 10^{-3}$)	RMSE ($\times 10^{-3}$)	MAPE ($\times 10^{-3}$)	R^2
Ridge	0.324	0.937	8.262	0.270
Lasso	0.299	1.109	7.396	-0.004
Decision tree	0.077	0.415	1.888	0.734

To verify the reliability of LLM-based interpretations, we designed six logic consistency questions derived from decision tree structures. As shown in Fig. 7, ChatGPT-o3 answered all questions correctly, demonstrating robust and repeatable understanding across tests. Deepseek-R1 followed closely with

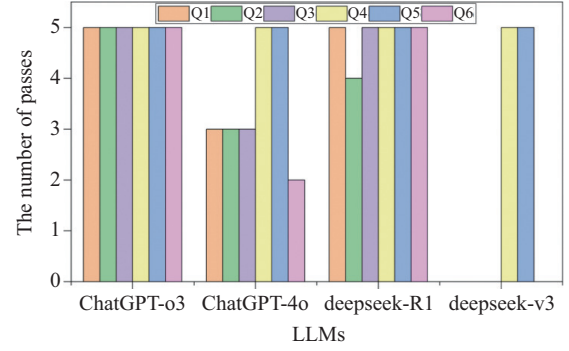


Fig. 7. The number of times different LLMs passed the consistency check under five tests.

five passes, while ChatGPT-4o and deepseek-v3 showed limited reliability. These results indicate that certain mainstream LLMs, especially ChatGPT-o3, have already acquired strong and stable capabilities in parsing and analyzing structured decision logic, supporting their integration into interpretability workflows in power system applications.

From the perspective of a single sample, we perform a local analysis of the specific decision-making logic governing unit 1's output. We extract the 75th sample point from this dataset and analyze it using LIME in conjunction with an LLM. Specifically, the decision tree's GV file is converted into a TXT format and then fed into ChatGPT. Subsequently, we tested GPT with the consistency-check questions from Fig. 2 and compared its results with those generated by the Python program. Only after confirming that GPT had passed all tests did we proceed to further summarize the decision logic.

Table V shows the dispatch patterns and dispatching principles distilled from the large language model's analysis.

C. Interpretable Analysis Under Random Disturbance

Figure 8 analyzes the contribution of the total dispatch command P_m to the output power of unit G_1 under step and random disturbances. Each point in the plot represents a single sample. The x -axis corresponds to the normalized total dispatch command, $P_m(0)$, at the current time—so its values range from 0 to 1—while the y -axis shows that sample's SHAP value, i.e. its contribution to the output (for example, the output of coal-fired Unit 1). Point's color encodes the value of a historical feature ($P_m(-1)$, $P_m(-2)$, $P_m(-3)$ or $P_m(-4)$), as indicated by the color bar on the right: redder hues denote a stronger influence on the current output, bluer hues a weaker one. Along the x -axis, the gray hatched bars depict the distribution of sample counts across equal-width bins—taller bars indicate more samples in that interval—and the hatched filling ensures these bars remain distinguishable in grayscale or monochrome printing. The scatter plot reveals a significantly different distribution compared to the step disturbance case, forming an hourglass-shaped pattern centered around (0, 0.5). This distribution arises because random disturbances fluctuate dynamically both upward and downward. The results show a generally positive correlation between P_m and the contribution to unit G_1 's output power. As the magnitude of P_m increases, its influence on the unit's power output becomes more promi-

TABLE V
LIME LOCAL EXPLANATION ASSISTED BY LLM

Item	Answers from LLM
Decision pathway	The primary factors include dispatching commands P_m, unit capacity limits, and each unit's historical output and cumulative values (y, y_{sum}). At each decision node, scenarios such as “high or low system demand”, “operational flexibility of the unit”, and “status of renewable generation/storage” are distinguished based on whether these features exceed predefined thresholds.
Splitting criteria	If past instructions are relatively low (left subtree), and no extremely high loads or abrupt changes have been observed recently, the predicted output for unit 1 typically falls within a moderate range (approximately 0.57 to 0.60). Should other units exhibit significant ramp-ups in the near term, unit 1's output may be slightly reduced; conversely, it may be marginally increased. If past instructions are considered ably high (right subtree), the predicted output for unit 1 fluctuates between 0.50 and 0.62 , contingent upon whether there is still a need for high load and whether other units are operating at maximum generation capacity.
Dispatching logic	This decision tree encapsulates the concept of “ overall system demand versus available operational capacity of the units. ” It simultaneously assesses whether the trend of dispatching instructions is increasing and whether the distribution among units has reached saturation or remains surplus. Historical output and cumulative values indicate “which units are already heavily generating and which can still increase their output”, thereby determining the subsequent adjustment of unit 1's output.
Conclusion	From an AGC dispatching perspective, the regression tree initially differentiates high and low dispatching demands based on historical P_m. It then examines capacity constraints, historical output, and cumulative conditions across units. If other units exhibit high renewable or storage outputs, there is no need for a significant increase in unit 1's output. Conversely, if demand is high and other units have not contributed substantially, unit 1's output is allowed to continue increasing. The entire logic employs multiple “threshold comparisons” to segment scenarios into distinct leaf nodes, each associated with a regression prediction value for “unit 1's output.” This partitioning approach minimizes the mean squared error on the training data and enables the model to make reasonably accurate estimations of unit 1's output under new operating conditions.

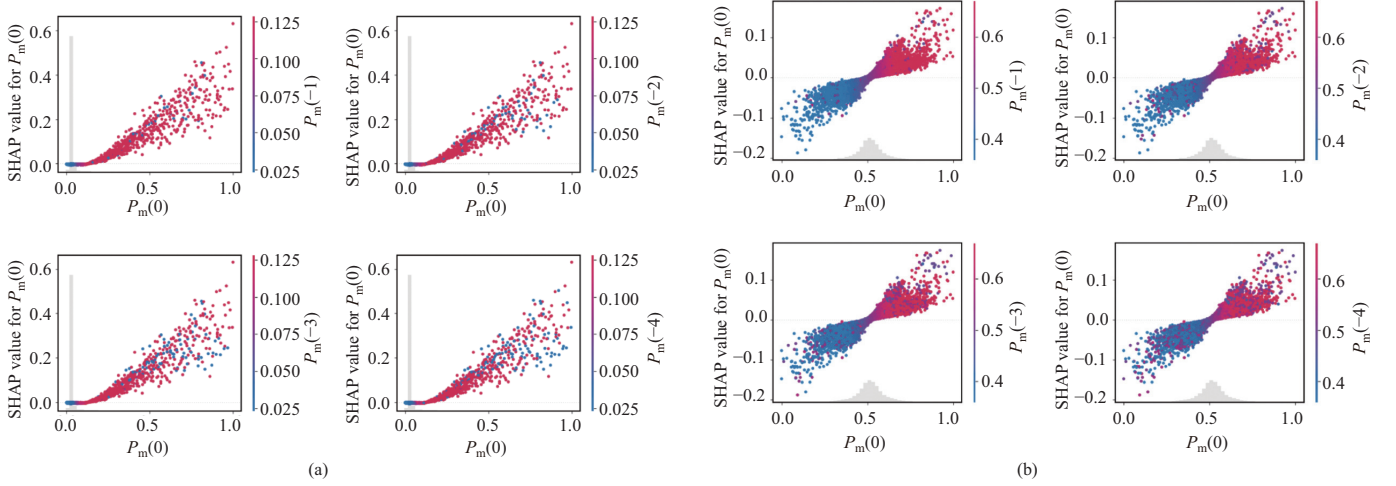


Fig. 8. The autocorrelation of the total dispatching command. (a) Step disturbance. (b) Random disturbance.

ment. When P_m lies in the range of 0 to 0.5, its contribution to the model's output tends to be negative. Additionally, the color gradient indicates that historical values $G_1 y_i(0)$ of P_m have a relatively minor impact on recent outputs when $P_m < 0.5$. However, when P_m exceeds 0.5, historical values contribute more significantly to the SHAP value, implying a stronger influence on current decisions. Furthermore, the plot demonstrates that older historical P_m commands progressively exert weaker influence on the contribution of the most recent P_m value to the model's output, highlighting a temporal decay in the relevance of past dispatch commands.

Figure 9(a) shows the top eight features affecting the power output of the coal-fired unit under random disturbances. Among these, the total dispatch command is the most critical, indicating that the model's decision-making relies on real-time dispatch requirements. Secondly, the unit's previous cycle output reflects a focus on its current state, which helps balance the continuous output and regulation capacity of base-load units. Additionally, the cumulative contribution of the remaining 116 (accounting for 92.8% of the total feature set) features is significant, with their total contribution value being

approximately 19 times that of the most important feature, P_m . However, the average contribution of these 116 features is only about 15.8% of P_m . The significant disparity underscores the need for optimized feature selection to remove redundancy and improve both model interpretability and computational efficiency.

Figure 9(b) examines the self-correlation of the latest historical outputs ($y_i(0)$) for six units and the impact of the total dispatch command P_m on their SHAP values. The results clearly show a positive correlation only for unit G_1 , where higher latest historical output ($y_1(0)$) correlates with increased future output. In contrast, the other units exhibit a slight negative correlation. This negative correlation suggests that under random disturbances, if the historical output of units other than unit G_1 is high, it may indicate a condition where further output from the coal-fired unit is unnecessary, potentially explaining the negative impact on the model's contribution. This observation aligns with the operational characteristics of coal-fired units and their slower ramping capabilities compared to fast-response units.

The SHAP analysis reveals that GCN effectively captures

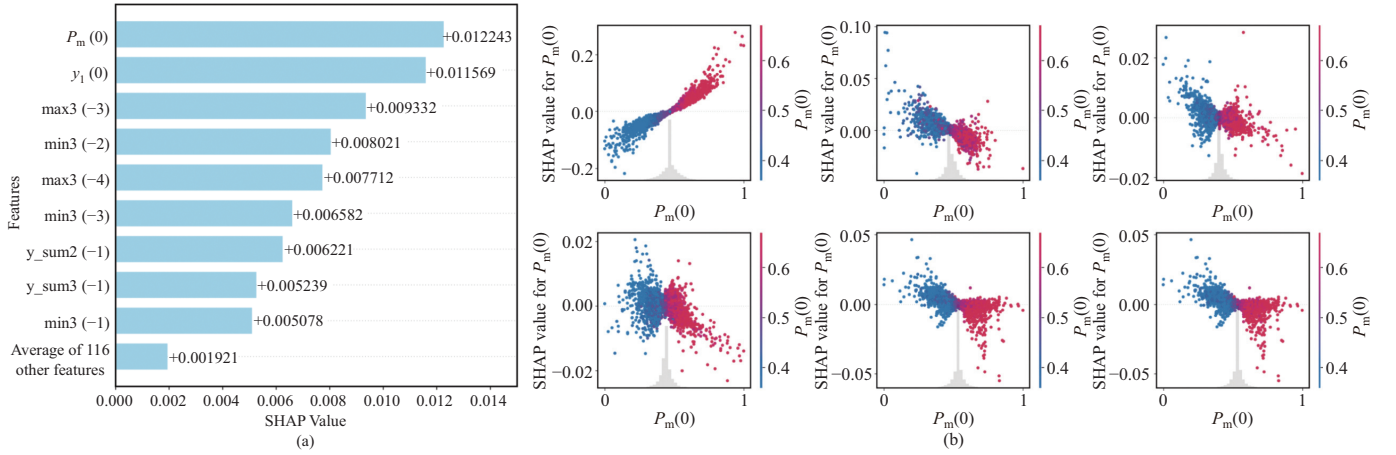


Fig. 9. The SHAP interpretable analysis of unit G_1 coal-fired generator under random disturbance. (a) Feature importance map. (b) The autocorrelation of the historical output $y_i(0)$ of six units.

the complementary relationships among generation units and the importance of total dispatch command P_m in determining unit outputs. The model prioritizes controllable units such as coal-fired generators for grid stability while dynamically adjusting wind and PV contributions based on real-time conditions. This behavior aligns with AGC principles and highlights the model's ability to generalize patterns from historical data.

V. CONCLUSION

This paper builds a dual-dimensional interpretability framework for LLM-assisted smart grid AGC strategies using SHAP and LIME methods. It analyzes the different impact mechanisms of total dispatch commands and unit capacity constraints. LLM technology is also used to refine the logic of dispatch strategies.

Globally, SHAP unveils actionable insights for AGC optimisation: in step disturbances the GCN policy allocates power adaptively by unit characteristics (Fig. 6(b)), while under random disturbances 92.8 % of features account for only 15.8 % of total importance. Feature-interaction analysis exposes a time-decay effect—each thermal unit's current output is positively tied only to its own previous output and negatively to other unit types.

Locally, an LLM-LIME framework—validated by consistency and electrical-constraint checks—recovers GCN decision paths and distills a single guiding rule: balance overall system demand against each unit's available operational capacity.

Although the framework already provides operator-friendly transparency, its practical use is still limited by SHAP's combinatorial cost and occasional LLM hallucinations; future work will therefore explore faster SHAP approximations and more reliable LLMs. In addition, we plan to explore lightweight fine-tuning methods (e.g., LoRA, Adapter) to enhance the domain alignment and stylistic consistency of LLM-generated explanations, while ensuring deployment efficiency and data security.

REFERENCES

- [1] P. Zhao, Z. Li, X. Bai, J. Su, and X. Chang, "Stochastic real-time dispatch considering AGC and electric-gas dynamic interaction: Fine-grained modeling and noniterative decentralized solutions," *Applied Energy*, vol. 375, p. 123976, Dec. 2024.
- [2] M. X. Xiang, Z. F. Yang, and J. Yu, "Hierarchical AGC dispatch with detailed modeling of energy storage system behaviors," *IEEE Transactions on Power Systems*, vol. 38, no. 2, pp. 1689–1701, Mar. 2023.
- [3] H. L. Ma, X. J. Luo, N. W. Gong, and Q. Wang, "A robust interval economic dispatch method accommodating significant wind power integration," in *2018 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, Kota Kinabalu, Malaysia, 2018, pp. 549–554.
- [4] J. Bian, Y. H. Song, C. Ding, J. N. Cheng, S. Li, and G. Q. Li, "Optimal bidding strategy for PV and BESSs in joint energy and frequency regulation markets considering carbon reduction benefits," *Journal of Modern Power Systems and Clean Energy*, vol. 12, no. 2, pp. 427–439, Mar. 2024.
- [5] J. N. Wang, F. X. Li, X. Fang, W. B. Wang, H. T. Cui, Q. W. Zhang, and B. X. She, "Electric vehicles charging time constrained deliverable provision of secondary frequency regulation," *IEEE Transactions on Smart Grid*, vol. 15, no. 4, pp. 3892–3903, Jul. 2024.
- [6] M. Kuang, Y. Tian, Q. Liu, M. Zhao, and X. Wu, "A review of control strategies for automatic generation control in power systems with renewable energy," *Progress in Energy*, vol. 6, no. 2, p. 022001, Feb. 2024.
- [7] Y. Zhang, P. Tiño, A. Leonardis, and K. Tang, "A survey on neural network interpretability," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 5, pp. 726–742, Oct. 2021.
- [8] S. Chakraborty, R. Tomsett, R. Raghavendra, D. Harborne, M. Alzantot, F. Cerutti, M. Srivastava, A. Preece, S. Julier, R. M. Rao, T. D. Kelley, D. Braines, M. Sensoy, C. J. Willis, and P. Gurrum, "Interpretability of deep learning models: a survey of results," in *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI)*, San Francisco, CA, USA, 2017, pp. 1–6.
- [9] N. H. Liu, Q. Z. Feng, and X. Hu, "Interpretability in graph neural networks," in *Graph Neural Networks: Foundations, Frontiers, and Applications*, L. F. Wu, P. Cui, J. Pei, and L. Zhao, Eds. Singapore: Springer Nature, 2022, pp. 121–147.
- [10] Y. V. Makarov, P. W. Du, M. C. W. Kintner-Meyer, C. L. Jin, and H. F. Illian, "Sizing energy storage to accommodate high penetration of variable energy resources," *IEEE Transactions on Sustainable Energy*, vol. 3, no. 1, pp. 34–40, Jan. 2012.
- [11] B. L. Cheng and W. B. Powell, "Co-optimizing battery storage for the frequency regulation and energy arbitrage using multi-scale dynamic programming," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1997–2005, May 2018.
- [12] Y. Z. Cheng, M. Tabrizi, M. Sahni, A. Povedano, and D. Nichols, "Dynamic available AGC based approach for enhancing utility scale

- energy storage performance,” *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 1070–1078, Mar. 2014.
- [13] D. Orihara and D. Iio, “Sensitivity-based AGC dispatching for mitigation of short-term power flow variation,” in *2018 IEEE Electronic Power Grid (eGrid)*, Charleston, SC, USA, 2018, pp. 1–6.
- [14] X. S. Zhang, T. Tan, T. Yu, B. Yang, and X. M. Huang, “Bi-objective optimization of real-time AGC dispatch in performance-based frequency regulation market,” *CSEE Journal of Power and Energy Systems*, vol. 10, no. 6, pp. 2360–2370, Nov. 2024.
- [15] Y. Zou, G. Q. Yu, K. M. Luo, H. X. Wang, M. L. Guo, and J. Y. Zheng, “Improved particle swarm optimization-based thermal power-energy storage combined AGC frequency regulation control,” in *2020 IEEE 3rd International Conference of Safe Production and Informatization (IICSPI)*, Chongqing City, China, 2020, pp. 569–574.
- [16] Z. W. Wen and B. Chen, “Load dispatch optimization of AGC system based on improved genetic algorithm,” in *2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, Ningbo, China, 2017, pp. 278–281.
- [17] L. Xi, L. P. Zhou, L. Liu, D. L. Duan, Y. C. Xu, L. Q. Yang, and S. Wang, “A deep reinforcement learning algorithm for the power order optimization allocation of AGC in interconnected power grids,” *CSEE Journal of Power and Energy Systems*, vol. 6, no. 3, pp. 712–723, Sep. 2020.
- [18] J. W. Li, T. Yu, H. X. Zhu, F. S. Li, D. Lin, and Z. H. Li, “Multi-agent deep reinforcement learning for sectional AGC dispatch,” *IEEE Access*, vol. 8, pp. 158067–158081, Aug. 2020.
- [19] J. W. Li and T. Yu, “Virtual generation alliance automatic generation control based on deep reinforcement learning,” *IEEE Access*, vol. 8, pp. 182204–182217, Oct. 2020.
- [20] J. W. Li and T. Yu, “Deep reinforcement learning based multi-objective integrated automatic generation control for multiple continuous power disturbances,” *IEEE Access*, vol. 8, pp. 156839–156850, Aug. 2020.
- [21] L. H. Xie, J. N. Wu, Y. Y. Li, Q. Y. Sun, and L. Xi, “Automatic generation control strategy for integrated energy system based on ubiquitous power internet of things,” *IEEE Internet of Things Journal*, vol. 10, no. 9, pp. 7645–7654, May 2023.
- [22] X. S. Zhang, Q. Li, T. Yu, and B. Yang, “Consensus transfer Q-learning for decentralized generation command dispatch based on virtual generation tribe,” *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 2152–2165, May 2018.
- [23] X. Y. Zhao, G. Yao, N. J. Tian, J. Dai, and H. Y. Su, “Regional power grid AGC control strategy research based on Q-Learning algorithm under WACPB mode,” in *2019 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)*, Chengdu, China, 2019, pp. 105–109.
- [24] X. S. Zhang, C. Z. Li, B. Xu, Z. N. Pan, and T. Yu, “Dropout deep neural network assisted transfer learning for bi-objective pareto AGC dispatch,” *IEEE Transactions on Power Systems*, vol. 38, no. 2, pp. 1432–1444, Mar. 2023.
- [25] L. F. Yin, Q. Gao, L. L. Zhao, and T. Wang, “Expandable deep learning for real-time economic generation dispatch and control of three-state energies based future smart grids,” *Energy*, vol. 191, pp. 116561, Jan. 2020.
- [26] T. Yu, X. S. Zhang, B. Zhou, and K. W. Chan, “Hierarchical correlated Q-learning for multi-layer optimal generation command dispatch,” *International Journal of Electrical Power & Energy Systems*, vol. 78, pp. 1–12, Jun. 2016.
- [27] D. Napolitano, L. Vaiani, and L. Cagliero, “Efficient neural network-based estimation of interval shapley values,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 12, pp. 8108–8119, Dec. 2024.
- [28] M. Yang, C. Y. Xu, Y. Y. Bai, M. M. Ma, and X. Su, “Investigating black-box model for wind power forecasting using local interpretable model-agnostic explanations algorithm,” *CSEE Journal of Power and Energy Systems*, vol. 11, no. 1, pp. 227–242, Jan. 2025.
- [29] K. Zhang, J. Zhang, P. D. Xu, T. L. Gao, and D. W. Gao, “Explainable AI in deep reinforcement learning models for power system emergency control,” *IEEE Transactions on Computational Social Systems*, vol. 9, no. 2, pp. 419–427, Apr. 2022.
- [30] Z. R. K. Rostam, S. Sznsi, and G. Kertsz, “Achieving peak performance for large language models: a systematic review,” *IEEE Access*, vol. 12, pp. 96017–96050, Jul. 2024.
- [31] D. Liu, Y. Z. Wu, Y. Q. Kang, C. Z. Li, X. T. Ji, and X. S. Zhang, “Graph convolution network for real-time AGC dispatch with high-penetration renewable energy,” *CSEE Journal of Power and Energy Systems*, doi: 10.17775/CSEEJPES.2022.07640.
- [32] S. M. Lundberg and S. I. Lee, “A unified approach to interpreting model predictions,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2017, pp. 4768–4777.
- [33] M. T. Ribeiro, S. Singh, and C. Guestrin, “‘Why Should I Trust You?’: explaining the predictions of any classifier,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Long Beach, California, USA, 2016, pp. 1135–1144.



Xiaoshun Zhang received the B.Eng. and Ph.D. degrees in Electrical Engineering from South China University of Technology, Guangzhou, China, in 2012 and 2017 respectively. Now, he is an Associate Professor in Foshan Graduate School of Innovation, Northeastern University, Foshan, China. His major research interests include artificial intelligence and evolutionary computation in operation of power systems and renewable energy.



Kun Zhang received the B.Eng. degree in Automation from Hebei University, Baoding, China, in 2021. He is currently a graduate student at Foshan Graduate School of Innovation, Northeastern University, Foshan, China. His major research interests include automatic generation control and interpretable machine learning.



Zhongxun Guo received the B.Eng. and M.Eng. degrees in Electrical Engineering from Kunming University of Science and Technology, Kunming, China, in 2020 and 2023, respectively. He is currently a Ph.D. student at the Foshan Graduate School of Innovation, Northeastern University, Foshan, China. His research interests include carbon emission reduction of distribution networks, dispatch and control of distribution networks, development and application of intelligent algorithms, and system optimization of renewable energy.



Penggen Wang graduated from the Department of Automation, Anhui Normal University in 2023, earning a Bachelor of Engineering degree. He is currently a master's graduate student at the School of Information Science and Engineering, Northeastern University. His main research interests include low-carbon scheduling and optimization of multi-microgrids and automatic generation control.