DiffFlex: A Diffusion-Based Model for Medium-Term Load Forecasting Over Flexible Intervals

Jia Ding, Ce Shi, Junyang Jin, and Zidong Wang, *Fellow, IEEE*

***Abstract*—Accurate medium-term load forecasting (MTLF) is critical for capacity planning, maintenance scheduling, and intelligent resource allocation in modern power systems. However, existing approaches are typically designed for fixed-horizon, contiguous prediction and often suffer from multistep error accumulation, limited spatiotemporal modeling capacity, and insufficient uncertainty quantification. This article proposes DiffFlex, a diffusion-based probabilistic forecasting framework that reformulates MTLF as a temporal outpainting problem. By decoupling forecasting from temporal contiguity, DiffFlex enables direct parallel prediction over arbitrary, noncontiguous future intervals without cumulative error propagation. The framework integrates fused historical blocks to capture cyclical consumption patterns, models spatiotemporal-interval dependencies across multivariate load series, and employs an interval-aware diffusion module to generate full predictive distributions. Experiments on the Independent System Operator New England, New York Independent System Operator, and European Network of Transmission System Operators for Electricity datasets across 1- to 12-week horizons show that DiffFlex outperforms state-of-the-art baselines in both point and probabilistic accuracy, achieving average mean absolute percentage errors of 8.88%, 9.71%, and 4.86%, respectively. These results indicate that DiffFlex can serve as a practical tool for scenario-based planning and risk-informed decision-making in power system operations.**

***Index Terms*—Diffusion models, medium-term load forecasting, noncontiguous prediction, probabilistic forecasting, temporal outpainting.**

Received 4 May 2026; revised 12 June 2026; accepted 4 July 2026. This work was supported by the National Natural Science Foundation of China under Grant 52205519 and Grant 62203182. Paper no. TII-26-4046. *(Corresponding author: Ce Shi.)*

Jia Ding is with the School of Economics and Management, East China Normal University, Shanghai 200062, China (e-mail: jding@fem.ecnu.edu.cn).

Ce Shi is with Yuanshi Intelligent Technology (Nantong) Company Ltd. Nantong 226009, China (e-mail: shice@yuanshi-autonomy.com).

Junyang Jin is with Yuanshi Intelligent Technology (Nantong) Company Ltd. Nantong 226009, China, and also with HUST Wuxi Research Institute, Wuxi 214174, China (e-mail: jinjunyang@yuanshi-autonomy.com).

Zidong Wang is with the Department of Computer Science, Brunel University London, UB8 3PH Uxbridge, U.K. (e-mail: Zidong.Wang@brunel.ac.uk).

Digital Object Identifier 10.1109/TII.2026.3712756

I. Introduction

A. Motivations

ACCURATE electric load forecasting is essential for the reliable and economical operation of modern power systems [1], [2]. Among various forecasting horizons, medium-term load forecasting (MTLF), typically covering one week to one year, plays a critical role in capacity planning, maintenance scheduling, and intelligent resource allocation [3], [4], [5]. Compared with short-term load forecasting, MTLF must capture longer load range temporal dependencies under greater uncertainty; compared with long-term load forecasting, it demands finer temporal resolution to support operational decisions. These requirements make MTLF a particularly challenging task that calls for models with strong long-range temporal modeling capability, robust spatiotemporal representation, and reliable uncertainty quantification.

Despite considerable advances in load forecasting methodology, most existing MTLF approaches are built around a common assumption of fixed, contiguous prediction horizons. As a result, practical scenarios that call for forecasts over flexible or noncontiguous future intervals remain less explored. This assumption also raises a modeling consideration, since conventional spatiotemporal formulations typically rely on fixed temporal positions, making them difficult to adapt to variable or irregular forecast intervals.

B. Literature Review

Existing research on load forecasting has progressed along three dimensions, namely model expressiveness, uncertainty quantification, and spatiotemporal modeling.

Early approaches relied on statistical methods, with ARMAX combined with FNNs incorporating weather variables to improve accuracy [19]. Deep learning subsequently enabled richer feature extraction across horizons [21]. For medium-term forecasting, a two-stage backpropagation neural network-radial basis function neural network (BPNN-RBFNN) framework mitigates multistep error propagation [16], and a multitask CNN-LSTM model jointly extracts weather and load features [17]. For shorter horizons, MSTL-VMD-TCN leverages multiscale temporal decomposition [13], a fuzzy LSTM with attention enhances robustness under unstable data quality [9], and a dynamic drift-adaptive LSTM (DA-LSTM)

TABLE I
COMPARISON OF THE PROPOSED METHOD WITH STATE-OF-THE-ART METHODS

Reference & Year	Forecasting Method	Method Type	Forecasting Horizon		Forecasting Type	Probabilistic	Spatiotemporal Modeling	Flexible Interval Support
Our method	Diffusion Transformer	Generative	MTLF	One week to three months	Multivariate	✓	✓	✓
[6] 2026	Latent Diffusion Model	Generative	MTLF	One month	Multivariate	✓	✓	×
[7] 2025	Diffusion U-Net	Generative	STLF	Half day to two days	Multivariate	✓	×	×
[8] 2025	COLNet	Discriminative	STLF	One day	Univariate	✓	×	×
[9] 2025	Fuzzy-LSTM-Attention	Discriminative	VSTLF	One hour	Univariate	×	×	×
[10] 2024	Transformer-RML	Discriminative	VSTLF, STLF	Half hour, half day	Univariate	✓	×	×
[11] 2024	CNNs-Transformer	Discriminative	STLF	One day	Univariate	✓	×	×
[12] 2024	MS-STGNN	Discriminative	VSTLF	One hour	Multivariate	×	✓	×
[13] 2024	MSTL-VMD-TCN	Discriminative	STLF	One day	Multivariate	×	×	×
[14] 2024	MIF-DNQR	Discriminative	VSTLF, STLF	One day	Univariate	✓	×	×
[15] 2024	DTformer	Discriminative	STLF, MTLF	One day, one month	Multivariate	×	×	×
[16] 2023	BPNN-RBFNN	Discriminative	MTLF	One month, one year	Univariate	×	×	×
[17] 2023	MTMV-CNN-LSTM	Discriminative	MTLF	10 days, three months	Multivariate	×	×	×
[18] 2023	DA-LSTM	Discriminative	VSTLF	One hour	Univariate	×	×	×
[19] 2023	ARMAX and FNN	Discriminative	STLF, MTLF	One day, half year	Multivariate	×	×	×
[20] 2023	AHLC-QRNN	Discriminative	STLF	One day	Univariate	✓	×	×

Note: MTLF = medium-term load forecasting; STLF = short-term load forecasting; VSTLF = very short-term load forecasting.
In the "Forecasting Horizon" column, a single value denotes a fixed horizon, comma-separated values (e.g., "one month, one year") denote the shortest and longest horizons across different model variants, and "to" (e.g., "one week to three months") denotes a continuous range supported by a single model.

adapts to concept drift in consumption patterns through active and passive model updates [18]. The DTformer further unifies short- and medium-term forecasting in integrated energy systems through transformer-based architectures [15]. Beyond temporal modeling, a multiscale spatiotemporal graph neural network (MS-STGNN) extends discriminative frameworks to multivariate settings by capturing intervariable dependencies through graph-based spatial modeling [12]. Despite these advances, the above methods produce deterministic outputs without quantifying predictive uncertainty, which is essential for risk-aware decision-making in power system operations.

To enable uncertainty quantification, a quantile regression neural network with an accumulated hidden layer connection structure (AHLC-QRNN) produces short-term interval forecasts [20], a multi-information fusion noncrossing quantile (MIF-DNQR) method generates coherent day-ahead probabilistic forecasts [14], and a complementary online learning network (COLNet) maintains calibrated forecasts under extreme-weather concept drift [8]. Transformer-based architectures have also been paired with Gaussian process residual modeling [10] and quantile regression loss [11] to quantify uncertainty under irregular load patterns. More recently, generative models have emerged as a more principled alternative by directly modeling the data-generating process. A latent diffusion model jointly synthesizes multivariate load and temperature scenarios with spatiotemporal structure but targets a fixed one-month horizon from the current time point [6]. A diffusion U-Net reformulates short-term forecasting as a temporal inpainting problem, completing unknown future values within a fixed window of half a day to two days by selecting target points inside it [7]. Despite this progression toward full distributional modeling, all of the above methods are limited to fixed, contiguous prediction horizons and cannot target arbitrary or noncontiguous future intervals.

As summarized in Table I, existing forecasting methods rarely provide explicit support for flexible interval prediction, whereas practical medium-term planning tasks often require forecasts for specific, and sometimes noncontiguous, future periods [22]. A common workaround is to forecast the entire horizon and then extract the required subintervals post hoc. However, this strategy has several inherent limitations. First, depending on the forecasting strategy, autoregressive models may accumulate errors through intermediate steps that are irrelevant to the target interval [23], while direct multioutput models may suffer from interference from task-irrelevant horizons. Second, full-horizon probabilistic inference introduces redundant computation when only a subset of future time steps is required. Third, training objectives defined over the entire horizon may dilute model capacity and lead to a mismatch between global sequence-level accuracy and interval-specific forecasting performance. Moreover, when the target interval varies in position and length, existing spatiotemporal formulations generally lack an explicit mechanism to adapt their representations to the requested interval [24]. These limitations indicate that flexible interval-selective forecasting and interval-aware spatiotemporal modeling remain jointly unaddressed in existing MTLF frameworks.

C. Contributions

To address the above gaps, we propose DiffFlex, a diffusion-based probabilistic forecasting framework that reformulates MTLF as a temporal outpainting problem. The main contributions of this study are as follows.

1) We reformulate MTLF as a flexible interval-selective forecasting problem that decouples the forecasting target from temporal contiguity. To the best of our knowledge, this is the first MTLF formulation supporting direct prediction over arbitrary, potentially noncontiguous future intervals.
2) We introduce the forecasting interval as an explicit modeling dimension alongside temporal points and spatial variables, capturing intradimensional dependencies and cross-dimensional interactions. This interval-aware design enables the model to adapt its spatiotemporal representations to different target intervals.
3) We design a probabilistic diffusion-based architecture integrating a historical block fusion mechanism, a spatiotemporal-interval conditioned fusion module, and an interval-embedded diffusion module. The fusion

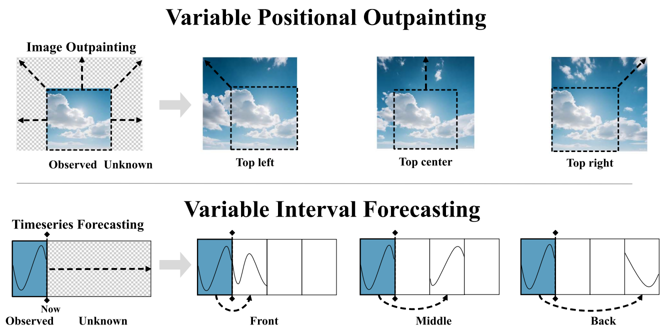


Fig. 1. Analogy between relative positional outpainting in images and targeted interval forecasting in time series.

mechanism aggregates multiscale temporal dependencies at daily, weekly, and monthly resolutions to condition the generative process, while the diffusion module generates a full predictive distribution rather than a single point estimate.

The remainder of this article is organized as follows. Section II presents the theoretical foundations of conditional diffusion models for forecasting. Section III details the overall framework of DiffFlex and its implementation specifics. Section IV presents the experimental setup and results using multiple real-world datasets. Section V concludes the article, summarizing key findings and limitations, and outlining directions for future research.

II. Methodology

A. From Positional Outpainting to Interval Forecasting

Outpainting, also known as image extrapolation, refers to the generation of new content that extends beyond the original boundaries of an image [25], [26], [27]. In contrast to inpainting, which focuses on filling in missing internal regions, outpainting aims to infer unseen areas outside the visible frame. Analogously, in time-series forecasting, this concept aligns with a broader and relatively underexplored paradigm in which future values may begin at arbitrary time points, thus enabling sparse, noncontiguous, and task-adaptive predictions beyond the immediate temporal horizon.

When conceptualizing a time series as a 1-D canvas, traditional forecasting resembles the inpainting of a continuous and fixed-length segment. However, many real-world applications demand greater flexibility, such as forecasting discrete or temporally distant values, where fixed-window approaches fall short. This generalized setting parallels the logic of outpainting, which allows for generation beyond adjacent or regularly spaced intervals.

To help visualize this analogy, Fig. 1 presents a comparison between relative positional outpainting in image generation and targeted interval forecasting in time series. Just as image outpainting can extend content from various reference positions (e.g., top-left, top-center, top-right), time series forecasting can benefit from generating values at different relative positions in the forecast horizon (e.g., front, middle, back).

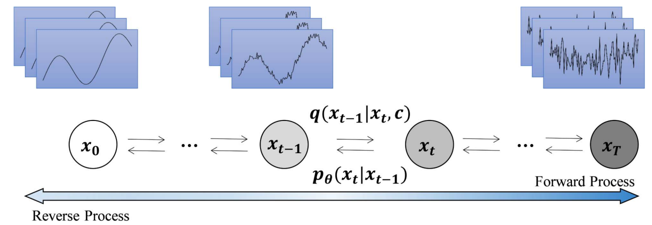


Fig. 2. Conditional diffusion model architecture illustrating the forward noising process, reverse denoising process, and conditional input integration.

The recent success of diffusion models in image outpainting highlights their potential for application in time-series forecasting [28]. Their ability to model complex distributions and quantify uncertainty makes them particularly well-suited for challenging prediction tasks. Building on this insight, we propose to apply diffusion models to flexible interval load forecasting scenarios.

B. Conditional Diffusion Model

1) Forward diffusion process and reverse (denoising) process: Conditional diffusion models introduce auxiliary information into the denoising process so that generation is guided by task-specific conditions. As shown in Fig. 2, the condition $\mathbf{c}$ encodes historical load observations and target interval information, while $\mathbf{x}_0$ denotes the target load sequence.

In the forward process, clean data are gradually perturbed by Gaussian noise according to a predefined variance schedule $\{\beta_t\}_{t=1}^T$ as

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad \bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s, \quad \alpha_t = 1-\beta_t \quad (1)$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This closed-form expression allows $\mathbf{x}_t$ to be sampled directly at any diffusion step.

The reverse process learns to remove the injected noise step by step. A neural network $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{c})$ predicts the noise component conditioned on the noisy sample, timestep, and load-related context. The model is optimized by the conditional noise prediction objective

$$\mathcal{L}_{\text{cond}} = \mathbb{E}_{t,\mathbf{x}_0,\boldsymbol{\epsilon},\mathbf{c}} \left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{c})\|^2 \right]. \quad (2)$$

By learning this denoising function across random timesteps, the model can transform Gaussian noise into load trajectories consistent with both historical patterns and the specified forecasting interval.

2) Model training and sampling methods: Following DDPMs [29], each training iteration samples $(\mathbf{x}_0, \mathbf{c})$ from the training set, draws $t \sim \mathcal{U}(\{1, \ldots, T\})$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, constructs $\mathbf{x}_t$ using the forward equation, and updates $\boldsymbol{\epsilon}_\theta$ with $\mathcal{L}_{\text{cond}}$.

For sampling, the process starts from $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and proceeds from T to 1. At each step, the network predicts

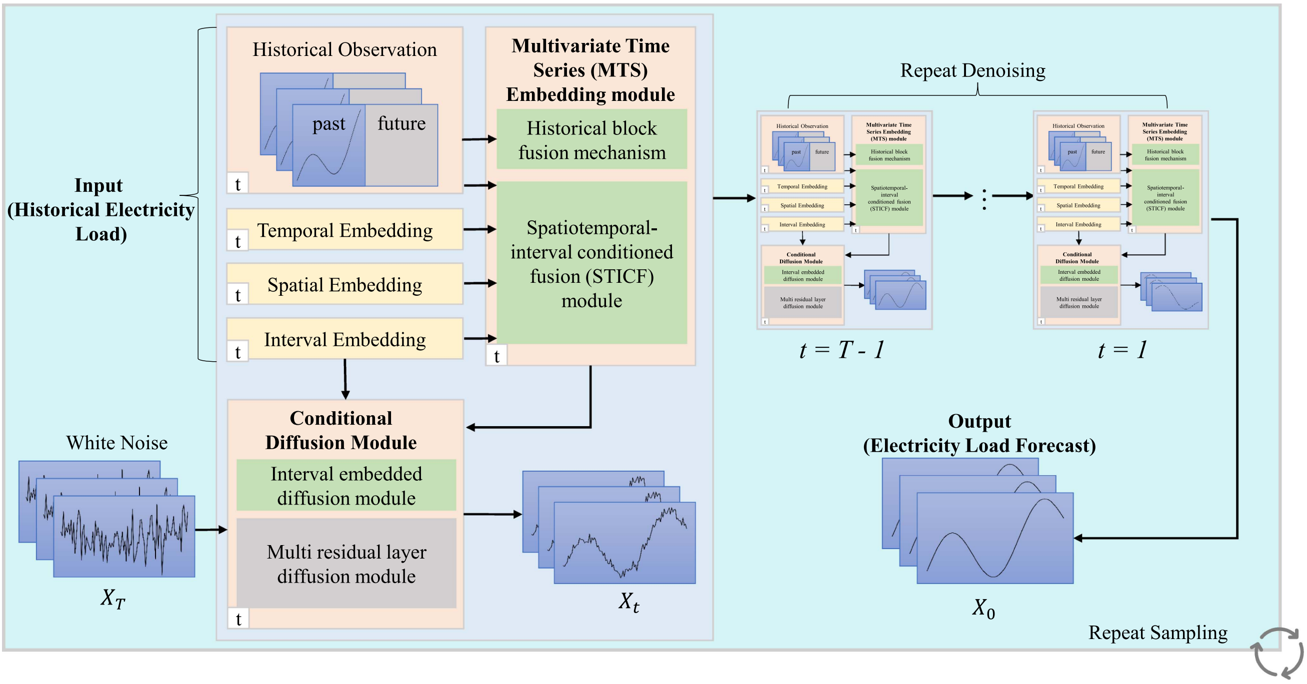


Fig. 3. Overall architecture of the DiffFlex framework.

$\hat{\boldsymbol{\epsilon}} = \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{c})$, and the reverse mean is computed as

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{c}) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\hat{\boldsymbol{\epsilon}}\right). \tag{3}$$

The previous state is then sampled by

$$\mathbf{x}_{t-1} = \boldsymbol{\mu}_\theta + \sigma_t \mathbf{z} \tag{4}$$

where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for $t > 1$ and $\mathbf{z} = \mathbf{0}$ for $t = 1$. The final $\mathbf{x}_0$ is used as the generated load forecast. The stochastic sampling process can produce multiple plausible trajectories, which provides a natural basis for uncertainty estimation.

III. Implementation of DiffFlex

Building on the theoretical foundations in Section II, this section details the practical implementation of DiffFlex for medium-term load forecasting.

A. Model Structure

Fig. 3 illustrates the overall architecture of DiffFlex, which takes historical electricity load data as input and generates interval-conditioned load forecasts. DiffFlex comprises two main components: the multivariate time series (MTS) embedding module and the conditional diffusion module. The three green modules in the figure represent our main contributions, namely the historical block fusion mechanism, the spatiotemporal-interval conditioned fusion (STICF) module, and the interval-embedded diffusion module, while the gray module follows TSDE [30].

In the MTS embedding module, the historical block fusion mechanism and the STICF module process complementary information separately. The former partitions past observations into temporal blocks to capture recurring demand cycles, while the latter extracts temporal, spatial, and interval-aware representations from the historical sequence and its embeddings. The block fusion mechanism is parameter-free and reorganizes the input sequence, whereas the STICF module is a trainable neural component learned end-to-end. Their outputs are concatenated at the end of the MTS module to strengthen the historical context, and the resulting embedding $\mathbf{H}_{\text{fused}}$ serves as the conditional context for the diffusion stage.

The conditional diffusion module is responsible for generating forecasts. It comprises two components. The first is an interval-embedded diffusion module that injects the target interval descriptor into the denoising process, and the second is a multiresidual layer diffusion module adapted from TSDE [30] for accurate noise estimation. In this way, the conditional diffusion module is conditioned jointly on $\mathbf{H}_{\text{fused}}$ and T_{interval}, enabling probabilistic forecasts $\mathbf{X}_{\text{pred}} \in \mathbb{R}^{K \times L_{\text{out}}}$ over user-specified horizons ranging from one to twelve weeks.

B. Historical Block Fusion Mechanism

The historical block fusion mechanism leverages cyclical demand patterns across multiple time scales, which is essential for capturing both short-term operational cycles and longer term seasonal trends.

The historical sequence $\mathbf{X}_{\text{hist}}$ is partitioned into M nonoverlapping blocks, where each block $\mathbf{B}_m$ spans the same duration as the target prediction length L_{out}. Formally,

$$\mathbf{B}_m = \mathbf{X}_{\text{hist}}[:, (m-1)\cdot L_{\text{out}} : m \cdot L_{\text{out}}], \quad m = 1, 2, \ldots, M \tag{5}$$

with $M = \lfloor L_{\text{in}} / L_{\text{out}} \rfloor$ denoting the maximum number of complete blocks.

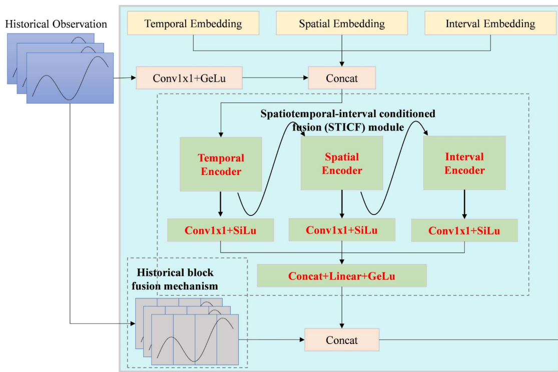


Fig. 4. Detailed structure of the STICF module showing temporal, spatial, and interval encoders with hybrid block fusion.

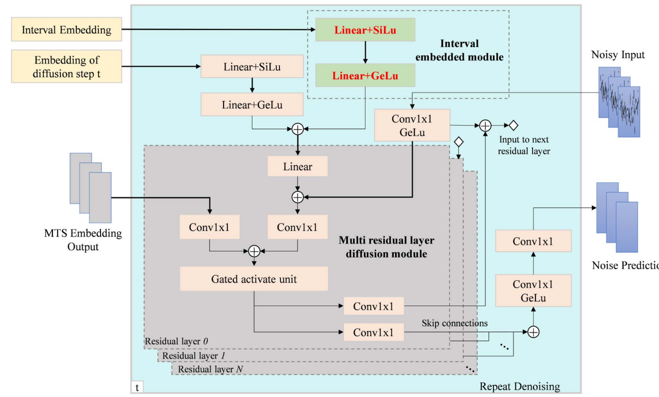


Fig. 5. Interval embedded diffusion process showing the integration of interval embeddings with the conditional noise prediction network.

This block-based design aligns historical segments with the temporal structure of the prediction target, allowing the model to exploit past conditions that resemble the forecast horizon. By capturing analogous weekly patterns, the mechanism provides rich historical context for forecasting across diverse horizons.

C. STICF Module

The spatiotemporal-interval conditioned fusion module shown in Fig. 4 constitutes the core feature integration stage of the STICF branch. It processes the historical sequence and its embedding representations through three specialized encoders to integrate temporal, spatial, and interval information. Specifically, the temporal encoder captures time-evolving dependencies and recurring patterns, the spatial encoder learns inter-regional correlations and cross-zone dependencies, and the interval encoder maps the target lead-time descriptor T_{interval} into dense embeddings to provide horizon-specific context.

The STICF module adopts a cascaded architecture with skip connections to enhance feature interactions for noncontiguous intervals. The historical sequence and embedding representations are passed through the temporal, spatial, and interval encoders in sequence, while skip connections preserve intermediate representations and facilitate cross-module information exchange. After fully connected transformations, the STICF output is concatenated with the historical block-enhanced representation, producing $\mathbf{H}_{\text{fused}}$ to guide the interval-conditioned diffusion process for arbitrary and potentially noncontiguous medium-term forecasts.

D. Interval Embedded Diffusion Module

As illustrated in Fig. 5, the interval-embedded diffusion module incorporates the target interval into the denoising process to support horizon-aware probabilistic generation. At each diffusion step, the noise prediction network is conditioned on the diffusion timestep and noise level, the interval embedding derived from T_{interval}, and the fused context $\mathbf{H}_{\text{fused}}$ from the MTS embedding module. These conditions allow the denoising process to remain aligned with the specified forecasting horizon while exploiting historical and spatiotemporal information.

During training, the module learns to estimate noise under the joint condition of historical context and horizon information. During sampling, it iteratively denoises Gaussian noise guided by the same condition, enabling forecasts to adapt to different medium-term horizons. Multiple generated trajectories are statistically aggregated to obtain point forecasts and prediction intervals, thereby retaining uncertainty information for downstream decision making.

IV. Experiments and Results

A. Experimental Design

1) Data sources and preprocessing: The datasets used in this study are obtained from three authoritative entities in the power system domain, namely Independent System Operator of New England (ISO-NE), the New York Independent System Operator (NYISO), and the European Network of Transmission System Operators for Electricity (ENTSO-E) [31]. They provide long-term regional electricity demand records that capture temporal fluctuations and seasonal variations. The ISO-NE data comprise hourly demand observations from 8 regions, spanning January 1, 2021 to March 10, 2025. The NYISO data comprise hourly observations from 11 regions spanning April 8, 2021 to March 28, 2025, resampled from the original 15-min resolution to an hourly resolution. The ENTSO-E data comprise hourly observations from 17 countries spanning January 1, 2016 to September 30, 2020, selected from 32 European countries whose missing data rate is below 0.1%. These records provide sufficient coverage for MTLF.

Data preprocessing follows a chronological split to avoid information leakage. For the ISO-NE and NYISO datasets, samples before September 1, 2024 are used for training, those from September 1 to October 31, 2024 for validation, and the rest for testing. For the ENTSO-E dataset, samples before January 1, 2020 are used for training, those from January 1 to April 30, 2020 for validation, and the rest for testing.

All series are standardized with training-set statistics:

$$\mathbf{x}_{\text{std}} = \frac{\mathbf{x} - \boldsymbol{\mu}_{\text{train}}}{\max(1 \times 10^{-5}, \boldsymbol{\sigma}_{\text{train}})} \tag{6}$$

where $\boldsymbol{\mu}_{\text{train}}$ and $\boldsymbol{\sigma}_{\text{train}}$ are computed only from the training set and then applied to all splits.

Training samples are constructed using a sliding-window mechanism for flexible interval forecasting. For each target interval $i \in \{1, \ldots, 12\}$, historical and target sequences are paired with the corresponding temporal offset, enabling forecasts from one to 12 weeks ahead.

During sample generation, temporal, spatial, and interval position indices are also produced for subsequent sinusoidal and learning-based representations.

2) Evaluation metrics: To evaluate forecasting performance from both deterministic and probabilistic perspectives, we adopt root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and continuous ranked probability score (CRPS). RMSE, MAE, and MAPE measure point prediction accuracy, whereas CRPS quantifies the discrepancy between the predictive distribution and the observed outcome.

In the following equations, $\hat{\mathbf{X}}$ denotes the predicted values and $\mathbf{X}$ represents the ground truth. Specifically, $\hat{x}_{i,k,l}$ and $x_{i,k,l}$ denote the predicted and actual values for the ith sample, kth feature (region), and the lth time step, respectively. Here, N is the number of test samples, K the number of features (regions), L the length of the forecasting sequence (time steps), and S the number of diffusion samples per (i, k, l)

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{K \times L}\sum_{k=1}^{K}\sum_{l=1}^{L}(\hat{x}_{i,k,l} - x_{i,k,l})^2\right)} \tag{7}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{K \times L}\sum_{k=1}^{K}\sum_{l=1}^{L}|\hat{x}_{i,k,l} - x_{i,k,l}|\right) \tag{8}$$

$$\text{MAPE} = \frac{100\%}{N}\sum_{i=1}^{N}\left(\frac{1}{K \times L}\sum_{k=1}^{K}\sum_{l=1}^{L}\left|\frac{\hat{x}_{i,k,l} - x_{i,k,l}}{x_{i,k,l} + 1}\right|\right). \tag{9}$$

For probabilistic evaluation, we report CRPS following [32], based on empirical quantiles of the diffusion samples. Given the quantile loss at level α,

$$\mathcal{L}_\alpha(q, x) = |(q - x)(\mathbf{1}\{x \le q\} - \alpha)|, \alpha \in [0, 1] \tag{10}$$

where q is the predicted quantile and x is the ground truth. Let $\hat{q}_{i,k,l}(\alpha_j)$ denote the empirical α_j-quantile of the S diffusion samples at (i, k, l), with $\alpha_j = 0.05j, j = 1, \ldots, M$, where $M = 19$,

$$\text{CRPS} = \frac{1}{M}\sum_{j=1}^{M}\frac{2\sum_{i=1}^{N}\sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{L}_{\alpha_j}(\hat{q}_{i,k,l}(\alpha_j), x_{i,k,l})}{\sum_{i=1}^{N}\sum_{k=1}^{K}\sum_{l=1}^{L}|x_{i,k,l}|}. \tag{11}$$

The CRPS of point prediction models is obtained by assuming a Gaussian predictive distribution centered at the point prediction with variance estimated from the training set, and then applying the CRPS for Gaussian distributions.

TABLE II
MAIN CONFIGURATION PARAMETERS OF THE DIFFFLEX MODEL

Parameter	Value
Residual blocks	4
Diffusion channels	96
Diffusion steps	50
Embedding dimension	128
Learning rate	1×10^{-3}
Weight decay	1×10^{-6}
History length	24×28 hours
Prediction length	24×7 hours
Interval range	[1, 12] weeks
Step size	24 hours
Batch size	32
Max epochs	500

3) Experimental setup: We compare DiffFlex against eight baseline models from five methodological categories, covering both deterministic and probabilistic forecasting. These categories include traditional machine learning, linear models, CNN-based models, transformer-based models, and diffusion-based generative models. XGBoost [33] is included as a representative traditional machine learning method that has been widely adopted in practical load forecasting. DLinear and NLinear [34] serve as strong linear baselines, as their simple linear mapping structures have demonstrated competitive forecasting performance on multiple benchmarks. TimesNet [35] is selected as a representative CNN-based model for temporal pattern extraction. TimeXer [36] and PatchTST [37] are included as representative transformer-based forecasting models and strong discriminative baselines. Diffusion-TS [38] and Time Series Diffusion Embedding (TSDE) [30] are adopted as diffusion-based probabilistic baselines to enable a controlled within-family comparison, thereby isolating the contribution of the proposed temporal outpainting formulation from the general benefits of diffusion modeling. The implementations of the compared models are obtained from their official repositories [30], [33], [38] or from TSLib [39].

The proposed DiffFlex framework adopts a spatiotemporal-interval fusion architecture in which historical observations are partitioned into temporal blocks aligned with the prediction interval. Feature refinement is performed through a cascaded attention mechanism, and the diffusion module is conditioned on interval information. Training is conducted with the Huber loss ($\delta = 1.0$). The main model, training, and data configurations are summarized in Table II.

All experiments were conducted on a server with a single 32 GB vGPU, a 12-core Intel Xeon Platinum 8352 V CPU at 2.10 GHz, and 90 GB RAM, using Ubuntu 22.04, Python 3.12, and PyTorch 2.5.1. For probabilistic evaluation, all diffusion-based models used the same set of 100 diffusion samples.

TABLE III
ABLATION RESULTS ON THE ISO-NE DATASET

Component	**RMSE** ↑%	**MAE** ↑%	**MAPE** ↑%	**CRPS** ↑%
Historical block	+14.56	+17.34	+14.83	+20.54
STICF	+11.48	+13.16	+10.37	+15.71
Interval-embedded	+2.60	+3.82	+4.15	+4.30

Note: Reported values indicate relative performance improvements when including each component.

B. Model Structure Analysis

To evaluate the contribution of each architectural component in DiffFlex, we conduct ablation experiments by selectively removing individual modules. These experiments are performed on the ISO-NE dataset, with results summarized in Table III.

The three components address distinct aspects of the forecasting task. The historical block fusion mechanism yields the largest individual gains (RMSE +14.56%, MAE +17.34%, MAPE +14.83%, CRPS +20.54%). This is expected, as medium-term load series exhibit strong weekly and seasonal periodicities that constitute the primary source of predictability in MTLF. Removing this mechanism deprives the model of these dominant cyclical cues.

The STICF module produces the second-largest improvements (RMSE +11.48%, MAE +13.16%, MAPE +10.37%, CRPS +15.71%). This module addresses a challenge unique to the flexible-interval setting, in which temporal positions, spatial variables, and heterogeneous forecasting intervals must be jointly modeled. Without it, the model treats all target intervals identically and cannot adapt its representations accordingly, confirming that interval-aware cross-dimensional fusion is indispensable for flexible-interval forecasting.

The interval embedded diffusion module contributes moderate but consistent gains (RMSE +2.60%, MAE +3.82%, MAPE +4.15%, CRPS +4.30%). Its relatively smaller marginal improvement is partly because interval information has already been captured by the STICF module. The interval embedded diffusion module further refines the conditioning at the generative stage. Its largest relative gain appears in CRPS (+4.30%), consistent with its design purpose of enhancing interval-aware probabilistic calibration.

Overall, the three components address complementary aspects of the forecasting task, covering periodicity extraction, cross-dimensional feature fusion, and interval-conditioned generation, and the ablation confirms that all three are necessary for DiffFlex to achieve its full performance.

C. Medium-Term Load Forecasting Performance of DiffFlex

To evaluate DiffFlex for MTLF, 12 direct forecasting tasks are considered, with lead times ranging from 1 to 12 weeks ahead. For each lead time, the model produces a full predictive distribution, from which the median serves as the point forecast and the 25%–75% and 5%–95% quantile intervals characterize predictive uncertainty.

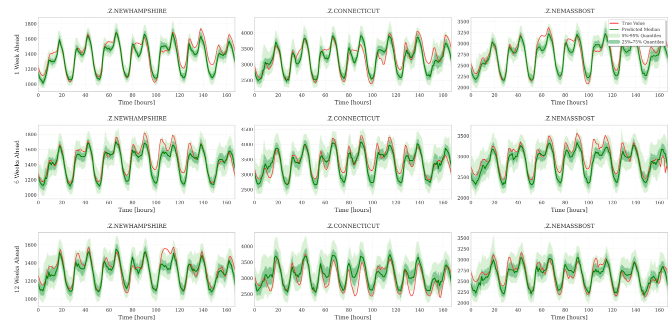


Fig. 6. Probabilistic load forecasting results across different horizons and regions on ISO-NE dataset.

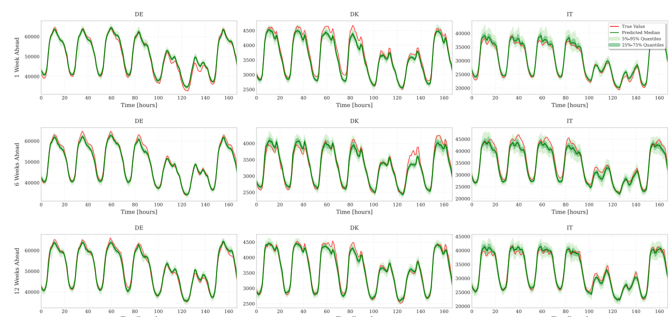


Fig. 7. Probabilistic load forecasting results across different horizons and regions on ENTSO-E dataset.

Figs. 6 and 7 present representative results for selected regions in the ISO-NE and ENTSO-E datasets. At the 1-week horizon, the predictive median follows the true load trajectory closely, reproducing the dominant daily cycle and the within-week variation together with the timing of major peaks and troughs. As the horizon extends to 6 and 12 weeks ahead, the median continues to recover the full seasonal waveform and keeps the cycles aligned in time, with no obvious phase distortion. The residual errors are confined to the amplitude of local extremes, where the 6-week median mildly flattens the strongest peaks and troughs, and the 12-week deviations fall on both sides of the true curve without a consistent directional bias. Across all horizons the predictive intervals widen with the lead time and the observed values largely remain within them, showing that DiffFlex absorbs the growing forecast difficulty into calibrated uncertainty rather than misleading point estimates.

The quantitative results support these visual observations. As shown in Table IV, the average RMSE, MAE, MAPE, and CRPS of DiffFlex over the 12 forecasting horizons are 211.87, 154.21, 8.88%, and 0.065 on ISO-NE, 202.58, 134.73, 9.71%, and 0.065 on NYISO, and 1373.37, 662.80, 4.86%, and 0.037 on ENTSO-E. DiffFlex exhibits robustness across datasets with distinct load scales and demand patterns.

To examine how accuracy evolves with the lead time, the horizon-level analysis uses ISO-NE and ENTSO-E as examples, with NYISO following a consistent pattern. In Figs. 8 and 9, the DiffFlex curve (dark blue) remains stable across horizons, rising to a mild peak around the middle horizons before easing back,

TABLE IV
AVERAGE PERFORMANCE COMPARISON OF DIFFERENT MODELS ACROSS THREE DATASETS (1–12 WEEKS AHEAD)

Dataset	Model	RMSE	MAE	MAPE (%)	CRPS
ISO-NE	DiffFlex	**211.87**	**154.21**	**8.88**	**0.065**
	XGBoost	238.60	178.75	10.72	0.100
	DLinear	255.81	194.37	11.43	0.109
	NLinear	249.79	179.14	11.11	0.100
	TimesNet	278.08	203.27	12.58	0.114
	TimeXer	287.13	209.84	13.22	0.118
	PatchTST	275.79	203.91	12.71	0.115
	Diffusion-TS	412.24	307.75	17.08	0.156
	TSDE	256.12	191.31	10.99	0.087
NYISO	DiffFlex	**202.58**	**134.73**	**9.71**	**0.065**
	XGBoost	331.29	204.76	14.08	0.129
	DLinear	235.39	157.19	11.62	0.099
	NLinear	308.50	202.57	15.59	0.128
	TimesNet	252.09	164.78	12.78	0.104
	TimeXer	331.39	216.42	16.44	0.137
	PatchTST	331.35	223.88	17.66	0.142
	Diffusion-TS	378.39	256.46	16.94	0.152
	TSDE	218.64	147.63	11.57	0.072
ENTSO-E	DiffFlex	**1373.37**	**662.80**	**4.86**	**0.037**
	XGBoost	2674.94	1439.88	9.75	0.102
	DLinear	1970.41	990.38	6.97	0.070
	NLinear	1821.76	916.64	6.73	0.065
	TimesNet	1983.22	1014.02	7.52	0.072
	TimeXer	2389.58	1179.96	7.82	0.083
	PatchTST	2575.39	1382.04	10.10	0.097
	Diffusion-TS	3445.54	1815.16	12.76	0.127
	TSDE	2123.13	1074.23	7.52	0.063

Note: Bold values represent the best performance for each dataset-metric combination.

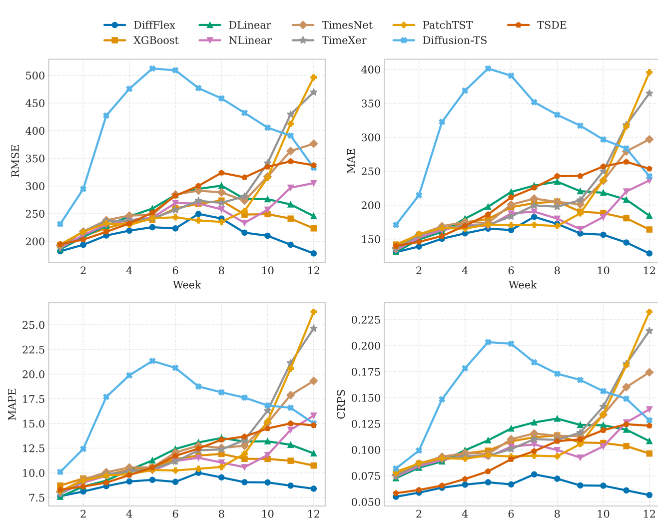


Fig. 8. Trends in forecasting accuracy metrics across models on ISO-NE dataset (1–12 weeks ahead).

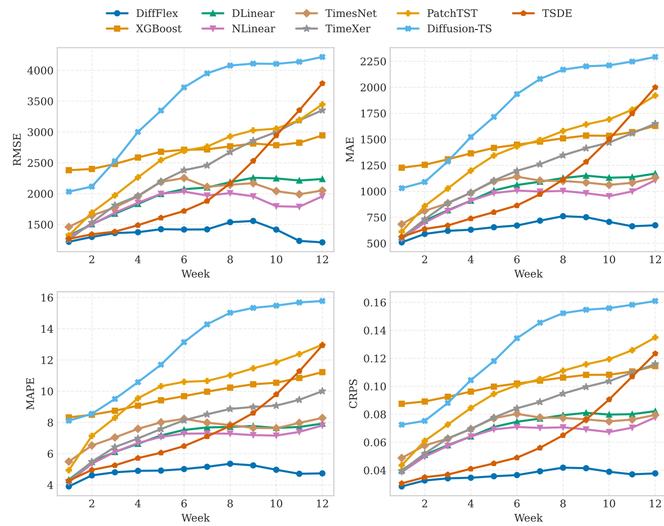


Fig. 9. Trends in forecasting accuracy metrics across models on ENTSO-E dataset (1–12 weeks ahead).

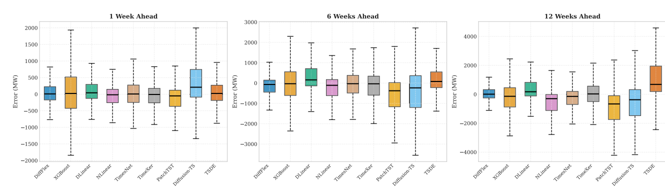


Fig. 10. Average forecasting error across different horizons on ENTSO-E dataset.

so its error does not grow monotonically with the lead time. This nonaccumulating profile reflects the direct forecasting design of DiffFlex, which produces each horizon directly and avoids the error propagation of recursive multistep prediction.

D. Comparative Performance Evaluation Against Baseline Models

1) Overall forecasting performance comparison: To comprehensively assess MTLF performance from 1 to 12 weeks ahead, DiffFlex is benchmarked against a diverse set of representative baselines covering different modeling approaches, with the average results summarized in Table IV. DiffFlex achieves the best average performance across all four metrics and all datasets, demonstrating a consistent advantage in both point and probabilistic prediction. Its advantage also strengthens as the horizon extends. As shown in Figs. 8 and 9, the error curves of DiffFlex (dark blue) rise much more slowly than those of the competing methods, and beyond approximately 6 to 8 weeks ahead the baselines degrade sharply while DiffFlex remains comparatively stable.

To further evaluate DiffFlex, the distributions of forecasting errors were examined at lead times of 1, 6, and 12 weeks on the ENTSO-E dataset. As shown in Fig. 10, DiffFlex maintains a relatively narrow interquartile range and a median close to zero across all three horizons. Compared with most baselines, it also exhibits fewer large deviations, especially at longer lead times.

TABLE V
DETAILED MEDIUM-TERM FORECASTING ACCURACY COMPARISON ACROSS MODELS ON ENTSO-E DATASET (4, 8, AND 12 WEEKS AHEAD)

Week	Metric	DiffFlex	XGBoost	DLinear	NLinear	TimesNet	TimeXer	PatchTST	Diffusion-TS	TSDE
4	RMSE	**1374.755**	2586.453	1833.648	1852.826	1958.467	1964.427	2264.267	3000.124	1490.533
	MAE	**630.418**	1365.247	910.705	910.227	985.761	986.336	1199.050	1521.924	739.306
	MAPE	**4.901**	9.074	6.633	6.685	7.592	6.978	9.547	10.578	5.724
	CRPS	**0.035**	0.096	0.064	0.064	0.069	0.070	0.085	0.104	0.041
8	RMSE	**1532.088**	2767.863	2186.329	2008.788	2145.359	2674.573	2928.797	4078.030	2173.331
	MAE	**758.425**	1509.546	1127.784	1004.300	1099.344	1346.491	1579.532	2169.869	1115.559
	MAPE	**5.345**	10.232	7.713	7.281	7.830	8.866	11.023	15.023	7.799
	CRPS	**0.042**	0.106	0.079	0.071	0.077	0.095	0.111	0.152	0.065
12	RMSE	**1212.656**	2944.605	2237.568	1958.478	2051.973	3352.748	3447.539	4216.951	3798.266
	MAE	**674.240**	1629.003	1170.171	1107.003	1132.047	1650.540	1920.386	2291.783	2006.125
	MAPE	**4.743**	11.221	7.926	7.806	8.289	10.005	12.976	15.775	12.978
	CRPS	**0.038**	0.114	0.082	0.078	0.080	0.116	0.135	0.161	0.123

Note: Bold values represent the best performance for each week-metric combination. DiffFlex performs direct parallel forecasting across multiple non-contiguous target weeks, whereas the baseline methods adopt a week-by-week rolling forecasting strategy.

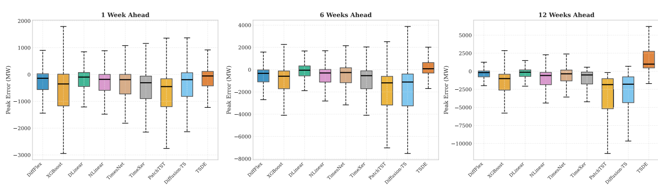


Fig. 11. Peak load forecasting error across different horizons on ENTSO-E dataset.

These results indicate that DiffFlex provides stable and accurate forecasts under medium-term forecasting settings. To further assess performance under high-demand conditions, near-peak loads were defined as the upper decile of regional load values. For each region, these upper-decile values were computed on a monthly basis (four-week windows) using actual load data. Upper-decile trends were then compared across regions and time intervals, and the corresponding errors are reported in Fig. 11. The peak-load errors are generally larger in magnitude than the average errors, and most baselines tend to underestimate peak demand. In contrast, DiffFlex keeps a compact error distribution and a limited bias across the three horizons. This advantage is particularly clear at the 12-week horizon, where PatchTST and Diffusion-TS show substantial underestimation, while TSDE tends to overestimate peak loads. These results suggest that DiffFlex can track peak-load conditions more reliably in MTLF.

2) Flexible-interval forecasting performance: Weeks 4, 8, and 12 are selected as noncontiguous target intervals, corresponding to one, two, and three months ahead, to evaluate DiffFlex under the flexible interval forecasting setting that motivates its design. As reported in Table V, DiffFlex attains the best point and probabilistic accuracy on every metric at every horizon, and its advantage becomes most pronounced as the target moves further into the future. At week 4 its MAPE of 4.901% already falls below the strongest baseline TSDE at 5.724%, and at week 12 its MAPE of 4.743% is 39.2% lower than the best competing model NLinear at 7.806%, while its CRPS of 0.038 is roughly half of NLinear at 0.078. This widening margin reflects a structural difference in forecasting strategy. Because the baselines reach distant targets through week-by-week rolling forecasting, their errors accumulate as the horizon increases.

TABLE VI
TIME COMPLEXITY AND INFERENCE SPEED OF DIFFUSION-BASED FORECASTING MODELS

Model	Time Complexity	1st week inference (in sec)	12th week inference (in sec)
DiffFlex	$O(TnN)$	14.25s	12s
Diffusion-TS	$O(MTnN)$	305.5s	2921s
TSDE	$O(MTnN)$	208s	1899s

Note: $N-$ Number of input samples; $M-$ Forecast horizon in weeks; $T-$ Number of diffusion timesteps; $n-$ Number of noise samples per input sample.

DiffFlex instead forecasts the three target weeks directly and in parallel, so it avoids error compounding entirely.

Fig. 12 confirms this behavior at the trajectory level. In panel (a), the DiffFlex forecast closely follows the ground truth across all three shaded windows, while the long unshaded gaps between windows indicate that these intervals are generated through direct noncontiguous prediction rather than a continuous roll-out. In panel (b), the baseline forecasts show markedly larger dispersion within the same target windows. Several baseline models overestimate or underestimate the peak magnitudes and exhibit phase shifts, with deviations becoming more pronounced at weeks 8 and 12 than at week 4. This trend is consistent with the error accumulation reported in Table V.

3) Computational Complexity and Inference Speed: Because diffusion-based models are inherently more computationally demanding than conventional point forecasters, the efficiency analysis focuses on the diffusion baselines, with results summarized in Table VI. Diffusion-TS and TSDE reach multihorizon targets through recursive week-by-week sampling, which gives them $O(MTnN)$ complexity and inference time that scales with the forecast horizon. DiffFlex instead achieves $O(TnN)$ complexity by generating all target weeks directly and in parallel, removing the horizon factor so that its inference time remains nearly constant across horizons and well below that of the diffusion baselines. Although conventional nondiffusion models remain faster at inference, this advantage is of limited practical relevance here, since model training is performed offline and MTLF operates on weekly to monthly lead times.

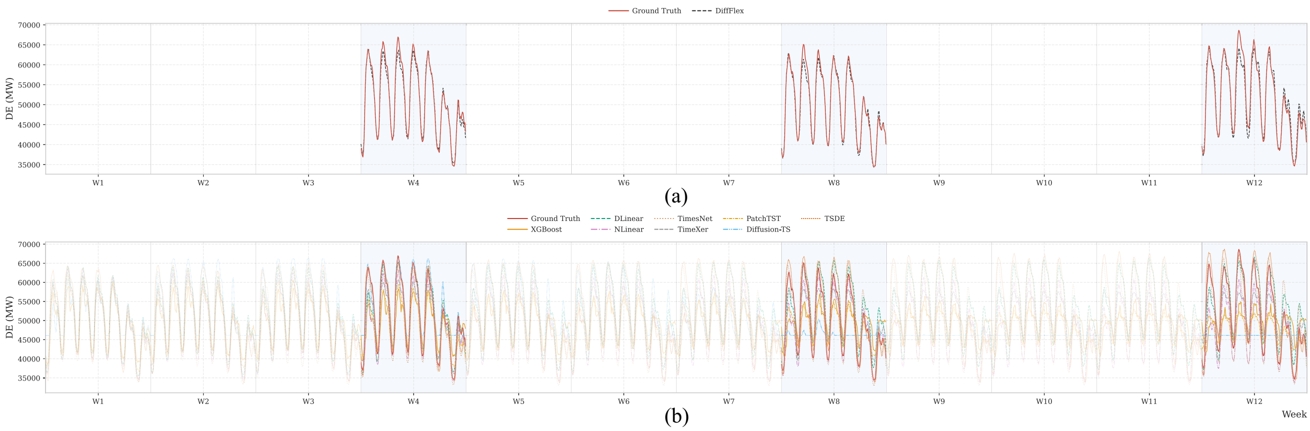


Fig. 12. Arbitrary and noncontiguous target intervals prediction on ENTSO-E dataset (4, 8, and 12 weeks ahead).

Under these moderate real-time requirements, the inference cost of DiffFlex remains practical.

V. Conclusion

This article proposes DiffFlex, a diffusion-based probabilistic framework that reformulates MTLF as a temporal outpainting problem, enabling direct prediction over arbitrary and potentially noncontiguous future intervals while quantifying predictive uncertainty.

Experiments on the ISO-NE, NYISO, and ENTSO-E datasets demonstrated that DiffFlex outperformed state-of-the-art baselines in both point and probabilistic accuracy, achieving average MAPEs of 8.88%, 9.71%, and 4.86%, respectively, while ablation studies confirmed the contribution of each proposed component. These results highlight the practical value of DiffFlex for risk-aware power system applications that require forecasts over flexible future intervals.

Future work will proceed in two directions. First, the model relies mainly on historical load observations and does not yet incorporate external drivers such as weather conditions and economic indicators, which may limit its accuracy when demand is strongly affected by such exogenous factors. Integrating these covariates into the model conditioning is therefore expected to further improve forecasting accuracy and operational interpretability. Second, the generalizability beyond electricity demand has not yet been validated. Testing the model on other domains, such as traffic flow, industrial process monitoring, or financial time series, and adapting the block design to their domain-specific temporal structures would help clarify the boundaries of its applicability.

References

[1] J. Ding, M. Wang, Z. Ping, D. Fu, and V. S. Vassiliadis, “An integrated method based on relevance vector machine for short-term load forecasting,” *Eur. J. Oper. Res.*, vol. 287, no. 2, pp. 497–510, 2020.

[2] L. Sun, W. An, Y. Chen, P. Zhao, and D. Ding, “An overview of distributed economic dispatch of microgrids: Advances and challenges,” *Syst. Sci. Control Eng.*, vol. 13, no. 1, 2025, Art. no. 2467077.

[3] M. Hasan et al., “A state-of-the-art comparative review of load forecasting methods: Characteristics, perspectives, and applications,” *Energy Convers. Manage., X*, vol. 26, 2025, Art. no. 100922.

[4] Y. Zhang, Z. Yuan, J. Ding, F. Guo, and J. Jin, “Intelligent and efficient fiber allocation strategy based on the dueling-double-deep Q-network,” *Front. Eng. Manage.*, vol. 12, no. 4, pp. 721–735, 2025.

[5] J. Liu, B. Qu, E. Wu, N. Li, Z. Wang, and D. Peng, “Analysis and prospect of wind power forecasting methods from multiple perspectives,” *Int. J. Netw. Dyn. Intell.*, vol. 4, no. 4, 2025, Art. no. 100024.

[6] H. Kim, A. Das, D. Wu, R. L. Kini, Z. Hou, and N. Lu, “Generative modeling for mid-term probabilistic load forecasting based on latent diffusion,” *Appl. Energy*, vol. 406, 2026, Art. no. 127273.

[7] L. Zhang, Z. Jiang, T. Ji, and Z. Chen, “Diffusion-based inpainting approach for multifunctional short-term load forecasting,” *Appl. Energy*, vol. 377, 2025, Art. no. 124442.

[8] R. Huang, P. Zhao, D. Cao, W. Hu, Q. Huang, and Z. Chen, “Complementary online learning network for probabilistic load forecasting against extreme weather,” *IEEE Trans. Ind. Inform.*, vol. 22, no. 4, pp. 3090–3101, Apr. 2026.

[9] J. Dong, M. Xu, J. Zhang, J. Li, Y. Sun, and S. Han, “Short-term power load forecasting under unstable data quality: A fuzzy deep neural network with LSTM and self-attention,” *IEEE Trans. Ind. Inform.*, vol. 21, no. 12, pp. 9945–9956, Dec. 2025.

[10] J. Hu et al., “Probabilistic net load forecasting based on transformer network and Gaussian process-enabled residual modeling learning method,” *Renewable Energy*, vol. 225, 2024, Art. no. 120253.

[11] Z. Tian, W. Liu, W. Jiang, and C. Wu, “CNNs-transformer based day-ahead probabilistic load forecasting for weekends with limited data availability,” *Energy*, vol. 293, 2024, Art. no. 130666.

[12] W. Zhuang, J. Fan, M. Xia, and K. Zhu, “A multi-scale spatial-temporal graph neural network-based method of multienergy load forecasting in integrated energy system,” *IEEE Trans. Smart Grid*, vol. 15, no. 3, pp. 2652–2666, May 2024.

[13] S. Zhou et al., “A load forecasting framework considering hybrid ensemble deep learning with two-stage load decomposition,” *IEEE Trans. Ind. Appl.*, vol. 60, no. 3, pp. 4568–4582, May/Jun. 2024.

[14] Y. Huang, H. Guo, E. Tian, and H. Chen, “Day-ahead probabilistic load forecasting: A multi-information fusion and noncrossing quantiles method,” *IEEE Trans. Ind. Inform.*, vol. 20, no. 8, pp. 10520–10529, Aug. 2024.

[15] J. Fan, W. Zhuang, M. Xia, W. Fang, and J. Liu, “Optimizing attention in a transformer for multihorizon, multienergy load forecasting in integrated energy systems,” *IEEE Trans. Ind. Inform.*, vol. 20, no. 8, pp. 10238–10248, Aug. 2024.

[16] A. Sharma and S. K. Jain, “A novel two-stage framework for mid-term electric load forecasting,” *IEEE Trans. Ind. Inform.*, vol. 20, no. 1, pp. 247–255, Jan. 2024.

[17] S. Zhang, R. Chen, J. Cao, and J. Tan, “A CNN and LSTM-based multi-task learning architecture for short and medium-term electricity load forecasting,” *Elect. Power Syst. Res.*, vol. 222, 2023, Art. no. 109507.

[18] F. Bayram, P. Aupke, B. S. Ahmed, A. Kassler, A. Theocharis, and J. Forsman, “DA-LSTM: A dynamic drift-adaptive learning framework for interval load forecasting with LSTM networks,” *Eng. Appl. Artif. Intell.*, vol. 123, 2023, Art. no. 106480.

[19] N. B. Behmiri, C. Fezzi, and F. Ravazzolo, “Incorporating air temperature into mid-term electricity load forecasting models using time-series regressions and neural networks,” *Energy*, vol. 278, 2023, Art. no. 127831.

[20] L. Luo, J. Dong, W. Kong, Y. Lu, and Q. Zhang, "Short-term probabilistic load forecasting using quantile regression neural network with accumulated hidden layer connection structure," *IEEE Trans. Ind. Inform.*, vol. 20, no. 4, pp. 5818–5828, Apr. 2024.

[21] Q. Ma, D. Wu, and X. Luo, "A review of deep learning-based power load forecasting methods," *Int. J. Netw. Dyn. Intell.*, vol. 4, no. 4, 2025, Art. no. 100027.

[22] N. Amjady and F. Keynia, "Mid-term load forecasting of power systems by a new prediction method," *Energy Convers. Manage.*, vol. 49, no. 10, pp. 2678–2687, 2008.

[23] B. Song, S. Zhao, L. Dang, H. Wang, and L. Xu, "A survey on learning from data with label noise via deep neural networks," *Syst. Sci. Control Eng.*, vol. 13, no. 1, 2025, Art. no. 2488120.

[24] Y. Zhan, R. Yang, J. You, M. Huang, W. Liu, and X. Liu, "A systematic literature review on incomplete multimodal learning: Techniques and challenges," *Syst. Sci. Control Eng.*, vol. 13, no. 1, 2025, Art. no. 2467083.

[25] W. Wang et al., "SinDiffusion: Learning a diffusion model from a single natural image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 5, pp. 3412–3423, May 2025.

[26] Y.-C. Cheng, C. H. Lin, H.-Y. Lee, J. Ren, S. Tulyakov, and M.-H. Yang, "InOut: Diverse image outpainting via GAN inversion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 11421–11430.

[27] L. Zhang, J. Wang, and J. Shi, "Multimodal image outpainting with regularized normalized diversification," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2020, pp. 3422–3431.

[28] S. Zhang et al., "Continuous-multiple image outpainting in one-step via positional query and a diffusion-based approach," in *Proc. Int. Conf. Learn. Representations*, 2024, vol. 2024, pp. 41204–41225.

[29] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 6840–6851.

[30] Z. Senane et al., "Self-supervised learning of time series representation via diffusion process and imputation-interpolation-forecasting mask," in *Proc. 30th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2024, pp. 2560–2571.

[31] F. Wiese et al., "Open power system data–frictionless data for electricity system modelling," *Appl. Energy*, vol. 236, pp. 401–409, 2019.

[32] K. Rasul, C. Seward, I. Schuster, and R. Vollgraf, "Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8857–8868.

[33] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 785–794.

[34] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?," in *Proc. AAAI Conf. Artif. Intell.*, 2023, vol. 37, pp. 11121–11128.

[35] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "TimesNet: Temporal 2D-variation modeling for general time series analysis," in *Proc. Int. Conf. Learn. Representations*, 2023, pp. 1–23.

[36] Y. Wang et al., "TimeXer: Empowering transformers for time series forecasting with exogenous variables," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 37, pp. 469–498.

[37] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *Proc. Int. Conf. Learn. Representations*, 2023, pp. 1–24.

[38] X. Yuan and Y. Qiao, "Diffusion-TS: Interpretable diffusion for general time series generation," in *Proc. Int. Conf. Learn. Representations*, 2024, pp. 1–30.

[39] Y. Wang et al., "Deep time series models: A comprehensive survey and benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, May 06, 2026, doi: 10.1109/TPAMI.2026.3690845.

Jia Ding received the bachelor's degree in business administration from Shanghai Jiao Tong University, Shanghai, China, in 2012, the master's degree in public policy from Georgetown University, Washington, DC, USA, in 2014, and the Ph.D. degree in business administration from Shanghai Jiao Tong University, Shanghai, China, in 2021.

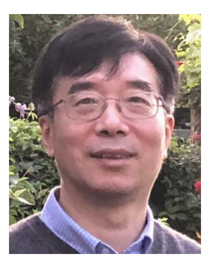
She is currently an Associate Professor with the School of Economics and Management, East China Normal University, Shanghai, China. Her research interests include business analytics, smart manufacturing, and operations management.

Ce Shi received the B.Eng. degree in building electricity and intelligence from Chang'an University, Xi'an, China, in 2018, and the M.Eng. degree in control engineering from Huazhong University of Science and Technology, Wuhan, China, in 2020.

He is currently a Senior Engineer with Yuanshi Intelligent Technology (Nantong) Company Ltd., Nantong, China. His research interests include AI for Science, smart manufacturing, and industrial embodied AI.

Junyang Jin received the B.S. degree in electrical and electronic engineering from the University of Liverpool, Liverpool, U.K., in 2013, the M.S. degree in control systems from Imperial College London, London, U.K., in 2014, and the Ph.D. degree in systems biology from the University of Cambridge, Cambridge, U.K., in 2018.

He is currently the CEO of Yuanshi Intelligent Technology (Nantong) Company, Ltd., Nantong, China, and the Director of the Institute of Industrial Data Analysis, HUST-Wuxi Research Institute, Wuxi, China. His research interests include system identification, network inference, machine learning, and smart manufacturing.



Zidong Wang (Fellow, IEEE) received the B.Sc. degree in mathematics from Suzhou University, Suzhou, China, in 1986, and the M.Sc. degree in applied mathematics and the Ph.D. degree in electrical engineering from Nanjing University of Science and Technology, Nanjing, China, in 1990 and 1994, respectively.

He is currently a Professor of dynamical systems and computing with the Department of Computer Science, Brunel University of London, Uxbridge, U.K. His research interests include dynamical systems, signal processing, bioinformatics, control theory, and applications.

Dr. Wang holds the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, and the William Mong Visiting Research Fellowship of Hong Kong. He was (or has served) the Editor-in-Chief for *International Journal of Systems Science*, *Neurocomputing*, and *Systems Science & Control Engineering*, and an Associate Editor for 12 international journals. He is a Member of the Academia Europaea and the European Academy of Sciences and Arts, an Academician of the International Academy for Systems and Cybernetic Sciences, and a Fellow of the Royal Statistical Society.