

G-Mamba: A Mechanism-Guided Graph State Space Network for Multi-Objective Prediction in Tyrosine Fermentation Process

Lihui Wang

University of Science and Technology of China

Chunyuan Wang

Suzhou Institute of Biomedical Engineering and Technology

Wenjing Li

Suzhou Institute of Biomedical Engineering and Technology

Min Chen

Tianjin Institute of Industrial Biotechnology

Kun Han

Suzhou Institute of Biomedical Engineering and Technology

Jianye Xia

Tianjin Institute of Industrial Biotechnology

Haixuan Sun

University of Science and Technology of China

Zhenying Zhao

zhaozy@sibet.ac.cn

University of Science and Technology of China

Article

Keywords: Fermentation Process Prediction, State Space Model (Mamba), Prior Knowledge Graph, Graph Convolutional Networks, Multi-Objective Forecasting

Posted Date: September 17th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10750845/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

G-Mamba: A Mechanism-Guided Graph State Space Network for Multi-Objective Prediction in Tyrosine Fermentation Process

Lihui Wang^{1,2}, Chunyuan Wang², Wenjing Li², Min Chen³, Kun Han², Jianye Xia³, Haixuan Sun^{1,2,*}, Zhenying Zhao^{1,2,*}

¹School of Biomedical Engineering (Suzhou), Division of Life Science and Medicine, University of Science and Technology of China, Hefei, 230026, China

²Suzhou Institute of Biomedical Engineering and Technology, Chinese Academy of Sciences, Suzhou, 215163, China

³Key Laboratory of Engineering Biology for Low-Carbon Manufacturing, Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences, Tianjin, 300308, China

*corresponding.author@email.: sunhx@sibet.ac.cn(H.Sun), zhaozy@sibet.ac.cn(Z.Zhao)

Abstract

Key core parameters in microbial fermentation processes typically rely on offline chemical analyses characterized by prolonged time delays, which fail to meet the imperatives of real-time, precision process control. Although existing data-driven modeling approaches have been deployed, they still suffer from the dual limitations of excessive computational overhead and a lack of mechanistic guidance when simultaneously capturing temporal evolution trends and complex, nonlinear spatial couplings among variables. To address these challenges, this paper proposes G-Mamba, a gray-box multi-objective prediction network that integrates prior mechanistic knowledge with the selective state space model. Structurally, the core of the model incorporates a bidirectional Mamba architecture to accurately capture the long-sequence temporal dynamics of fermentation processes while maintaining linear computational complexity. Concurrently, a topological adjacency matrix is explicitly formulated by incorporating prior knowledge of cellular metabolic pathways and operational control logic, which operates in tandem with an adaptive adjacency matrix and a graph convolutional network (GCN) to model both explicit kinetics and dynamic latent spatial couplings among variables. Furthermore, a lightweight gated fusion mechanism is designed to adaptively merge deep temporal and spatial features, effectively decoupling inter-variable metabolic competition and synergy. We conducted biomass and product concentration prediction experiments using a multi-batch dataset from an *Escherichia coli* tyrosine fermentation process. The experimental results demonstrate that G-Mamba consistently outperforms mainstream baseline models across both point prediction and sequence forecasting tasks under various input lengths, exhibiting significant synergistic performance enhancements in multi-objective forecasting scenarios while maintaining superior computational efficiency.

Key words: Fermentation Process Prediction, State Space Model (Mamba), Prior Knowledge Graph, Graph Convolutional Networks, Multi-Objective Forecasting

1. Introduction

As a core pillar of modern biomanufacturing and a vital component of global strategic emerging industries, the biological fermentation industry has extensively penetrated key sectors such as pharmaceuticals, food processing, biochemical engineering, and environmental protection. In industrial fermentation processes, maintaining cells in their optimal metabolic state to maximize the synthesis

efficiency of target products, while ensuring batch-to-batch quality consistency, constitutes the primary objective of fermentation process optimization and control^{1,2}. Achieving real-time monitoring of the fermentation state is the prerequisite for implementing precision dynamic regulation³. Among various process variables, key core parameters that directly reflect fermentation efficiency and cell growth status—such as product and biomass concentrations—typically serve as the essential baseline for dynamic control strategies⁴. Consequently, achieving accurate predictions of these critical parameters holds paramount significance for the construction of closed-loop control systems and the optimization of operational control policies^{5,6}.

Depending on specific control requirements, time-series forecasting tasks in fermentation processes are primarily categorized into "soft sensing" at the current timestamp^{7,8} and sequence forecasting of future trends^{9,10}. Current modeling methodologies are generally divided into mechanistic models and data-driven models¹¹. Mechanistic models are typically constructed based on the laws of mass, energy, and momentum conservation, as well as kinetic rate equations¹². However, when confronted with practical fermentation processes characterized by complex biochemical mechanisms and numerous unknown kinetic parameters, such models are often difficult to accurately reproduce, thereby limiting their practical viability in industrial production. Conversely, data-driven approaches rely entirely on historical process data, utilizing machine learning algorithms to achieve regression predictions of target variables^{13,14}. If mechanistic models are viewed as fully interpretable "white-box models," data-driven models fall into the category of "black-box models" that lack intrinsic physical mechanisms. In recent years, numerous studies have demonstrated that "gray-box models," constructed by incorporating prior mechanistic knowledge into data-driven algorithms, frequently exhibit superior and more robust predictive performance^{15,16}.

Physicochemical parameters in fermentation processes typically exhibit strong time-series characteristics, presenting prominent dynamic evolution patterns as the fermentation progresses; concurrently, complex nonlinear spatial couplings exist among different variables. In recent years, temporal networks based on the Transformer architecture have demonstrated excellent performance in forecasting tasks across various domains, as their self-attention mechanism can comprehensively capture both the temporal and spatial features of time-series data¹⁷. However, as the sequence length increases, the computational complexity of the Transformer grows quadratically due to its inherent architectural design, severely limiting its computational efficiency for long sequences. Recently, the state space model (Mamba)¹⁸ has demonstrated a massive advantage with its linear complexity in long-sequence modeling. Currently, numerous scholars have developed variant models based on Mamba and validated their tremendous application potential in the fields of text^{19,20} and image^{21,22} processing, thereby providing novel insights for the long-term temporal forecasting of fermentation processes.

To address the dual challenges of "computational inefficiency for long time-series" and "lack of mechanistic guidance for nonlinear spatial coupling" in the aforementioned fermentation processes, this paper proposes a Graph State Space Network (G-Mamba) model for the multi-objective joint prediction of the tyrosine fermentation process. The core motivation of this model lies in utilizing a bidirectional Mamba architecture to capture long-term temporal evolution trends while maintaining linear complexity; concurrently, a graph convolutional network (GCN) is introduced and combined with prior mechanistic knowledge to construct an adjacency matrix, explicitly modeling the spatial coupling among variables. Within the G-Mamba architecture, data from each temporal node are first mapped onto high-dimensional features through a linear layer, and deep temporal features are subsequently extracted using a Mamba encoding layer. Simultaneously, the input data are fed in parallel into a GCN branch to encode the spatially correlated features among process variables. Finally, the spatiotemporal features extracted by these two

parallel branches are dynamically integrated via a gated fusion mechanism, and the prediction results are output through a feedforward network. The primary contributions of this study are as follows:

- (1) We introduce a bidirectional Mamba architecture to capture the long-term temporal evolution of fermentation data with linear computational complexity.
- (2) Prior mechanistic knowledge is integrated to construct an explicit adjacency matrix, leveraging GCN to model the spatial coupling relationships among different process variables; concurrently, an adaptive adjacency matrix is introduced to mine the latent implicit correlations among variables within the data.
- (3) A lightweight gated fusion mechanism is designed to adaptively integrate deep temporal evolutionary features with spatially decoupled characteristics.
- (4) The effectiveness of the model is validated using a real-world tyrosine fermentation process as a case study. Through comparative experiments, the superiority of G-Mamba across various time-series forecasting tasks is demonstrated.

2. Related work

2.1 Fermentation Process Prediction

Considerable effort has been dedicated to applying machine learning and deep learning methodologies to parameter prediction tasks within fermentation processes. Li et al.²³ utilized a deep sparse autoencoder neural network for feature extraction on fermentation process data, establishing a multi-output support vector machine to conduct multi-objective modeling of the penicillin fermentation process. Barton et al.²⁴ constructed a Gaussian process regression model to predict yield in the simulation processes of penicillin and yeast fermentations. Conversely, several studies have attempted to incorporate mechanistic knowledge into data-driven approaches. Sun et al.¹⁰ proposed DSGraphformer, which extracts temporal dynamic feature graphs of fermentation data via a self-attention mechanism that slides along the time series, while simultaneously utilizing a graph attention algorithm to extract spatial static feature graphs from the data based on a prior knowledge graph. This gray-box methodology demonstrated excellent performance in predicting the chemical potency of the erythromycin fermentation process, providing an important reference for the integration of mechanistic knowledge and deep learning in this study. However, most temporal backbone networks in existing fermentation prediction frameworks rely on conventional architectures, which incur massive computational resource consumption when confronting extended industrial fermentation sequences, and still fall short in mining the latent implicit correlations among process variables.

2.2 Time-Series Forecasting Networks and State Space Models

2.2.1 Temporal Forecasting Networks

Early temporal forecasting networks were predominantly dominated by recurrent neural networks (RNNs) and their variants. For instance, Long Short-Term Memory (LSTM)²⁵ and Gated Recurrent Units (GRU)²⁶ mitigated the issues of long-term dependencies and vanishing gradients by introducing gating mechanisms, respectively. However, constrained by the inherent sequential data-processing structure of RNNs, these models struggle to achieve efficient parallel computing, resulting in high computational costs and slow convergence rates during long-sequence training.

Since its inception, the Transformer architecture has become a research hotspot in academia owing to its natural parallel processing capability and outstanding capacity to capture global dependencies. Nevertheless, the intrinsic quadratic time and space complexity ($O(N^2)$) of its self-attention mechanism consumes substantial computational and memory resources. To optimize its computational efficiency,

Informer¹⁷ introduced the ProbSparse self-attention mechanism. Autoformer²⁷ proposed a trend and seasonal feature extraction architecture based on series decomposition. Subsequent works, such as PatchTST²⁸ and PETformer²⁹, attempted to utilize patching mechanisms to optimize the computational efficiency of Transformers. Despite these advancements, when the input sequence becomes excessively long, attention-based models still face the risk of a rapid decline in forecasting robustness, and their positional encoding mechanisms encounter severe challenges within complex fermentation time series.

To counter the high complexity of Transformers, Multi-Layer Perceptron (MLP) architectures have regained considerable attention in time-series forecasting. Several studies have demonstrated that simple linear layers can even outperform complex Transformer models when predicting time-series data characterized by stability and periodicity³⁰. However, in the presence of high volatility, non-periodicity, and non-stationary patterns, MLPs struggle to effectively capture nonlinear dynamic evolutions³¹.

Recently, the inception of state space models (SSMs), particularly the Mamba architecture, has opened up new avenues for long-sequence prediction, achieving breakthroughs in natural language processing and computer vision. Numerous studies have attempted to adapt Mamba for time-series forecasting tasks. Ahamed and Cheng³² proposed TimeMachine, a Mamba-based framework that captures long-term dependencies and context across different scales through a two-stage embedding. S-Mamba³³ utilizes Mamba to learn inter-channel correlations while combining it with a feedforward network (FNN) to handle the temporal features of the data. Similarly, ms-Mamba³⁴ integrates multiple parallel Mamba modules with varying sampling rates to capture multi-scale temporal dependencies. Furthermore, MambaTS³⁵ optimizes the reconstruction efficiency of inter-variable channel features by eliminating redundant convolutional layers. Collectively, these pioneering works fully substantiate the tremendous potential of Mamba in long-term, high-efficiency forecasting tasks.

2.2.2 Graph Neural Networks and Multivariate Spatial Modeling

Graph Neural Networks (GNNs) have been widely adopted in recent years to process data with complex spatial geometric or topological features, playing a pivotal role in multivariate time-series forecasting. ST-GCN³⁶ innovatively combined GCN and RNN architectures to synergistically learn the complex coupling relationships within spatiotemporal data, demonstrating the application potential of GNNs in time-series problems such as traffic forecasting. Graph Attention Networks (GATs)³⁷ dynamically learn the spatial correlations among nodes via attention coefficients. Furthermore, Temporal Graph Networks (TGNs)³⁸ introduced evolving graph topologies to learn the dynamic, time-varying associations among nodes. These research achievements in spatiotemporal graph networks have provided the core inspiration for explicitly modeling the spatial coupling among variables in the fermentation process in this study.

2.3 Chapter Summary

In summary, despite the significant progress achieved by existing temporal networks and graph neural networks in multivariate forecasting, their application to complex industrial fermentation processes still confronts dual challenges. On the one hand, conventional temporal networks, such as Transformers, face bottlenecks of excessive computational complexity when capturing the long-term evolution trends of fermentation. On the other hand, existing spatiotemporal models struggle to simultaneously accommodate explicit mechanistic constraints based on biochemical kinetics and implicit association mining derived from data among fermentation variables. Consequently, how to construct a specialized forecasting architecture that possesses both linear computational efficiency and deep integration of spatiotemporal features remains an unresolved gap in the current field of intelligent fermentation. Addressing this research opportunity, this paper proposes the G-Mamba model, aiming to fuse state space models and graph

convolutional networks to provide an efficient gray-box solution for the precise multi-objective prediction of the tyrosine fermentation process.

3. Methodology

Figure 1 illustrates the overall architecture of G-Mamba, which primarily consists of three components. Within the Mamba temporal encoding module, the input data are first embedded via a linear layer and subsequently encoded temporally using a bidirectional Mamba block. In the GCN branch, a graph convolution operation is performed on the input data utilizing a prior knowledge adjacency matrix to extract the correlation features among variables. Ultimately, the outputs from both components are integrated through a gated fusion mechanism and processed by a feedforward network to generate the final output results.

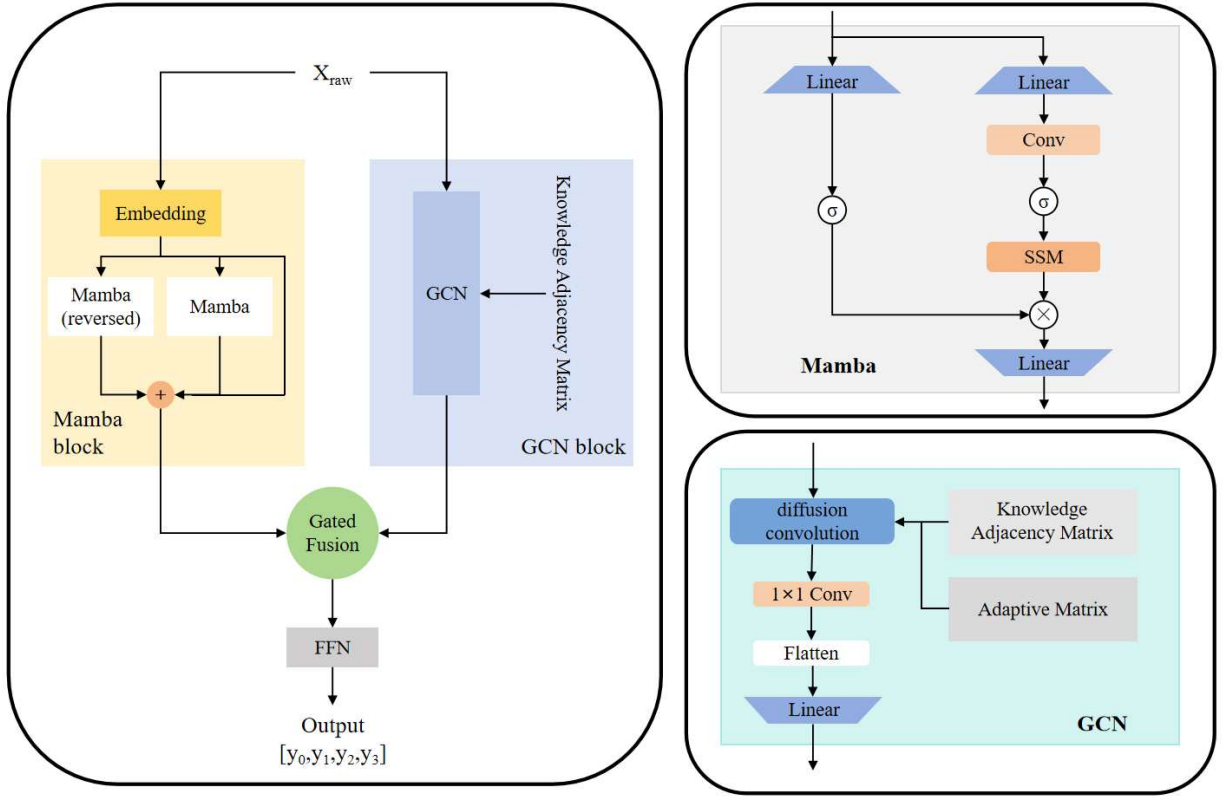


Figure. 1: Architecture of G-Mamba.

3.1 Mamba Temporal Encoding Block

The input variables are first transformed through an embedding layer. Given an input sequence $X \in \mathbb{R}^{L \times D}$ with a time step length of L and containing D variables, we apply a linear transformation along the variable dimension, as formulated in Eq. (1).

$$E = \text{Embedding}(X) \in \mathbb{R}^{L \times D_m}, \quad (1)$$

where D_m denotes the embedding dimension. This transformation projects the variable dimension into a higher-dimensional space while completely preserving the temporal features of the input sequence.

The Mamba layer handles temporal dependencies via a learnable step size (sampling rate) Δ .

$$E_m = \text{Mamba}(E) = \text{SSM}\left(\sigma\left(\text{Conv}\left(\text{Linear}(E)\right)\right)\right) \otimes \sigma\left(\text{Linear}(E)\right), \quad (2)$$

where SSM represents the State Space Model. Its specific computational operations are formulated as

follows:

$$h_t = \bar{A}h_{t-1} + \bar{B}x_t, \quad (3)$$

$$y_t = Ch_t + Dx_t, \quad (4)$$

where x_t represents the t -th element in the input sequence of the state space model, y_t denotes the corresponding output, h_t signifies the current hidden state, and h_{t-1} represents the preceding hidden state in the sequence. A , B , C , and D are learnable parameters, $\bar{A} = \exp(\Delta A)$, and $\bar{B} = (\Delta A)^{-1}(\exp(\Delta A) - I) \cdot \Delta B$, where Δ denotes the learnable step size (sampling rate), and I represents the identity matrix.

3.2 GCN Block

In this layer, the input data first undergoes diffusion convolution utilizing both the prior knowledge adjacency matrix and the adaptive matrix. Subsequently, a 1×1 convolution kernel is employed to perform high-dimensional channel mapping. Finally, the features across all channels are flattened and projected onto the designated dimension via a linear layer.

$$G = \text{Concat}_{k=0}^K (P_f^k X, P_b^k X, M_{apt}^k X), \quad (5)$$

$$P_f = M / \text{rowsum}(M), \quad (6)$$

$$P_b = M^T / \text{rowsum}(M^T), \quad (7)$$

$$E_g = \text{Linear}(\text{Flatten}(\text{Conv}(G))), \quad (8)$$

where M denotes the prior knowledge adjacency matrix, $\text{rowsum}(M)$ is normalized by its row sum. P_f represents the forward transition matrix, and P_b signifies the backward transition matrix. M_{apt} denotes the adaptive adjacency matrix. We refer to the method for constructing the adaptive matrix in GWnet³⁹, which is formulated as follows:

$$M_{apt} = \text{Softmax}(\text{ReLU}(E_1 E_2^T)), \quad (9)$$

where $E_1, E_2 \in \mathbb{R}^{N \times c}$, N represents the number of variables, c is a settable constant. A ReLU activation function is employed to eliminate weak connections, and a SoftMax function is applied to normalize the adaptive adjacency matrix.

In this module, the construction process of the prior knowledge adjacency matrix warrants particular attention. We formulate this matrix by combining known metabolic reaction pathways, bioreaction mechanisms, and fermentation control strategies. For instance, during the fermentation process of *Escherichia coli* for tyrosine production, side reactions generate a small amount of acetate, and respiration expels carbon dioxide, both of which lead to a decrease in the pH of the fermentation broth. However, among the measured variables in the experiment, there is no direct parameter that explicitly represents biological activity. Consequently, this study designates all dynamic process variables that significantly affect microbial biological activity as source nodes to construct directed edges pointing toward the pH response node. Furthermore, certain variables within the system are mathematically derived from other variables, such as the Oxygen Uptake Rate (OUR) and Carbon Dioxide Evolution Rate (CER), which are formulated as follows:

$$OUR = AirFlow \cdot \left(IN_{O_2} - \frac{IN_{N_2} \cdot OUT_{O_2}}{OUT_{N_2}} \right) / Volume, \quad (10)$$

$$CER = AirFlow \cdot \left(\frac{IN_{N_2} \cdot OUT_{CO_2}}{OUT_{N_2}} - IN_{CO_2} \right) / Volume, \quad (11)$$

Accordingly, it is necessary to construct directed edges pointing toward the OUR node from the following variables: aeration rate, inlet oxygen fraction, inlet nitrogen fraction, exhaust oxygen fraction,

exhaust nitrogen fraction, and fermentation broth volume. A similar operation is performed for the CER variable. Additionally, during the fermentation process, pH is regulated by adjusting the acid and alkali feeding rates; thus, both feeding rate variables point toward the pH variable. Similarly, the aeration rate, agitation speed, and vessel pressure are manipulated to regulate the dissolved oxygen (DO) concentration; hence, the dependencies of these three variables pointing toward the DO variable must also be established. All the aforementioned directed edges are set to 1 in the adjacency matrix, while the remaining entries are set to 0; specifically, if a directed edge exists from variable i to variable j , then $M_{ij} = 1$, otherwise $M_{ij} = 0$. Following these rules, the construction of the prior knowledge adjacency matrix is completed.

3.3 Gated Fusion Module

To achieve a deep integration of the temporal features extracted by the Mamba temporal encoding module and the inter-variable correlation features extracted by the GCN module, we employ a gated architecture that allows the model to adaptively combine these two distinct feature sets. The specific computational operations are formulated as follows:

$$E_f = zE_m + (1 - z)E_g, \quad (12)$$

$$z = \text{Softmax}\left(\text{FFN}(\text{Flatten}(X))\right), \quad (13)$$

where E_f represents the final fused feature, while E_m and E_g denote the features extracted by the Mamba temporal encoding module and the GCN module, respectively, and z signifies the gating vector. The gating weights are calculated directly from the original input sequence X via a lightweight linear projection. This design explicitly preserves the complete spatiotemporal information of the original input while significantly reducing the additional computational overhead of the gating mechanism.

4. Experimental studies

4.1 Dataset

The dataset utilized in this study is a private repository constructed specifically for the development of an intelligent monitoring model for *Escherichia coli* fermentation. The empirical data comprise 30 historical batches of the fermentation process of *Escherichia coli* for tyrosine production. The fermentation experiments were conducted in a 7.5 L bioreactor using the strain *Escherichia coli* ATCC 8739, wherein a variety of feeding strategies were implemented across different batches. The dataset encompasses 30 distinct process variables, featuring online variables sampled at 1-minute intervals and offline variables sampled at 2-hour intervals. Each batch within the dataset spans a total duration of 48 hours, which is partitioned into 24 data segments based on the sampling interval of the offline parameters. The specific variables and their corresponding definitions are detailed in Table 1.

4.2 Overview of Tyrosine Fermentation Process

Tyrosine is an aromatic amino acid produced through the fermentation of recombinant *Escherichia coli* modified via metabolic engineering, and it is widely utilized in the pharmaceutical, food, and chemical industries. The preparation process of tyrosine encompasses several sequential stages, including strain activation, primary seed cultivation, secondary seed amplification, deep submerged cultivation in the fermenter, cell separation, and product purification. The fermenter cultivation stage represents the critical phase of tyrosine production, and the level of process control within this link directly determines the accumulation efficiency and product quality of the final output.

The biomass (cell concentration) and tyrosine concentration in the fermentation broth serve as the core biological indicators for guiding process regulation. Achieving real-time prediction of these key

metrics is of profound significance for controlling the replenishment of various substrates during fermentation and modulating microbial metabolism. To enhance the controllability of the fermentation process and identify the optimal operating window, this study aims to integrate existing mechanistic knowledge with online process sensing data to construct a time-series forecasting model for the tyrosine fermentation process.

Table 1: Process Variables of The *Escherichia coli* Fermentation Dataset.

Num.	Category	Description	Units
1	Online	Time since the start of fermentation	h
2		pH	-
3		Dissolved oxygen(DO)	%
4		Temperature	°C
5		Stirring rate	Rmp
6		Airflow	L/min
7		Weight	kg
8		Weight of the feed tank	g
9		Weight of the base tank	g
10		Vessel Press	bar
11		Base-Flow	L/min
12		Acid-Flow	L/min
13		Antifoaming-Flow	L/min
14		Feed-Flow	L/min
15		Water-Flow	L/min
16		IN-CO ₂	%
17		IN-O ₂	%
18		IN-N ₂	%
19		OUT-CO ₂	%
20		OUT-O ₂	%
21		OUT-N ₂	%
22		MeOH	%
23		EtOH	%
24		Ar	-
25		Oxygen uptake rate(OUR)	mmol/(L·h)
26		Carbon dioxide evolution rate(CER)	mmol/(L·h)
27		Respiratory quotient(RQ)	-
28		Root mean square(RMS) of mass spectrometry baseline	-
29	Offline	Optical Density at 600nm(OD ₆₀₀)	-
30		Tyrosine concentration	g/L

4.3 Baseline Comparison and Result Analysis

The algorithm in this study was developed using Python 3.10, and the deep learning framework employed was PyTorch 2.1.0. All experiments were conducted on an NVIDIA GeForce RTX 3090 GPU. To ensure a fair evaluation, we compared our model against eight benchmarking time-series forecasting models across three distinct architectural families. These baselines include Transformer-based models:

Informer¹², Autoformer²², PatchTST²³, and PETformer²⁴; Linear-based models: DLinear and NLinear²⁵; and Mamba-based models: S-Mamba²⁸ and MambaTS³⁰. In our experiments, the dataset was partitioned into training, validation, and testing sets at a batch-wise ratio of 7:1:2. Mean Absolute Error (MAE) and Mean Squared Error (MSE) were adopted as the evaluation metrics for model performance. Assuming y represents the ground-truth value and $\hat{y}$ denotes the predicted value, the evaluation metrics are formulated as follows:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|, \quad (14)$$

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (15)$$

According to the control requirements of actual industrial production, we primarily established two categories of forecasting tasks. The first category is the point forecasting task, in which a set of online parameters is utilized to predict the corresponding set of offline parameters. For instance, given the online parameters from hour 0 to hour 2 as inputs, the model outputs the OD_{600} value and tyrosine concentration at the 2nd hour. The purpose of this forecasting task is to assist in the real-time adjustment of control strategies. The second category is the sequence forecasting task, which uses a segment of fermentation process data to predict the subsequent evolutionary trends of offline parameters. To evaluate the performance of the models in handling long sequences, sequence forecasting tasks under three distinct input lengths were configured. Specifically, these three tasks utilize 4, 6, and 8 hours of online data as inputs, respectively; meanwhile, all outputs correspond to the offline parameter values at 0, 2, 4, and 6 hours after the given input time window. That is, 2, 3, and 4 groups of online data are used to predict 4 groups of offline data, respectively. Table 2 presents the performance of the different algorithms across the aforementioned forecasting tasks.

Table 2: Experimental Results of Point and Sequence Forecasting Tasks.

Task	1->1		2->4		3->4		4->4		Avg	
Matrix	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Informer	<u>0.319</u>	0.400	0.390	0.421	0.379	0.429	<u>0.368</u>	0.421	0.364	0.418
PatchTST	0.454	0.514	0.524	0.559	0.561	0.612	0.570	0.604	0.527	0.572
PETformer	0.432	0.555	0.772	0.663	0.547	0.594	0.817	0.642	0.642	0.614
Autoformer	0.429	0.445	0.434	0.421	0.421	0.443	0.511	0.495	0.449	0.451
NLinear	0.624	0.546	1.636	1.086	1.422	0.963	2.157	1.113	1.460	0.927
DLinear	0.464	0.561	<u>0.327</u>	0.434	0.342	0.448	0.378	0.438	0.378	0.470
S-Mamba	<u>0.319</u>	<u>0.359</u>	0.343	<u>0.365</u>	<u>0.311</u>	<u>0.352</u>	0.371	<u>0.376</u>	<u>0.336</u>	<u>0.363</u>
MambaTS	0.428	0.423	0.525	0.493	0.546	0.514	0.529	0.514	0.507	0.486
G-Mamba	0.214	0.313	0.264	0.345	0.261	0.345	0.275	0.356	0.254	0.340

[1]The best results are highlighted in bold, and the second-best results are underlined.

[2]Avg denotes the average value of each method in the subtask.

As indicated in Table 2, G-Mamba consistently outperforms the alternative models across all tasks, establishing the efficacy of integrating Mamba-based temporal forecasting with GCN-driven prior knowledge spatial modeling. In the point forecasting task, architectures that incorporate inter-variable relationship modeling—such as Informer, S-Mamba, and G-Mamba—exhibit a clear performance advantage over the remaining models. This phenomenon highlights the critical importance of spatial modeling within multivariate, deeply coupled fermentation processes. Furthermore, by introducing a prior

knowledge adjacency matrix into spatial modeling, G-Mamba achieves superior predictive performance over other spatiotemporal models.

In the sequence forecasting tasks, it is noteworthy that the linear model, DLinear, delivers highly competitive results. Specifically, while its MAE performance is particularly outstanding, its MSE performance remains relatively mediocre. This discrepancy indicates that DLinear's predictions suffer from instability and a relatively high variance. Meanwhile, S-Mamba, which is also based on the Mamba architecture, demonstrates commendable performance; however, its framework utilizes the Mamba block to model inter-variable relationships, thereby failing to fully exploit the inherent advantages of Mamba in sequential modeling. In comparison, G-Mamba yields significantly superior and more robust predictive performance.

To comprehensively evaluate the performance of the model in multi-objective forecasting tasks, we compared the individual objective metrics under both multi-objective forecasting and separate single-objective forecasting configurations for S-Mamba, Informer, DLinear, and G-Mamba, all of which exhibited superior comprehensive performance. Table 3 presents the experimental results.

Table 3: Comparison of Multi-task and Single-task Forecasting Results

Task			1->1		2->4		3->4		4->4	
Matrix			MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
single	Objective 1 (OD ₆₀₀)	Informer	<u>0.179</u>	<u>0.301</u>	<u>0.194</u>	<u>0.284</u>	<u>0.202</u>	<u>0.311</u>	<u>0.184</u>	0.288
		DLinear	0.428	0.477	0.258	0.338	0.307	0.373	0.298	0.373
		S-Mamba	0.125	0.238	0.161	0.268	0.153	0.266	0.177	<u>0.295</u>
		G-Mamba	0.247	0.307	0.232	0.302	0.242	0.313	0.253	0.349
	Objective 2 (Tyrosine concentration)	Informer	0.440	0.477	0.578	0.621	0.696	0.678	0.636	0.601
		DLinear	<u>0.326</u>	<u>0.417</u>	0.317	0.395	0.363	0.419	0.346	0.395
		S-Mamba	0.575	0.571	0.540	0.468	0.747	0.512	0.588	<u>0.482</u>
		G-Mamba	0.315	0.369	<u>0.422</u>	<u>0.447</u>	<u>0.421</u>	<u>0.469</u>	<u>0.583</u>	0.580
Multi	Objective 1 (OD ₆₀₀)	Informer	<u>0.155</u>	0.294	<u>0.158</u>	<u>0.273</u>	<u>0.186</u>	<u>0.307</u>	<u>0.191</u>	<u>0.304</u>
		DLinear	0.487	0.592	0.248	0.377	0.308	0.401	0.364	0.386
		S-Mamba	0.119	0.210	0.133	0.240	0.137	0.232	0.155	0.250
		G-Mamba	0.189	<u>0.286</u>	0.242	0.312	0.249	0.313	0.254	0.327
	Objective 2 (Tyrosine concentration)	Informer	0.484	<u>0.505</u>	0.626	0.570	0.567	0.551	0.541	0.537
		DLinear	<u>0.440</u>	0.529	<u>0.406</u>	<u>0.491</u>	<u>0.375</u>	0.495	<u>0.391</u>	<u>0.490</u>
		S-Mamba	0.518	0.508	0.553	<u>0.491</u>	0.485	<u>0.471</u>	0.586	0.502
		G-Mamba	0.239	0.340	0.285	0.377	0.273	0.377	0.297	0.385

*The best results are highlighted in bold, and the second-best results are underlined.

As observed in Table 3, during single-objective forecasting, S-Mamba and Informer achieve superior accuracy in predicting biomass growth (OD₆₀₀), as OD₆₀₀ exhibits a relatively smooth trajectory that can be effectively captured purely by temporal autoregression. Conversely, predicting target product concentration (tyrosine) presents higher difficulty due to complex metabolic flux changes and operational feeding disturbances. In this context, DLinear achieves lower MAE but suffers from instability and high variance (poor MSE), while G-Mamba yields relatively balanced yet non-dominant metrics across single-task setups. Intriguingly, when transitioning to the multi-objective joint forecasting scenario, a distinctive architectural advantage of G-Mamba emerges. While S-Mamba and Informer maintain their

leading precision for OD_{600} , G-Mamba demonstrates a substantial performance leap in tyrosine prediction, outperforming all baseline models by a clear margin while maintaining highly competitive OD_{600} forecasting capability. This physiological phenomenon directly aligns with the intrinsic bio-mechanisms of amino acid fermentation: biomass density (OD_{600}) acts as the foundational dynamic capacity ("cellular factory") for target metabolite synthesis, whereas product accumulation exerts a potential metabolic burden back on cell growth. The GCN branch and prior knowledge matrix within G-Mamba effectively model this asymmetric metabolic dependency. By jointly optimizing both objectives, G-Mamba leverages OD_{600} as an informative auxiliary state to constrain and guide the nonlinear feature extraction for tyrosine concentration. Consequently, G-Mamba successfully overcomes the single-task bottleneck for difficult target parameters, achieving optimal multi-objective tradeoff and overall superiority in complex bioprocess soft-sensing.

4.4 Ablation Study

To verify the effectiveness of each individual module within G-Mamba in enhancing model performance, we conducted the following ablation studies:

- (1) All-ones Adjacency Matrix with Adaptive Matrix: Utilizing an all-ones matrix as the prior knowledge adjacency matrix input in conjunction with the adaptive matrix.
- (2) Prior Knowledge Adjacency Matrix Only: Utilizing solely the prior knowledge adjacency matrix without the adaptive matrix.
- (3) Adaptive Matrix Only: Utilizing solely the adaptive matrix without the prior knowledge adjacency matrix.
- (4) W/O GCN: Eliminating the GCN module entirely.
- (5) W/O Mamba Temporal Encoding: Eliminating the Mamba temporal encoding module entirely.
- (6) Unidirectional Mamba: Employing only a unidirectional Mamba block within the Mamba temporal encoding module.

The experimental results are summarized in Table 4.

Table 4: Ablation Results of G-Mamba.

Task	1->1		2->4		3->4		4->4		Avg	
Matrix	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
G-Mamba	<u>0.214</u>	0.313	0.264	0.345	0.261	0.345	<u>0.275</u>	<u>0.356</u>	0.254	0.340
(1)	0.225	0.326	0.283	0.358	0.311	0.369	0.314	0.372	0.283	0.356
(2)	0.236	0.313	0.326	0.385	0.335	<u>0.359</u>	0.349	0.414	0.311	0.367
(3)	0.229	0.323	0.333	0.394	0.386	0.432	0.239	0.336	0.297	0.371
(4)	0.212	0.323	0.359	0.438	0.363	0.423	0.377	0.433	0.328	0.404
(5)	0.313	0.409	0.417	0.499	0.423	0.496	0.504	0.555	0.414	0.490
(6)	0.216	<u>0.315</u>	<u>0.269</u>	<u>0.348</u>	<u>0.262</u>	0.345	0.285	0.379	<u>0.258</u>	<u>0.347</u>

[1]The best results are highlighted in bold, and the second-best results are underlined.

[2]Avg denotes the average value of each method in the subtask.

As indicated by the table, the introduction of the GCN module yields a significant improvement in the model's predictive accuracy. Model (3), which exclusively utilizes the adaptive adjacency matrix, exhibits a pronounced performance enhancement in sequence tasks with certain input lengths, yet delivers relatively mediocre results in alternative tasks. This performance instability is remarkably mitigated upon incorporating the prior knowledge adjacency matrix, enabling the model to achieve a well-balanced performance profile across all forecasting tasks. Compared with Model (4) that completely eliminates the

GCN module, G-Mamba reduces the average MSE and MAE by 22.6% and 15.8%, respectively. Meanwhile, Model (2) (prior knowledge adjacency matrix only) achieves average reductions of 5.2% and 9.2%, and Model (3) (adaptive adjacency matrix only) yields average reductions of 9.5% and 8.2%. These findings robustly demonstrate the efficacy and necessity of spatial relationship modeling, while highlighting the critical roles of both prior knowledge integration and adaptive adjacency matrix learning. Fig. 2 illustrates the heatmap representation of the prior knowledge adjacency matrix utilized in this study, alongside its superimposed visualization with the adaptive adjacency matrix learned by the model.

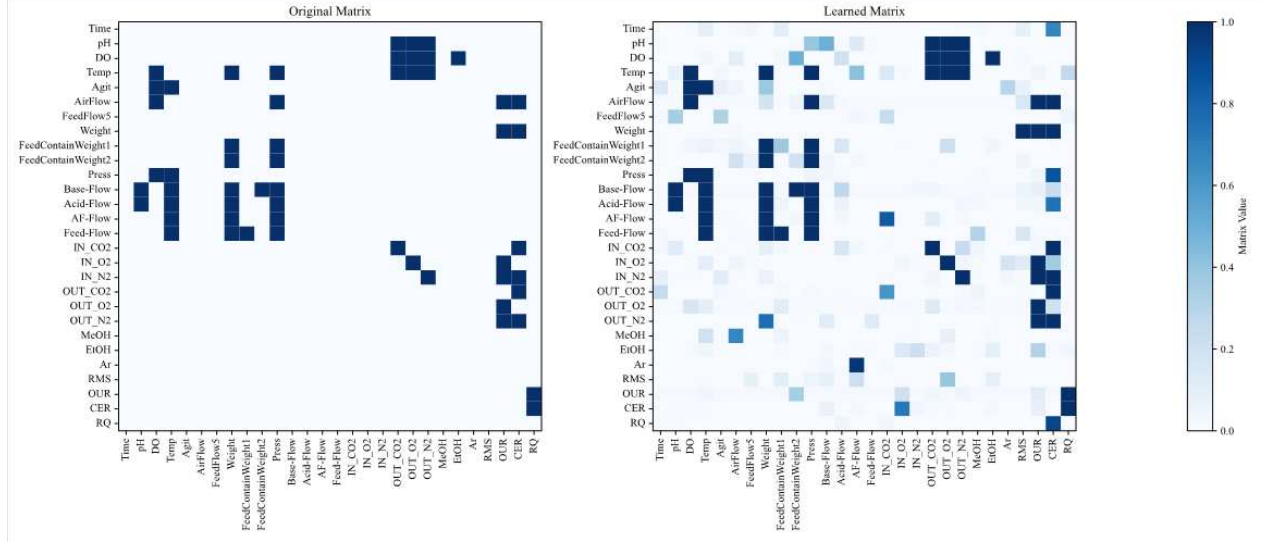


Fig. 2: Prior Knowledge Adjacency Matrix and The Learned Adjacency Matrix.

On the other hand, compared with Model (5) that removes the entire Mamba temporal encoding module, Model (6) (which employs a unidirectional Mamba encoder) achieves reductions of 37.7% and 29.2% in MSE and MAE, respectively. Furthermore, G-Mamba achieves even greater average reductions of 38.6% and 30.6%. These results comprehensively validate the superiority of the Mamba architecture in sequential data processing, forcefully demonstrating that the Mamba temporal encoding module serves as the core backbone for time-series forecasting in fermentation processes.

4.5 Model Efficiency

To evaluate the computational efficiency of the proposed model, we compared the GPU memory utilization and computational complexity of G-Mamba against other baseline models. All models were trained independently on a single NVIDIA GeForce RTX 3090 GPU with a consistent batch size of 24. Figure 3 illustrates the experimental results of the various models in the point forecasting task. A bubble chart is employed to visualize the measurements, where the vertical axis denotes the Mean Squared Error (MSE), the horizontal axis represents the computational complexity quantified in GFLOPs, and the bubble size is proportional to the allocated GPU memory.

The visualization results reveal that models based on the Mamba architecture generally exhibit lower computational complexity and memory footprints compared to Transformer-based models. Although Linear-layer-based models consume the least memory and computational resources, their predictive accuracy remains unsatisfactory. Overall, G-Mamba achieves highly precise forecasting while maintaining a relatively low consumption of computational resources.

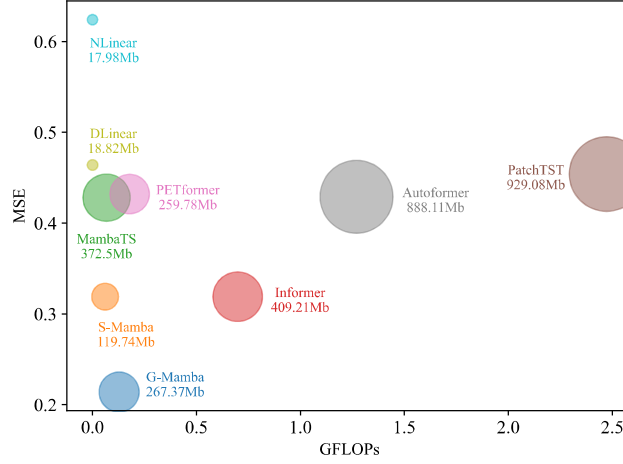


Fig. 3: Performance and Efficiency Comparison across Models

Additionally, Figure 4 presents a line chart illustrating the variation in computational complexity for various models as the input time-series length increases from 2 hours to 8 hours. Notably, since S-Mamba performs embedding along the variable dimension rather than the temporal dimension, its processing sequence length remains constant, resulting in an almost unchanged computational workload. As observed from the figure, the Mamba-based models consistently exhibit lower computational complexity than Transformer-based models across all input sequence lengths, and their growth rate in computational cost remains relatively marginal as the input sequence scales. Specifically, when the input time-series length increases from 2 hours to 8 hours, the increment in G-Mamba's computational complexity is only 18.4% of that of Informer, 5.3% of PatchTST, and 10.1% of Autoformer.

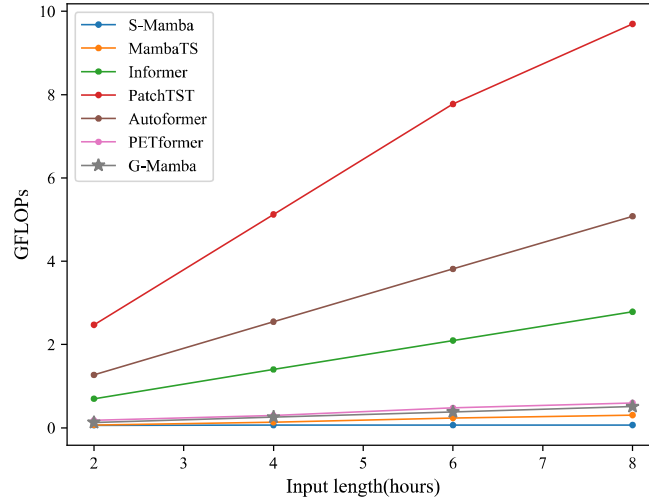


Fig. 4: Computational Complexity Evolution with Input Time Length

5. Conclusion

To address the dual bottlenecks of complex inter-variable nonlinear coupling and high computational complexity over long sequential windows in industrial fermentation processes, this paper proposes G-Mamba, a novel gray-box predictive architecture that integrates prior biochemical mechanistic knowledge with selective state space models. The proposed model was empirically evaluated on a real-world *Escherichia coli* tyrosine fermentation process. By leveraging a bidirectional Mamba temporal backbone, G-Mamba captures extended temporal evolution trends with linear computational complexity,

significantly outperforming conventional attention-based models in resource efficiency. By incorporating metabolic pathway topology and dynamic control dependencies into an explicit prior adjacency matrix—complemented by adaptive matrix learning—the GCN module successfully resolves complex dynamic dynamic inter-variable interactions. Furthermore, a lightweight gated fusion module ensures seamless integration of global temporal trends and decoupled spatial characteristics.

Empirical results demonstrate that G-Mamba achieves state-of-the-art predictive accuracy across both point sensing and multi-step sequence forecasting tasks under various time horizons. Crucially, in multi-objective prediction tasks, G-Mamba effectively captures the underlying biological metabolic coupling and resource competition between biomass growth (OD_{600}) and target product accumulation (tyrosine concentration), enabling cross-task performance reinforcement. Computational efficiency benchmarks further confirm that G-Mamba delivers high predictive precision with lightweight memory consumption and minimal FLOPs growth as input length increases.

Future work will focus on validating the generalizability of G-Mamba across a broader range of industrial bioprocesses and larger-scale multi-strain datasets, as well as investigating the closed-loop integration of G-Mamba predictions with model predictive control (MPC) frameworks for active fermentation dynamic optimization.

References

1. Zhu, X., Rehman, K. U., Wang, B. & Shahzad, M. Modern Soft-Sensing Modeling Methods for Fermentation Processes. *Sensors* **20**, 1771 (2020).
2. Meleiro, L. A. da C., Costa, A. C. da & Maciel Filho, R. Non-linear multivariable predictive control of an alcoholic fermentation process using functional link networks. *Braz. arch. biol. technol.* **48**, 7–18 (2005).
3. Feng, Y. et al. Real-time and on-line monitoring of ethanol fermentation process by viable cell sensor and electronic nose. *Bioresour. Bioprocess.* **8**, 37 (2021).
4. Liu, Z., Zheng, L., Yu, X., Zhuang, Y. & Wang, G. Multi-omics elucidation of resource allocation for enhanced ethanol production via precise glucose control in anaerobic *Saccharomyces cerevisiae* fermentation. *Synthetic and Systems Biotechnology* **12**, 20–31 (2026).
5. Wang, B. et al. Soft-sensor modeling for l-lysine fermentation process based on hybrid ICS-MLSSVM. *Sci Rep* **10**, 11630 (2020).
6. Foss, B. A., Johansen, T. A. & Sørensen, A. V. Nonlinear predictive control using local models — applied to a batch fermentation process. *Control Engineering Practice* **3**, 389–396 (1995).
7. He, Y.-L., Li, X.-Y., Ma, J.-H., Zhu, Q.-X. & Lu, S. Attribute-relevant distributed variational autoencoder integrated with LSTM for dynamic industrial soft sensing. *Engineering Applications of Artificial Intelligence* **119**, 105737 (2023).
8. Zhu, K., Zhao, S., Luan, X. & Liu, F. Optimal modeling of fermentation process using hybrid support vector regression. *Journal of Process Control* **150**, 103429 (2025).
9. Bolmanis, E., Dubencovs, K., Suleiko, A. & Vanags, J. Model Predictive Control—A Stand Out among Competitors for Fed-Batch Fermentation Improvement. *Fermentation* **9**, 206 (2023).
10. Sun, Y. & Yan, X. Spatial-Temporal relation inference Transformer combined with dynamic relationship and static causality for batch process modeling and the application of erythromycin fermentation. *Engineering Applications of Artificial Intelligence* **146**, 110271 (2025).
11. Chen, M. & Xia, J. Chapter 3 - Optimization parameters for efficient scale-up of fermentation process.

- in Scale-up and Chemical Process for Microbial Production of Plant-Derived Bioactive Compounds (eds Wei, Y., Ji, X.-J. & Cao, M.) 29–42 (Academic Press, 2024). doi:10.1016/B978-0-443-15584-0.00002-1.
12. Mears, L., Stocks, S. M., Albaek, M. O., Sin, G. & Gernaey, K. V. Mechanistic Fermentation Models for Process Design, Monitoring, and Control. *Trends in Biotechnology* **35**, 914–924 (2017).
 13. Xia, J., Long, D., Chen, M. & Chen, A. Optimization of fermentation processes in intelligent biomanufacturing: on online monitoring, artificial intelligence, and digital twin technologies. *Sheng Wu Gong Cheng Xue Bao* **41**, 1179–1196 (2025).
 14. Xia, J., Liu, J. & Zhuang, Y. Opportunities and challenges for fermentation optimization and scale-up technology in the artificial intelligence era. *Sheng Wu Gong Cheng Xue Bao* **38**, 4180–4199 (2022).
 15. Talib, A., Park, S., Im, P. & Joe, J. Grey-box and ANN-based building models for multistep-ahead prediction of indoor temperature to implement model predictive control. *Engineering Applications of Artificial Intelligence* **126**, 107115 (2023).
 16. Rogers, A. W., Song, Z., Ramon, F. V., Jing, K. & Zhang, D. Investigating ‘greyiness’ of hybrid model for bioprocess predictive modelling. *Biochemical Engineering Journal* **190**, 108761 (2023).
 17. Zhou, H. et al. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* **35**, 11106–11115 (2021).
 18. Gu, A. & Dao, T. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. Preprint at <https://doi.org/10.48550/arXiv.2312.00752> (2024).
 19. Anthony, Q., Tokpanov, Y., Glorioso, P. & Millidge, B. BlackMamba: Mixture of Experts for State-Space Models. Preprint at <https://doi.org/10.48550/arXiv.2402.01771> (2024).
 20. Liu, F., Liu, S., Chai, Y. & Zhu, Y. Enhanced Mamba model with multi-head attention mechanism and learnable scaling parameters for remaining useful life prediction. *Sci Rep* **15**, 7178 (2025).
 21. Zhong, X., Lu, G. & Li, H. Vision Mamba and xLSTM-UNet for medical image segmentation. *Sci Rep* **15**, 8163 (2025).
 22. Wang, Y., Jin, J., Chen, X., Wu, Z. & Zhang, L. A lightweight crack segmentation network based on the importance-enhanced Mamba model. *Sci Rep* **15**, 41549 (2025).
 23. Li, L. et al. Multi-output soft sensor modeling approach for penicillin fermentation process based on features of big data. *Expert Systems with Applications* **213**, 119208 (2023).
 24. Barton, M., Duran-Villalobos, C. A. & Lennox, B. Multivariate batch to batch optimisation of fermentation processes to improve productivity. *Journal of Process Control* **108**, 148–156 (2021).
 25. Hochreiter, S. & Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **9**, 1735–1780 (1997).
 26. Chung, J., Gulcehre, C., Cho, K. & Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. Preprint at <https://doi.org/10.48550/arXiv.1412.3555> (2014).
 27. Wu, H., Xu, J., Wang, J. & Long, M. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. in *Advances in Neural Information Processing Systems* vol. 34 22419–22430 (Curran Associates, Inc., 2021).
 28. Nie, Y., Nguyen, N. H., Sinthong, P. & Kalagnanam, J. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. Preprint at <https://doi.org/10.48550/arXiv.2211.14730> (2023).
 29. Lin, S., Lin, W., Wu, W., Wang, S. & Wang, Y. PETformer: Long-Term Time Series Forecasting via Placeholder-Enhanced Transformer. *IEEE Transactions on Emerging Topics in Computational Intelligence* **9**, 1189–1201 (2025).
 30. Zeng, A., Chen, M., Zhang, L. & Xu, Q. Are Transformers Effective for Time Series Forecasting?

- Proceedings of the AAAI Conference on Artificial Intelligence* **37**, 11121–11128 (2023).
31. Shao, Z. et al. Exploring Progress in Multivariate Time Series Forecasting: Comprehensive Benchmarking and Heterogeneity Analysis. *IEEE Transactions on Knowledge and Data Engineering* **37**, 291–305 (2025).
 32. Ahamed, M. A. & Cheng, Q. TimeMachine: A Time Series is Worth 4 Mambas for Long-Term Forecasting. *ECAI 2024* (2024) 392, 1688–1695 (2024).
 33. Wang, Z. et al. Is Mamba effective for time series forecasting? *Neurocomputing* **619**, 129178 (2025).
 34. Karadag, Y. M., Talaz, I., Gursel Dino, I. & Kalkan, S. ms-mamba: Multi-scale mamba for time-series forecasting. *Neurocomputing* **680**, 133226 (2026).
 35. Cai, X., Zhu, Y., Wang, X. & Yao, Y. MambaTS: Improved Selective State Space Models for Long-term Time Series Forecasting. Preprint at <https://doi.org/10.48550/arXiv.2405.16440> (2024).
 36. Yu, B., Yin, H. & Zhu, Z. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence* 3634–3640 (2018). doi:10.24963/ijcai.2018/505.
 37. Veličković, P. et al. Graph Attention Networks. Preprint at <https://doi.org/10.48550/arXiv.1710.10903> (2018).
 38. Rossi, E. et al. Temporal Graph Networks for Deep Learning on Dynamic Graphs. Preprint at <https://doi.org/10.48550/arXiv.2006.10637> (2020).
 39. Wu, Z., Pan, S., Long, G., Jiang, J. & Zhang, C. Graph WaveNet for Deep Spatial-Temporal Graph Modeling. Preprint at <https://doi.org/10.48550/arXiv.1906.00121> (2019).

Competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We express our gratitude to the editor and the esteemed reviewers for the invaluable feedback provided on our manuscript.

Data Availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and design: Zhenying Zhao, Lihui Wang; data collection: Min Chen, Jianye Xia, Wenjing Li, Lihui Wang, Kun Han; analysis and interpretation of results: Lihui Wang, Haixuan Sun, Chunyuan Wang; draft manuscript preparation: Lihui Wang; Funding acquisition: Haixuan Sun, Kun Han. All authors reviewed the results and approved the final version of the manuscript.

Funding

This study was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDC0250202) and the Key Core Technology "Open Competition" Program of the Suzhou Science and Technology Plan Project (Grant No. SYG2024155).