國立東華大學應用數學系

碩士論文

基於結構化狀態空間擴散模型與自適應固定秩克利金法之未知區域時空預測

Spatiotemporal Prediction of Unknown Areas Based on Structured State Space Diffusion and Adaptive Fixed Rank Kriging

研究生：許堯智
指導教授：黃灝勻 博士

中華民國 115 年 6 月

國立東華大學應用數學系

碩士論文

基於結構化狀態空間擴散模型與自適應固定秩克利金法之未知區域時空預測

Spatiotemporal Prediction of Unknown Areas Based on Structured State Space Diffusion and Adaptive Fixed Rank Kriging

研究生：許堯智

指導教授：黃灝勻 博士

中華民國 115 年 6 月

學位考試委員會審定書

國立東華大學＿＿＿＿＿＿＿＿＿＿系所

研究生　＿＿＿＿＿＿＿＿君所提之論文

經本委員會審查並舉行口試，認為

符合碩士學位標準。

學位考試委員會召集人＿＿＿＿＿＿＿＿簽章

指導教授＿＿＿＿＿＿＿＿簽章

委　　員＿＿＿＿＿＿＿＿簽章

委　　員＿＿＿＿＿＿＿＿簽章

委　　員＿＿＿＿＿＿＿＿簽章

系 主 任＿＿＿＿＿＿＿＿簽章
（所長）

中華民國　　年　　月　　日

國立東華大學
NATIONAL DONG HWA UNIVERSITY

學位論文原創性聲明書
DECLARATION OF THESIS/DISSERTATION ORIGINALITY

學位論文題目：
Thesis/Dissertation Title：

本人在此聲明，所呈交的學位論文是在指導教授黃灝勻的指導下，由個人獨立研究所完成之最終版本。本人對論文內容負責，除了文中已經標註引用處的內容外，論文不包含任何其他人已經發表或撰寫過的研究成果。對本研究及學位論文做出重要貢獻的個人和組織，均已在文中以明確方式標明。

該論文內容如有違反學術道德或學術規範的行為，如造假、變造、抄襲、研究成果重複發表或未適當引註、以違法或不當手段影響論文審查、不當作者列名等，本人願意承擔由此而產生的法律責任和法律後果。

I declare that the thesis/dissertation herein is the final version of my work, which is composed and accomplished individually under the guidance of my supervisor, Prof. Hao-Yun Huang. I am responsible for the contents of this thesis/dissertation: It contains no research result that was previously published or written by another person. Information derived from published and unpublished work of others has been acknowledged in the text, and a list of references is given . Any contribution made by other individual or organization is explicitly acknowledged in the thesis/dissertation.

If any research misconduct, including fabrication, falsification, or plagiarism in proposing, performing, or reviewing research, or in reporting research results, is discovered in my thesis/dissertation, I am willing to bear corresponding legal responsibilities and all the results therefrom.

聲明人 Declarant：

日期 Date：_____(yyyy/mm/dd)

1140303修訂

摘要

*** 本研究針對大規模時空觀測資料常見之缺失問題，提出一套整合結構化狀態空間擴散模型（Structured State Space Diffusion, SSSD^{S4}）與自適應固定秩克利金法（Adaptive Fixed Rank Kriging, AFRK）之時空建模框架。該方法旨在於缺漏比例高、資料結構複雜以及長序列預測等情境下，兼具時間依賴性刻畫能力與空間相關結構建模能力。研究首先利用 SSSD^{S4} 擷取長期時間序列的潛在動態，再結合 AFRK 以有效描述不同地點間的空間共變異結構，使模型能同時於時間與空間維度上強化資料重建品質。爲量化預測準確度，本研究採用平均平方預測誤差（Mean Squared Prediction Error, MSPE）作爲核心評估指標。實驗結果顯示，所提出之時空整合方法相較於僅進行時間建模後再以空間方法補全之傳統流程，於多個資料集皆可達成更低之 MSPE，證明其於長序列重建與時空依存結構提取方面具備顯著優勢。此結果亦反映 SSSD^{S4} 與 AFRK 的結合能有效強化模型的預測穩健性與泛化能力，爲未知位置之未來時空資料預測與插補研究提供一具可延伸性之框架。

關鍵字：時空模型、資料缺失、結構化狀態空間模型、擴散模型、自適應固定秩克利金法

Abstract

Keywords:

Contents

1 Introduction

With the rapid advancement of sensing technologies and digital infrastructure, researchers and the general public can now more easily access diverse types of data, such as traffic flow, water quality monitoring, and satellite remote sensing, through online platforms, real-time sensors, and various open-access databases. However, during the data collection process, instruments may still experience failures, malfunctions, or maintenance downtime, resulting in missing observations and thus incomplete spatiotemporal datasets. When attempting to predict future trends at locations with missing values, an important challenge arises that how to simultaneously account for the characteristics of temporal dynamics and the structural dependencies inherent in spatial data (Decorte et al., 2024).

To address this issue, this study proposes a methodology that integrates time series forecasting with spatial statistical modeling. We employ a Structured State Space Diffusion model (SSSD) (Alcaraz and Strodthoff, 2023) to capture temporal dependencies in the data, while incorporating Adaptive Fixed Rank Kriging (AFRK) (Tzeng and Huang, 2018) to characterize spatial correlations. Through this joint spatiotemporal framework, we aim to enhance the reconstruction of missing data and improve the predictive accuracy of future trends.

2 Related Works

Spatiotemporal data analysis plays a crucial role in many fields, including transportation engineering, hydrological monitoring, and environmental science. With the advancement of observation technologies and the widespread deployment of sensing devices, large-scale and high-resolution spatiotemporal data can now be continuously collected. However, during the data acquisition process, missing values and outliers may still arise due to sensor malfunctions, communication interruptions, or maintenance operations, which increase the difficulty of data analysis and predictive modeling (Decorte et al., 2024; Little and Rubin, 2002).

Previous studies have often focused separately on either temporal models or spatial models. Nevertheless, because spatiotemporal data simultaneously exhibit temporal dependencies and spatial correlation structures, recent research has increasingly developed spatiotemporal modeling approaches that integrate both temporal and spatial information in order to improve the accuracy of prediction and estimation (N. Cressie and Wikle, 2011; Shi et al., 2015).

In terms of temporal modeling, traditional approaches include the Autoregressive Integrated Moving Average (ARIMA) model (Box and Jenkins, 1976) and the Long Short-Term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997) etc. These methods are capable of capturing temporal dependencies in data; however, they still encounter limitations when dealing with long-term time series or complex nonlinear structures. More recently, the State Space Model (SSM) (Kalman, 1960) and the Structured State Space Diffusion (SSSD) model (Alcaraz and Strodthoff, 2023) have been proposed. These approaches demonstrate advantages in modeling long time series, offering capabilities for data imputation and forecasting while maintaining computational efficiency.

On the spatial modeling side, the Kriging method in spatial statistics (N. A. C. Cressie, 1993) has been widely applied to spatial interpolation and missing value estimation. However, when dealing with large-scale datasets, the computational cost of Kriging can be substantial, limiting its feasibility for real-time applications. To address this issue, Fixed Rank Kriging (FRK) (N. Cressie and Johannesson, 2008) and its extension, Adaptive Fixed Rank Kriging (AFRK), have been proposed (N. Cressie and Johannesson, 2008; Tzeng and Huang, 2018). These methods effectively reduce computational complexity and improve computational efficiency.

Temporal models emphasize the dynamic evolution of data, whereas spatial models capture the dependency structures among neighboring locations. Recent studies have increasingly recognized the importance of integrating these two dimensions, with applications in areas such as transportation demand forecasting, meteorological simulation, and environmental monitoring. Motivated by this context, the present study proposes a spatiotemporal framework that combines SSSD and AFRK to enhance predictive accuracy and robustness under scenarios with missing data.

2.1 Time Series Models

2.1.1 State Space Model

The State Space Model (SSM) is a class of mathematical models that describes dynamic systems or sequential data through a latent state vector. Initially proposed by Kalman (1960) within the fields of control theory and filtering, SSMs were designed to address optimal filtering and prediction problems for linear dynamic systems. Subsequently, researchers such

as Gu, Goel, and Ré (2022) extended this concept to deep learning architectures for long-sequence time series modeling, demonstrating that SSMs outperform traditional RNNs and LSTMs in capturing long-range dependencies and maintaining stable gradients (Gu, Goel, and Ré, 2022).

Given a one-dimensional input signal sequence $\boldsymbol{u}(t)$ and a one-dimensional output signal sequence $\boldsymbol{y}(t)$, the basic formulation of an SSM is

$$\begin{aligned} \boldsymbol{x}'(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t); \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t), \end{aligned} \tag{1}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^N$ is an N-dimensional latent state that maps the input $\boldsymbol{u}(t)$, $\boldsymbol{x}'(t) = \frac{d}{dt}\boldsymbol{x}(t)$ denotes its time derivative, $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ is the state matrix, and $\boldsymbol{B} \in \mathbb{R}^{N\times 1}$ and $\boldsymbol{C} \in \mathbb{R}^{1\times N}$ are the input and output matrices, respectively, characterizing how the input influences the state and how the state maps to the output. The term $\boldsymbol{D} \in \mathbb{R}$ is the feedthrough matrix, allowing the input to directly affect the output, and is typically set to zero. In deep learning contexts, these parameters are generally learned via gradient descent.

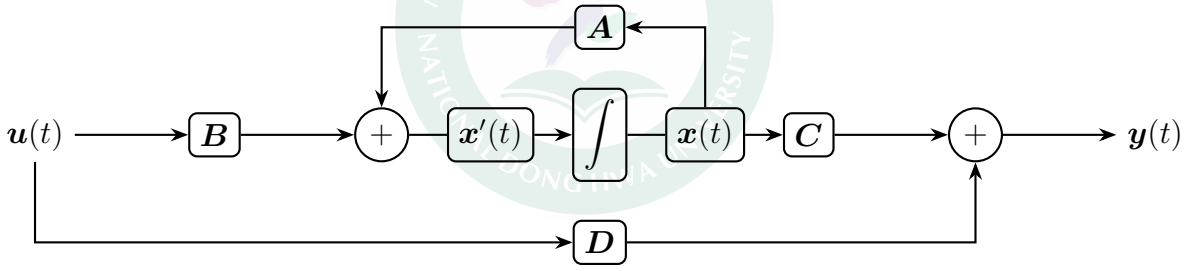

Figure 1: Typical State Space Model.

To address the practical issue in which the gradients of SSMs may increase or decrease exponentially with sequence length, Gu, Goel, and Ré (2022) introduced the HiPPO (High-order Polynomial Projection Operators) matrix (Gu, Dao, et al., 2020) to replace the original random matrix $\boldsymbol{A}$ in (1).

$$\boldsymbol{A}_{nk} = -\begin{cases} (2n+1)^{1/2}(2k+1)^{1/2}, & \text{if } n > k; \\ n+1, & \text{if } n = k; \\ 0, & \text{if } n < k. \end{cases} \tag{2}$$

As shown in (2), the HiPPO matrix dynamically evaluates the importance of each past time step as time evolves, enabling adaptive memory updates and retaining all historical information. By replacing the original random matrix $\boldsymbol{A}$ with the HiPPO matrix, the latent state $\boldsymbol{x}(t)$ can effectively store the historical information of the input sequence $\boldsymbol{u}(t)$ while avoiding gradient explosion or vanishing. Experimental results demonstrate that this design not only enhances computational stability but also significantly improves performance in long-sequence forecasting tasks (Gu, Goel, and Ré, 2022).

2.1.2 Structured State Space Model

Building upon the theory discussed in the previous section, Gu, Goel, and Ré (2022) proposed the Structured State Space Sequence Model (S4). This model aims to discretize the continuous-time State Space Model and embed it within deep learning frameworks for handling long sequential data.

S4 is grounded in the SSM formulation and discretizes (1) to enable its application to discrete input sequences. Let the step size be defined as Δ, then the discretized SSM can be written as follows:

$$\begin{aligned} \boldsymbol{x}_k &= \overline{\boldsymbol{A}}\boldsymbol{x}_{k-1} + \overline{\boldsymbol{B}}\boldsymbol{u}_k; \\ \boldsymbol{y}_k &= \overline{\boldsymbol{C}}\boldsymbol{x}_k, \end{aligned} \tag{3}$$

where $\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}}, \overline{\boldsymbol{C}}$ are the discrete approximations of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$, respectively.

$$\begin{aligned} \overline{\boldsymbol{A}} &= (\boldsymbol{I} - \Delta/2 \cdot \boldsymbol{A})^{-1}(\boldsymbol{I} + \Delta/2 \cdot \boldsymbol{A}); \\ \overline{\boldsymbol{B}} &= (\boldsymbol{I} - \Delta/2 \cdot \boldsymbol{A})^{-1}\Delta\boldsymbol{B}; \\ \overline{\boldsymbol{C}} &= \boldsymbol{C}. \end{aligned} \tag{4}$$

To reduce the computational cost of matrix operations, S4 diagonalizes the discretized matrices, expressing them in an equivalent form under a different basis:

$$(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}) \sim (\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}, \boldsymbol{V}^{-1}\boldsymbol{B}, \boldsymbol{C}\boldsymbol{V}), \tag{5}$$

where $\boldsymbol{V}$ is the basis transformation matrix. Moreover, the discretized SSM can be reformu-

lated in a convolutional form to enhance parallel computation efficiency:

$$\boldsymbol{y} = \overline{\boldsymbol{K}} * \boldsymbol{u}; \tag{6}$$

$$\overline{\boldsymbol{K}} \in \mathbb{R}^L := (\overline{\boldsymbol{C}}\overline{\boldsymbol{B}}, \overline{\boldsymbol{C}}\overline{\boldsymbol{A}}\overline{\boldsymbol{B}}, \cdots, \overline{\boldsymbol{C}}\overline{\boldsymbol{A}}^{L-1}\overline{\boldsymbol{B}}), \tag{7}$$

where $\overline{\boldsymbol{K}}$ denotes the SSM convolution kernel and L represents the convolution length.

To further reduce computational complexity, S4 employs the Normal Plus Low-Rank (NPLR) parameterization to express the matrix $\overline{\boldsymbol{A}}$ as:

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^* - \boldsymbol{P}\boldsymbol{Q}^\top = \boldsymbol{V}(\boldsymbol{\Lambda} - (\boldsymbol{V}^*\boldsymbol{P})(\boldsymbol{V}^*\boldsymbol{Q})^*)\boldsymbol{V}^*, \tag{8}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix, $\boldsymbol{P}, \boldsymbol{Q} \in \mathbb{R}^{N\times r}$ are low-rank matrices, and $\boldsymbol{V} \in \mathbb{C}^{N\times N}$ is a unitary matrix.

Based on the above theoretical components, S4 integrates the HiPPO matrix, discretization, diagonalization, convolution, and NPLR parameterization, enabling efficient computation and strong performance on long-sequence tasks (Gu, Goel, and Ré, 2022).

2.1.3 Diffusion Model

Diffusion models are a class of generative models that learn data distributions through a dual process consisting of a *forward diffusion process* and a *reverse denoising process* (Sohl-Dickstein et al., 2015). In the forward process, Gaussian noise is gradually injected into the original data until it approaches a standard normal distribution. The model is then trained to learn the reverse mapping of this process in order to reconstruct the data distribution. In recent years, diffusion models have been applied to time series imputation, where diffusion and denoising are performed only on missing segments to recover complete sequences under conditional observations (Alcaraz and Strodthoff, 2023).

Let $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$ denote an original data sample. The forward process is defined as a fixed-parameter Gaussian Markov chain that simulates progressively perturbed data generation:

$$\begin{cases} q(\boldsymbol{x}_1|\boldsymbol{x}_0) = \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}); \\ q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t\boldsymbol{I}), \end{cases} \tag{9}$$

where β_t is the variance schedule controlling noise intensity, and $\mathcal{N}$ denotes the normal distribution.

To reconstruct the data, the model must learn the reverse mapping of this process. The reverse process is defined as:

$$\begin{cases} p_\theta(x_0) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t); \\ p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \end{cases} \tag{10}$$

where $p(\boldsymbol{x}_T) = \mathcal{N}(\boldsymbol{x}_T; 0, \boldsymbol{I})$ denotes a standard normal distribution, while μ_θ and Σ_θ represent the mean vector and covariance matrix parameterized by a neural network with parameters θ, respectively.

However, directly modeling the mean of the reverse process, μ_θ, is often difficult to optimize and may lead to unstable convergence in practice. To address this issue, Ho, Jain, and Abbeel (2020) proposed a parameterization known as the Denoising Diffusion Probabilistic Model (DDPM), which reparameterizes $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ as

$$\mu_\theta(\boldsymbol{x}_t, t) = \frac{1}{\sqrt{\alpha_t}} \Big(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) \Big), \tag{11}$$

$$\Sigma_\theta(\boldsymbol{x}_t, t) = \sigma_t^2 \boldsymbol{I}, \qquad \sigma_t^2 = \beta_t \text{ or } \sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \beta_t, \tag{12}$$

where $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$. Under this framework, $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$ is used to estimate the random Gaussian noise added to $\boldsymbol{x}_t$ during the forward diffusion process, and $\boldsymbol{x}_{t-1}$ is reconstructed by removing the estimated noise component from $\boldsymbol{x}_t$.

This parameterization avoids the need to directly model complex high-dimensional data distributions, thereby substantially simplifying the training objective and improving numerical stability. Consequently, the sample at any diffusion step t can be expressed as a linear combination of the original data $\boldsymbol{x}_0$ and Gaussian noise:

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t} \boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t} \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{13}$$

which allows the model to randomly sample the time step t and noise $\boldsymbol{\epsilon}$ during training without iteratively computing intermediate diffusion steps. Since this representation transforms the original problem of directly fitting the data distribution into the estimation of Gaussian noise,

the training objective can be further simplified as

$$\min_{\theta} = \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{\epsilon}} \left[\left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta \left(\sqrt{\bar{\alpha}_t} \boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t} \boldsymbol{\epsilon}, t \right) \right\|^2 \right], \tag{14}$$

which corresponds to minimizing the Mean Squared Error (MSE) between the predicted noise and the injected noise.

For time series imputation, Alcaraz and Strodthoff (2023) further proposed a Conditional Diffusion Model that applies diffusion and denoising operations only to missing segments. During the training stage, the model receives partially observed sequences as conditional inputs to learn how to reconstruct the complete sequence. During the generation stage, the observed portions are fixed while the reverse denoising process is performed, allowing the missing segments to be reconstructed while preserving temporal consistency. By leveraging the stability and high-quality generation capability of diffusion models, this approach can effectively handle time series data with long-term dependencies or structurally missing segments.

2.1.4 Structured State Space Diffusion Model with S4 Layers

The Structured State Space Diffusion (SSSD) model, proposed by Alcaraz and Strodthoff (2023), is designed to integrate the generative stability of the DiffWave-based diffusion architecture (Kong et al., 2021) with the long-range dependency modeling capability of S4. The model is formulated as a conditional diffusion framework for time series imputation, in which noise is applied exclusively to the missing segments while the observed portions remain unperturbed. This strategy effectively prevents information leakage and preserves conditional consistency. By learning the reverse denoising process at each diffusion step, SSSD progressively reconstructs the missing data under fixed observations, achieving high-quality imputation and generation performance.

Building upon this foundation, Alcaraz and Strodthoff (2023) further introduce the Structured State Space Diffusion model with S4 layers (SSSD^{S4}) as an enhanced variant tailored for time series tasks. SSSD^{S4} retains the conditional diffusion design of SSSD but replaces the Bidirectional Dilated Convolutional Layers in the original DiffWave architecture with Structured State Space (S4) layers, thereby improving the model's ability to capture long-range temporal dynamics. The overall architecture is illustrated in Figure 2. Experimental

results demonstrate that this variant achieves more stable and accurate imputation across diverse missing patterns, outperforming diffusion models based on conventional convolutional or Transformer architectures (Alcaraz and Strodthoff, 2023).

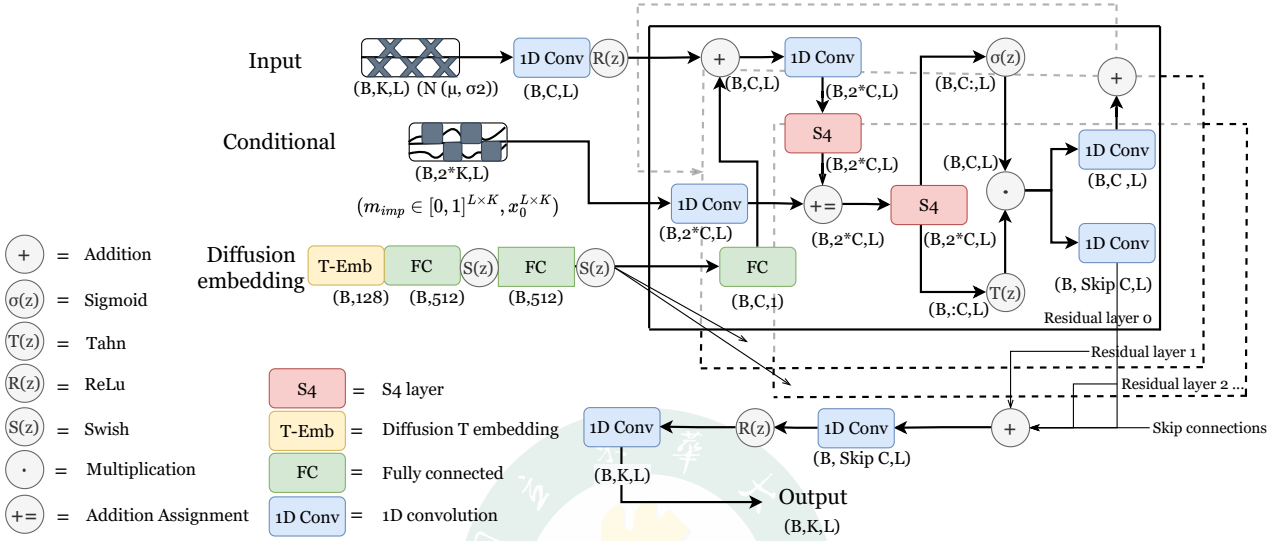

Figure 2: SSSD^{S4} model architecture.

2.2 Spatial Statistics

2.2.1 Kriging

Kriging originates from spatial statistics in geosciences and is a linear interpolation method used to estimate the values of a random field at unobserved spatial locations based on observed data (N. A. C. Cressie, 1993). The observed value $Z(\boldsymbol{s})$ at location $\boldsymbol{s}$ is modeled as:

$$Z(\boldsymbol{s}) = Y(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}), \qquad \boldsymbol{s} \in D \subset \mathbb{R}^D, \tag{15}$$

where $Y(\boldsymbol{s}) = \mu(\boldsymbol{s}) + \xi(\boldsymbol{s})$ represents a linear mean structure varying over space, and $\varepsilon(\boldsymbol{s})$ is a zero-mean random noise term that is uncorrelated with $Y(\boldsymbol{s})$. The covariance function of the noise is given by $C(\boldsymbol{s}, \boldsymbol{s}') = \mathrm{Cov}(\varepsilon(\boldsymbol{s}), \varepsilon(\boldsymbol{s}'))$, which may correspond to a non-stationary spatial covariance structure.

Traditional Kriging relies on the inversion of a full covariance matrix, leading to computational costs that scale cubically with the number of observations n, thereby creating a significant computational bottleneck as n increases (N. Cressie and Johannesson, 2008). Fixed Rank Kriging (FRK), along with its adaptive extension Adaptive Fixed Rank Kriging (AFRK), was introduced in this context to alleviate the computational burden associated with large-scale spatial data analysis.

2.2.2 Fixed Rank Kriging

To reduce the computational burden of Kriging, N. Cressie and Johannesson (2008) proposed Fixed Rank Kriging (FRK), which represents the random field using a finite set of basis functions, thereby approximating high-dimensional spatial random effects with low-dimensional random coefficients:

$$Y(\boldsymbol{s}) = \mu(\boldsymbol{s}) + \boldsymbol{f}(\boldsymbol{s})^\top \boldsymbol{w} + \xi(\boldsymbol{s}), \tag{16}$$

where $\boldsymbol{f}(\boldsymbol{s}) = (f_1(\boldsymbol{s}), \ldots, f_K(\boldsymbol{s}))^\top$ is a pre-specified K-dimensional vector of basis functions with $K \leq n$, $\boldsymbol{w} \sim N(\boldsymbol{0}, \boldsymbol{M})$ with $\boldsymbol{M}$ is an unknown nonnegative-definite matrix, and $\xi(\boldsymbol{s}) \sim \mathcal{N}(0, \sigma_\xi^2)$ represents fine-scale random noise. The corresponding covariance matrix can then be expressed as:

$$\mathrm{Cov}(Y(\boldsymbol{s}), Y(\boldsymbol{s}')) = \boldsymbol{f}(\boldsymbol{s})^\top \boldsymbol{M} \boldsymbol{f}(\boldsymbol{s}') + \sigma_\xi^2 \boldsymbol{I}(\boldsymbol{s} = \boldsymbol{s}'). \tag{17}$$

Since the rank of $\boldsymbol{f}\boldsymbol{M}\boldsymbol{f}^\top$ is typically much smaller than the number of observation points n, FRK can significantly reduce the computational cost of covariance matrix inversion, making it particularly suitable for large-scale remote sensing and environmental monitoring data.

2.2.3 Adaptive Fixed Rank Kriging

Building upon FRK, Tzeng and Huang (2018) further proposed Adaptive Fixed Rank Kriging (AFRK). The core idea is to allow the resolution of the basis functions to automatically adapt to the spatial distribution of the data, thereby providing greater flexibility in capturing spatial heterogeneity. This adaptive mechanism enables the model to allocate appropriate spatial resolution across different regions according to the underlying spatial variability.

The basis functions adopted in AFRK are multi-resolution spline basis functions (MRTS), which are constructed from thin-plate splines (TPS). TPS is a commonly used smoothing spline method that produces smooth functions by minimizing the sum of squared errors together with a smoothness penalty term (Wahba and Wendelberger, 1980; Green and Silverman, 1993). Based on TPS, Tzeng and Huang (2018) further constructed an ordered set of basis functions at multiple resolutions via eigen-decomposition, referred to as the MRTS.

To automatically adapt to the spatial distribution of the data and spatial heterogeneity, AFRK selects the number of basis functions according to the magnitude of the corresponding eigenvalues. Only the bases that explain most of the spatial variability are retained, allowing the model to capture the dominant spatial variations using a relatively small number of basis functions while improving computational efficiency.

In AFRK, the MRTS functions are defined as:

$$f_k(\boldsymbol{s}) = \begin{cases} 1, & \text{if } k = 1; \\ x_{k-1}, & \text{if } k = 2, \ldots, d+1; \\ \lambda_{k-d-1}^{-1} \times \left\{ \phi(\boldsymbol{s}) - \boldsymbol{\Phi}\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{x} \right\}' \boldsymbol{v}_{k-d-1}, & \text{if } k = d+2, \ldots, n, \end{cases} \tag{18}$$

where f_k is the k-th basis in $\boldsymbol{f}$, $\boldsymbol{X} \in \mathbb{R}^{N \times (d+1)}$ is the design matrix, with each row corresponding to the intercept and coordinates of an observation location $\boldsymbol{s}_i$, and $\boldsymbol{x} = (1, \boldsymbol{s}')' = (1, x_1, \ldots, x_d)'$. The matrix $\boldsymbol{\Phi}$ is defined as

$$J(f) = \boldsymbol{\alpha}'\boldsymbol{\Phi}\boldsymbol{\alpha}, \tag{19}$$

an $n \times n$ matrix with entries $\phi_j(\boldsymbol{s}_i)$, where $\phi(\boldsymbol{s})$ is defined by:

$$\phi_i(\boldsymbol{s}) = \begin{cases} \frac{1}{12}\|\boldsymbol{s} - \boldsymbol{s}_i\|^3, & \text{if } d = 1; \\ \frac{1}{8\pi}\|\boldsymbol{s} - \boldsymbol{s}_i\|^2 \log(\|\boldsymbol{s} - \boldsymbol{s}_i\|), & \text{if } d = 2; \\ -\frac{1}{8}\|\boldsymbol{s} - \boldsymbol{s}_i\|, & \text{if } d = 3; \end{cases} \tag{20}$$

and $\boldsymbol{v}_k$ denotes the k-th row of matrix $\boldsymbol{V}$, where $\boldsymbol{V}$ diag$(\lambda_1, \ldots, \lambda_n)\boldsymbol{V}'$ is the eigendecomposition of $\boldsymbol{Q}\boldsymbol{\Phi}\boldsymbol{Q}$, with $\boldsymbol{Q} = \boldsymbol{I} - \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'$. This approach adjusts the basis resolution according to data density, producing higher-resolution bases in densely sampled regions while maintaining smoother structures in sparse regions.

By retaining the computational advantages of low-rank approximation while introducing data-driven multi-resolution bases, AFRK effectively handles irregular sampling and regions with significant local variability. Compared to conventional FRK, AFRK demonstrates superior predictive performance on non-uniform and non-stationary spatial data.

3 Methodology

In this study, we propose a spatiotemporal forecasting framework that integrates deep time series modeling with spatial statistical modeling. The objective is to simultaneously exploit temporal and spatial information in order to improve reconstruction quality under missing data scenarios and enhance prediction accuracy for future time points.

The proposed framework utilizes the SSSD^{S4} model to capture the dynamic structure of the data along the temporal dimension, thereby extracting sequential temporal features. During model training, spatial dependence is incorporated through AFEK, allowing the model to account for spatial correlations among different locations.

In the inference stage, the temporal features predicted by SSSD^{S4} together with the spatial coordinates of unobserved locations are provided as inputs to AFRK. By integrating both temporal dynamics and spatial dependence, AFRK estimates the target values at unobserved locations.

The remainder of this section describes the problem formulation, the proposed spatiotemporal modeling framework, and the overall algorithmic procedure of the integrated method.

3.1 Problem Formulation

Consider a spatial domain consisting of a set of locations $\mathcal{S}$, which can be partitioned into the subset of observed locations $\mathcal{S}_{\text{observed}}$ with available measurements and the subset of unobserved locations $\mathcal{S}_{\text{unobserved}}$ with no recorded observations. For each observed location $\boldsymbol{s} \in \mathcal{S}_{\text{observed}}$, the target variable is fully observed over the temporal horizon $t \in \{1, \ldots, T\}$, and the corresponding observations are denoted as $y_t(\boldsymbol{s})$. In contrast, for each unobserved location $\boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}}$, no historical measurements exist within this time interval. The objective of this study is to construct a predictive model that leverages the historical spatiotemporal

features from the observed locations, denoted by $\boldsymbol{Y}_{1:T}(\mathcal{S}_{\text{observed}})$, in order to perform temporal extrapolation and spatial estimation of the target variable $\hat{y}_t(\boldsymbol{s}^*)$ at unobserved locations for future time points $t > T$.

The objective can be formulated as learning a mapping function

$$\hat{y}_t(\boldsymbol{s}^*) = f\left(\boldsymbol{Y}_{1:T}(\mathcal{S}_{\text{observed}}), \boldsymbol{s}^*\right), \qquad \boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}},\ t > T, \tag{21}$$

where $\boldsymbol{Y}_{1:T}(\mathcal{S}_{\text{observed}})$ denotes the collection of observed sequences from all known locations up to time T, and $\hat{y}_t(\boldsymbol{s}^*)$ represents the estimated value at an unknown location $\boldsymbol{s}^*$ for a future time step t.

3.2 Spatiotemporal Modeling Approach

To achieve the aforementioned mapping objective, this study first employs the SSSD^{S4} model to capture the dynamic dependencies along the temporal dimension, and subsequently integrates AFRK to fuse the extracted temporal features with spatial coordinates, enabling spatial interpolation at unobserved locations as well as future forecasting.

3.2.1 Temporal Modeling Based on SSSD^{S4}

To effectively capture the temporal dependencies inherent in the data, the SSSD^{S4} model is adopted as the temporal feature extractor. Its objective is to learn latent temporal features that simultaneously encode long-term dependencies and local dynamics from incomplete or noisy time series, leveraging the combination of the diffusion model and the S4 layers.

Using the observations at the observed locations $\mathcal{S}_{\text{observed}}$ as the training basis, the data are first standardized as

$$\tilde{y}_t(\boldsymbol{s}) = \frac{y_t(\boldsymbol{s}) - \mu(\boldsymbol{s})}{\sigma(\boldsymbol{s})}, \qquad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}, \tag{22}$$

where $\mu(\boldsymbol{s})$ and $\sigma(\boldsymbol{s})$ denote the mean and standard deviation of the time series at location $\boldsymbol{s}$,

defined as

$$\mu(\boldsymbol{s}) = \frac{1}{T}\sum_{t=1}^{T} y_t(\boldsymbol{s}), \tag{23}$$

$$\sigma(\boldsymbol{s}) = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(y_t(\boldsymbol{s}) - \mu(\boldsymbol{s})\right)^2}. \tag{24}$$

The standardized input sequences $\tilde{y}_t(\boldsymbol{s})$ are then fed into the SSSD$^{\text{S4}}$ model to learn the temporal dependency structure.

According to Ho, Jain, and Abbeel (2020), during the parameter optimization phase, a diffusion model approximates the true Gaussian noise $\boldsymbol{\epsilon}$ added in the forward diffusion process by the noise prediction term $\boldsymbol{\epsilon}_\theta$ output by the neural network, and updates the model parameters by minimizing the objective

$$\mathcal{L} = \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\|^2. \tag{25}$$

In this study, during the reverse diffusion process, $\boldsymbol{\epsilon}_\theta$ is further adjusted based on its spatial adjacency structure. Specifically, prior to the gradient update, AFRK is applied to the predictions of $\boldsymbol{\epsilon}_\theta$ at the same spatial locations to enforce spatial consistency and smoothness, thereby enhancing the ability of the reverse diffusion process to capture spatial structures accurately.

$$\hat{\boldsymbol{\epsilon}}_\theta = \text{AFRK}(\boldsymbol{\epsilon}_\theta) = \mu_{\boldsymbol{\epsilon}_\theta} + \boldsymbol{f}_{\boldsymbol{\epsilon}_\theta}^\top \hat{\boldsymbol{w}}_{\boldsymbol{\epsilon}_\theta} + \hat{\xi}_{\boldsymbol{\epsilon}_\theta}. \tag{26}$$

The predictions of the SSSD$^{\text{S4}}$ model adjusted via AFRK can be expressed as

$$\hat{\tilde{y}}_t(\boldsymbol{s}) = g_{\theta_{\text{S4+AFRK}}}\big(\tilde{y}_{1:t}(\boldsymbol{s})\big), \qquad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}, \tag{27}$$

where $g_{\theta_{\text{S4+AFRK}}}(\cdot)$ denotes the SSSD model employing the S4 architecture and adjusted via AFRK, and $\hat{\tilde{y}}_t(\boldsymbol{s})$ represents the model predictions on the standardized scale. Since the model is trained and evaluated on standardized data, the final predictions need to be transformed back to the original data scale via the inverse standardization:

$$\hat{y}_t(\boldsymbol{s}) = \sigma(\boldsymbol{s})\hat{\tilde{y}}_t(\boldsymbol{s}) + \mu(\boldsymbol{s}), \tag{28}$$

where $\mu(\boldsymbol{s})$ and $\sigma(\boldsymbol{s})$ denote the mean and standard deviation of the time series at location $\boldsymbol{s}$, respectively. This transformation restores the model outputs to the original measurement scale, facilitating subsequent analyses and spatial interpolation applications.

The SSSD model is trained using the MSE as the loss function, which iteratively updates the parameters $\theta_{\text{S4+AFRK}}$ to ensure that the extracted temporal features are stable and predictive. For tasks such as missing value imputation and multi-step future forecasting, the temporal representations learned by SSSD$^{\text{S4}}$ provide discriminative sequential embeddings, while the spatial regularization imposed by AFRK during training further enhances the overall accuracy and reliability of spatiotemporal estimation.

3.2.2 Spatial Modeling Based on AFRK

Within the proposed integrated framework, AFRK treats the temporal predictions at observed locations generated by SSSD$^{\text{S4}}$, denoted as $\hat{y}_t(\boldsymbol{s})$ for $\boldsymbol{s} \in \mathcal{S}_{\text{observed}}$, as input information along the temporal dimension. To remove scale differences across locations, the predictions at a given time point t are first standardized along the spatial dimension:

$$\tilde{y}_t(\boldsymbol{s}) = \frac{\hat{y}_t(\boldsymbol{s}) - \mu_t}{\sigma_t}, \qquad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}, \tag{29}$$

where μ_t and σ_t denote the spatial mean and standard deviation at time t over all observed locations $\mathcal{S}_{\text{observed}}$.

The standardized values $\tilde{y}_t(\boldsymbol{s})$ are then used as input to AFRK to perform conditional estimation at unobserved locations $\boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}}$. On the standardized scale, the predictions at unobserved locations can be expressed as

$$\hat{\tilde{y}}_t(\boldsymbol{s}^*) = \mu(\boldsymbol{s}^*) + \boldsymbol{f}(\boldsymbol{s}^*)^\top \hat{\boldsymbol{w}}_t + \hat{\xi}_t(\boldsymbol{s}^*), \tag{30}$$

where $\hat{\boldsymbol{w}}_t$ and $\hat{\xi}_t(\boldsymbol{s}^*)$ are the spatial conditional distribution parameters estimated by AFRK based on the observed location information $\tilde{y}_t(\boldsymbol{s})$.

Finally, the spatial predictions are transformed back to the original measurement scale via inverse standardization:

$$\hat{y}_t(\boldsymbol{s}^*) = \sigma_t \hat{\tilde{y}}_t(\boldsymbol{s}^*) + \mu_t. \tag{31}$$

thereby completing the reconstruction and forecasting of the global spatiotemporal field.

3.3 Model Training and Inference Algorithms

This section summarizes the operational workflow of the aforementioned spatiotemporal integration framework. Algorithm 1 describes the training procedure of SSSD^{S4} combined with AFRK, while Algorithm 2 presents the complete steps for model inference.

Algorithm 1 SSSD-AFRK Training Stage.

Require: $y_t(\boldsymbol{s}), \boldsymbol{s} \in \mathcal{S}_{\text{observed}}$; Max steps T; Learning rate η

$\tilde{y}_0(\boldsymbol{s}) = (y_t(\boldsymbol{s}) - \mu(\boldsymbol{s}))\,/\sigma(\boldsymbol{s})$
repeat
  $t \sim \text{Uniform}(\{1, \ldots, T\}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$
  $\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\tilde{y}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$
  $\mathbf{h} = \text{S4-Layer}(\boldsymbol{x}_t, t)$
  $\boldsymbol{\epsilon}_\theta = \mathbf{W}_o\mathbf{h} + \mathbf{b}_o$
  $\hat{\boldsymbol{\epsilon}}_\theta = \text{AFRK}(\boldsymbol{\epsilon}_\theta) = \mu_{\boldsymbol{\epsilon}_\theta} + \boldsymbol{f}_{\boldsymbol{\epsilon}_\theta}^\top \hat{\boldsymbol{w}}_{\boldsymbol{\epsilon}_\theta} + \hat{\xi}_{\boldsymbol{\epsilon}_\theta}$
  $\mathcal{L} = \|\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}_\theta\|^2$
  $\theta \leftarrow \theta - \eta\nabla_\theta\mathcal{L}$
until convergence

Here, $\boldsymbol{h}$ denotes the latent temporal features obtained after processing through the S4 layer, and $\mathbf{W}_o$ and $\mathbf{b}_o$ represent the learnable weight matrix and bias vector of the output projection layer, respectively.

Algorithm 2 SSSD-AFRK Inference Stage.

Require: Known samples $y_t(\boldsymbol{s})$; Unknown locations $\mathcal{S}_{\text{unobserved}} = \{\boldsymbol{s}_1^*, \ldots, \boldsymbol{s}_m^*\}$; Model θ

{Stage 1: Temporal Sequence Generation (at $\mathcal{S}_{\text{observed}}$)}

$\tilde{y}_0(\boldsymbol{s}) = (y_t(\boldsymbol{s}) - \mu(\boldsymbol{s}))/\sigma(\boldsymbol{s})$

$\boldsymbol{x}_T \sim \mathcal{N}(0, \boldsymbol{I})$

for $t = T, \ldots, 1$ **do**

  $\mathbf{h} = \text{S4-Layer}(\boldsymbol{x}_t, t)$

  $\boldsymbol{\epsilon}_\theta = \mathbf{W}_o\mathbf{h} + \mathbf{b}_o$

  $\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\right) + \sigma_t\mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}(0, \boldsymbol{I})$

end for

$\hat{y}_t(\boldsymbol{s}) = \sigma(\boldsymbol{s})\boldsymbol{x}_0 + \mu(\boldsymbol{s}), \quad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}$

{Stage 2: Spatial Interpolation via AFRK (at $\mathcal{S}_{\text{unobserved}}$)}

$\tilde{y}_t(\boldsymbol{s}) = (\hat{y}_t(\boldsymbol{s}) - \mu_t)/\sigma_t$

$\hat{\tilde{y}}_t(\boldsymbol{s}^*) = \text{AFRK}(\tilde{y}_t(\boldsymbol{s}) \mid \boldsymbol{s}^*), \quad \forall \boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}}$

$\hat{y}_t(\boldsymbol{s}^*) = \sigma_t\hat{\tilde{y}}_t(\boldsymbol{s}^*) + \mu_t$

return Full field $\hat{y}_t(\mathcal{S}_{\text{observed}} \cup \mathcal{S}_{\text{unobserved}})$

4 Experiments

This chapter presents the experimental design and setup for the spatiotemporal integration framework, aiming to evaluate its effectiveness and feasibility in forecasting future spatiotemporal values at unobserved locations. The experiments are conducted using two primary datasets, with detailed descriptions of model configurations, training and inference procedures, as well as the characteristics and partitioning of the datasets, in order to demonstrate the framework's performance under different data conditions.

4.1 Datasets

Two representative meteorological datasets are selected for this study, covering both ground-based observation station scales and global reanalysis scales, to assess the model's applicability and generalization ability under different spatiotemporal dynamics.

4.1.1 Weather2K

Weather2K is a recently proposed multivariate ground-based observation benchmark dataset, comprising measurements from thousands of meteorological stations across China, with a temporal resolution of 3 hours and covering near-surface meteorological variables such as air temperature, air pressure, humidity, and wind speed (Zhu et al., 2023). The open-source version, Weather2K-R, contains 1,866 stations and 13,632 consecutive time steps, featuring complete and regularly sampled time series without missing values. The dataset also provides constant location information (latitude, longitude, and elevation), which facilitates spatiotemporal modeling.

The Weather2K dataset exhibits heterogeneous spatial distribution and uneven station density, encompassing various geographical environments and climatic conditions, including urban areas, plains, and mountainous regions, thus providing rich spatiotemporal variation signals. These characteristics make Weather2K suitable for spatiotemporal interpolation, short-term sequence forecasting, and evaluations of model generalization and robustness.

In this study, observations from Weather2K-R between 00:00 on March 1, 2021, and 21:00 on August 31, 2021, are selected. Four variables, Air Temperature, Maximum Temperature, Minimum Temperature, and Relative Humidity, are used as the data sources for the experiments. Detailed descriptions of the variables are provided in Appendix A.

4.1.2 MERRA-2

MERRA-2 (Modern-Era Retrospective Analysis for Research and Applications, Version 2) is a global atmospheric reanalysis dataset provided by the NASA Goddard Earth Sciences Data and Information Services Center (GES DISC). It reconstructs the state of the global atmosphere since 1980 through data assimilation techniques that integrate numerical weather prediction models with multi-source observations, primarily from satellites (GMAO, 2015). In this study, we use the hourly, single-level, instantaneous assimilation diagnostic product M2I1NXASM (Version 5.12.4) as the analysis dataset.

MERRA-2 is archived and managed by the Distributed Active Archive Center (DAAC) at NASA Goddard Space Flight Center, providing globally consistent reanalysis data. Its data assimilation process integrates satellite, ground-based, and remote sensing observations, with

significant improvements over its predecessor MERRA in representing physical processes such as aerosols, radiative balance, and the hydrological cycle. It has been widely applied in climate trend analysis, extreme event studies, energy balance diagnostics, and numerical model evaluation. The M2I1NXASM product offers hourly instantaneous data with high resolution and representative climate signals, making it suitable for short-term spatiotemporal forecasting and statistical feature analysis.

For this study, data from the MERRA-2 M2I1NXASM (Version 5.12.4) product between 00:00 on July 1, 2023, and 23:00 on December 31, 2023, are selected. The Surface Skin Temperature variable is used as the primary variable for analysis. Detailed descriptions of the variables are provided in Appendix B. The experimental region is defined within the rectangular spatial domain of $43.0°$N to $55.0°$N latitude and $0.0°$E to $30.0°$E longitude, from which 1,000 observation points are selected as the data sources.

4.2 Environment and Computational Resources

To ensure the feasibility and reliability of training and inference experiments of the proposed spatiotemporal integration framework on large-scale datasets, a unified computational environment was established. Multiple existing software packages were integrated to support model development, training, and evaluation.

The SSSD model has been implemented in Python by its original authors (AI4HealthUOL, 2023), which is capable of effectively capturing long-range dependency structures in time-series data. AFRK was implemented in the R programming language by Wen-Ting Wang and released as the `autoFRK` package (Tzeng, Huang, Wang, Nychka, et al., 2021), providing stable and scalable spatial interpolation capabilities. To integrate these functionalities, this study reimplemented and encapsulated `autoFRK` as a Python package (Tzeng, Huang, Wang, and Hsu, 2025). The algorithm was further integrated with the original SSSD implementation, enabling a complete spatiotemporal modeling workflow within the PyTorch framework in Python.

All experiments, including model training, validation, and inference, were conducted on the Taiwania 2 high-performance computing platform (NCHC, 2018). Taiwania 2 provides high-performance GPU computing resources, along with large-capacity memory and high-speed storage systems. These resources enable efficient processing of high-resolution datasets

and long time series while ensuring computational stability and reproducibility of the experimental results.

4.3 Experimental Design

To clearly present the configurations of SSSD^{S4} and `autoFRK`, the following sections summarize the hyperparameters for training and inference as shown in Table 1.

Table 1: Model Hyperparameter Settings.

Model	**Hyperparameter**	**Value**
Training Configuration		
	Batch size	100
	Learning rate	0.0005
	Only generate missing	true
	Masking	forecast
	Missing k	—
SSSD^{S4}		
WaveNet	Input channels	—
	Output channels	—
	Residual layers	20
	Residual channels	20
	Skip channels	20
Diffusion step embedding	Input dimension	64
	Hidden dimension	128
	Output dimension	128
S4	Max sequence length	—
	State dimension	128
	Dropout	0.2
	Bidirectional	true
	Layer normalization	true

Model	Hyperparameter	Value
Diffusion	Diffusion steps (T)	100
	β_0	0.0001
	β_T	0.01
autoFRK		
	Method	fast
	Thin-plate spline method	rectangular

Table 2 details the parameter settings for each experiment, aiming to compare the model's performance under different training conditions. The input sequences are represented as $\boldsymbol{X} \in \mathbb{R}^{N\times T\times C}$, where N denotes the number of spatial locations, T is the length of the time series, and C corresponds to the input variable dimensions or channels. In the datasets, the temporal split between training and testing time steps is 0.9 and 0.1, respectively, while the spatial split between observed and unobserved locations is 0.8 and 0.2.

The experimental design considers whether AFRK is integrated during the training phase, and a temporal reshaping mechanism applied to the input variables. By comparing the baseline SSSD^{S4} model with the AFRK-integrated SSSD$^{S4\,+\,AFRK}$ model, we evaluate whether incorporating AFRK during training enhances spatial dependency and examine its impact on imputing unobserved locations during inference. The temporal reshaping mechanism transforms the original sequence of shape (N, T, C) into $(N, T/p, C \times p)$, i.e., aggregating consecutive p time steps into the feature channels. This procedure is used to verify whether converting temporal information into the channel dimension can improve the model's capability in capturing both short- and long-term spatiotemporal dependencies.

Table 2: Experimental Parameter Configurations (Weather2K / MERRA-2).

Control Variables	Expt. 1	Expt. 2	Expt. 3	Expt. 4
Iterations	4,000	4,000	4,000	4,000
Training Strategy	SSSD$^{S4\,+\,AFRK}$	SSSD^{S4}	SSSD$^{S4\,+\,AFRK}$	SSSD^{S4}
Observed Locations	1,492 / 960	1,492 / 960	1,492 / 960	1,492 / 960
Unobserved Locations	374 / 240	374 / 240	374 / 240	374 / 240

Control Variables	Expt. 1	Expt. 2	Expt. 3	Expt. 4
Temporal Reshaping	false	false	true	true
Aggregated Time Steps p	—	—	8 / 24	8 / 24
Input Channels	4 / 1	4 / 1	32 / 24	32 / 24
S4 Max Sequence Length	1,328 / 3,984	1,328 / 3,984	166 / 166	166 / 166
Missing k	144 / 432	144 / 432	18 / 18	18 / 18

The spatial distribution of observed and unobserved stations in the datasets is illustrated in Figure 3 and Figure 4. Among the sampled locations, blue circles represent observed stations, while green triangles denote unobserved stations.

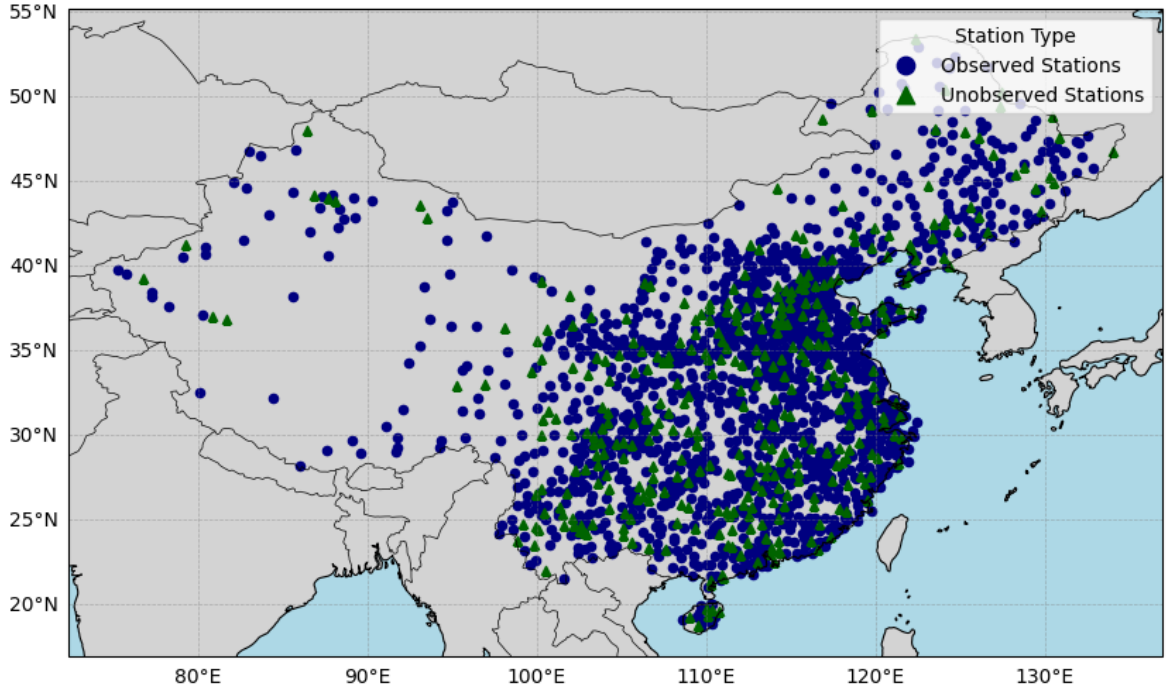

Figure 3: Spatial distribution of Weather2K 3-Hourly observation stations.

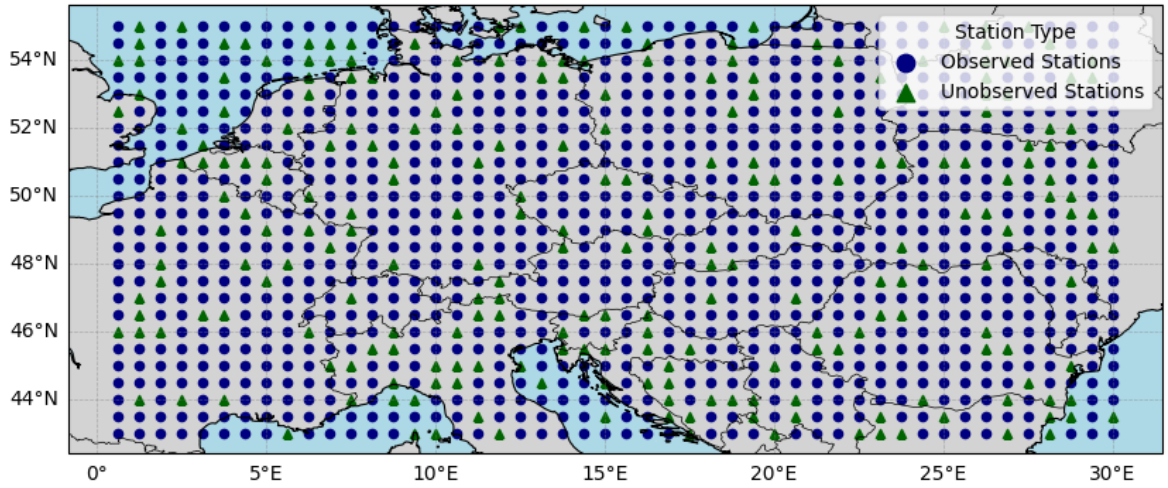

Figure 4: Spatial distribution of MERRA-2 Hourly observation stations.

5 Experimental Results

In this study, the Mean Squared Prediction Error (MSPE) is employed as the primary evaluation metric to assess the accuracy of the model's predictions at unobserved locations for future time points. Table 3 presents the MSPE performance of the integrated spatiotemporal framework, $\text{SSSD}^{\text{S4 + AFRK}}$, in comparison with the approach that employs only SSSD^{S4} followed by AFRK for spatial imputation. The table also reports the number of MRTS basis functions selected by AFRK during the inference phase.

Table 3: MSPE for different temporal predictions and the number of MRTS basis functions selected by AFRK during inference for each dataset.

Metrics	**Expt. 1**	**Expt. 2**	**Expt. 3**	**Expt. 4**
Weather2K				
Unobserved & Future	**73.0718**	75.0860		100.2486
Unobserved & Past	23.4334	**19.1816**		21.2375
Observed & Future	**78.6897**	85.0839		274.7339
# of MRTS Basis	382	382		178
MERRA-2				

Metrics	Expt. 1	Expt. 2	Expt. 3	Expt. 4
Unobserved & Future	1616.0968	1555.5420		**1373.6186**
Unobserved & Past	1.5091	**1.1911**		3.4276
Observed & Future	1616.0968	**1566.5607**		1987.3689
# of MRTS Basis	301	301		111

***Table 3 所示，整合的 $SSSD^{S4+AFRK}$ 時空框架，相較於單獨使用 $SSSD^{S4}$ 再以 AFRK 填補空間欄位的策略，於兩資料集上皆獲得更低的 MSPE 。此結果顯示，本研究提出的整合方法能有效捕捉複雜的時空依賴關係，展現高度的可靠性與穩健性。

*** 表 3 的結果表明，單獨使用 $SSSD^{S4}$ 或 AFRK 在特定任務上表現尚可，但在未觀測位置的未來預測上，結合兩者的時空整合框架普遍獲得最低的 MSPE。這反映出整合方法能同時捕捉時間序列中的長程依賴性與空間結構的相關性，顯著提升未觀測位置的預測精準度。不同實驗間的比較亦顯示，整合框架在兩個資料集上皆保持穩定優勢，證明其在大規模時空氣象資料建模中的實務可行性與泛化能力。

6 Conclusion

*** 本研究提出一套結合時間序列生成模型與空間統計插值方法的整合式時空預測框架，由 $SSSD^{S4}$ 負責時間結構建模，並以 AFRK 建立低秩空間基底與跨區域插值機制，XXXX。透過在 Weather2K 與 MERRA-2 兩個異質性高、空間分佈差異顯著的資料集上驗證，本研究顯示該整合架構能在未觀測地點的未來時序預測任務中取得穩定且優於基線模型的表現，特別是在稀疏觀測與跨區域泛化的情境下展現良好適應性。此外，本研究亦指出 $SSSD^{S4}$ 能有效捕捉長期時間依賴，而 AFRK 所引入的空間低秩結構可提供額外的空間一致性補償，使整合模型在高不確定性或觀測缺失的狀況下仍維持合理預測。本研究驗證了結構化時間模型與低秩空間模型結合的可行性，並爲未來建構可泛化、可解釋且跨場景適用的時空預測系統奠定基礎。

6.1 未來工作

儘管本研究提出的時空整合框架已在未觀測地點的未來時序預測任務中展現優異性能，仍存在多個潛在改進方向，以進一步提升模型的廣泛適用性與理論價值。

6.1.1 跨區域泛化能力

本研究主要驗證了模型於中國地區地面觀測資料及全球再分析資料的表現。未來可擴展至更多區域性資料集，包括歐洲、北美與東南亞等，藉以檢驗模型在不同氣候型態、地形環境以及資料品質條件下的穩定性與泛化能力，並分析不同地理與氣候條件對 $SSSD^{S4}$ 時間建模與 AFRK 空間插值效果的影響。

6.1.2 空間建模擴展

AFRK 目前所構建的空間結構具有一定彈性，但仍爲固定形式。未來研究可考慮引入動態核函數或可調節的空間權重機制，使其能隨時間變化捕捉局部或區域性氣象結構的轉變。此外，非平穩空間模型或區域化建模方法亦可進一步改善 AFRK 對高度異質化或稀疏觀測網格的適應性，從而增強整合框架在極端氣候事件或非典型時空模式下的表現。

6.1.3 缺失機制與模型穩健性

除了既有的未來時序預測（Time Forecasting, TF）外，後續可延伸探討 $SSSD^{S4}$ 與 AFRK 在多種缺失機制下的表現差異，包括隨機缺失（Random Missing, RM）、隨機區塊缺失（Random Block Missing, RBM）以及大範圍遮蔽型缺失（Blackout Missing, BM）。在 RM 與 RBM 中，缺失具有高隨機性與局部破碎性，使時空相關結構受到更強破壞；AFRK 的低秩空間結構可望提供額外的空間一致性約束，以補償 $SSSD^{S4}$ 在時序生成中所面臨的高不確定性。相較之下，BM 屬於大範圍且多時間步的連續遮蔽，對純時間模型與純空間模型皆構成挑戰，因此未來可進一步設計結合 AFRK 空間基底與 $SSSD^{S4}$ 擴散機制的協同重建策略，以提升模型於不同缺失模式下的穩健性。

6.1.4 時空協同與生成模型整合

針對 SSSD^{S4} 與 AFRK 的結合，未來可探索更緊密的時空協同策略。例如，可在 SSSD^{S4} 的時間序列輸出中引入基於 AFRK 空間結構的正則化項，以強化時間預測與空間插值的相互約束，或透過可微分空間插值模塊實現端到端訓練，使模型在未觀測位置的預測精度進一步提升。此外，隨著大規模生成式氣象模型的發展，整合擴散模型或其他大型 AI 氣象模型的生成能力，亦可在高缺失率資料補全、物理一致性預測及極端事件建模上提供額外增益。

6.1.5 模型可解釋性與不確定性量化

模型的可解釋性與不確定性量化仍爲重要研究方向。未來可進一步開發時空特徵重要度分析方法，以評估 SSSD^{S4} 的時間依賴表達與 AFRK 的空間插值貢獻，同時估計未觀測位置預測的不確定性與罕見事件的敏感度，從而提升模型在科學研究及實務氣象預報系統中的可靠性與可操作性。

References

AI4HealthUOL (2023). *SSSD: Official Implementation of Diffusion-based Time Series Imputation and Forecasting with Structured State Space Models*. https://github.com/AI4HealthUOL/SSSD. Accessed: 2025-11-28.

Alcaraz, Juan Miguel Lopez and Nils Strodthoff (2023). *Diffusion-based Time Series Imputation and Forecasting with Structured State Space Models*. arXiv: 2208.09399 [cs.LG]. URL: https://arxiv.org/abs/2208.09399.

Box, George E. P. and Gwilym M. Jenkins (1976). *Time Series Analysis: Forecasting and Control*. Holden-Day series in time series analysis and digital processing. Holden-Day. ISBN: 9780816211043. URL: https://books.google.com.tw/books?id=1WVHAAAAMAAJ.

Cressie, Noel and Gardar Johannesson (2008). "Fixed rank kriging for very large spatial data sets". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 70.1, pp. 209–226. DOI: https://doi.org/10.1111/j.1467-9868.2007.00633.x. eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2007.00633.x. URL: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2007.00633.x.

Cressie, Noel and Christopher K. Wikle (2011). *Statistics for Spatio-Temporal Data*. CourseSmart Series. Wiley. ISBN: 9780471692744. URL: https://books.google.com.tw/books?id=-kOC6D0DiNYC.

Cressie, Noel A. C. (1993). *Statistics for Spatial Data*. John Wiley & Sons, Inc. DOI: 10.1002/9781119115151. URL: https://doi.org/10.1002/9781119115151.

Decorte, Thomas et al. (2024). "Missing Value Imputation of Wireless Sensor Data for Environmental Monitoring". In: *Sensors* 24.8. ISSN: 1424-8220. DOI: 10.3390/s24082416. URL: https://www.mdpi.com/1424-8220/24/8/2416.

Global Modeling and Assimilation Office (GMAO) (2015). *MERRA-2 inst1_2d_asm_Nx: 2d, 1-Hourly, Instantaneous, Single-Level, Assimilation, Single-Level Diagnostics V5.12.4*. Accessed: 2025-11-01. Greenbelt, MD, USA. DOI: 10.5067/3Z173KIE2TPD.

Green, P. J. and B. W. Silverman (1993). *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*. 1st ed. Chapman and Hall/CRC. DOI: 10.1201/b15710.

Gu, Albert, Tri Dao, et al. (2020). *HiPPO: Recurrent Memory with Optimal Polynomial Projections*. arXiv: 2008.07669 [cs.LG]. URL: https://arxiv.org/abs/2008.07669.

Gu, Albert, Karan Goel, and Christopher Ré (2022). *Efficiently Modeling Long Sequences with Structured State Spaces*. arXiv: 2111.00396 [cs.LG]. URL: https://arxiv.org/abs/2111.00396.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel (2020). *Denoising Diffusion Probabilistic Models*. arXiv: 2006.11239 [cs.LG]. URL: https://arxiv.org/abs/2006.11239.

Hochreiter, Sepp and Jürgen Schmidhuber (Nov. 1997). "Long Short-Term Memory". In: *Neural Computation* 9.8, pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. eprint: https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf. URL: https://doi.org/10.1162/neco.1997.9.8.1735.

Kalman, R. E. (Mar. 1960). "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1, pp. 35–45. ISSN: 0021-9223. DOI: 10.1115/1.3662552. eprint: https://asmedigitalcollection.asme.org/fluidsengineering/article-pdf/82/1/35/5518977/35_1.pdf. URL: https://doi.org/10.1115/1.3662552.

Kong, Zhifeng et al. (2021). *DiffWave: A Versatile Diffusion Model for Audio Synthesis*. arXiv: 2009.09761 [eess.AS]. URL: https://arxiv.org/abs/2009.09761.

Little, Roderick J. A. and Donald B. Rubin (2002). *Statistical Analysis with Missing Data*. John Wiley & Sons, Inc. DOI: 10.1002/9781119013563. URL: https://doi.org/10.1002/9781119013563.

National Center for High-Performance Computing (NCHC) (2018). *Taiwania 2 / Taiwan Computing Cloud (TWCC) High-Performance AI Cloud Platform*. National Center for High-Performance Computing, Taiwan. Supercomputer: Taiwania 2; 9 PFLOPS, 2,016 NVIDIA Tesla V100 GPUs; Operated via TWCC.

Shi, Xingjian et al. (2015). "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting". In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes et al. Vol. 28. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2015/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf.

Sohl-Dickstein, Jascha et al. (2015). *Deep Unsupervised Learning using Nonequilibrium Thermodynamics*. arXiv: 1503.03585 [cs.LG]. URL: https://arxiv.org/abs/1503.03585.

Tzeng, ShengLi and Hsin-Cheng Huang (2018). "Resolution Adaptive Fixed Rank Kriging". In: *Technometrics* 60.2, pp. 198–208. DOI: 10.1080/00401706.2017.1345701. eprint: https://doi.org/10.1080/00401706.2017.1345701. URL: https://doi.org/10.1080/00401706.2017.1345701.

Tzeng, ShengLi, Hsin-Cheng Huang, Wen-Ting Wang, and Yao-Chih Hsu (2025). *autoFRK-python: Automatic Fixed Rank Kriging. The Python version with PyTorch*. Python package version 1.2.3. URL: https://pypi.org/project/autoFRK/.

Tzeng, ShengLi, Hsin-Cheng Huang, Wen-Ting Wang, Douglas Nychka, et al. (2021). *autoFRK: Automatic Fixed Rank Kriging*. R package version 1.4.3. URL: https://CRAN.R-project.org/package=autoFRK.

Wahba, Grace and James Wendelberger (1980). "Some New Mathematical Methods for Variational Objective Analysis Using Splines and Cross Validation". In: *Monthly Weather Review* 108.8, pp. 1122–1143. DOI: 10.1175/1520-0493(1980)108<1122:SNMMFV>2.0.CO;2. URL: https://journals.ametsoc.org/view/journals/mwre/108/8/1520-0493_1980_108_1122_snmmfv_2_0_co_2.xml.

Zhu, Xun et al. (2023). *Weather2K: A Multivariate Spatio-Temporal Benchmark Dataset for Meteorological Forecasting Based on Real-Time Observation Data from Ground Weather Stations*. arXiv: 2302.10493 [cs.LG]. URL: https://arxiv.org/abs/2302.10493.

A Variable List of Weather2K

Weather2K-R 資料集所包含之變數如下。本研究使用其地表溫度（Air Temperature）變數作爲主要分析對象。

Table 4: Variable List of Weather2K. Description of variables, their abbreviations, and measurement units.

Variable	**Short Name**	**Unit**
Latitude	lat	degrees east
Longitude	lon	degrees north
Altitude	alt	m
Air Pressure	ap	hPa
Air Temperature	t	°C
Maximum Temperature	mxt	°C
Minimum Temperature	mnt	°C
Relative Humidity	rh	%
Precipitation in 3h	p3	mm
Wind Direction	wd	degrees
Wind Speed	ws	$\mathrm{m\ s^{-1}}$
Maximum Wind Direction	mwd	degrees
Maximum Wind Speed	mws	$\mathrm{m\ s^{-1}}$

B Variable List of MERRA-2

MERRA-2 資料集所包含之變數如下。本研究使用其大氣柱總奇氧量（total column odd oxygen）變數作爲主要分析對象。

Table 5: Variable List of MERRA-2. Description of variables, their abbreviations, and measurement units.

Variable	**Short Name**	**Unit**
Longitude	lon	degrees east
Latitude	lat	degrees north
Time	time	minutes since 2024-06-01 00:00:00
2-Meter Air Temperature	t2m	K
Total Precipitable Liquid Water	tql	kg m^{-2}
Total Column Odd Oxygen	tox	kg m^{-2}
2-Meter Eastward Wind	u2m	m s^{-1}
Surface Pressure	ps	Pa
Tropopause Temperature Using Blended TROPP Estimate	tropt	K
Northward Wind at 50 Meters	v50m	m s^{-1}
Zero Plane Displacement Height	disph	m
Total Column Ozone	to3	Dobsons
Surface Skin Temperature	ts	K
10-Meter Air Temperature	t10m	K
Tropopause Pressure Based on Thermal Estimate	troppt	Pa

Variable	Short Name	Unit
Total Precipitable Ice Water	tqi	kg m^{-2}
Sea Level Pressure	slp	Pa
Tropopause Pressure Based on Blended Estimate	troppb	Pa
Total Precipitable Water Vapor	tqv	kg m^{-2}
2-Meter Northward Wind	v2m	m s^{-1}
Tropopause Specific Humidity Using Blended TROPP Estimate	tropq	kg kg^{-1}
10-Meter Northward Wind	v10m	m s^{-1}
Eastward Wind at 50 Meters	u50m	m s^{-1}
10-Meter Eastward Wind	u10m	m s^{-1}
2-Meter Specific Humidity	qv2m	kg kg^{-1}
Tropopause Pressure Based on EPV Estimate	troppv	Pa
10-Meter Specific Humidity	qv10m	kg kg^{-1}