Decoupled Spatiotemporal Forecasting from Extreme Sparse Observations via Quantized Latent Space

Zhongnan Weng[1], Yue Hong[1], Hang Yu[1], Jiayi Que[1], Juan Liu[2*], Xiangrong Liu[1,3*]

[1]Department of Computer Science and Technology, Xiamen University
[2]Pen-Tung Sah Institute of Micro-Nano Science and Technology, Xiamen University
[3]National Institute for Data Science in Health and Medicine, Xiamen University
{wengzhongnanw, hongy, hangyu203, quejiayi}@stu.xmu.edu.cn, {cecyliu, xrliu}@xmu.edu.cn

Abstract

Predicting spatiotemporal fields governed by partial differential equations (PDEs) from sparse sensor data is a critical and long-standing challenge in science and engineering. Recent deep learning approaches, particularly neural operators, have shown considerable promise in solving PDEs. However, their performance degrades significantly in the demanding regime of extreme sparsity, characterized by spatial sensor coverage of less than 1% and limited temporal observations. To overcome this limitation, we propose SparQT, a novel framework that decouples the task into two stages: spatial reconstruction and temporal extrapolation. In the first stage, rather than reconstructing the high-dimensional physical field directly, our model learns to reconstruct the complete latent features from sparse observations—features that would otherwise be extracted from a dense field. This process is stabilized by a Vector Quantization (VQ) bottleneck, which discretizes the latent space. In the second stage, a decoder-only Transformer performs temporal extrapolation by autoregressively predicting the future sequence of these discrete latent indices. This design inherently allows the model to generalize to new initial conditions and varying forecast horizons, akin to standard autoregressive models. We validate our framework on three challenging benchmarks, achieving state-of-the-art (SOTA) performance under severe sparsity constraints. Furthermore, we introduce a challenging benchmark dataset based on fire dynamics simulations. On this benchmark, our model successfully forecasts the field's evolution 30 frames into the future from a single timeframe with less than 0.1% spatial observations—a result that pushes well beyond the capabilities of existing methods.

Code — https://github.com/wznwznwzn123/SparQT
Datasets — https://zenodo.org/records/17626533

Introduction

Predicting the evolution of spatiotemporal fields governed by physical laws is a fundamental and enduring challenge in science and engineering(Brunton and Kutz 2022; Baker et al. 2019). This predictive capability is crucial in numerous applications, such as industrial design, where fluid dynamics simulations are used to optimize efficiency and safety(Martins and Ning 2021); and in emergency response scenarios like fire forecasting(Jain et al. 2020), where understanding fire spread is crucial for effective response. The evolution of these systems is almost universally described by partial differential equations (PDEs). Therefore, the problem of predicting these fields can be formulated as solving the underlying PDEs, which involves inferring initial conditions, operator parameters, and the solution from a given set of observational data(Tarantola 2005). Traditional numerical methods, such as the finite element method (FEM)(Zienkiewicz, Taylor, and Zhu 2005) and the finite volume method (FVM)(LeVeque 2002), have long been standard approaches. However, they are often computationally expensive, requiring extensive domain knowledge and vast computing resources, which limits their application in real-time scenarios.

With the rapid development of machine learning in recent years(Jordan and Mitchell 2015), large-scale machine learning has provided a natural solution to this problem. These data-driven approaches offer a powerful alternative, promising a better balance between accuracy, speed, and flexibility(Karniadakis et al. 2021). Broadly speaking, these methods can be divided into two major research directions: one is to solve the forward problem, that is, to find the solution of the PDE given its initial conditions and parameters; the other is to solve the inverse problem, that is, to infer the initial conditions or parameters from partial observations of the solution. Neural operators(Li et al. 2021; Lu et al. 2021), as an powerful class of deep learning models, have shown extraordinary potential in learning solution operators for PDE families and can achieve zero-shot generalization to new initial conditions. These methods, as well as some unified perspective frameworks, have introduced a new paradigm for predicting the evolution of spatiotemporal fields(Brunton, Noack, and Koumoutsakos 2020; Pathak et al. 2018).

Although the above methods have made significant progress, they still face two key challenges in practical applications. (1) **Extreme sparsity of observations.** In the real world, data is usually collected from a limited number of sensors. This is especially true in applications such as fire prediction, where it is not feasible to deploy dense sensor networks. When the spatial coverage of known data points is extremely low (for example, less than 0.1%), existing models often struggle because they cannot accurately reconstruct

*Juan Liu and Xiangrong Liu are corresponding authors.

the complete initial state of the system from such sparse information. (2) **Temporal Extrapolation.** Many applications require not only a snapshot in time but also long-term predictions of the system's evolution(Lian et al. 2024). The challenge is to accurately predict the complete spatiotemporal field for a long period of time in the future (e.g., a time span greater than 10 times the length of the initial observation window) based only on sparse observations of the initial state. Most existing methods are not designed for this task; forward solvers require a complete initial field, and unified methods that attempt to simultaneously perform reconstruction and prediction suffer from severe accuracy degradation under such sparse initial conditions. For example, a model recently proposed by Steeven et al(Steeven et al. 2024). attempts to address both problems, but its performance degrades significantly in scenarios of extreme sparsity.

To overcome these fundamental limitations, we propose an innovative framework that performs spatial reconstruction and temporal extrapolation separately in a quantized latent space. Our core insight is to decouple this complex problem into two more manageable sequential stages: spatial reconstruction and temporal extrapolation. And unlike previous work, for spatial reconstruction, our model not only reconstructs the high-dimensional physical field but also learns to reconstruct its complete, compressed latent representation from sparse observations. This process is stabilized by a vector quantization (VQ) bottleneck, which discretizes the continuous latent space. In the second stage, we perform temporal prediction directly in this discretized latent space. We treat the sequence of quantized latent indices at consecutive time steps as a sequence of indices. We then train a decoder-only Transformer to perform the "next index prediction" task, autoregressively predicting the future evolution of the system in the latent space domain. This decoupled, discretized approach effectively mitigates model compounding errors and enables robust, long-term prediction from a single, sparsely observed time frame.

We demonstrate through extensive experiments on three benchmark datasets that our approach achieves state-of-the-art performance. In particular, on our new fire dynamics dataset, our model successfully predicts the field evolution 30 frames into the future from a single, sparse initial observation, significantly outperforming existing methods. The main contributions of this paper are summarized as follows:

- We propose a novel two-stage decoupled framework that effectively addresses the challenges of combining extreme spatial sparsity and long-term prediction in PDE-controlled systems. - We introduce a spatial reconstruction method that maps sparse observations into a complete representation in a quantized latent space, significantly improving the robustness and quality of reconstruction from very small amounts of data.
- We design a temporal extrapolation model that uses a Transformer to autoregressively predict future states as a sequence of discrete latent indices, thereby preventing error propagation and achieving stable long-range predictions.
- We introduce a new, more challenging benchmark dataset based on fire dynamics simulations to test model performance under real-world conditions with extreme sparsity ($<0.1\%$ spatial observations).

Related Work

Solving PDE Inverse Problems from Sparse Observations

Recovering complete physical fields from partial or sparse data is a long-standing problem. Recently, neural networks have emerged as a powerful approach. Physics-Informed Neural Networks (PINNs) (Raissi, Perdikaris, and Karniadakis 2019) constrain network training with PDE residuals. To address the challenge of highly incomplete data, recent work has focused on latent-space and generative methods. For instance, DiffusionPDE (Huang et al. 2024) employs a generative diffusion model to jointly infer missing inputs and solve the governing PDE. Another line of work embeds the system's dynamics in a learned latent manifold. To handle inverse problems, LNO (Wang and Wang 2024) combines operator learning frameworks like DeepONet (Lu et al. 2021) and FNO (Li et al. 2021). The Senseiver framework (Santos et al. 2023) focuses on reconstruction by using cross-attention to encode sparse sensor inputs into a fixed-size latent representation, which is then decoded into a complete field. These advances highlight a clear trend: mapping sparse observations to a complete latent representation before evolving the dynamics. Our work extends this philosophy. We integrate vector quantization (VQ) to establish a robust mapping from extremely sparse inputs, through a semi-sparse latent representation, to the complete geometric space.

Spatiotemporal Field Prediction from Sparse Observations

Beyond static reconstruction, many applications require forecasting the temporal evolution of a system from a sparsely observed initial state. Some methods bridge this gap by jointly performing spatial completion and temporal forecasting. For example, MAgNet (Boussif et al. 2022) interpolates the observation graph in a latent space before using a Graph Neural Network (GNN) for prediction. DINo (Yin et al. 2023) encodes initial observations into an implicit neural representation and evolves this latent state with a learned ordinary differential equation (ODE). DeepMind's MeshGraphNet (Pfaff et al. 2021) and its variants (Han et al. 2022) first perform graph completion to infer missing node values, then use a GNN-based simulator for autoregressive updates. Steeven et al. (Steeven et al. 2024) proposed a "dual observation" framework to connect dynamics at sparse sensor locations with those on a continuous domain. While effective for semi-sparse data, the performance of these methods degrades significantly under the extreme sparsity regimes we consider, as shown in Section Experiments. Our framework partially addresses this limitation by decomposing the problem into two stages. We first obtain vector-quantized latent indices from the sparse observations, and then employ a Transformer to perform autoregressive temporal prediction

in this discrete latent space, thereby mitigating the impact of reconstruction errors.

Methodology

Our proposed framework tackles the challenge of spatiotemporal forecasting from extremely sparse observations by decoupling the problem into two distinct stages: (1) Spatial imputation through latent space based on Perceiver IO(Jaegle et al. 2021). (2) Stable temporal extrapolation with vector quantization and a Transformer.

Problem Setting

We aim to learn a neural operator(Kovachki et al. 2021) that can predict the evolution of a dynamical system governed by a latent Partial Differential Equation (PDE). Let the system's state be denoted by $u(x,t)$ for a spatial position $x \in \Omega$ and a time $t \in [0, \mathcal{T}]$. The evolution is determined by an unknown initial condition $u(x,0) = u_0(x)$.

Our central goal is to learn an operator $\mathcal{G}_\theta$, which takes two arguments as input: a sparse representation of the initial condition, and a specific spatiotemporal coordinate (x,t). The operator then outputs the predicted state at that precise coordinate. We formulate this as:

$$\hat{u}(x,t) = \mathcal{G}_\theta(a_0, x, t), \tag{1}$$

where $\hat{u}(x,t)$ is the predicted state. The input a_0 encapsulates our entire knowledge of the initial state. Crucially, this knowledge is extremely sparse. It consists of the state values measured only at a small, fixed set of N_s sensor locations $P = \{p_i\}_{i=1}^{N_s} \subset \Omega$. Formally, the input initial condition is represented as a set of value-coordinate pairs:

$$a_0 = (u_0(p_i), p_i)_{i=1}^{N_s}. \tag{2}$$

The challenge lies in the fact that N_s is orders of magnitude smaller than the number of points required to describe the full field, forcing the operator $\mathcal{G}_\theta$ to learn the underlying dynamics from severely incomplete information. The query coordinate (x,t) can be any point within the continuous domain $\Omega \times [0, \mathcal{T}]$, requiring the operator to interpolate and extrapolate in both space and time. We train the neural operator $\mathcal{G}_\theta$ on a dataset $\mathcal{D}$ composed of K distinct trajectories, where each trajectory corresponds to a different initial condition:

$$\mathcal{D} = (a_0^{(k)}, \mathcal{S}^{(k)})_{k=1}^{K}. \tag{3}$$

For each k-th trajectory, $a_0^{(k)}$ is the sparse initial observation. $\mathcal{S}^{(k)}$ represents the ground truth data used for supervision. Crucially, our training methodology relies on *decoupling* of the ground truth data. While the encoder only observes the sparse initial state $a_0^{(k)}$ (that is, values at the small set of sensor locations P) as input, the learning objective is to reconstruct the entire dense field. The supervision signal is therefore derived from the complete and dense field represented by $\mathcal{S}^{(k)}$. The decoder is trained to predict the state $u(x,t)$ for any query coordinate (x,t), and the loss is computed against the true values of $\mathcal{S}^{(k)}$.

Model Architecture

As illustrated in Fig. 1, our method achieves spatio-temporal prediction from sparse data via a two-stage process. First, we learn a robust spatial representation by using a Perceiver-IO-based VQ-Autoencoder to map sparse observations into a discrete, quantized latent space. This process is trained to align the latent features of sparse data with those of semi-sparse data, enabling high-fidelity reconstruction. Second, we treat the sequence of these quantized indices as tokens and use an autoregressive Transformer to predict future sequences. Finally, a shared Perceiver decoder maps these predicted indices back to complete spatial fields, enabling end-to-end forecasting from sparse initial conditions.

Input Embedding and Positional Encoding The attention mechanism (Vaswani et al. 2017), central to our architecture, is permutation invariant and therefore does not have an inherent notion of spatial location. To provide the model with this crucial information, we explicitly encode the coordinates of both the sparse sensor inputs and the dense query points for the final reconstruction.

Let a single sparse observation at a given timestep t (omitted hereafter for brevity) be represented by a set of N_s sensors $\{s_i, p_i\}_{i=1}^{N_s}$, where $s_i \in \mathcal{S}$ is the physical value and $p_i \in P$ is its spatial coordinate. Following prior work (Santos et al. 2023), we use a fixed sine-cosine(Vaswani et al. 2017) positional encoding. For each spatial dimension, a set of frequencies is used to create a feature vector that uniquely represents each location. This method requires no learnable parameters.

The final input vectors are formed by directly concatenating the sensor values s_i with their corresponding positional encodings $P_E(p_i)$:

$$\mathbf{Z}_i = \text{concat}(s_i, P_E(p_i)). \tag{4}$$

Perceiver IO Encoder Our encoder design is based on the Senseiver(Santos et al. 2023) architecture, with the core objective of efficiently encoding sparse observations into a compact, fixed-size latent space.

First, we represent the sparse input observations as a set $Z = \{(p_i, P_E(p_i))\}_{i=1}^{N_s}$. Concurrently, we initialize a learnable, fixed-size latent array, $L_{sparse} \in \mathbb{R}^{M \times D}$. Here, M is the number of latent vectors, and D is the dimension of each latent vector.

The core of the encoding process is the *Cross-Attention*(Vaswani et al. 2017) mechanism. We use the latent array L_{sparse} as the Query, while the sparse input data Z serves as the Key and Value.

$$L_{sparse} = \text{CrossAttention}(Q = L_{sparse}, K = V = Z). \tag{5}$$

Subsequently, the updated latent array L_{sparse} is fed into a deep processor composed of *Self-Attention*(Vaswani et al. 2017) blocks for information integration and refinement. This yields the final output latent representation $L_{sparse} \in \mathbb{R}^{M \times D}$:

$$L_{sparse} = \text{SelfAttentionBlock}(L_{sparse}). \tag{6}$$

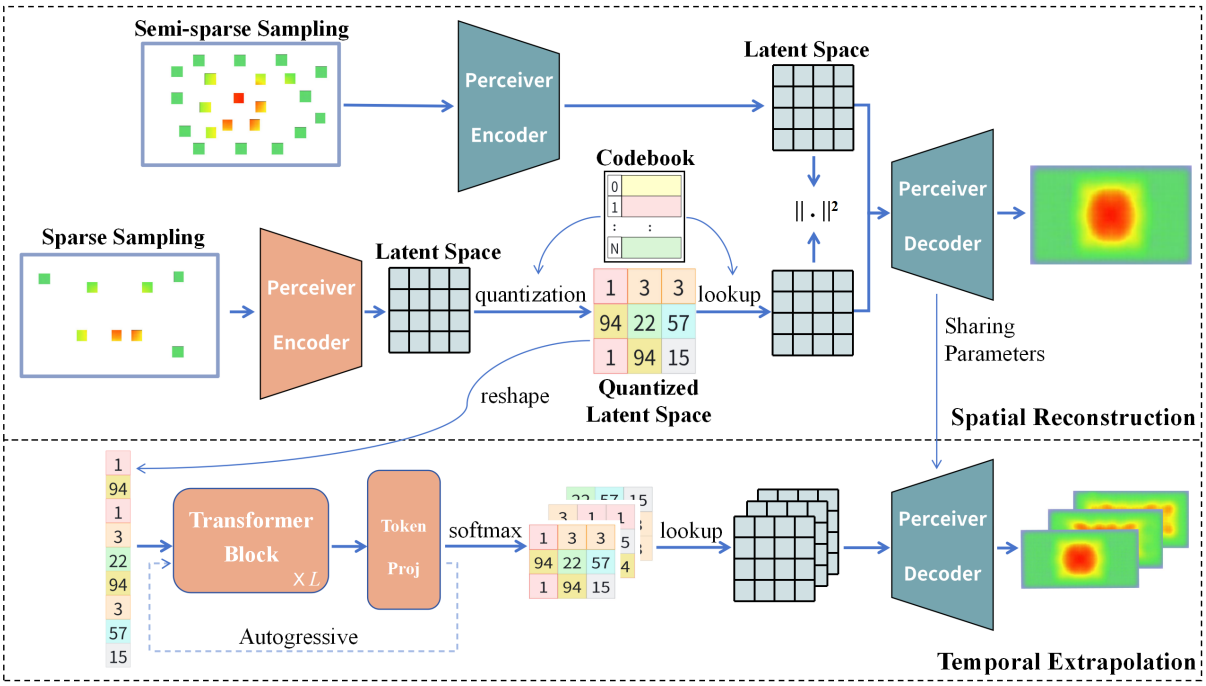


Figure 1: Overview of the decoupled spatiotemporal forecasting framework for extreme sparse observations. The model first learns a quantized spatial representation from sparse inputs using a Perceiver-VQ architecture (Top, Spatial Reconstruction). Then, an autoregressive Transformer predicts the future sequence of these quantized indices, which a shared decoder uses to generate the complete future fields (Bottom, Temporal Extrapolation).

Additional Semi-Sparse Encoder However, differing from Senseiver's single-pathway design(Santos et al. 2023), we introduce a critical addition: a *Semi-Sparse Observation Encoder*. This encoder shares the same architecture with the *sparse encoder* but maintains separate weights.

The primary purpose of this semi-sparse encoder is to act as a "teacher" model that provides supervision during the training phase. It takes denser observation data, Z_{denser}, as input and generates a "gold-standard" latent representation, L_{denser}, through the same process.

Our key hypothesis is that L_{denser}, generated from semi-sparse data, more comprehensively captures the global information of the physical field. In this manner, the *semi-sparse encoder* guides the *sparse encoder* and the *Vector Quantizer(Razavi, Van den Oord, and Vinyals 2019)*, which will be introduced later, to learn how to infer a globally and physically plausible latent structure from only a small amount of information.

Perceiver IO Decoder The decoder's role is to map the latent representation L back to the physical space using *Cross-Attention*. Given a set of query coordinates $\{p_j\}_{j=1}^{\mathcal{N}}$ for the entire space domain, it uses cross-attention to project the information from the latent array L (as Key and Value) onto the positionally-encoded query points (as Query) to produce the reconstructed field values $\hat{u}_j$. Unlike the encoder, the decoder's weights are shared for reconstructions from both the sparse and semi-sparse pathways, ensuring a consistent mapping from the latent space to the physical domain.

Vector Quantization and Latent Reconstruction Loss To stabilize the latent space and discretize it for the subsequent temporal prediction task, we introduce a Vector Quantized Variational AutoEncoder (VQ-VAE) layer(Razavi, Van den Oord, and Vinyals 2019). A learnable codebook, or embedding space $e_v, v = 1, \ldots, V$ is maintained, where V is the codebook size and each e_v is a code vector in $\mathbb{R}^C$.

Given the continuous latent representation L_{sparse} from the sparse encoder, it is encoded by the encoder of VQ-VAE layer to a set of new vectors $\mathbf{z}_{\text{sparse}} = E_{\text{vq}}(L_{\text{sparse}})$. Each vector in this set is then quantized by replacing it with its nearest neighbor from the codebook. This produces the quantized vector set $\mathbf{z}_{q,\text{ sparse}}$:

$$\mathbf{z}^{(i)}_{q,\text{ sparse}} = e_k \text{ where } k = \arg\min_l \|\mathbf{z}^{(i)}_{\text{sparse}} - e_l\|. \quad (7)$$

And the indices of quantized codes are also obtained from this process, which we will note as I. $\mathbf{z}_{\text{sparse}}$, $\mathbf{z}_{\text{q, sparse}}$ and I have the same length here and L_{sparse} does not.

The training objective includes a standard VQ-VAE reconstruction loss

$$\mathcal{L}_{\text{rec}} = \|L_{\text{sparse}} - D(\mathbf{z}_{q,\text{ sparse}})\|_2^2, \quad (8)$$

where D is the VQ-VAE decoder function The VQ-VAE codebook and commitment losses

$$\mathcal{L}_{\text{vq}} = \sum_i \|\text{sg}[\mathbf{z}^{(i)}_{\text{sparse}}] - e\|_2^2 + \beta\|\mathbf{z}^{(i)}_{\text{sparse}} - \text{sg}[e]\|_2^2, \quad (9)$$

where e is the quantized code for $\mathbf{z}^{(i)}_{\text{sparse}}$, $\text{sg}[\cdot]$ denotes the stop-gradient operator and β is a hyperparameter which controls the reluctance to change the code corresponding to the

encoder output(Razavi, Van den Oord, and Vinyals 2019). And a novel **latent reconstruction loss**

$$\mathcal{L}_{\text{denser}} = \|L_{\text{sparse}} - L_{\text{denser}}\|_2^2. \tag{10}$$

This loss enforces that the reconstructed latent with quantized vectors from a *sparse* input should approximate the continuous latent grid from the corresponding *semi-sparse* input. This forces the sparse encoder and the VQ-VAE layer to learn to impute the "correct" latent features that a model with full information would have extracted.

Stage 2: Temporal Extrapolation with Latent Space Dynamics

Once Stage 1 is trained, we have a robust mechanism for encoding a spatiotemporal field from a sparse observation into a discrete set of indices corresponding to the codebook vectors. The second stage focuses on learning the temporal dynamics directly in this compressed, discrete latent space.

Autoregressive Prediction of Latent Indices We employ a decoder-only Transformer(Radford et al. 2019) to model the temporal evolution of the system. The Transformer is trained on a standard next-token prediction task. For each position in the sequence, it predicts a probability distribution over the entire vocabulary of V possible codebook indices for the next token. This is achieved through stacked Transformer blocks followed by a final softmax layer. Each block consists of two main sub-layers: masked multi-head self-attention and a position-wise feed-forward network. The masked self-attention is computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + \mathbf{M}\right)V, \tag{11}$$

where Q, K, V are the query, key, and value matrices, and $\mathbf{M}$ is a mask matrix that sets the attention scores for future positions to $-\infty$ to ensure causality. And the FFN (Feed-Forward Network)(Vaswani et al. 2017) is computed as:

$$\text{FFN}(X) = \text{MLP}(\text{LayerNorm}(X)) + X. \tag{12}$$

The initial input to the first Transformer block, X_0, is obtained by feeding the sequence of discrete codebook indices into an embedding layer and reshaping the resulting embeddings to the Transformer's required dimension. The computation for the entire Transformer blocks can then be summarized as:

$$\begin{aligned} \hat{X}_l &= \text{Attention}(\text{LayerNorm}(X_{l-1})) + X_{l-1} \\ X_l &= \text{FFN}(\hat{X}_l), \end{aligned} \tag{13}$$

At stage 2, the cross-entropy loss is used as the training objective:

$$\mathcal{L}_{\text{ce}} = -\sum_{v=1}^{V} y_v \log(\hat{y}_v), \tag{14}$$

where V is the vocabulary size, y_v is the ground-truth label (1 for the true token and 0 for all others), and $\hat{y}_v$ is the predicted probability for token v.

Scheduled Sampling and Inference Procedure A common issue in autoregressive models is the exposure bias between training (where ground-truth inputs are provided, i.e., "teacher forcing") and inference (where the model consumes its own predictions). To bridge this gap, we employ *Scheduled Sampling*(Bengio et al. 2015), a mixed-ratio training strategy. During training, we probabilistically feed the model either the ground-truth previous token or the model's own sampled prediction from the previous step. This makes the model more robust to its own errors.

During inference, the model autoregressively generates the entire future sequence of indices, one token at a time. The generated indices are reshaped and mapped back to the VQ latent vectors, which are finally reconstructed by the VQ decoder and passed to the shared Perceiver IO Decoder to render the full-resolution physical field.

Experiments

To systematically validate the effectiveness of our proposed two-stage framework for spatio-temporal field prediction under extremely sparse conditions, we conduct comprehensive experiments on two representative datasets and one novel dataset. The experiments are designed to answer two key research questions:

1. How does our method perform compared to state-of-the-art baselines under varying levels of sparsity?
2. Can our framework work in complex, realistic scenarios, such as fire dynamics simulation?

Datasets

We briefly introduce the datasets used in our experiments. **Navier-Stokes** (Yin et al. 2023) simulates the vorticity of a viscous, incompressible flow on a 64×64 grid over 20 timesteps. We use 256 sequences for training and 16 for validation and testing. **Shallow Water** (Yin et al. 2023) models the velocity of shallow water on a 128×64 grid. The dataset provides 80 sequences of 20 timesteps each, with 64 used for training and 8 each for validation and testing. **Fire Dynamics** is a custom dataset we generated using the Fire Dynamics Simulator (FDS) (McGrattan et al. 2013) to simulate complex 3D smoke propagation. It includes 30 training sequences and 2 sequences each for validation and testing, with each sequence consisting of 30 timesteps. This dataset is designed to mimic real-world scenarios with sparse sensor coverage.

Baselines

We compare our proposed framework against a representative state-of-the-art method designed for sparse PDE learning:

- **MGN**(Pfaff et al. 2021): A multi-layered GNN used auto-regressively, extended to spatiotemporal continuity via physics-agnostic interpolation.
- **MAgNet**(Yin et al. 2023): First interpolates the initial conditions in latent space at query positions before applying the MGN architecture.

Method	Navier-Stokes			Shallow Water			TSR
	0.5%	1%	5%	0.5%	1%	5%	
MAgNet	955.7	188.4	57.21*	381.4	211.7	30.55*	1/1
	1370.1	201.9	63.40	425.6	253.2	65.81	1/4
MGN	602.3	45.60	14.80*	142.1	92.5	8.187*	1/1
	715.4	51.28	17.95	168.3	105.9	28.34	1/4
DINo	568.1	16.35	5.493*	58.72	67.33	21.55*	1/1
	642.5	19.52	6.810	125.18	131.60	48.95	1/4
Steeven et al.	560.5	69.90	<u>0.560*</u>	12.06	10.34	2.800*	1/1
	497.1	67.21	1.321	13.19	10.56	3.122	1/4
Steeven et al.-D.C.	<u>5.301</u>	<u>1.420</u>	**0.511**	<u>0.432</u>	<u>0.361</u>	**0.154**	1/1
	<u>12.18</u>	<u>5.430</u>	<u>1.126</u>	<u>0.547</u>	<u>0.374</u>	**0.182**	1/4
OURS	**1.060**	**1.330**	0.623	**0.379**	**0.318**	<u>0.287</u>	1/1
	0.756	**0.518**	**0.461**	**0.509**	**0.304**	<u>0.254</u>	1/4

Table 1: Combined results on Navier-Stokes and Shallow Water Datasets (MSE $\times 10^{-3}$). Results marked with an asterisk (*) are taken from the original paper (Steeven et al. 2024).

- **DINo**(Boussif et al. 2022): models the solution as an Implicit Neural Representation (INR) modulated by a time-dependent context vector whose dynamics are learned by a Neural-ODE.
- **Steeven et al.'s** (Steeven et al. 2024): A continuous neural operator that directly maps sparse spatiotemporal observations to future fields. It is chosen for its explicit focus on sparse inputs and its demonstrated temporal continuity.

Different from our model's decoupled scheme for supervision, the original Steeven et al.'s framework trains and evaluates using only sparse observations. Therefore, to create a fair and comprehensive comparison, we add another baseline into our experiments:

- **Steeven et al.'s-D.C. (Decoupled Correction)**: We modify the original Steeven et al.'s model to adopt our decoupled supervision strategy. Specifically, the model will get input from the sparse field and obtain the labels from dense observations. This ensures a fair comparison between our model and the baseline.

Evaluation Metrics and Protocol

- **Mean Squared Error (MSE):** The primary metric, defined as the average squared difference between the predicted and ground-truth field values over all spatial grid points and time steps:

$$\text{MSE} = \frac{1}{N \cdot T} \sum_{t=1}^{T} \sum_{i=1}^{N} (\hat{u}_{i,t} - u_{i,t})^2 \qquad (15)$$

where N is the number of spatial grid points, T is the number of time steps, $u_{i,t}$ is the ground-truth value at grid point i and time t, and $\hat{u}_{i,t}$ is the corresponding predicted value.

- **Qualitative Visualization:** For the Fire Dynamics dataset, we provide visual comparisons of the predicted and ground-truth smoke propagation to highlight structural consistency and long-term coherence.

All models are trained on 4 NVIDIA A800 GPUs. For the baseline, we adopt the hyperparameters recommended by the original authors and perform additional tuning on the Fire Dynamics dataset to achieve their best performance.

Main Results

In the results, the time split rate (TSR) indicates the proportion of data uniformly sampled from the temporal dimension of the training samples. In each column, the **bold** numbers represent the best performance (lowest error) among the three models, while the <u>underlined</u> numbers denote the second-best performance.

Regarding **spatial sparsity**, our model demonstrates exceptional performance when handling spatially sparse observational data. As Tab. 1 indicates, it accurately reconstructs complete and high-fidelity dynamic fields even under stringent conditions where observation points are extremely scarce (e.g., with only 0.5% or 1% of the data available). This demonstrates our architecture's ability to effectively capture and comprehend the system's intrinsic physical laws and spatial correlations, enabling the inference of the entire continuous field state from only a few discrete points.

Regarding **temporal sparsity and long-Term prediction stability** as is depicted in Tab. 1, even with only 1/4 of the temporal data in training samples, our model still performed well in predicting future frames using the autoregressive Transformer. A common challenge in autoregressive models is the accumulation of errors over prediction time steps, which can significantly degrade the accuracy of long-term

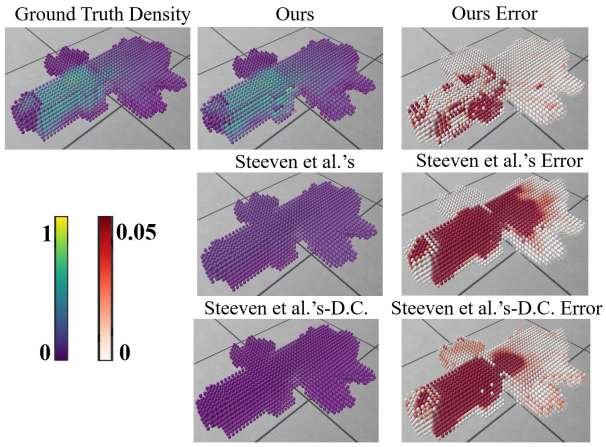


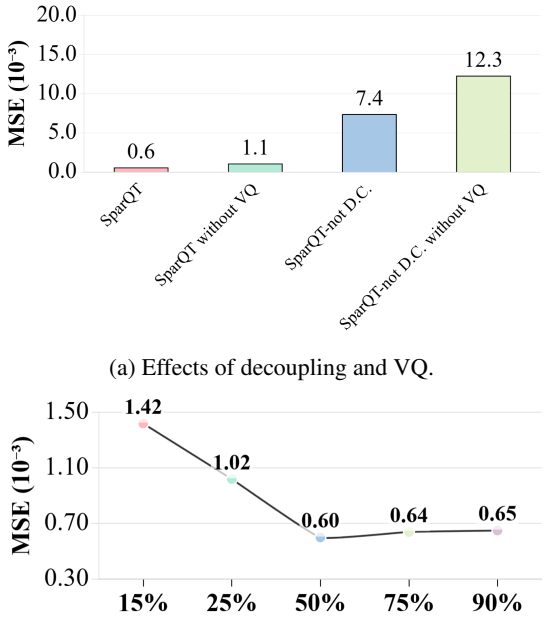
Figure 2: Comparison of ground-truth smoke density with predictions from our model and two baselines on the Fire Dynamics dataset (t=10), including pointwise squared error. Color bars denote normalized density (left) and unnormalized squared error (right).

forecasts. To address this, we incorporate a Vector Quantization (VQ) module into our model. This design choice effectively curtails the propagation and amplification of errors across the time series by mapping continuous hidden states to a discrete codebook. Each quantization step can be viewed as a "correction" of the system state, which prevents the compounding effect of minor deviations(Van Den Oord, Vinyals et al. 2017). The experimental results validate the effectiveness of this design, showing that our model maintains higher stability and accuracy in long-term forecasting.

To further assess the model's generalization capabilities and **performance in real-world scenario**, we evaluated it on our custom-generated Fire Dynamics dataset. As visualized in Fig. 2, a stark contrast emerges. Quantitatively, our model achieves a significantly lower MSE over all spatial grid points and time steps of **0.78** ($\times 10^{-3}$), outperforming both the Steeven et al.'s (3.53) and Steeven et al.'s-D.C. (3.22) models under the 0.1%, 1/1 condition. More critically, while the MSEs of the other two model might seem acceptable, their relative L2 error is excessively high at 0.93. The visualization also reveals their failure to grasp the physical semantics of the smoke field; they merely regress towards a uniform state to minimize average error. In sharp contrast, our model adeptly captures the crucial movement trends and structural information, maintaining high fidelity to the ground truth in both macroscopic structures and detailed textures. This superior performance highlights that our model is not only applicable to theoretical fluids but also possesses significant potential for practical applications in engineering and environmental forecasting.

Ablation Study

We conducted ablation studies to validate SparQT's key components. As shown in Fig. 3a, an end-to-end trained variant proved unstable due to noisy spatial gradients, while removing the Vector Quantization (VQ) module caused severe error accumulation in long-term forecasts. Furthermore, our analysis of semi-sparse supervision (Fig. 3b) reveals an optimal trade-off, with performance peaking when supervision data coverage is around 50%. These results confirm our decoupled design is vital for stable training and the VQ module is essential for long-term accuracy, validating our architectural choices.



(a) Effects of decoupling and VQ.

(b) Effect of teacher model sparsity.

Figure 3: Ablation study results on the Navier-Stokes dataset with 5% input spatial sparsity and 1/1 TSR (averaged).

Conclusion

In this paper, we introduced a novel two-stage framework for spatiotemporal forecasting of systems governed by partial differential equations, specifically designed to address the critical challenge of extreme observational sparsity. Our approach successfully decouples the complex problem into two manageable stages: robust spatial reconstruction and stable temporal extrapolation. We have demonstrated through extensive experiments on multiple challenging benchmarks, including a new, complex fire dynamics dataset, that our model can produce high-fidelity, long-range forecasts from observations covering less than 0.1% of the spatial domain.

A key limitation of our method is error propagation in autoregressive predictions, especially for ultra-long-range extrapolation. Furthermore, the model's sensitivity to noise has not been sufficiently explored. These issues represent key directions for our future research.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant nos. 62072384 and 62372391),

Fujian Provincial Major Science and Technology Project (2022YZ040011) and the National Key Research and Development Program of China (2023YFF1205600 and 2024YFF1206204).

References

Baker, N.; Kapur, A.; ; et al. 2019. Workshop report: The 2018 workshop on artificial intelligence and deep learning for advancing sustainable development. arXiv:1906.01258.

Bengio, S.; Vinyals, O.; Jaitly, N.; and Shazeer, N. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in neural information processing systems*, 1171–1179.

Boussif, O.; Assouline, D.; Benabbou, L.; and Bengio, Y. 2022. MAgNet: Mesh Agnostic Neural PDE Solver. In *Thirty-sixth Conference on Neural Information Processing Systems (NeurIPS 2022)*.

Brunton, S. L.; and Kutz, J. N. 2022. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge: Cambridge University Press.

Brunton, S. L.; Noack, B. R.; and Koumoutsakos, P. 2020. Machine Learning for Fluid Mechanics. *Annual Review of Fluid Mechanics*, 52: 477–508.

Han, X.; Gao, H.; Pfaff, T.; Wang, J.-X.; and Liu, L.-P. 2022. Predicting Physics in Mesh-Reduced Space with Temporal Attention. In *International Conference on Learning Representations (ICLR)*.

Huang, J.; Yang, G.; Wang, Z.; and Park, J. J. 2024. DiffusionPDE: Generative PDE-Solving Under Partial Observation. In *Thirty-eighth Conference on Neural Information Processing Systems (NeurIPS 2024)*.

Jaegle, A.; Borgeaud, S.; Alayrac, J.-B.; Doersch, C.; Lenc, K.; Brock, A.; Lakshminarayanan, B.; Hubert, T.; Vinyals, O.; and Zisserman, A. 2021. Perceiver IO: A general architecture for structured inputs & outputs. In *International Conference on Learning Representations*.

Jain, M. J.; Coogan, S. C. P.; Subramanian, S. G.; Crowley, M.; Taylor, S.; and Flannigan, M. D. 2020. A review of machine learning applications in wildfire science and management. *Environmental Reviews*, 28(4): 478–505.

Jordan, M. I.; and Mitchell, T. M. 2015. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245): 255–260.

Karniadakis, G. E.; Kevrekidis, I. G.; Lu, L.; Perdikaris, P.; Wang, S.; and Yang, L. 2021. Physics-informed machine learning. *Nature Reviews Physics*, 3(6): 422–440.

Kovachki, N.; Li, Z.; Liu, B.; Mistry, K.; Nelsen, B.; Anandkumar, A.; Bhattacharya, K.; and Lusch, B. 2021. Neural operators: A comprehensive overview. *arXiv preprint arXiv:2108.08481*.

LeVeque, R. J. 2002. *Finite Volume Methods for Hyperbolic Problems*. Cambridge: Cambridge University Press.

Li, Z.; Kovachki, N.; Azizzadenesheli, K.; Liu, B.; Bhattacharya, K.; Stuart, A.; and Anandkumar, A. 2021. Fourier Neural Operator for Parametric Partial Differential Equations. In *International Conference on Learning Representations (ICLR)*.

Lian, S.; Liu, J.; Liu, Y.; Wang, Y.; and Lu, C. 2024. GNOT: A General Neural Operator for Transferable BE-PDE Simulation. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Lu, L.; Jin, P.; Pang, G.; Zhang, Z.; and Karniadakis, G. E. 2021. Learning nonlinear operators via DeepONet. *Nature Machine Intelligence*, 3(3): 218–229.

Martins, J. R. R. A.; and Ning, A. 2021. *Engineering Design Optimization*. Cambridge: Cambridge University Press.

McGrattan, K.; Hostikka, S.; McDermott, R.; Floyd, J.; Weinschenk, C.; and Overholt, K. 2013. Fire dynamics simulator user's guide. *NIST special publication*, 1019(6): 1–339.

Pathak, J.; Hunt, B.; Girvan, M.; Lu, Z.; and Ott, E. 2018. Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach. *Physical Review Letters*, 120(2): 024102.

Pfaff, T.; Fortunato, M.; Sanchez-Gonzalez, A.; and Battaglia, P. W. 2021. Learning Mesh-Based Simulation with Graph Networks. In *International Conference on Learning Representations (ICLR)*.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Raissi, M.; Perdikaris, P.; and Karniadakis, G. 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378: 686–707.

Razavi, A.; Van den Oord, A.; and Vinyals, O. 2019. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32.

Santos, J.; Fox, Z.; Mohan, A.; Howe, N.; Azizpour, H.; kuberry, S.; Baydin, A.; and Gal, Y. 2023. Development of the Senseiver for efficient field reconstruction from sparse observations. *Nature Machine Intelligence*, 5(11): 1317–1325.

Steeven, J.; Nadri, M.; Digne, J.; and Wolf, C. 2024. Space and Time Continuous Physics Simulation From Partial Observations. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Tarantola, A. 2005. *Inverse Problem Theory and Methods for Model Parameter Estimation*. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM).

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. In *Advances in neural information processing systems*, 6306–6315.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention Is All You Need. arXiv:1706.03762.

Wang, T.; and Wang, C. 2024. Latent Neural Operator for Solving Forward and Inverse PDE Problems. In *Thirty-eighth Conference on Neural Information Processing Systems (NeurIPS 2024)*.

Yin, Y.; Kirchmeyer, M.; Franceschi, J.-Y.; Rakotomamonjy, A.; and Gallinari, P. 2023. Continuous PDE Dynamics Forecasting with Implicit Neural Representations. In *International Conference on Learning Representations (ICLR)*.

Zienkiewicz, O. C.; Taylor, R. L.; and Zhu, J. Z. 2005. *The Finite Element Method: Its Basis and Fundamentals*. Burlington, MA: Elsevier, 6th edition.