ViT-K: A Vision Transformer–Koopman Framework for Coupled Stokes/Navier–Stokes–Darcy Systems*

Mengjia Chen†, Changxin Qiu†

November 18, 2025

Abstract

The Stokes–Darcy and Navier–Stokes–Darcy systems provide fundamental models for coupled free–porous flow, but their nonlinear interface coupling poses significant difficulties for long-time numerical prediction. We therefore develop ViT-K, a Vision Transformer–Koopman framework that captures the essential spatio–temporal structures and yields a stable data-driven evolution of these systems. Numerical experiments on benchmark Stokes–Darcy and Navier–Stokes–Darcy problems demonstrate that ViT-K achieves high reconstruction accuracy and remains stable over prediction horizons far beyond the training interval. These results indicate that ViT-K offers an efficient and robust surrogate modeling approach for complex coupled flow systems.

1 Introduction

1. Stokes/Navier-Stokes-Darcy systems background

The coupled Stokes–Darcy and Navier–Stokes–Darcy systems form a fundamental mathematical model describing the interaction between viscous flow in a free region and seepage flow through a porous medium. These coupled flow phenomena arise in a variety of scientific and engineering contexts, such as groundwater hydrology[1, 2], petroleum recovery[3], biofluid transport[4], and heat or mass transfer in composite materials[5].Owing to their nonlinear coupling and multiscale nature, these systems have become canonical benchmarks for studying multiphysics flow interactions and for developing advanced numerical and data-driven modeling techniques.

2. numerical methods for Stokes/Navier-Stokes-Darcy systems

Over the past several decades, extensive research has been conducted on the numerical simulation of coupled Stokes–Darcy and Navier–Stokes–Darcy systems. To ensure accurate interface coupling and stable numerical approximation, numerous numerical schemes have been proposed and rigorously analyzed. Classical approaches include finite element and mixed finite element methods, which realize the continuity of momentum and flux between the free-flow and porous regions through appropriate discretizations in each subdomain [6–8]. Lagrange multiplier and mortar methods [9–11] were introduced to enforce interface constraints on nonmatching meshes, improving flexibility in complex domains. Domain decomposition techniques [12–15] have also been extensively studied, allowing the Stokes and Darcy subproblems to be solved independently while ensuring consistent interface exchange through iterative or Robin-type transmission conditions. In addition, multigrid and two-grid strategies have been employed to accelerate convergence for large-scale steady problems [16]. These traditional methods have established a rigorous mathematical foundation for the coupled flow models and demonstrated reliable accuracy and

*Corresponding author: Changxin Qiu, qiuchangxin@nbu.edu.cn

†School of Mathematics and Statistics, Ningbo University, Ningbo 315211, P.R. China. @nbu.edu.cn

stability. Nevertheless, their high computational cost and difficulties in handling strongly nonlinear or time-dependent systems have motivated the search for reduced-order and data-driven alternatives.

3. machine learning method for Stokes/Navier-Stokes-Darcy systems With the rapid development of machine learning, an increasing number of methods have emerged to solve Stokes-Darcy and Navier-Stokes-Darcy problems. Raissi et al. [17] proposed the Physics-Informed Neural Network (PINN) method, which integrates physical loss into the loss function of a neural network, enabling the effective combination of physical laws with data-driven learning. This approach has been successfully applied to fluid dynamics and porous media flow problems [18–21]. Li et al. [22]introduced the Fourier Neural Operator (FNO).It learns mappings between function spaces and efficiently solves parametrized partial differential equations, achieving significant acceleration in the solution process for complex systems like Navier-Stokes-Darcy Systerm[23–25]. Ref. [26, 27] employs the DeepONet method to learn the solution operator of parametric partial differential equations (PDEs), mapping input functions (such as boundary conditions and source terms) to the corresponding output solutions.Ref.[28] proposes a method that combines the U-Net architecture with neural operator learning. By introducing encoding and decoding structures along with skip connections, it achieves more efficient parameter learning.

Despite the growing potential of machine learning methods, such as Physics-Informed Neural Networks (PINNs) and Neural Operators/operator learning models, they still face significant challenges. These methods are increasingly used to solve complex partial differential equations (PDEs). However, their practical application to coupled fluid–porous media systems remains problematic.,PINNs often suffer from poorly conditioned loss functions when dealing with strong nonlinearities, interface jumps, or coupled subdomain problems, leading to training failure or a severe decline in accuracy[29].Second, operator learning models face substantial error accumulation and stability issues during long-term predictions for time-varying systems. When these models are used for multi-step extrapolation beyond the training range, prediction errors grow exponentially[30]. Third, these methods still require significant data and computational resources in large-scale, high-dimensional, and multi-scale coupled domains (e.g., free-flow region + porous medium region), limiting their practical deployment in engineering applications[31].

4. our paper: motivation(ViT-K), how to generate this method to solve this system.

To address the challenges mentioned above, we propose a novel algorithm, ViT-K, which combines Vision Transformer (ViT) and Koopman operators to simulate and predict the state of nonlinear systems in the context of Navier-Stokes-Darcy (NSD) flow problems. As a deep learning model that has achieved remarkable success in image processing and time series analysis, ViT excels at capturing complex spatial features. Its self-attention mechanism aids in understanding spatial dependencies between fluid and porous medium regions.Ref [32, 33] utilizes ViT to extract global spatial features from flow data. It demonstrates its advantages in fluid dynamics by effectively capturing complex flow patterns and enhancing the accuracy of flow field prediction. On the other hand, Koopman operator offer a method for linearizing nonlinear dynamical systems, which is key for long-term stability in predictive modeling. By transforming the nonlinear dynamics of the NSD system into a higher-dimensional linear system, Koopman operators mitigate the issue of error propagation over time, a common challenge in long-term predictions of complex fluid systems. The Koopman operator enhances prediction stability by allowing for the use of learnable, frequency-based dynamics within the system [34]. By combining ViT's spatial feature extraction capabilities with the Koopman operator's temporal dynamic modeling capabilities, ViT-K eliminates the need for physical constraints, reduces computational costs, accelerates numerical computation efficiency, and transforms nonlinear evolution into linear evolution in high-dimensional space, thereby enhancing long-term prediction stability.

5. The rest of the paper is organized as follows. Section 2 presents the coupled Stokes–Darcy and Navier–Stokes–Darcy models considered in this work. Section 3 introduces the Vision Transformer, the Koopman operator, and the proposed ViT-K framework. Section 4 develops the structured Koopman formulation and analyzes its stability properties. Section 5 provides numerical experiments demonstrating

the accuracy and long-term stability of ViT-K. Section 6 concludes the paper.

2 Model

In this section, we briefly introduce the time-dependent Stokes-Darcy model and Navier-Stokes-Darcy model with different interface condition.

2.1 Stokes-Darcy Model

We first consider the time-dependent Stokes-Darcy problem on a bounded domain $\Omega = \Omega_D \cup \Omega_S \subset \mathbb{R}^d$ $(d = 2, 3)$, where Ω_D and Ω_S denote the porous-medium and free-flow regions, respectively, separated by the interface $\Gamma = \overline{\Omega}_D \cap \overline{\Omega}_S$.

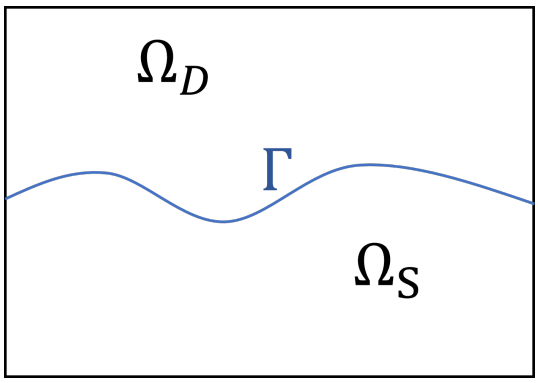

Figure 1: A sketch of the porous median domain Ω_D , fluid domain Ω_S , and the interface Γ

In the porous media region Ω_D, the flow is governed by Darcy's law and mass conservation:

$$\vec{u}_D = -\mathbb{K}\nabla\phi_D, \tag{1}$$

$$S_0\frac{\partial\phi_D}{\partial t} + \nabla\cdot\vec{u}_D = f_D, \tag{2}$$

where $\phi_D = z + \frac{p_D}{\rho g}$ denotes the hydraulic head (with p_D the dynamic pressure, z the height, ρ the density, and g the gravity constant), $\vec{u}_D$ represents the filtration velocity, $\mathbb{K} = k\mathbb{I}$ is the hydraulic conductivity tensor ($k = 1$), S_0 is the mass storativity coefficient ($S_0 = 1$ for simplicity), and f_D is the sink/source term.

Substituting (1) into (2), we obtain the second-order form:

$$S_0\frac{\partial\phi_D}{\partial t} - \nabla\cdot(\mathbb{K}\nabla\phi_D) = f_D. \tag{3}$$

In the fluid region Ω_S, the velocity-pressure pair $(\vec{u}_S, p_S)$ satisfies the transient Stokes system:

$$\frac{\partial\vec{u}_S}{\partial t} - \nabla\cdot\mathbb{T}(\vec{u}_S, p_S) = \vec{f}_S, \tag{4}$$

$$\nabla\cdot\vec{u}_S = 0, \tag{5}$$

where $\mathbb{T}(\vec{u}_S, p_S) = 2\nu\mathbb{D}(\vec{u}_S) - p_S\mathbb{I}$ is the stress tensor, $\nu = 1$ is the kinematic viscosity, and $\mathbb{D}(\vec{u}_S) = \frac{1}{2}(\nabla\vec{u}_S + (\nabla\vec{u}_S)^{\mathsf{T}})$ is the rate of strain tensor.

At the interface Γ, we impose the following standard interface conditions:

- **Continuity of normal flux:**

$$\vec{u}_S \cdot \vec{n}_S = -\vec{u}_D \cdot \vec{n}_D, \tag{6}$$

This ensures the conservation of mass, where $\vec{n}_S$ and $\vec{n}_D$ are the unit normal vectors to the fluid region Ω_S and the porous-medium region Ω_D, respectively.

- **Normal stress balance:**

$$-\vec{n}_S \cdot \mathbb{T}(\vec{u}_S, p_S) \cdot \vec{n}_S = g(\phi_D - z), \tag{7}$$

where g is the gravitational constant, z is the height at the interface.

- **Simplified Beavers-Joseph condition:**

$$\vec{\tau}_j \cdot \vec{u}_S = 0, \quad j = 1, \ldots, d-1, \tag{8}$$

where $\{\vec{\tau}_j\}$ denotes the orthonormal tangent vectors on Γ, and $\vec{n}_S$, $\vec{n}_D$ are the unit normal vectors pointing outward from Ω_S and Ω_D, respectively.

Assume that the hydraulic head ϕ_D and the fluid velocity $\vec{u}_S$ satisfy homogeneous Dirichlet boundary conditions except on Γ, i.e., $\phi_D = 0$ on the boundary $\partial\Omega_D \setminus \Gamma$ and $\vec{u}_S = 0$ on the boundary $\partial\Omega_S \setminus \Gamma$.

Assume that the hydraulic head ϕ_D and the fluid velocity $\vec{u}_S$ satisfy the following initial conditions:

$$\phi_D(0, x, y) = \phi_0(x, y) \quad \text{and} \quad \vec{u}_S(0, x, y) = \vec{u}_0(x, y). \tag{9}$$

2.2 Navier-Stokes-Darcy Model

We now extend the previous model to the Navier-Stokes-Darcy case, where the Stokes system is replaced by the Navier-Stokes equations to account for convective effects.

In the porous media region Ω_D, the governing equations remain identical to the Stokes-Darcy case, as given by equations (1)-(3).

In the fluid region Ω_S, the velocity-pressure pair $(\vec{u}_S, p_S)$ now satisfies the transient Navier-Stokes equations:

$$\frac{\partial \vec{u}_S}{\partial t} + (\vec{u}_S \cdot \nabla)\vec{u}_S - \nabla \cdot \mathbb{T}(\vec{u}_S, p_S) = \vec{f}_S, \tag{10}$$

$$\nabla \cdot \vec{u}_S = 0, \tag{11}$$

with the stress tensor $\mathbb{T}(\vec{u}_S, p_S)$ and deformation tensor $\mathbb{D}(\vec{u}_S)$ defined as before.

At the interface Γ, the conditions for continuity of normal flux (6) and normal stress balance (7) remain unchanged. However, the simplified Beavers-Joseph condition (8) is replaced by the full Beavers-Joseph condition [35]:

$$-\vec{\tau}_j \cdot (\mathbb{T}(\vec{u}_S, p_S) \cdot \vec{n}_S) = \frac{\alpha\nu\sqrt{d}}{\sqrt{\text{trace}(\Pi)}} \vec{\tau}_j \cdot (\vec{u}_S - \vec{u}_D), \quad j = 1, \ldots, d-1, \tag{12}$$

where α is the Beavers-Joseph coefficient, $\Pi = \frac{\mathbb{K}\nu}{g}$ is the permeability tensor, and $\{\vec{\tau}_j\}$ are the orthonormal tangent vectors on Γ.

The system is closed by the initial conditions specified in (9). Regarding the boundary conditions on the external boundary $\partial\Omega \setminus \Gamma$, Dirichlet conditions are imposed. For problems involving a manufactured solution, such as the numerical example considered in this work, these are typically inhomogeneous Dirichlet conditions where the boundary values are prescribed by the exact solution itself.

3 Vision Transformer with Koopman (ViT-K)

This section presents a ViT-K method, which combines the Vision-Transformer (ViT) approach with the Koopman operator. The ViT-K method integrates the spatial information capture capability of ViT and the temporal extrapolation capability of the Koopman operator, enabling the prediction of the state of Stokes/Navier-Stokes-Darcy systems.

3.1 ViT Structure

The Vision Transformer (ViT), first introduced by Dosovitskiy et al. [**?**], applies the Transformer architecture to image-like data by treating an image as a sequence of patch embeddings rather than using convolutional filters. Unlike CNN-based encoders, which extract features through localized receptive fields, ViT captures global spatial dependencies via multi-head self-attention, making it suitable for learning long-range interactions in complex physical fields [36, 37].

Given a standard 2D input image $\mathbf{X} \in \mathbb{R}^{H\times W\times C}$, where H, W, and C are the height, width, and number of channels of the image, respectively, the first step of ViT is to partition the image into a sequence of N flattened patches $\mathbf{X}_p \in \mathbb{R}^{N\times(P^2\cdot C)}$, where P is the spatial dimension of each patch, and $N = \frac{HW}{P^2}$ is the number of patches, also representing the effective input sequence length for the Transformer.

To retain positional information of the patches, a learnable positional embedding $\mathbf{E}_{pos}$ is added to the patch embeddings. Additionally, a special `[class]` token embedding, denoted as $\mathbf{x}_{class}$, is prepended to the patch sequence. The state of this token at the output of the Transformer encoder serves as the global image representation $\mathbf{z}_L^0$ used for final classification. The input sequence $\mathbf{z}_0$, incorporating the `[class]` token and positional embeddings, is calculated as follows:

$$\mathbf{z}_0 = [\mathbf{x}_{class}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \ldots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos}, \quad \mathbf{E} \in \mathbb{R}^{(P^2\cdot C)\times D}, \quad \mathbf{E}_{pos} \in \mathbb{R}^{(N+1)\times D},$$

where $\mathbf{E}$ is a linear projection matrix mapping the flattened patches into a D-dimensional latent space.

The embedded sequence $\mathbf{z}_0$ is then processed by a standard Transformer encoder consisting of L layers. Each layer l contains a Multi-Headed Self-Attention (MSA) module and a Feed-Forward Network (FFN), with Layer Normalization (LN) applied before each module and residual connections added after each module, formulated as:

$$\mathbf{z}'_l = \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{l-1})) + \mathbf{z}_{l-1}, \quad l = 1, \ldots, L,$$

$$\mathbf{z}_l = \mathrm{FFN}(\mathrm{LN}(\mathbf{z}'_l)) + \mathbf{z}'_l, \quad l = 1, \ldots, L.$$

Here, FFN typically consists of two linear layers with a GELU activation in between.

The output of the Transformer encoder is a sequence of embedding vectors $\mathbf{z}_L \in \mathbb{R}^{(N+1)\times D}$. We take the first element of this sequence, corresponding to the `[class]` token $\mathbf{z}_L^0 \in \mathbb{R}^{1\times D}$, as the final representation of the input image. A linear classification head is then applied to this representation to produce the output logits:

$$\mathbf{y} = \mathrm{Linear}(\mathrm{LN}(\mathbf{z}_L^0)),$$

where $\mathbf{y} \in \mathbb{R}^K$ and K is the number of classes.

The ViT model is trained by minimizing a loss function, typically the cross-entropy loss between the ground-truth labels and the predictions:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \sum_{k=1}^{K} y_{i,k} \log(\hat{y}_{i,k}),$$

where $y_{i,k}$ is the ground-truth label (one-hot encoded) for the i-th sample, and $\hat{y}_{i,k}$ is the predicted probability of the i-th sample belonging to the k-th class, obtained by applying the softmax function to $\mathbf{y}$. By minimizing $\mathcal{L}_{\text{CE}}$ using an optimizer (such as AdamW) to update the model parameters θ (which include the projection matrix $\mathbf{E}$, positional embeddings $\mathbf{E}_{pos}$, and all weights and biases in the Transformer encoder and classification head), the model learns the mapping from the input image to the output class.

The typical ViT framework is shown in Figure 2.

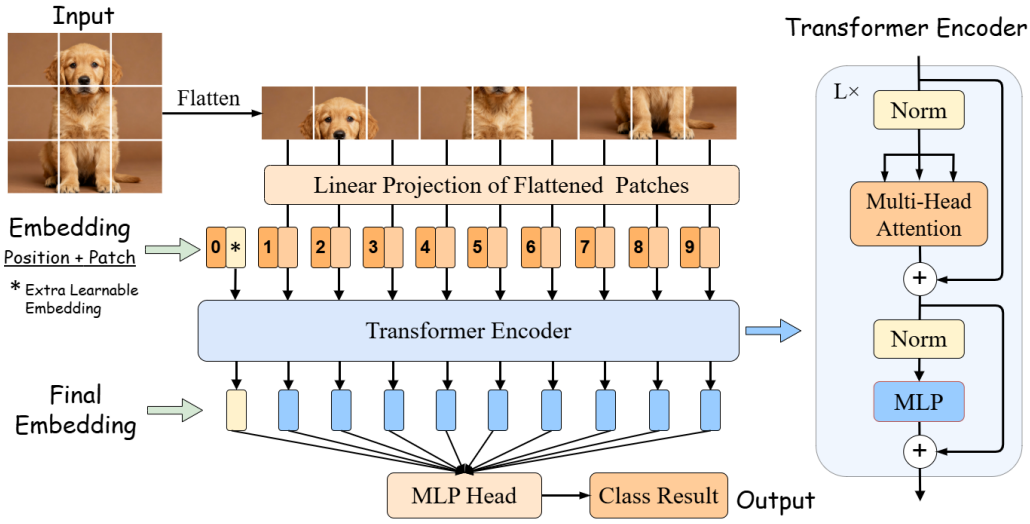

Figure 2: ViT Framework

.

3.2 Koopman Operator

For continuous–time dynamical systems arising from Stokes-Darcy or Navier–Stokes–Darcy (NSD) flows, the state evolution

$$x(t + \Delta t) = \Phi_{\Delta t}\big(x(t)\big) \tag{13}$$

is generally nonlinear, which makes direct long–horizon extrapolation in the state space difficult. The Koopman framework provides a linear representation of this evolution in an associated observable space, first introduced by Koopman (1931) [38] and later expanded in the modern framework for dynamic systems [39].

Let $\mathcal{X}$ be the state space and $\{\Phi_t\}_{t\geq 0}$ the (nonlinear) solution semigroup. For a Banach space $\mathcal{G}$ of scalar observables $g : \mathcal{X} \to \mathbb{C}$, the discrete–time Koopman operator is defined as

$$(\mathcal{K}_{\Delta t} g)(x) = g(\Phi_{\Delta t}(x)) \,, \tag{14}$$

and, in continuous time, the corresponding generator $\mathcal{L}$ satisfies

$$\frac{d}{dt}\, g(\Phi_t(x)) = (\mathcal{L}g)(\Phi_t(x)), \qquad \mathcal{K}_{\Delta t} = e^{\Delta t\, \mathcal{L}}. \tag{15}$$

Thus, while Φ_t is nonlinear in $\mathcal{X}$, both $\mathcal{K}_{\Delta t}$ and $\mathcal{L}$ act linearly on observables $g \in \mathcal{G}$.

If $\mathcal{L}\phi = \omega\phi$ (equivalently $\mathcal{K}_{\Delta t}\phi = e^{\omega \Delta t}\phi$), then

$$\phi(\Phi_t(x)) = e^{\omega t}\,\phi(x), \tag{16}$$

so any observable that admits an expansion in Koopman eigenfunctions evolves as a superposition of exponentially scaled (possibly oscillatory) modes.

3.3 ViT-K for Stokes/Navier-Stokes-Darcy systems

Accurate prediction of the spatio-temporal evolution in the Stokes/Navier–Stokes–Darcy (NSD) system remains a challenging task in multiphysics flow modeling. The nonlinear coupling among convective transport, viscous diffusion, and porous-medium flow introduces strong multiscale interactions that often lead to numerical instability and long-term error accumulation in conventional simulations.

We develop a Vision Transformer–Koopman (ViT-K) framework that combines data-driven representation learning with linear dynamical modeling. The proposed method extracts dominant spatio-temporal structures of the NSD system and enables stable predictions of complex flow dynamics.The model mapping the nonlinear spatio-temporal dynamics of flow fields into a latent observable space governed by linear dynamics. This enables stable and interpretable predictions of fluid behavior.The overall architecture of the ViT-K model is illustrated in Figure 3.

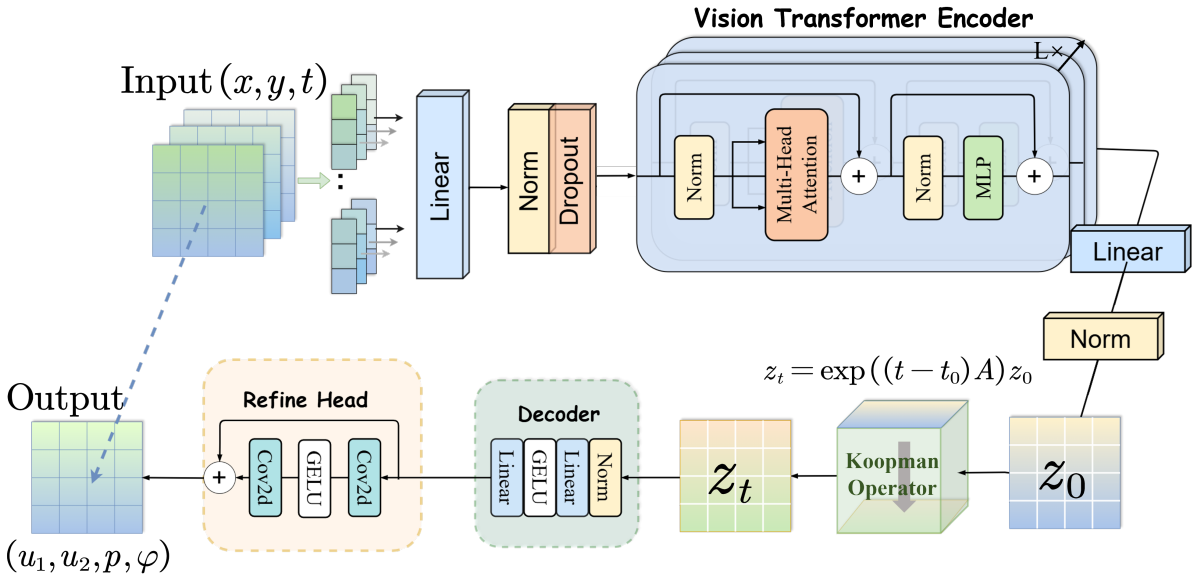

Figure 3: ViT-K framework for solving the PDEs.

The encoder employs a Vision Transformer structure. The input flow field is divided into spatial patches, and multi-head self-attention captures global correlations across regions. The resulting latent representations effectively encode the spatial coupling among velocity, pressure, and potential fields, providing a compact and physically meaningful basis for temporal evolution.

In the latent space, ViT-K introduces a learnable linear operator as a finite-dimensional approximation of the Koopman generator. This operator advances the latent states linearly in time, replacing the nonlinear dynamics of the original PDE system. Such linear evolution ensures stability and interpretability and allows extrapolation over long temporal horizons.

The decoder maps the evolved latent variables back to the physical domain, reconstructing the velocity components, pressure field, and Darcy potential at each time step. A combination of linear projection and local convolutional refinement ensures consistency between global structures and fine spatial details.

To balance prediction accuracy across different physical subdomains, a domain-weighted reconstruction loss is introduced. Let Ω_S and Ω_D denote the Stokes/Navier–Stokes and Darcy subdomains, with N_{Ω_S} and N_{Ω_D} grid points, respectively. Denote the predicted fields by $\hat{u}_1, \hat{u}_2, \hat{p}, \hat{\phi}$ and the reference fields by u_1, u_2, p, ϕ. The total loss is defined as

$$\mathcal{L} = w_{u_1} \,\mathrm{MSE}_{\Omega_S}(\hat{u}_1, u_1) + w_{u_2} \,\mathrm{MSE}_{\Omega_S}(\hat{u}_2, u_2) + w_p \,\mathrm{MSE}_{\Omega_S}(\hat{p}, p) + w_\phi \,\mathrm{MSE}_{\Omega_D}(\hat{\phi}, \phi),$$

where the discrete mean-square error over domain Ω is

$$\mathrm{MSE}_\Omega(\hat{f}, f) = \frac{1}{N_\Omega} \sum_{(i,j)\in\Omega} (\hat{f}_{i,j} - f_{i,j})^2.$$

Here N_Ω denotes the number of grid points in domain Ω. The indicator functions $\mathbf{1}_{\Omega_S}$ and $\mathbf{1}_{\Omega_D}$ take the value 1 within their respective subdomains and 0 elsewhere, masking non-target regions during training.

This loss computes weighted mean-square errors in each subdomain, ensuring balanced accuracy and physical consistency in both fluid and porous regions. Minimizing $\mathcal{L}$ jointly optimizes the encoder, the linear operator, and the decoder, achieving unified modeling of NSD dynamics and stable temporal extrapolation.

Overall, ViT-K combines the global spatial representation power of Transformers with the linear and interpretable dynamics of the Koopman framework, enabling accurate, stable, and physically consistent long-term predictions for coupled flow systems.

4 A Structured Koopman Operator for Long-Term Prediction

A central challenge in data-driven modeling of dynamical systems is ensuring long-term stability and physical plausibility. Standard approaches that learn a dense, unconstrained Koopman generator often suffer from overfitting to spurious, unstable modes and lack clear physical interpretability. To overcome these limitations, we introduce a powerful *inductive bias* by imposing a physically-motivated structure on the Koopman generator $\mathbf{A}$, from which the continuous-time evolution operator $e^{t\mathbf{A}}$ is derived. Our design is guided by the principle that fluid systems, governed by inertial and viscous effects, are inherently dissipative and should not spontaneously generate energy.

4.1 A General Stable Structure via Operator Decomposition

Any real square matrix $\mathbf{A}$ can be uniquely decomposed into its symmetric and skew-symmetric parts: $\mathbf{A} = \mathbf{S} + \mathbf{W}$, where $\mathbf{S} = \frac{1}{2}(\mathbf{A} + \mathbf{A}^\top)$ is the symmetric part and $\mathbf{W} = \frac{1}{2}(\mathbf{A} - \mathbf{A}^\top)$ is the skew-symmetric part. From a dynamical systems perspective, the symmetric part $\mathbf{S}$ governs the contraction or expansion of the state space volume (related to energy dissipation), while the skew-symmetric part $\mathbf{W}$ governs energy-preserving rotations or oscillations.

To enforce stability, we constrain the symmetric part to be negative semi-definite, which ensures that the system's energy can only dissipate or be conserved. This leads to our general stable generator structure:

- $\mathbf{S}$ is a symmetric negative semi-definite matrix ($\mathbf{S} = \mathbf{S}^\top$, $\mathbf{z}^\top \mathbf{S} \mathbf{z} \leq 0$ for all $\mathbf{z}$). This component models **dissipation**.
- $\mathbf{W}$ is a skew-symmetric matrix ($\mathbf{W} = -\mathbf{W}^\top$). This component models **oscillation**.

This construction provides a formal guarantee for the stability of the learned dynamics.

Proposition 4.1 (Stability of the General Generator)**.** *Let the Koopman generator be constructed as* $\mathbf{A} = \mathbf{S} + \mathbf{W}$*, where* $\mathbf{S}$ *is symmetric negative semi-definite and* $\mathbf{W}$ *is skew-symmetric. Then, for any time* $t > 0$*, the spectral radius of the evolution operator* $e^{t\mathbf{A}}$ *satisfies* $\rho(e^{t\mathbf{A}}) \leq 1$*.*

Proof. Let λ be an eigenvalue of $\mathbf{A}$ with corresponding eigenvector $\mathbf{v}$. The real part of λ, $\mathrm{Re}(\lambda)$, determines the stability. Consider the quadratic form $\mathbf{v}^*\mathbf{A}\mathbf{v}$, where $\mathbf{v}^*$ is the conjugate transpose.

$$\mathbf{v}^*\mathbf{A}\mathbf{v} = \mathbf{v}^*(\mathbf{S} + \mathbf{W})\mathbf{v} = \mathbf{v}^*\mathbf{S}\mathbf{v} + \mathbf{v}^*\mathbf{W}\mathbf{v}. \tag{17}$$

Also, $\mathbf{v}^*\mathbf{A}\mathbf{v} = \lambda\|\mathbf{v}\|^2$, so its real part is $\mathrm{Re}(\lambda)\|\mathbf{v}\|^2$. For the right-hand side, since $\mathbf{S}$ is real symmetric, $\mathbf{v}^*\mathbf{S}\mathbf{v}$ is real and, due to its negative semi-definite property, $\mathbf{v}^*\mathbf{S}\mathbf{v} \leq 0$. Since $\mathbf{W}$ is real skew-symmetric, $\mathbf{v}^*\mathbf{W}\mathbf{v}$ is purely imaginary. Thus, $\mathrm{Re}(\mathbf{v}^*\mathbf{A}\mathbf{v}) = \mathbf{v}^*\mathbf{S}\mathbf{v} \leq 0$. Equating the real parts gives $\mathrm{Re}(\lambda)\|\mathbf{v}\|^2 \leq 0$, which implies $\mathrm{Re}(\lambda) \leq 0$. By the spectral mapping theorem, the eigenvalues of $e^{t\mathbf{A}}$ are $\mu_i = e^{t\lambda_i}$, and their magnitude is $|\mu_i| = e^{t\mathrm{Re}(\lambda_i)} \leq 1$. □

4.2 A Decoupled Dissipative-Oscillatory Structure

While the general structure in Sec. 4.1 allows for full modal coupling, a stronger and more interpretable inductive bias can be achieved by assuming that the latent dynamics are largely decoupled. This is a reasonable assumption if the encoder successfully identifies the dominant, quasi-orthogonal modes of the system. We therefore parameterize the generator $\mathbf{A} \in \mathbb{R}^{d\times d}$ as a block-diagonal matrix composed of $d/2$ independent 2×2 blocks:

$$\mathbf{A} = \bigoplus_{i=1}^{d/2} \mathbf{A}_i, \quad \text{where} \quad \mathbf{A}_i = \begin{bmatrix} -\gamma_i & -\omega_i \\ \omega_i & -\gamma_i \end{bmatrix}. \tag{18}$$

Each block $\mathbf{A}_i$ is a special case of the stable $\mathbf{S}+\mathbf{W}$ form, with $\mathbf{S}_i = \mathrm{diag}(-\gamma_i, -\gamma_i)$ and $\mathbf{W}_i = \begin{pmatrix} 0 & -\omega_i \\ \omega_i & 0 \end{pmatrix}$. The learnable parameters $\{\gamma_i, \omega_i\}$ directly represent the decay rates and frequencies of the latent modes. To enforce stability, we constrain the physical decay rates $\gamma_i \geq 0$, typically by parameterizing them via a non-negative function (e.g., softplus). This design provides a formal and direct guarantee of stability.

Proposition 4.2 (Inherent Stability of Decoupled Structure)**.** *Let the Koopman generator* $\mathbf{A}$ *be constructed as in Eq. (18) with the constraint* $\gamma_i \geq 0$ *for all* i*. Then, for any time* $t > 0$*, the spectral radius of the evolution operator satisfies* $\rho(e^{t\mathbf{A}}) \leq 1$*.*

Proof. The eigenvalues of each block $\mathbf{A}_i$ are $\lambda_i = -\gamma_i \pm i\omega_i$. By the spectral mapping theorem, the eigenvalues of the evolution operator $e^{t\mathbf{A}}$ are $\mu_i = e^{t\lambda_i}$. Their magnitude is given by:

$$|\mu_i| = |e^{t(-\gamma_i \pm i\omega_i)}| = |e^{-t\gamma_i}| \cdot |e^{\pm it\omega_i}| = e^{-t\gamma_i}. \tag{19}$$

Since $\gamma_i \geq 0$ and $t > 0$, we have $e^{-t\gamma_i} \leq 1$. Thus, all eigenvalues of the evolution operator lie within or on the unit circle. □

4.3 Theoretical Justification of the Learning Process

In this section, we provide a theoretical argument to justify that the learned Koopman generator, denoted as $\mathbf{A}$, is a principled approximation of the true system dynamics projected onto the latent space discovered by the Vision Transformer (ViT) encoder. Our approach can be interpreted as an end-to-end, nonlinear extension of the Extended Dynamic Mode Decomposition (EDMD) framework [40].

Let the true, unknown Koopman generator of the continuous-time dynamical system be denoted by the infinite-dimensional linear operator $\mathcal{L}$. Our goal is to find a finite-dimensional matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$ that best approximates the action of $\mathcal{L}$ on a carefully chosen set of observables.

The key to this approximation lies in the choice of observables. Our ViT encoder, $\Psi_{\text{enc}} : \mathcal{X} \to \mathbb{R}^d$, provides a nonlinear mapping from the high-dimensional state space $\mathcal{X}$ to a d-dimensional latent space of observables. Let this vector of observables be $\mathbf{g}(\mathbf{x}) = \Psi_{\text{enc}}(\mathbf{x})$. These observables span a finite-dimensional subspace, $\mathcal{G}_d = \text{span}\{g_1, \ldots, g_d\}$, within the full space of all possible observables.

The action of the true generator $\mathcal{L}$ on our vector of observables $\mathbf{g}$ can be decomposed into a component that lies within our chosen subspace $\mathcal{G}_d$ and a residual component $\mathbf{r}(\mathbf{x})$ that is orthogonal to it:

$$\mathcal{L}\mathbf{g}(\mathbf{x}) = \mathbf{A}^*\mathbf{g}(\mathbf{x}) + \mathbf{r}(\mathbf{x}). \tag{20}$$

Here, $\mathbf{A}^*$ is the *optimal* finite-dimensional generator that represents the action of $\mathcal{L}$ projected onto the subspace $\mathcal{G}_d$. This is, in essence, a Galerkin projection of the infinite-dimensional operator $\mathcal{L}$ onto the subspace spanned by the learned basis functions $\{\mathbf{g}_i\}$. The term $\mathbf{r}(\mathbf{x})$ represents the intrinsic **projection error** (or truncation error), which is the part of the dynamics that cannot be captured by any linear model within the chosen subspace $\mathcal{G}_d$.

Our objective is to demonstrate that the training process, which minimizes a data-driven loss function, yields a matrix $\mathbf{A}$ that is a good approximation of the ideal operator $\mathbf{A}^*$.

The core of our training involves minimizing loss terms related to the linear evolution in the latent space. Let's consider a one-step prediction loss in its continuous form. The time evolution of observables is governed by $\frac{d}{dt}\mathbf{g}(\mathbf{x}(t)) = (\mathcal{L}\mathbf{g})(\mathbf{x}(t))$. A first-order Euler discretization yields:

$$\mathbf{g}(\mathbf{x}_{k+1}) \approx \mathbf{g}(\mathbf{x}_k) + \Delta t \cdot (\mathcal{L}\mathbf{g})(\mathbf{x}_k). \tag{21}$$

Substituting the projection (20) into (21), we get:

$$\mathbf{g}(\mathbf{x}_{k+1}) \approx \mathbf{g}(\mathbf{x}_k) + \Delta t \cdot (\mathbf{A}^*\mathbf{g}(\mathbf{x}_k) + \mathbf{r}(\mathbf{x}_k)). \tag{22}$$

Rearranging this, we see that the ideal finite-dimensional operator $\mathbf{A}^*$ should ideally satisfy:

$$\mathbf{A}^*\mathbf{g}(\mathbf{x}_k) \approx \frac{\mathbf{g}(\mathbf{x}_{k+1}) - \mathbf{g}(\mathbf{x}_k)}{\Delta t} - \mathbf{r}(\mathbf{x}_k). \tag{23}$$

This forms the basis of the classic EDMD algorithm, which solves a least-squares problem over a dataset of snapshots $\{(\mathbf{x}_k, \mathbf{x}_{k+1})\}$ to find the best linear fit.

Our model's training objective, particularly the linearity loss term, is an end-to-end, deep learning analogue of this process. The loss function, when summed over a large dataset of M snapshot pairs, aims to find parameters for Ψ_{enc} and the generator $\mathbf{A}$ that minimize:

$$\mathcal{L}_{\text{linearity}} = \frac{1}{M}\sum_{k=1}^{M} \left\| \Psi_{\text{enc}}(\mathbf{x}_{k+1}) - e^{\Delta t_k \mathbf{A}} \Psi_{\text{enc}}(\mathbf{x}_k) \right\|^2. \tag{24}$$

Minimizing this loss with respect to $\mathbf{A}$ for a fixed encoder Ψ_{enc} is equivalent to solving for the best linear operator that maps the encoded states forward in time.

We can now formalize the quality of our learned approximation. The total one-step prediction error in the latent space can be bounded. Let $\mathbf{g}_k = \Psi_{\text{enc}}(\mathbf{x}_k)$. The error is given by $\left\|\mathbf{g}_{k+1} - e^{\Delta t \mathbf{A}}\mathbf{g}_k\right\|$. Using the first-order approximation $e^{\Delta t\mathbf{A}} \approx \mathbf{I} + \Delta t\mathbf{A}$ for small Δt, the error is approximately $\|(\mathbf{g}_{k+1} - \mathbf{g}_k)/\Delta t - \mathbf{A}\mathbf{g}_k\|$.

Proposition 4.3 (Error Decomposition)**.** *The total one-step approximation error for the latent dynamics can be bounded by the sum of two primary error sources: a regression error and a projection error.*

$$\mathbb{E}_{\mathbf{x}} \left\| \frac{\mathbf{g}(\mathbf{x}_{k+1}) - \mathbf{g}(\mathbf{x}_k)}{\Delta t} - \mathbf{A}\mathbf{g}_k \right\| \leq \mathbb{E}_{\mathbf{x}} \left\| (\mathbf{A}^* - \mathbf{A})\mathbf{g}_k \right\| + \mathbb{E}_{\mathbf{x}} \left\| \mathbf{r}(\mathbf{x}_k) \right\|. \tag{25}$$

Proof. By adding and subtracting the term $\mathbf{A}^*\mathbf{g}_k$ and applying the triangle inequality:

$$\begin{aligned}\left\|\frac{\mathbf{g}_{k+1}-\mathbf{g}_k}{\Delta t}-\mathbf{A}\mathbf{g}_k\right\| &= \left\|\left(\frac{\mathbf{g}_{k+1}-\mathbf{g}_k}{\Delta t}-\mathbf{A}^*\mathbf{g}_k\right)+(\mathbf{A}^*-\mathbf{A})\mathbf{g}_k\right\| \\ &\leq \left\|\frac{\mathbf{g}_{k+1}-\mathbf{g}_k}{\Delta t}-\mathbf{A}^*\mathbf{g}_k\right\|+\|(\mathbf{A}^*-\mathbf{A})\mathbf{g}_k\| \\ &\approx \|\mathbf{r}(\mathbf{x}_k)\|+\|(\mathbf{A}^*-\mathbf{A})\mathbf{g}_k\|\,.\end{aligned}$$

The approximation in the last step comes from ignoring higher-order terms in the discretization. Taking the expectation over the data distribution yields the result. □

Interpretation of the Error Terms:

- **Regression Error:** The term $\mathbb{E}\,\|(\mathbf{A}^*-\mathbf{A})\mathbf{g}_k\|$ represents how well our learned generator $\mathbf{A}$ approximates the ideal projected generator $\mathbf{A}^*$. This error is primarily a function of the amount and quality of the training data. As the number of training samples $M\to\infty$, standard results from linear system identification suggest that $\mathbf{A}\to\mathbf{A}^*$. Our training process is explicitly designed to minimize this term.
- **Projection Error:** The term $\mathbb{E}\,\|\mathbf{r}(\mathbf{x}_k)\|$ is the fundamental error introduced by approximating an infinite-dimensional system with a finite one. Its magnitude depends entirely on the quality of the subspace $\mathcal{G}_d$ learned by the ViT encoder. A more powerful and well-trained encoder will find a set of observables $\mathbf{g}$ where the dynamics are "more linear," thus minimizing this residual.

Conclusion of the Argument: Our end-to-end training procedure is justified because it simultaneously performs two critical tasks:

1. It optimizes the ViT encoder Ψ_{enc} to find a latent space that minimizes the **projection error**.
2. It optimizes the generator $\mathbf{A}$ to minimize the **regression error** within that learned space.

Furthermore, by imposing the dissipative-oscillatory structure on $\mathbf{A}$, we ensure that this entire approximation process is constrained to a stable manifold, guaranteeing the long-term stability of the resulting surrogate model.

5 Numerical experiments

In this section, we present a series of numerical experiments to validate the accuracy and long-term stability of the proposed ViT-K framework.

5.1 Example 1: Stokes–Darcy Problem

We first evaluate our ViT-K framework on a time-dependent Stokes-Darcy problem, as defined in Section 2.1. The problem is set on the domain $\Omega=[0,\pi]\times[-1,1]$, with the porous-medium region $\Omega_D=[0,\pi]\times[0,1]$ and the free-flow region $\Omega_S=[0,\pi]\times[-1,0]$. The exact solutions for the hydraulic head (ϕ_D), fluid velocity ($\vec{u}_S$), and pressure (p_S) are given by:

$$\phi_D=(e^y-e^{-y})\sin(x)e^{-t},$$

$$\vec{u_S}=\left[\frac{k}{\pi}\sin(2\pi y)\cos(x)e^{-t},\ \left(-2k+\frac{k}{\pi^2}\sin^2(\pi y)\right)\sin(x)e^{-t}\right]^T,$$

$$p_S = 0,$$

with physical parameters set to $\nu = 1, g = 1, z = 0$, and $k = 1$. For this experiment, we test the model's ability to learn the dissipative dynamics and extrapolate stably. The training data is generated from 20 snapshots over the time interval $t \in [0, 1.0]$, and the model is evaluated on 40 snapshots from an extended interval $t \in [0, 2.0]$. The physical fields are discretized on a uniform 96×96 grid. Each physical channel is normalized by its maximum amplitude, and the binary subdomain mask is used as an additional input feature.

Then, we could follow the training process in Figure 3 to train the VIT-K method.

Our ViT-K model is configured with a ViT encoder that uses a patch size of 16×16, an embedding dimension of 192, and consists of 6 Transformer layers, each with 6 attention heads and a feedforward dimension of 384. To handle the dissipative nature of this problem, we employ the *purely dissipative Koopman structure* (as outlined in Section 4). The temporal evolution in the latent space is governed by this continuous-time Koopman operator, implemented via the matrix exponential $K(\Delta t) = \exp(\Delta t \mathbf{A})$ to ensure linearly stable propagation. The model is trained for 1000 epochs with a batch size of 16 using an 80/20 training/validation split. We use the AdamW optimizer with a learning rate of 5×10^{-4}, weight decay of 2×10^{-5}, and a cosine annealing schedule with a 10-epoch linear warmup. Gradient clipping with a maximum norm of 0.5 is applied during training. The training objective is a domain-weighted Mean Squared Error (MSE) loss with weights set to $w_{u_1} = 2.0, w_{u_2} = 5.0, w_p = 0.1$, and $w_\phi = 1.0$.

Model performance is quantitatively assessed using three standard metrics. For each quantity $f \in \{u_1, u_2, p, \phi\}$ at a given time t_n, we define:

$$\text{MSE} = \frac{1}{N}\sum_{i,j}(\hat{f}_{i,j} - f_{i,j})^2, \quad \text{MAE} = \frac{1}{N}\sum_{i,j}|\hat{f}_{i,j} - f_{i,j}|, \quad \text{Rel}\, L^2 = \frac{\|\hat{f} - f\|_2}{\|f\|_2},$$

where N is the number of grid points in the corresponding domain.

Table 1: Stokes-Darcy Model Prediction Quality Assessment

Channel	Time	MSE	MAE	Max Error	Relative Error(%)
u_1	$t = 0.0$	3.297×10^{-5}	4.3997×10^{-3}	2.2015×10^{-2}	1.1484%
	$t = 0.5$	1.309×10^{-5}	2.7564×10^{-3}	1.4307×10^{-2}	1.2084%
	$t = 1.0$	8.03×10^{-6}	2.1114×10^{-3}	1.4498×10^{-2}	1.5808%
	$t = 1.5$	1.761×10^{-5}	3.2086×10^{-3}	2.4394×10^{-2}	3.7132%
	$t = 2.0$	4.520×10^{-5}	5.2711×10^{-3}	3.4467×10^{-2}	9.9344%
u_2	$t = 0.0$	3.393×10^{-5}	4.6971×10^{-3}	2.0912×10^{-2}	0.8491%
	$t = 0.5$	1.437×10^{-5}	3.0789×10^{-3}	1.4911×10^{-2}	0.9227%
	$t = 1.0$	6.30×10^{-6}	1.9532×10^{-3}	1.0190×10^{-2}	1.0205%
	$t = 1.5$	3.852×10^{-5}	5.5740×10^{-3}	1.5741×10^{-2}	4.0032%
	$t = 2.0$	1.7425×10^{-4}	1.2444×10^{-2}	2.5514×10^{-2}	14.2185%
p	$t = 0.0$	1.253×10^{-5}	2.6463×10^{-3}	2.1233×10^{-2}	-
	$t = 0.5$	2.82×10^{-6}	1.2956×10^{-3}	7.3977×10^{-3}	-
	$t = 1.0$	5.05×10^{-6}	1.6247×10^{-3}	1.5721×10^{-2}	-
	$t = 1.5$	9.51×10^{-6}	2.2394×10^{-3}	1.9518×10^{-2}	-
	$t = 2.0$	1.564×10^{-5}	2.9479×10^{-3}	2.3157×10^{-2}	-
ϕ	$t = 0.0$	2.761×10^{-5}	3.9402×10^{-3}	2.3406×10^{-2}	1.3594%
	$t = 0.5$	9.54×10^{-6}	2.4101×10^{-3}	1.6368×10^{-2}	1.3341%
	$t = 1.0$	6.97×10^{-6}	2.0843×10^{-3}	8.9245×10^{-3}	1.9042%
	$t = 1.5$	2.446×10^{-5}	4.1397×10^{-3}	1.6538×10^{-2}	5.6604%
	$t = 2.0$	8.381×10^{-5}	8.0832×10^{-3}	2.4294×10^{-2}	15.4993%

Table 1 provides a quantitative assessment of the predictive performance of our proposed ViT-K model on the Stokes-Darcy problem. The table details the prediction errors for the velocity components (u_1, u_2), pressure (p), and hydraulic head (ϕ) at various time instances.

The results demonstrate that within the training domain ($t \in [0, 1.0]$), the model can reconstruct the system's dynamics with high fidelity. The Mean Squared Error (MSE) for all physical quantities remains at a low level, on the order of 10^{-5} to 10^{-6}, and the relative errors for the velocity components and hydraulic head are generally below 2%. This confirms the model's strong learning and fitting capabilities.

In the unseen prediction domain ($t \in (1.0, 2.0]$), we observe a gradual accumulation of error as the prediction horizon increases. For instance, the relative error for u_2 grows from 1.02% at $t = 1.0$ to 14.22% at $t = 2.0$. For example, the relative error of u_2 increases from 1.02% at $t = 1.0$ to 14.22% at $t = 2.0$. However, this error growth exhibits stable and bounded characteristics, showing no signs of catastrophic failure or numerical divergence. The absolute error metric consistently remains at a low level, confirming the effectiveness of the Kupman operator in ensuring long-term stability and suppressing false modal amplification.The relative error for the pressure field p is not listed. This is because the analytical

solution for pressure is always zero, rendering the relative L^2 error metric mathematically undefined. However, the absolute error metric remains informative. The consistently low mean squared error of the pressure term (approximately 10^{-5} magnitude) validates the model's ability to accurately predict near-zero pressure fields, precisely capturing the core physical characteristics of this solution.

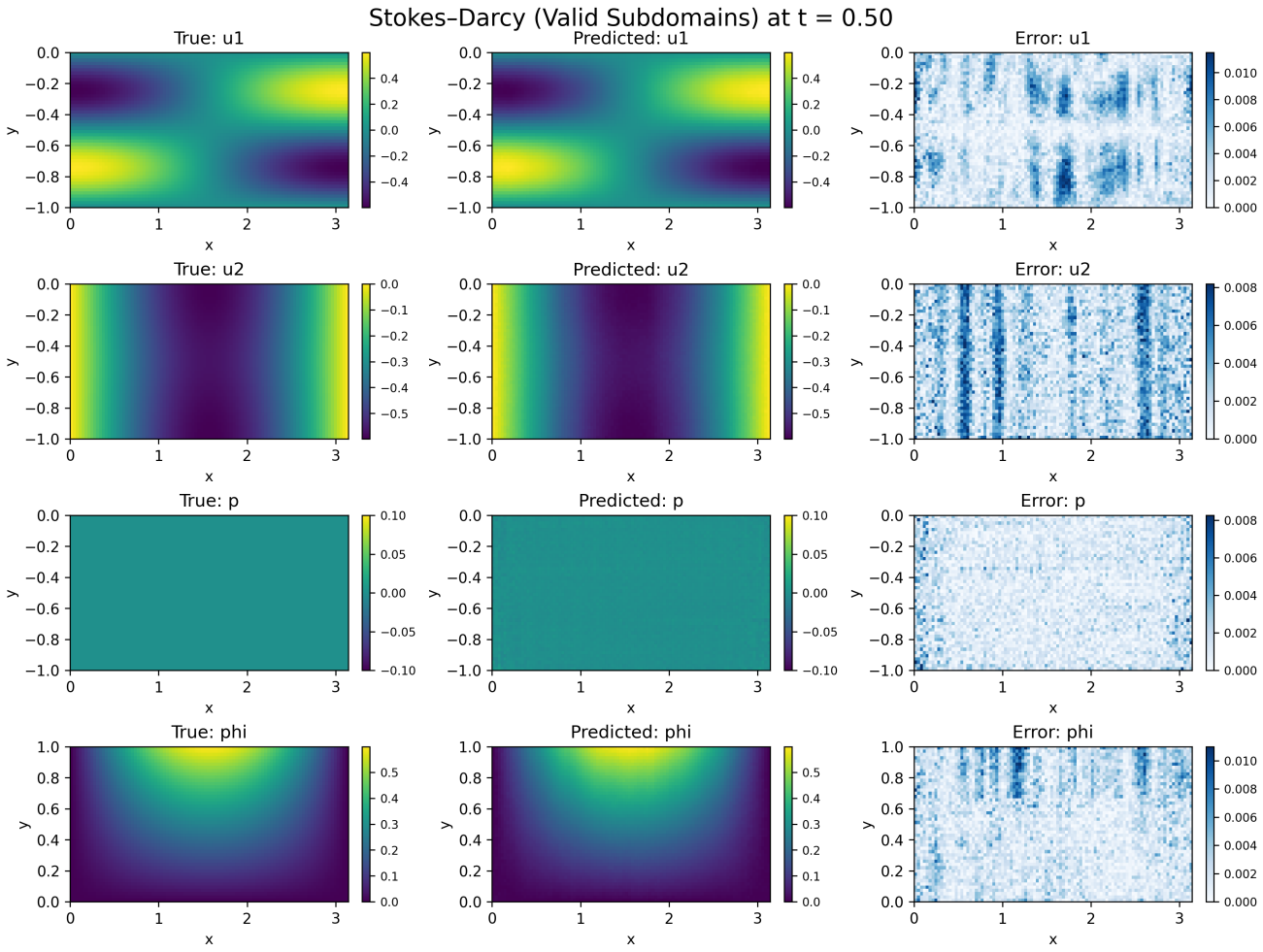

Figure 4: The exact solution (left), the approximation from VIT (middle) and the absolute error (right) at $t = 0.5$.

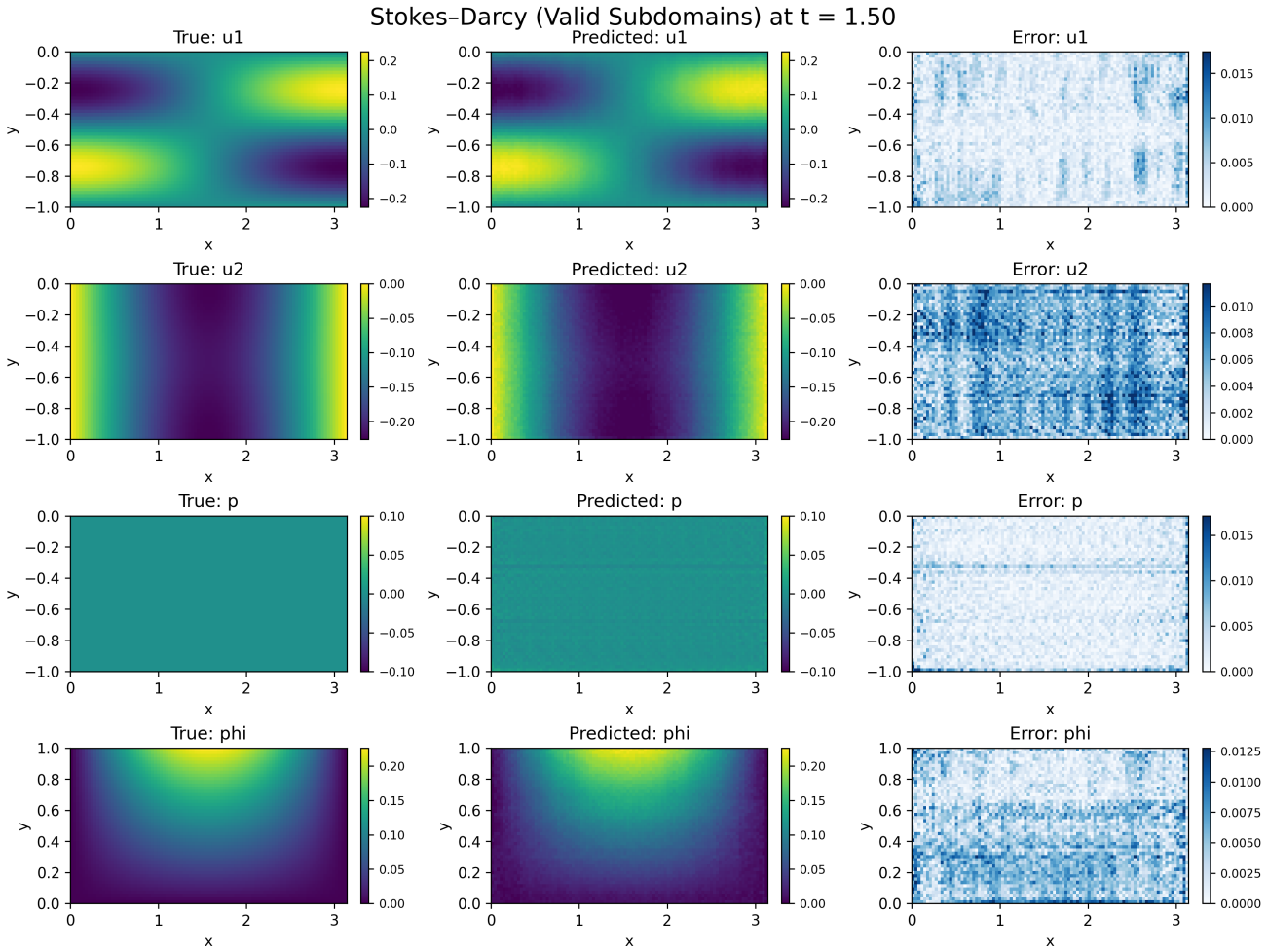

Figure 5: The exact solution (left), the approximation from VIT (middle) and the absolute error (right) at $t = 0.5$.

Figure 4 displays the predicted fields at $t = 0.5$, a time instance within the training domain. The predicted solution is visually indistinguishable from the ground truth, and the absolute error is negligible across the entire domain, confirming the model's high reconstruction fidelity. In contrast, Figure 5 shows the model's performance in the extrapolation regime at $t = 1.5$. Here, the model successfully captures the macroscopic structure and magnitude of all physical fields.

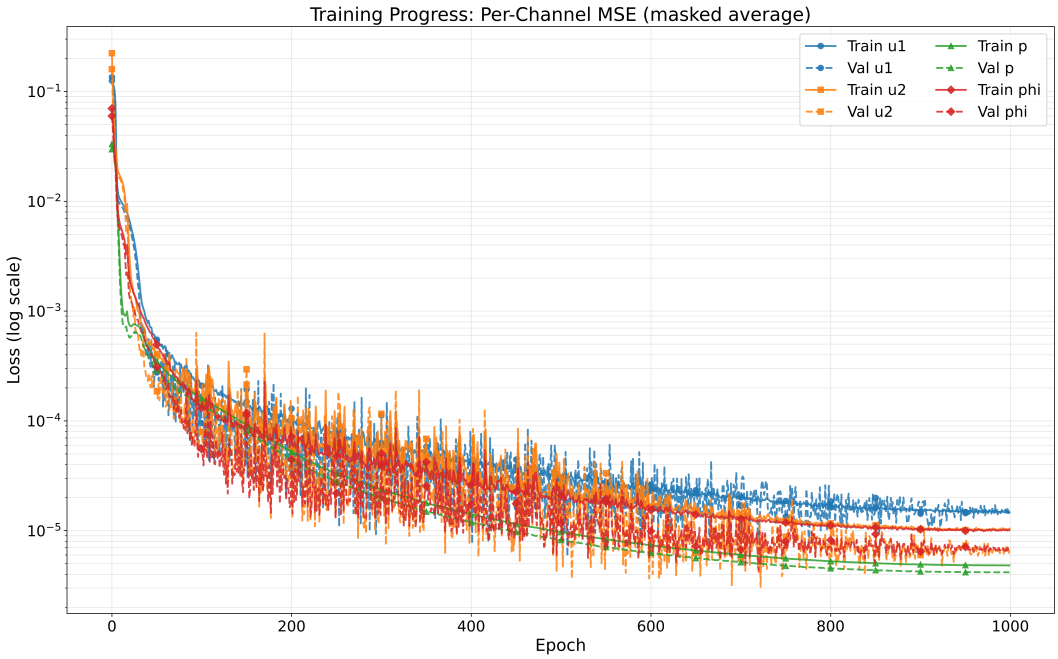

Figure 6: Training curves of total loss and component-wise losses for u_1, u_2, p, and ϕ_D.

The training process of the proposed ViT-K method is illustrated in Figure 6, which depicts the evolution of the losses for u_1, u_2, p, and ϕ_D. A rapid decrease in loss is observed during the early training epochs, indicating that the model quickly learns the dominant spatiotemporal patterns of the coupled Stokes–Darcy dynamics. As the training progresses, both total and component losses steadily decline and eventually converge to small values, demonstrating stable optimization and effective representation learning in the latent Koopman space. These results confirm the reliability and convergence of the ViT-K framework.

5.2 Example 2: Navier-Stokes-Darcy Problem

We next consider the time-dependent Navier–Stokes–Darcy (NSD) problem on the domain $\Omega = [0,1]\times[-0.25,0.75]$, where the Darcy region is $\Omega_D = [0,1]\times[0,0.75]$ and the Stokes region is $\Omega_S = [0,1]\times[-0.25,0]$. This system incorporates the nonlinear convection term in the Navier-Stokes equations. The model parameters are chosen as $\alpha = 1$, $\nu = 1$, $g = 1$, $z = 0$, and $\mathbb{K} = kI$ with $k = 1$, where I denotes the identity tensor.

The boundary conditions and source terms are specified such that the exact solutions are given by

$$\phi_D(x,y,t) = [\,2-\pi\sin(\pi x)\,]\,[\,-y+\cos(\pi(1-y))\,]\cos(2\pi t), \tag{26}$$

$$\vec{u}_S(x,y,t) = \begin{bmatrix} x^2y^2+e^{-y} \\ -\frac{2}{3}xy^3+2-\pi\sin(\pi x)\end{bmatrix}\cos(2\pi t), \tag{27}$$

$$p_S(x,y,t) = -[\,2-\pi\sin(\pi x)\,]\cos(2\pi y)\cos(2\pi t), \tag{28}$$

which satisfy the interface conditions, including the Beavers–Joseph condition.

The numerical settings and examples adopt the same configuration as Example 1, including spatial discretization, network architecture and evaluation metrics.

The key distinction in this experiment lies in the parameterization of the Kupman operator. The generator $\mathbf{A}$ is composed of block diagonal matrices from a 2×2 rotation generator, thereby enforcing

energy-conserving harmonic evolution in the latent space. Loss weights are adjusted to $w_{u_1} = 3.0, w_{u_2} = 3.0, w_p = 1.0$, and $w_\phi = 1.0$ to balance contributions across dimensions under the new dynamic paradigm. Model performance evaluation retains the previously defined metric set.

Table 2: Navier-Stokes-Darcy Model Prediction Quality Assessment

Channel	Time	MSE	MAE	Max Error	Relative Error(%)
u_1	$t = 0.0$	3.81×10^{-6}	1.5141×10^{-3}	1.0123×10^{-2}	0.2284%
	$t = 0.5$	4.40×10^{-6}	1.6278×10^{-3}	8.3809×10^{-3}	0.2460%
	$t = 1.0$	3.85×10^{-6}	1.5208×10^{-3}	1.0902×10^{-2}	0.2325%
	$t = 1.5$	4.57×10^{-6}	1.6534×10^{-3}	8.0236×10^{-3}	0.2508%
	$t = 2.0$	3.93×10^{-6}	1.5320×10^{-3}	1.0478×10^{-2}	0.2319%
u_2	$t = 0.0$	1.316×10^{-5}	3.0053×10^{-3}	1.1028×10^{-2}	0.7415%
	$t = 0.5$	1.395×10^{-5}	3.0055×10^{-3}	1.1447×10^{-2}	0.7659%
	$t = 1.0$	1.282×10^{-5}	2.9519×10^{-3}	1.0598×10^{-2}	0.7415%
	$t = 1.5$	1.396×10^{-5}	3.0107×10^{-3}	1.1452×10^{-2}	0.7662%
	$t = 2.0$	1.314×10^{-5}	3.0016×10^{-3}	1.0915×10^{-2}	0.7410%
p	$t = 0.0$	8.37×10^{-6}	2.3027×10^{-3}	1.0017×10^{-2}	0.8441%
	$t = 0.5$	8.68×10^{-6}	2.2931×10^{-3}	1.2537×10^{-2}	0.8626%
	$t = 1.0$	8.23×10^{-6}	2.2838×10^{-3}	1.0442×10^{-2}	0.8481%
	$t = 1.5$	8.74×10^{-6}	2.2939×10^{-3}	1.1767×10^{-2}	0.8653%
	$t = 2.0$	8.41×10^{-6}	2.3118×10^{-3}	1.0391×10^{-2}	0.8462%
ϕ	$t = 0.0$	2.360×10^{-5}	3.9349×10^{-3}	1.9068×10^{-2}	1.3777%
	$t = 0.5$	2.642×10^{-5}	4.1706×10^{-3}	2.1905×10^{-2}	1.4626%
	$t = 1.0$	2.298×10^{-5}	3.8779×10^{-3}	1.8884×10^{-2}	1.3773%
	$t = 1.5$	2.628×10^{-5}	4.1532×10^{-3}	2.1771×10^{-2}	1.4586%
	$t = 2.0$	2.356×10^{-5}	3.9276×10^{-3}	1.9639×10^{-2}	1.3765%

Table 2 summarizes the quantitative prediction errors for each physical variable at five representative time instances up to $t = 2.0$. The prediction errors for all physical variables are extremely small and exhibit remarkable stability throughout the entire extrapolation range $t > 1.0$.

For instance, the relative L^2 error of the velocity component u_1 consistently remained at an extremely low level between 0.23% and 0.25% from $t = 0.0$ to $t = 2.0$, showing no discernible trend of long-term error accumulation. Moreover, the error metric exhibits slight periodic fluctuations, peaking near $t = 0.5$ and $t = 1.5$. This indicates that while the model captures the system's dominant frequency, minor systematic phase or amplitude deviations may exist, most pronounced at points of maximum velocity or pressure variation. Despite this subtle behavior, the absolute magnitude of the error remains exceptionally low.

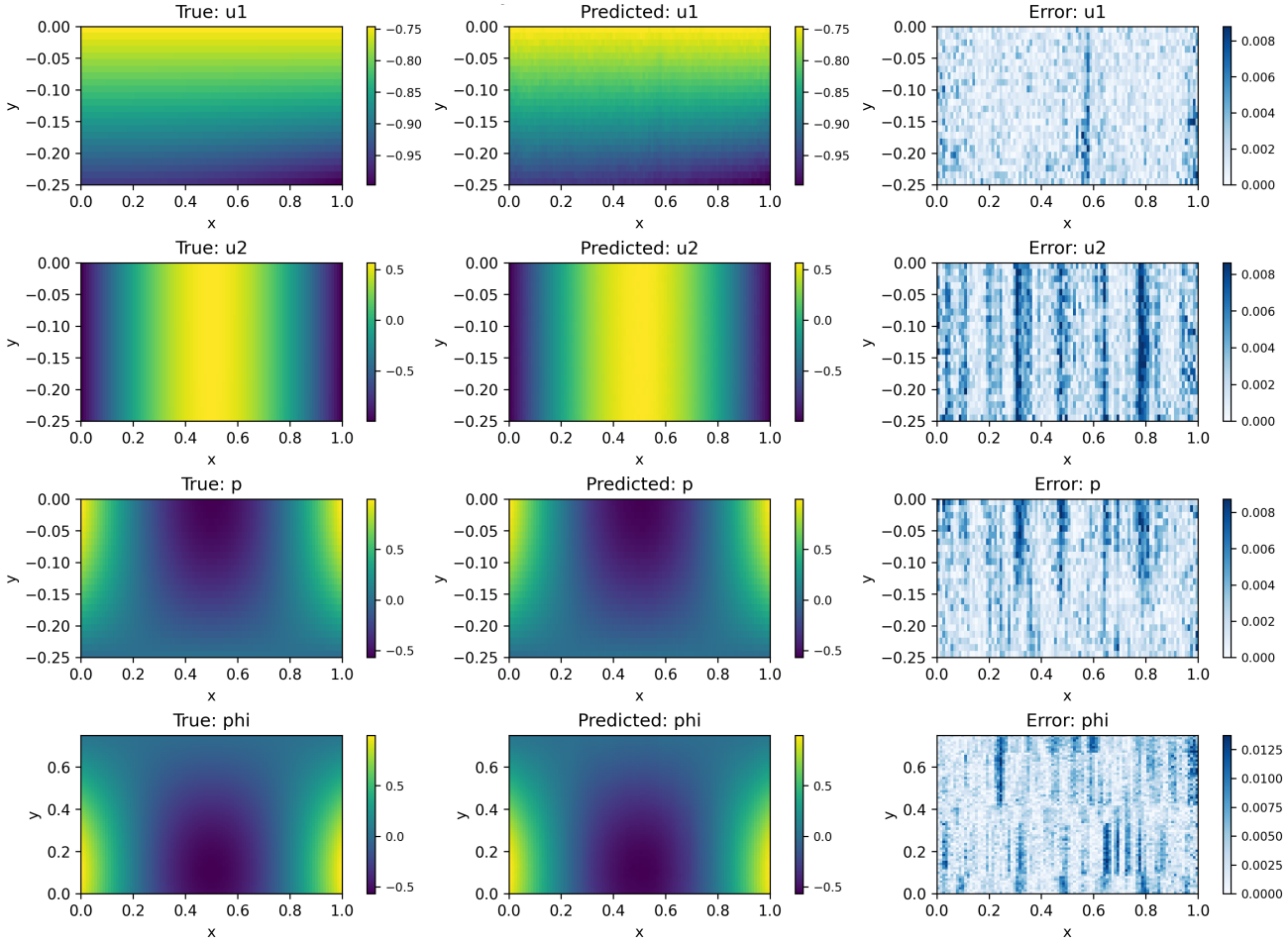

Figure 7: The exact solution (left), the approximation from VIT (middle) and the absolute error (right) at $t = 0.5$.

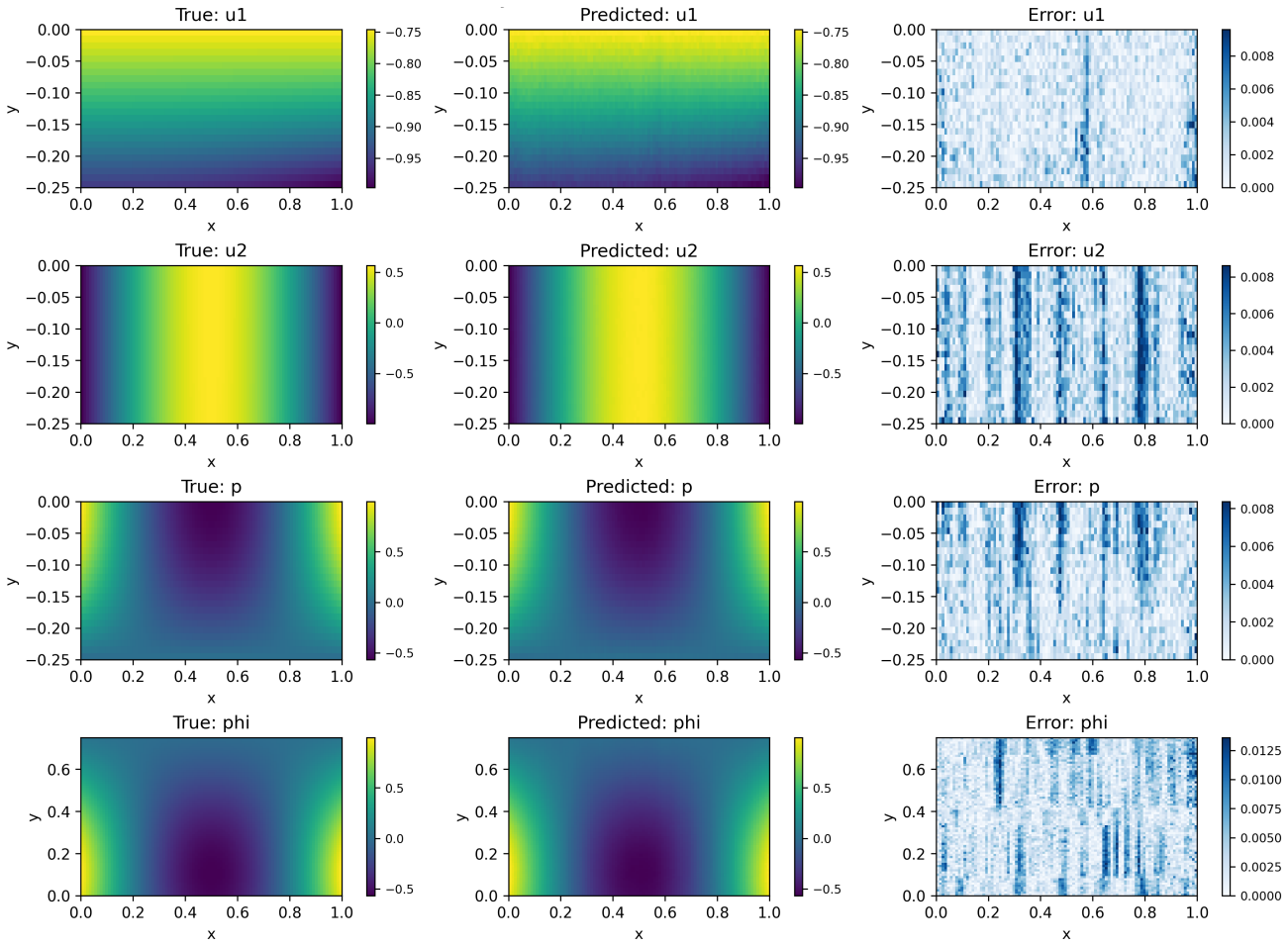

Figure 8: The exact solution (left), the approximation from VIT (middle) and the absolute error (right) at $t = 1.5$.

Figures 7 and 8 show the exact solution, the predicted field, and the absolute error at $t = 0.5$ and $t = 1.5$, respectively. The predicted results closely match the analytical solution. This indicates that the model accurately captures the coupling relationship between the Navier-Stokes domain and the Darcy domain.

The error fields exhibit faint vertical stripe patterns, which mainly originate from the patch-wise spatial encoding of the Vision Transformer. Since the input domain is divided into 16×16 patches, local discontinuities between adjacent patches can lead to structured reconstruction artifacts along the y-direction. In addition, the Fourier spatial features and the Koopman latent dynamics may introduce mild phase misalignment across neighboring patches, producing periodic banded errors. These artifacts remain small in magnitude and do not affect the overall accuracy or stability of the prediction.

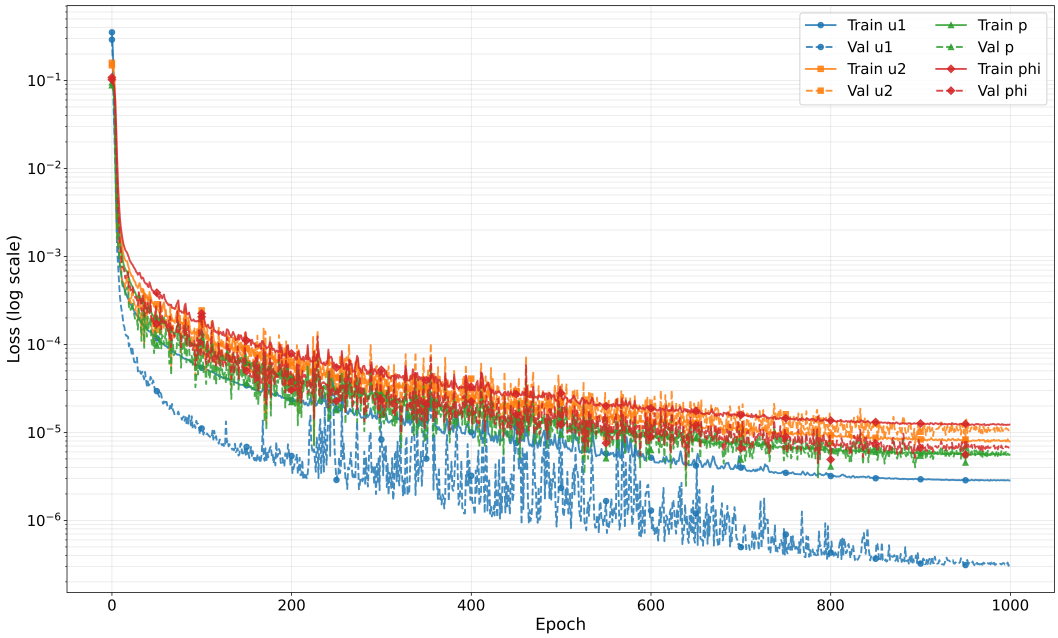

Figure 9: Training curves of total loss and component-wise losses for u_1, u_2, p, and ϕ_D.

Figure 9 illustrates the trend of the loss function. The loss value decreases rapidly during the initial training cycles, then achieves stable convergence across all channels. This behavior validates the stability and reliability of the optimization process, and confirms the effectiveness of the ViT-K model in complex coupled fluid systems.

To further investigate the physical fidelity of our model across all components of the coupled system, we conduct a spectral analysis of the predicted fields. Figure 10 presents the one-dimensional, spatially-averaged Power Spectral Densities (PSDs) for all four physical quantities (u_1, u_2, p, ϕ) at the predicted time $t = 2.0s$. The PSDs reveal how the energy of each field is distributed across the spatial wavenumbers, from large-scale structures (low k) to fine-scale features (high k).

The plots demonstrate a high degree of agreement between the predicted and actual spectral lines, particularly in the energy-dense low-wavenumber region. For all four variables, the predicted spectral lines nearly coincide with the measured values when $k \lesssim 10$. This confirms that the ViT-based encoder can accurately capture the low-frequency characteristics of the coupled system.

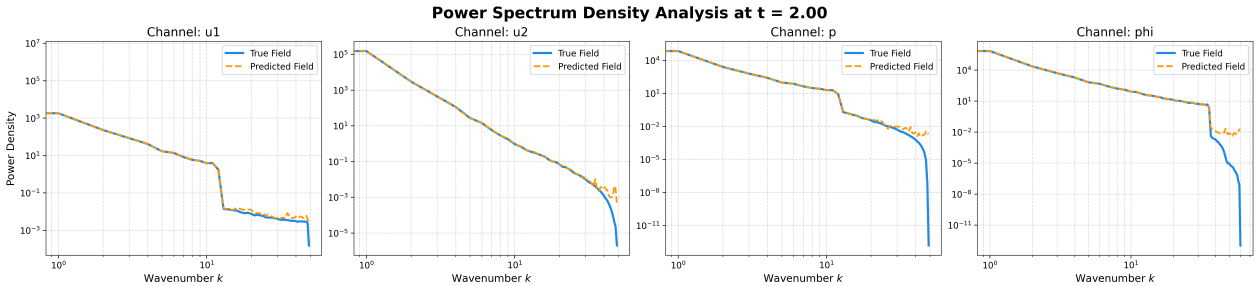

Figure 10: Power Spectral Density analysis of predicted versus true fields at $t = 2.0s$.

In the high-wavenumber region, representing fine-scale details, the model successfully captures the overall decay trend of all field quantities, avoiding the over-smoothing phenomenon common in data-driven models. For the velocity components (u_1, u_2), the matching accuracy remains remarkably close

across the entire spectral range. For the pressure (p) and potential energy (ϕ) fields, we observe a slight overestimation of energy in the highest frequency bands. This suggests the model may introduce minor high-frequency numerical artifacts in specific fields. However, this deviation is far smaller than the magnitude of the principal mode energy.

To validate the long-term stability of this framework, we conducted a long-term extrapolation experiment on the Navier-Stokes-Darcy system. Trained solely on data within the interval $t \in [0, 1.0]$, the system predicted the state at $t = 100.0s$—a prediction horizon extending to **100 times** the length of the training data.

Figure 11 plots the evolution of the relative L^2 error for each physical variable across the entire time span.

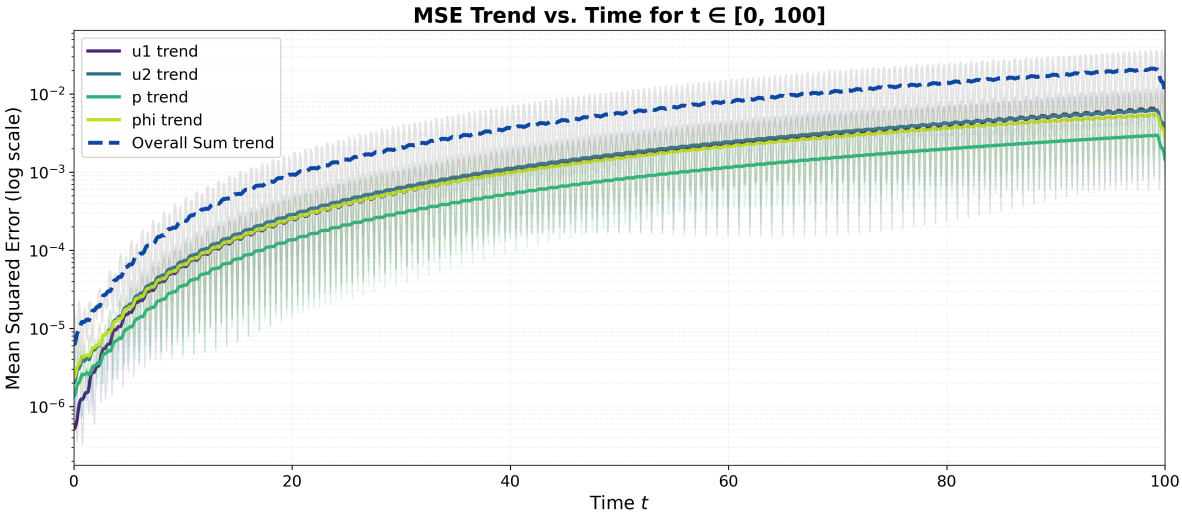

Figure 11: The exact solution (left), the approximation from VIT (middle) and the absolute error (right) at $t = 50.0$.

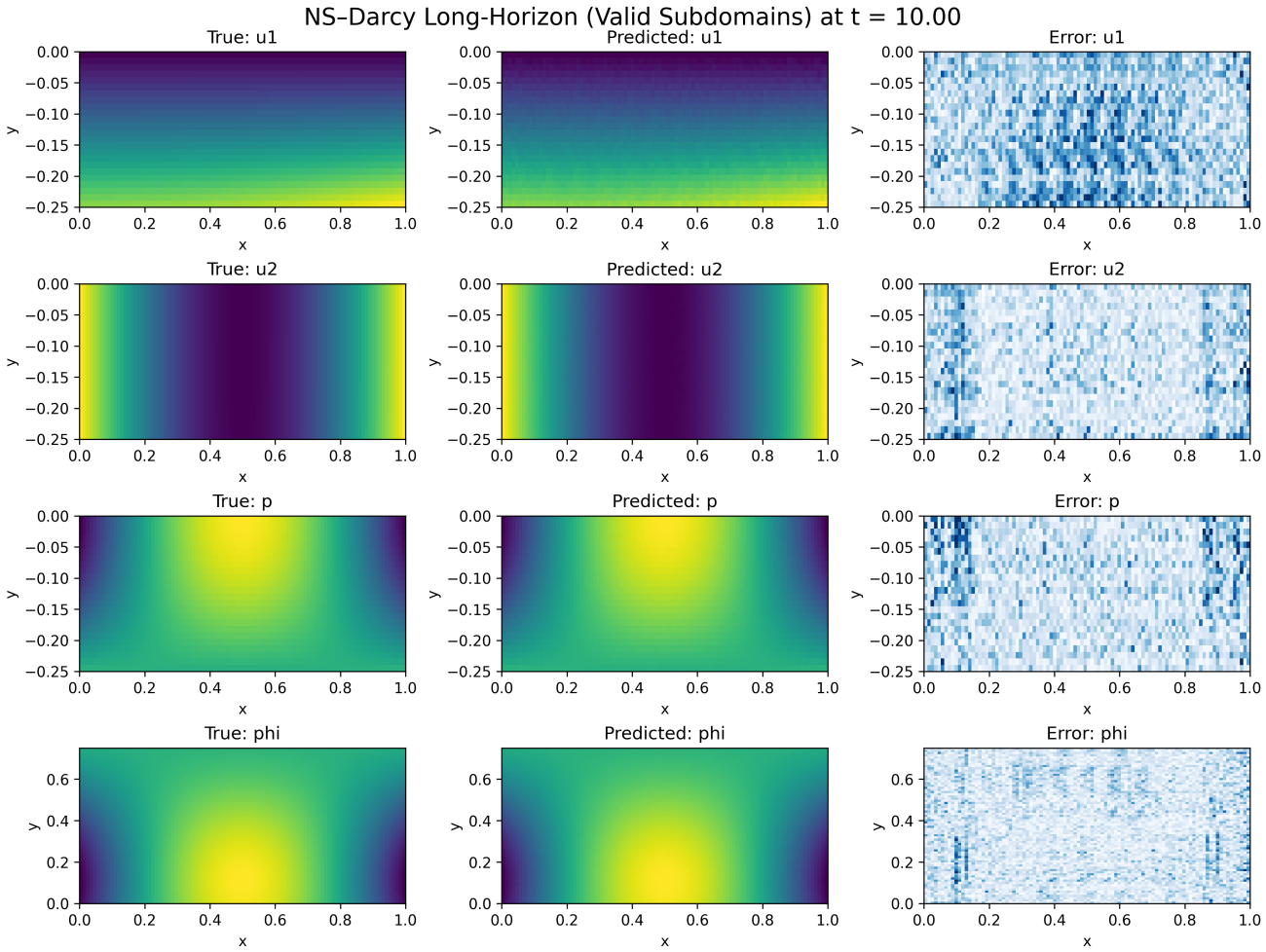

Figure 12: The exact solution (left), the approximation from VIT (middle) and the absolute error (right) at $t = 10.0$.

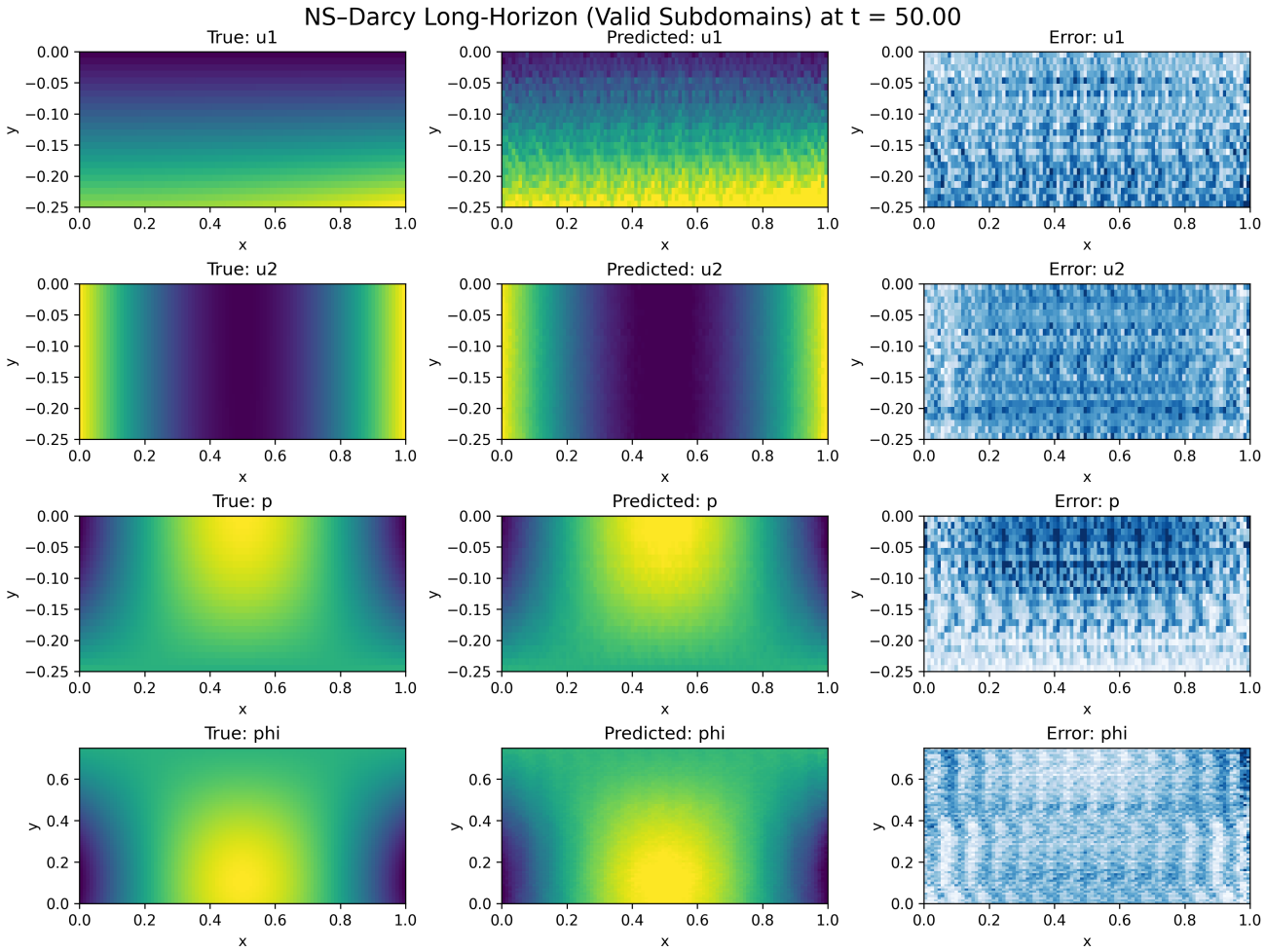

Figure 13: The exact solution (left), the approximation from VIT (middle) and the absolute error (right) at $t = 50.0$.

Figure 12 and Figure 13 respectively show the prediction results at t=10 and t=50.At t=10s, the predicted solution is still relatively close to the true solution. For the case at t=50s, despite the accurate capture of low-frequency macroscale structures in the flow field, high-frequency spurious oscillations with grid-like structures are observed.

6 Conclusion

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant (No. 12201327), Ningbo Natural Science Foundation (No. 2022J087).

Declarations

Conflict of interest

The authors declare that they have no conflict of interests.

Data availability statements

All data generated or analyzed during this study are included in this article.

References

[1] Yuanyi Li, Dekui Yuan, Binliang Lin, and Fang-Yenn Teo. A fully coupled depth-integrated model for surface water and groundwater flows. *Journal of Hydrology*, 542:172–184, 2016.

[2] Marco Discacciati. *Domain decomposition methods for the coupling of surface and groundwater flows*. PhD thesis, Verlag nicht ermittelbar, 2004.

[3] NS Hanspal, AN Waghode, V Nassehi, and RJ Wakeman. Numerical analysis of coupled stokes/darcy flows in industrial filtrations. *Transport in porous media*, 64(1):73–101, 2006.

[4] Alberto Zingaro, Christian Vergara, Luca Dede', Francesco Regazzoni, and Alfio Quarteroni. A comprehensive mathematical model for cardiac perfusion. *Scientific Reports*, 13(1):14220, 2023.

[5] A-RA Khaled and Kambiz Vafai. The role of porous media in modeling flow and heat transfer in biological tissues. *International Journal of Heat and Mass Transfer*, 46(26):4989–5003, 2003.

[6] Béatrice Rivière and Ivan Yotov. Locally conservative coupling of stokes and darcy flows. *SIAM Journal on Numerical Analysis*, 42(5):1959–1977, 2005.

[7] William J Layton, Friedhelm Schieweck, and Ivan Yotov. Coupling fluid flow with porous media flow. *SIAM Journal on Numerical Analysis*, 40(6):2195–2218, 2002.

[8] Yanzhao Cao, Max Gunzburger, Xiaolong Hu, Fei Hua, Xiaoming Wang, and Weidong Zhao. Finite element approximations for stokes–darcy flow with beavers–joseph interface conditions. *SIAM Journal on Numerical Analysis*, 47(6):4239–4256, 2010.

[9] Christine Bernardi, Tomás Chacón Rebollo, Frédéric Hecht, and Zoubida Mghazli. Mortar finite element discretization of a model coupling darcy and stokes equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(3):375–410, 2008.

[10] Vincent J Ervin, Eleanor W Jenkins, and Shuyu Sun. Coupling nonlinear stokes and darcy flow using mortar finite elements. *Applied numerical mathematics*, 61(11):1198–1222, 2011.

[11] Juan Galvis and Marcus Sarkis. Feti and bdd preconditioners for stokes–mortar–darcy systems. *Communications in Applied Mathematics and Computational Science*, 5(1):1–30, 2009.

[12] Marco Discacciati, Alfio Quarteroni, and Alberto Valli. Robin–robin domain decomposition methods for the stokes–darcy coupling. *SIAM Journal on Numerical Analysis*, 45(3):1246–1268, 2007.

[13] Yizhong Sun, Feng Shi, Haibiao Zheng, Heng Li, and Fan Wang. Two-grid domain decomposition methods for the coupled stokes–darcy system. *Computer Methods in Applied Mechanics and Engineering*, 385:114041, 2021.

[14] Wenbin Chen, Max Gunzburger, Fei Hua, and Xiaoming Wang. A parallel robin–robin domain decomposition method for the stokes–darcy system. *SIAM Journal on Numerical Analysis*, 49(3):1064–1084, 2011.

[15] Changxin Qiu, Xiaoming He, Jian Li, and Yanping Lin. A domain decomposition method for the time-dependent navier-stokes-darcy model with beavers-joseph interface condition and defective boundary condition. *Journal of Computational Physics*, 411:109400, 2020.

[16] Mo Mu and Jinchao Xu. A two-grid method of a mixed stokes–darcy model for coupling fluid flow with porous media flow. *SIAM journal on numerical analysis*, 45(5):1801–1813, 2007.

[17] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. 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, 2019.

[18] Zhiping Mao, Ameya D Jagtap, and George Em Karniadakis. Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering*, 360:112789, 2020.

[19] Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks (pinns) for fluid mechanics: A review. *Acta Mechanica Sinica*, 37(12):1727–1738, 2021.

[20] Salvatore Cuomo, Vincenzo Schiano Di Cola, Fabio Giampaolo, Gianluigi Rozza, Maziar Raissi, and Francesco Piccialli. Scientific machine learning through physics–informed neural networks: Where we are and what's next. *Journal of Scientific Computing*, 92(3):88, 2022.

[21] Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. Physics-informed neural operator for learning partial differential equations. *ACM/IMS Journal of Data Science*, 1(3):1–27, 2024.

[22] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[23] Gege Wen, Zongyi Li, Kamyar Azizzadenesheli, Anima Anandkumar, and Sally M Benson. U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 163:104180, 2022.

[24] Kai Zhang, Yuande Zuo, Hanjun Zhao, Xiaopeng Ma, Jianwei Gu, Jian Wang, Yongfei Yang, Chuanjin Yao, and Jun Yao. Fourier neural operator for solving subsurface oil/water two-phase flow partial differential equation. *Spe Journal*, 27(03):1815–1830, 2022.

[25] Byoung-Ju Choi, Hong Sung Jin, and Bataa Lkhagvasuren. Applications of the fourier neural operator in a regional ocean modeling and prediction. *Frontiers in Marine Science*, 11:1383997, 2024.

[26] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021.

[27] Pan Huang, Yifei Leng, Cheng Lian, and Honglai Liu. Porous-deeponet: Learning the solution operators of parametric reactive transport equations in porous media. *Engineering*, 39:94–103, 2024.

[28] Md Ashiqur Rahman, Zachary E Ross, and Kamyar Azizzadenesheli. U-no: U-shaped neural operators. *arXiv preprint arXiv:2204.11127*, 2022.

[29] Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in neural information processing systems*, 34:26548–26560, 2021.

[30] Phillip Lippe, Bas Veeling, Paris Perdikaris, Richard Turner, and Johannes Brandstetter. Pde-refiner: Achieving accurate long rollouts with neural pde solvers. *Advances in Neural Information Processing Systems*, 36:67398–67433, 2023.

[31] Wei Wang, Maryam Hakimzadeh, Haihui Ruan, and Somdatta Goswami. Time-marching neural operator–fe coupling: Ai-accelerated physics modeling. *Computer Methods in Applied Mechanics and Engineering*, 446:118319, 2025.

[32] Hyoeun Kang, Yongsu Kim, Thi-Thu-Huong Le, Changwoo Choi, Yoonyoung Hong, Seungdo Hong, Sim Won Chin, and Howon Kim. A new fluid flow approximation method using a vision transformer and a u-shaped convolutional neural network. *AIP Advances*, 13(2), 2023.

[33] Yuvarajendra Anjaneya Reddy, Joel Wahl, and Mikael Sjödahl. Twins-pivnet: Spatial attention-based deep learning framework for particle image velocimetry using vision transformer. *Ocean Engineering*, 318:120205, 2025.

[34] Steven L Brunton, Marko Budišić, Eurika Kaiser, and J Nathan Kutz. Modern koopman theory for dynamical systems. *arXiv preprint arXiv:2102.12086*, 2021.

[35] Gordon S Beavers and Daniel D Joseph. Boundary conditions at a naturally permeable wall. *Journal of fluid mechanics*, 30(1):197–207, 1967.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[37] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[38] Bernard O Koopman. Hamiltonian systems and transformation in hilbert space. *Proceedings of the National Academy of Sciences*, 17(5):315–318, 1931.

[39] Igor Mezić. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41(1):309–325, 2005.

[40] Matthew O. Williams, Ioannis G. Kevrekidis, and Clarence W. Rowley. A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25:1307–1346, 2015.