COMPOL: A Scalable Neural Operator Framework for Multi-Physics Simulations

Junqi Qu[†], Tao Wang[*], Yushun Dong[†], Hewei Tang[*], and Shibo Li[†]
[*]*Hildebrand Dept. of Petroleum and Geosystems Engineering*
The University of Texas at Austin, Austin, TX, USA
{tao.wang, hewei.tang}@austin.utexas.edu
[†]*Department of Computer Science*
Florida State University, Tallahassee, FL, USA
{jq24b, yushun.dong}@fsu.edu, shiboli@cs.fsu.edu

***Abstract*—Multi-physics simulations play an essential role in accurately modeling complex interactions across diverse scientific and engineering domains. Although neural operators, particularly the Fourier Neural Operator (FNO), have significantly improved computational efficiency, they often fail to capture the intricate correlations inherent in coupled physical processes. To address this limitation, we introduce COMPOL, a novel coupled multi-physics operator learning framework. COMPOL extends conventional operator architectures by incorporating a sophisticated attention-based aggregation mechanism that effectively models interdependencies among interacting physical processes within latent feature spaces. Our approach is architecture-agnostic and seamlessly integrates into various neural operator frameworks involving latent space transformations. Extensive experiments on diverse benchmarks, including biological reaction-diffusion systems, pattern-forming chemical reactions, multiphase geological flows, and thermo-hydro-mechanical processes, demonstrate that COMPOL consistently achieves superior predictive accuracy compared to state-of-the-art methods. Our code and data are available at https://github.com/AriaQJ/COMPOL.**

***Index Terms*—Operator Learning, Fourier Neural Operators, Multi-Physics Simulations**

I. Introduction

Physical simulations governed by partial differential equations (PDEs) are fundamental tools across scientific and engineering fields, including aerospace engineering, fluid mechanics, chemical processes, and environmental modeling [1, 2, 3]. These simulations leverage core physical principles, such as conservation laws and symmetries, to predict complex phenomena. Traditional numerical methods, including finite difference, finite element, and finite volume techniques, face significant computational challenges when addressing high-dimensional or coupled multi-physics problems [4, 5]. Recent advances in machine learning have introduced neural operator methods [6, 7, 8], which directly approximate solution mappings between infinite-dimensional function spaces, significantly enhancing computational efficiency. Despite their promise, existing neural operator methods often inadequately capture the intricate correlations present in coupled multi-physics systems.

Modeling multi-physics systems presents unique challenges due to the dynamic interactions between multiple distinct physical processes, each governed by its own set of PDEs. These interactions often span multiple spatial and temporal scales, creating significant computational complexity [9]. Examples include fluid-structure interactions, chemical reaction-diffusion systems, and multiphase geological flows [10, 11]. Current neural operator frameworks typically treat these processes independently or apply overly simplistic coupling strategies, resulting in insufficient representation of inter-process dynamics. Consequently, capturing nuanced interactions and accurately predicting system behaviors remains a critical unresolved challenge.

To address these challenges, we propose COMPOL, a novel coupled multi-physics operator learning framework (see Fig. 1) explicitly designed to represent complex interactions among multiple physical processes. COMPOL introduces a sophisticated attention-based mechanism [12] for latent feature aggregation, capturing detailed inter-process interactions within latent representation spaces. Notably, our framework is architecture-agnostic, capable of integration with various neural operator architectures involving latent feature transformations.

Our contributions are threefold. First, we propose a versatile, architecture-agnostic operator learning framework tailored for coupled multi-physics systems that enhances traditional operator layers with sophisticated feature aggregation. Second, we introduce an attention-based latent aggregation technique that robustly captures dynamic inter-process dependencies across an arbitrary number of physical processes. Third, we demonstrate COMPOL's effectiveness and scalability across diverse multi-physics benchmarks, including predator-prey dynamics, chemical reactions, multiphase flows, and thermo-hydro-mechanical processes, consistently achieving substantial improvements in predictive accuracy compared to state-of-the-art approaches.

II. Background and Related Work

a) Operator Learning: Operator learning [6] directly approximates solution operators of partial differential equations (PDEs) by learning mappings between infinite-dimensional function spaces. Consider a PDE defined as: $\mathfrak{L}(u)(x) = f(x), x \in \Omega; u(x) = 0, x \in \partial\Omega$ where $\mathfrak{L}$ is a differential

*Corresponding author: Shibo Li (shiboli@cs.fsu.edu).

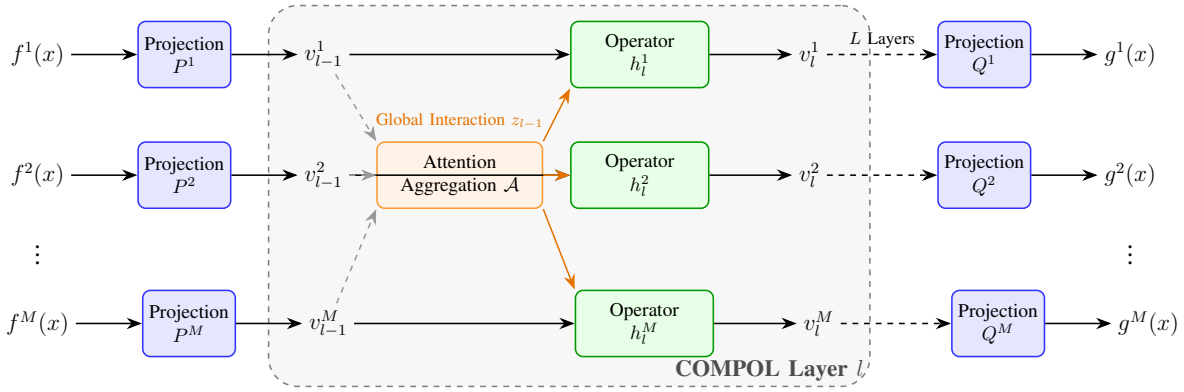

Fig. 1: Architecture of the Coupled Multi-Physics Operator Learning (COMPOL) framework. The model processes M interacting physical systems in parallel. Initial physical input functions $f^m(x)$ are mapped into a latent space via process-specific linear projections P^m. Within each of the L COMPOL layers, the latent representations v_{l-1}^m undergo transformations via independent neural operators h_l^m (e.g., FNO layers). Simultaneously, an attention-based aggregation mechanism $\mathcal{A}$ evaluates the intermediate states to compute a global interaction feature z_{l-1}. This aggregated state explicitly captures the nonlinear cross-process dependencies and is fed into the respective neural operators to produce the updated latent states v_l^m. After L iterative layers, dedicated linear projections Q^m reconstruct the latent representations back into the final physical solution spaces $g^m(x)$.

operator, u is the solution function, f is the input function, Ω is the problem domain and $\partial\Omega$ represents its boundary. Operator learning aims to approximate the inverse operator $\mathfrak{L}^{-1}$ mapping inputs f to u through a parametric operator $\psi_\theta : \mathcal{H} \to \mathcal{Y}$, minimizing the empirical risk over function pairs $\{(f_n, u_n)\}_{n=1}^N$: $\theta^* = \operatorname{argmin}_\theta \frac{1}{N}\sum_{n=1}^N \|\psi_\theta(f_n) - u_n\|_{\mathcal{Y}}^2$ where $\mathcal{H}$ and $\mathcal{Y}$ represent suitable function spaces, such as Banach or Hilbert spaces.

b) Neural Operator Architectures: Several neural operator architectures have been proposed. The Fourier Neural Operator (FNO) [8] approximates solution operators using spectral convolutions: $v(x) \leftarrow \sigma\left(Wv(x) + \mathcal{F}^{-1}(R(k)\cdot\mathcal{F}(v(x)))\right)$, where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the Fourier transform and its inverse, and $R(k)$ is a learned spectral kernel. The U-shaped FNO (UFNO) [13] extends FNO with a U-Net-style encoder-decoder for improved multi-scale feature extraction. The Graph Neural Operator (GNO) [14] combines Nyström approximation with graph neural networks for function convolution. The Deep Operator Net (DeepONet) [7] employs a dual-network architecture with branch and trunk networks, later refined into POD-DeepONet [15] using PCA bases for improved stability. Transolver [16] introduces a geometry-aware transformer-based operator for PDEs on general meshes. Recent advances have also developed mesh-agnostic approaches [17, 18] that reformulate PDE simulation as ODE-solving problems. Training neural operators often requires extensive high-quality data, motivating multi-fidelity modeling [19] and active learning approaches [20, 21].

c) Multi-Physics Neural Operators: Operator learning extends traditional surrogate modeling that focused on mapping system parameters to solutions [22, 23]. Recent work has explored neural operators for multi-physics settings [24, 25, 26], though these target general physics pre-training rather than interactions within individual coupled systems. Closely related is the Coupled Multiwavelet Neural Operator (CMWNO) [27], which models coupled PDEs using wavelet-based methods but has limitations including rigid decomposition schemes and difficulty scaling to multiple interacting processes or higher-dimensional domains. Our method builds upon flexible FNO layers, and the introduced attention-based aggregation mechanism integrates seamlessly with any latent-feature neural operator framework while naturally supporting complex interactions across multiple physical processes.

III. Method

A. Multi-Physics Simulation

We consider multi-physics systems governed by coupled partial differential equations (PDEs), which naturally emerge in diverse scientific and engineering disciplines. These systems involve multiple interacting physical processes, each described by its own PDE and interconnected through nonlinear coupling terms [28]. Such interactions substantially influence overall system behavior, creating considerable modeling and computational challenges. Formally, a coupled PDE system involving M interacting physical processes is given by:

$$\begin{aligned} \mathfrak{L}_m\left(u_1, u_2, \ldots, u_M\right)(x,t) &= f_m(x,t), \quad m = 1, \ldots, M, \\ u_m(x,t) &= 0, \quad x \in \partial\Omega_m, \end{aligned} \tag{1}$$

where $\mathfrak{L}_m$ represents differential operators, potentially including spatial and temporal derivatives and nonlinear terms capturing interactions among physical processes. Here, $u_m(x,t)$ are the unknown state variables associated with the m-th physical process, and $f_m(x,t)$ represent known source terms, external inputs, or boundary conditions.

B. Coupled Multi-Physics Operator Learning (COMPOL)

Neural operator learning has shown great promise for modeling physical phenomena governed by PDEs. However, most existing approaches assume systems are governed by a single

set of PDEs, overlooking the reality where multiple processes interact across different scales [9]. Existing techniques like feature concatenation [13] and cross-overlaying [27] attempt to model these interactions but are often insufficient for sophisticated multi-physics scenarios.

To overcome this limitation, we propose COMPOL, a coupled multi-physics neural operator $\mathcal{MPO}(\mathcal{O}_1, \cdots, \mathcal{O}_M)$ designed to capture intricate correlations among individual operators $\mathcal{O}_1, \cdots, \mathcal{O}_M$. COMPOL approximates solution operators for coupled PDE systems as mappings between function spaces: $\mathcal{G}_\theta : \mathcal{H}_1 \times \mathcal{H}_2 \times \cdots \times \mathcal{H}_M \to \mathcal{Y}_1 \times \mathcal{Y}_2 \times \cdots \times \mathcal{Y}_M$, where each $\mathcal{H}_m$ denotes the input function space and $\mathcal{Y}_m$ denotes the output solution function space for the m-th physical process. Given input functions $f^m(x)$ representing initial conditions or external inputs, we first embed each process into a shared latent space: $v_0^m(x) = P^m(f^m(x))$, where P^m are linear projection operators mapping inputs into a common latent feature space.

Subsequently, latent features undergo iterative transformations using nonlinear multi-physics neural operator layers: $v_l^m(x) = h_l^m\left(v_{l-1}^m(x), z_{l-1}(x)\right), \quad l = 1, \ldots, L$, where each layer h_l^m comprises a nonlinear transformation representing internal process dynamics, augmented by interaction-aware aggregated features $z_{l-1}(x)$. The latent representation $v_l^m(x)$ captures process-specific evolution and local dynamics independently for each physical process, while the aggregated feature $z_{l-1}(x)$ explicitly represents inter-process interactions, integrating global context across different processes. This separation is a crucial design choice, allowing the model to independently refine representations of each process while explicitly modeling nonlinear interactions among them.

The aggregated interaction features are computed using an attention-based aggregation function: $z_{l-1}(x) = \mathcal{A}(v_{l-1}^1(x), v_{l-1}^2(x), \ldots, v_{l-1}^M(x))$, ensuring coherent representation of inter-process dependencies. Finally, each process-specific solution is reconstructed via dedicated linear projection operators: $g^m(x) = Q^m\left(v_L^m(x)\right)$, yielding predicted solutions in their respective physical spaces.

C. Aggregation of Multi-Physics Operator Layers

We now present the formulation using discretized function notation. Consider a coupled physical system with M processes, each associated with discretized input functions $\mathbf{f}^1, \ldots, \mathbf{f}^M$. Each process m maps its input $\mathbf{f}^m$ to a latent representation $\mathbf{v}_0^m$ via a channel-wise linear layer $P^m : \mathbb{R}^{d_m} \to \mathbb{R}^{d_h}$, where d_m and d_h are input and latent dimensions, respectively. By stacking L neural operator layers, each layer h_l^m transforms latent representations $\mathbf{v}_{l-1}^m$ into $\mathbf{v}_l^m$. A final linear layer $Q^m : \mathbb{R}^{d_h} \to \mathbb{R}^{d_m}$ maps the latent representation $\mathbf{v}_L^m$ back to the output function $\mathbf{g}^m$. To capture inter-process interactions, we compute an aggregated latent state $\mathbf{z}_l = \mathcal{A}(\{\{\mathbf{v}_j^m\}_{m=1}^M\}_{j=0}^l)$ at each layer using an attention-based mechanism [12].

For notational convenience, let $\mathcal{V} = \{\{\mathbf{v}_l^m\}_{m=1}^M\}_{l=0}^L$ represent the complete collection of intermediate latent features across M processes and $L+1$ layers. We define $\mathbf{V}_l$ as the $M \times d_h$ matrix obtained by extracting the l-th slice along the first dimension of $\mathcal{V}$, and $\mathcal{V}_l = \{\mathbf{V}_j\}_{j=0}^l$ represents the sequence of latent functions from layer 0 through layer l.

a) Attention-Based Aggregation: We compute the global latent interaction state through scaled dot-product attention tailored for multi-physics systems. The latent features from each process are linearly projected into query (Q), key (K), and value (A) representations via process-specific transformations: $Q = \Phi_Q(\mathcal{V}_l)$, $K = \Phi_K(\mathcal{V}_l)$, and $A = \Phi_A(\mathcal{V}_l)$. The aggregation state $\mathbf{z}_l$ is derived as a weighted sum of value vectors, where each weight α_j quantifies the significance of interactions involving the j-th process:

$$z_l = \sum_{j=1}^{M} \alpha_j A_j, \quad \alpha_j = \frac{\exp\left(QK_j^\top/\sqrt{d_k}\right)}{\sum_{i=1}^{M} \exp\left(QK_i^\top/\sqrt{d_k}\right)} \tag{2}$$

with d_k representing the dimensionality of key vectors. This approach explicitly encodes process-specific interactions, dynamically identifying the most influential cross-process features at each layer.

b) Complexity Analysis: The computational complexity of COMPOL includes three components. Projection layers have complexity $O(Md_md_h)$, with M processes, input dimension d_m, and latent dimension d_h. Each FNO layer involves FFT operations with complexity $O(N \log N)$ per channel [29], where N denotes spatial discretization points, yielding total complexity $O(LMd_h \cdot N \log N)$ for L layers. The attention aggregation contributes $O(LM^2d_h)$. Overall complexity is dominated by Fourier operations in typical scenarios where $N \log N \gg M, d_h$.

c) Generalizability to Other Neural Operator Frameworks: COMPOL builds upon FNO due to its proven efficiency, but the proposed aggregation mechanism can be seamlessly integrated with other frameworks, including DeepONet [7], GNO [14], and transformer-based operators [16].

D. Training

Given training data from a coupled multi-physics system, denoted as $\{\{\mathbf{f}_n^m, \mathbf{y}_n^m\}_{n=1}^{N_m}\}_{m=1}^M$, where $\mathbf{f}_n^m$ and $\mathbf{y}_n^m$ represent the n-th input and output functions from the m-th process, we optimize our coupled multi-physics neural operator by minimizing the empirical risk: $\mathcal{L}_{\mathcal{MPO}} = \mathbb{E}_{m\sim\pi}\mathbb{E}_{f^m\sim\mu^m}||\mathcal{MPO}(f) - y^m|| \approx \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N_m}\sum_{n=1}^{N_m}||\mathbf{g}_n^m - \mathbf{y}_n^m||$ where $\mathbf{g}_n^m$ is the prediction of $\mathbf{f}_n^m$. We use gradient-based optimization to minimize $\mathcal{L}_{\mathcal{MPO}}$.

IV. Experiment

a) Benchmarks: We evaluated our framework on five coupled multi-physics PDE systems spanning diverse scientific applications. (1) The *1-D Lotka-Volterra equations* [30] model predator-prey dynamics through two coupled reaction-diffusion processes with diffusion coefficients $D_u = D_v = 0.01$ and interaction parameters $a = b = c = d = 0.01$, initialized with Gaussian random fields (length-scale $l = 0.1$) under periodic boundary conditions. (2) The *1-D Belousov-Zhabotinsky equations* [31] describe chemical oscillations with

Method	N_{train}	Lotka-Volterra	Belousov-Zhab.	Gray-Scott	Multiphase	THM
FNO_c	512	0.3251±0.0507	0.0058±1.4e-4	0.0056±1.0e-4	0.0232±5.0e-4	0.1826±0.0088
	1024	0.1499±0.0097	0.0022±4.4e-5	0.0042±7.8e-5	0.0684±0.1071	0.1577±0.0073
CMWNO	512	0.3989±0.0422	0.0375±0.0015	–	–	–
	1024	0.3876±0.0357	0.0336±0.0019	–	–	–
UFNO_c	512	0.2519±0.0518	0.0239±7.7e-4	0.0083±3.0e-4	0.0241±4.0e-4	0.1824±0.0094
	1024	0.1677±0.0120	0.0122±5.1e-4	0.0044±9.8e-5	0.0137±0.0010	0.1496±0.0086
Transolver_c	512	0.2606±0.0515	0.0329±0.0089	0.0251±9.0e-4	0.0503±0.0311	0.2176±0.0123
	1024	0.1262±0.0348	0.0151±0.0023	0.0174±9.0e-4	0.0185±0.0037	0.1916±0.0126
DeepONet_c	512	0.4024±0.0143	0.6491±0.0199	0.7655±0.0039	0.1123±0.0051	0.2867±0.0199
	1024	0.3005±0.0285	0.5810±0.0352	0.6700±0.0240	0.0766±0.0049	0.2388±0.0125
COMPOL	512	**0.0918**±0.0042	**0.0016**±2.7e-5	**0.0050**±0.0016	**0.0182**±0.0011	**0.1647**±0.0092
	1024	**0.0510**±0.0048	**0.0008**±2.6e-5	**0.0035**±7.2e-5	**0.0110**±9.0e-4	**0.1442**±0.0066

TABLE I: Comparison of relative L_2 prediction errors (mean ± standard deviation over 5 runs), defined as $\frac{\|\hat{u}-u\|_2}{\|u\|_2}$, where $\hat{u}$ and u denote the predicted and ground-truth solutions, respectively for COMPOL against baseline neural operators (FNO, UFNO, Transolver, DeepONet, CMWNO) across five multi-physics benchmark datasets, evaluated with 512 and 1024 training samples. Bold values indicate the best performance for each case.

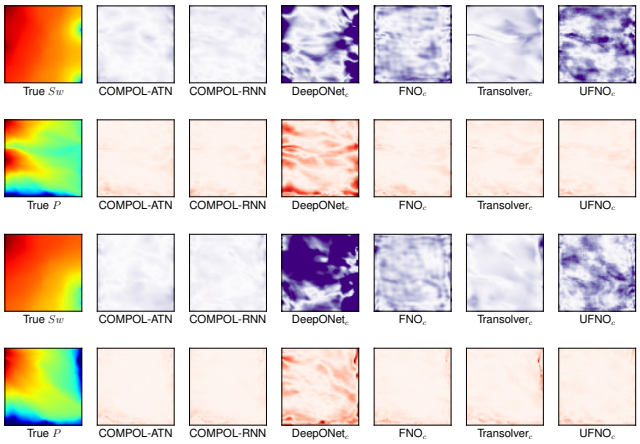

Fig. 2: Element-wise solution error of coupled multiphase flow. The left most column is the ground-truth phase pressure P and saturation S_w. The other columns are the absolute errors of each method. The lighter the color, the smaller the error.

three reactants (u, v, w) governed by nonlinear coupling terms including uv and u^2, using diffusion coefficients $\epsilon_1 = \epsilon_2 = 10^{-2}$ and $\epsilon_3 = 5 \times 10^{-3}$, simulated from $t = 0$ to $t = 0.5$. (3) The *2-D Gray-Scott equations* [32] model pattern formation in chemical reactions with activator-inhibitor dynamics, using $D_u = 0.12$, $D_v = 0.06$, feeding rate $F = 0.054$, and removal rate $k = 0.063$, evolved to $T = 20$. (4) The *2-D multiphase flow problem* [33] simulates oil-water two-phase flow in porous media with permeability ranging from 1mD to 1000mD sampled from fractal distributions, tracking phase pressure and saturation over 7.5×10^6 seconds using GEOS simulator. (5) The *2-D Thermo-Hydro-Mechanical (THM) problem* [34] couples five processes (pore pressure, temperature, and three strain components) governed by momentum balance, fluid mass balance, and heat transfer equations, with permeability fields generated using fractal algorithms ($k \in [10, 200]$ D). Training datasets were generated using numerical solvers with 256-point meshes for 1-D cases and 64×64 meshes for 2-D cases. We trained models on two dataset sizes (512 and 1024 samples) and evaluated on 200 independent test examples. Coefficients and boundary conditions remained constant to isolate effects of varying initial conditions. The primary objective was to assess our framework's capability to map initial conditions ($t = 0$) to final solution fields ($t = T$), demonstrating its ability to model complex multi-physics interactions.

b) Competing Methods: We compare COMPOL against state-of-the-art neural operators: FNO [8], UFNO [13], Transolver [16], DeepONet [7], and CMWNO [27], each using their official implementations. Except for CMWNO, these baselines are not inherently designed for multi-physics modeling; we adapt them by concatenating process inputs and outputs across channels (denoted with subscript 'c', e.g., FNO_c). All models were implemented in PyTorch [35] and trained with the Adam optimizer (learning rate = 0.001). CMWNO employed a step scheduler following its original implementation, while all other models used cosine annealing [36]. Models were trained for 500 epochs, sufficient for convergence except DeepONet which required 10,000 epochs. Experiments were conducted on NVIDIA A100 GPUs with 5-fold cross-validation, and performance was evaluated using average relative L_2 error with standard deviations. CMWNO's wavelet-based coupling relies on a fixed multiwavelet decomposition scheme that is implemented only for 1-D domains in its released codebase; extending it to 2-D grids is non-

Method	Modes	Width	Layers	Params (M)	BZ	LV
FNO_c (default)	16	64	4	0.58	0.0022±4.4e-5	0.1499±0.0097
FNO_c	96	64	4	3.20	0.0018±4.9e-5	0.1183±0.0106
FNO_c	64	64	6	3.23	0.0020±4.4e-5	0.1303±0.0152
FNO_c	48	64	8	3.25	0.0032±1.0e-4	0.1488±0.0160
FNO_c	16	224	2	3.62	0.0014±2.6e-5	0.1757±0.0102
FNO_c	16	128	6	3.48	0.0018±3.8e-5	0.1761±0.0099
FNO_c	16	160	4	3.64	0.0016±4.0e-5	0.1698±0.0090
COMPOL	16	64	4	2.10 / 1.49	**0.0008**±2.6e-5	**0.0510**±0.0048

TABLE II: Comparison of relative L_2 errors for parameter-matched FNO variants versus COMPOL on Belousov–Zhabotinsky (BZ) and Lotka–Volterra (LV) with 1024 training samples. COMPOL's parameter counts are reported as BZ / LV. Despite having fewer parameters, COMPOL outperforms all FNO variants, demonstrating that its advantage stems from explicit cross-process coupling rather than increased capacity.

Method	$N_{\text{train}} = 512$	$N_{\text{train}} = 1024$
FNO	0.0715±0.0033	0.0422±0.0037
CMWNO	0.4441±0.0134	0.4509±0.0168
UFNO	0.1043±0.0061	0.0565±0.0012
Transolver	0.1264±0.0733	0.4066±0.4248
DeepONet	0.4007±0.0216	0.3749±0.0181
COMPOL	**0.0200**±0.0021	**0.0106**±0.0006

TABLE III: Comparison of prediction accuracy (relative L_2 error) for the single-process Burgers' equation trained with 512 and 1024 examples. Results averaged over 5 runs, demonstrating that COMPOL's attention-based aggregation improves performance even in single-process scenarios.

Method	BZ (u, v, w)			GS (u, v)	
	u	v	w	u	v
CMWNO	0.6517	0.7629	0.5010	–	–
DeepONet	2.0934	3.4638	2.7937	31.14	30.57
FNO	0.1475	0.0930	0.1182	0.4024	0.2256
Transolver	0.3899	0.4728	0.2775	0.4859	0.3413
UFNO	0.2210	0.1786	**0.1072**	0.4056	0.2734
COMPOL	**0.1403**	**0.0672**	0.1091	**0.4037**	**0.2147**

TABLE IV: MAE $= \frac{1}{K}\sum_{k=1}^{K} |E_{\hat{u}}(k) - E_u(k)|$ of spectral energy distributions for Belousov–Zhabotinsky (BZ) and Gray–Scott (GS) systems. Lower values indicate better agreement with ground-truth spectra; best per column in **bold**.

trivial and beyond the scope of this work. We therefore report CMWNO results only on the two 1-D benchmarks.

A. Analysis of Predictive Performance

The experimental results in Table I present the predictive performance of COMPOL compared to baseline neural operators (FNO, UFNO, Transolver, DeepONet, and CMWNO), evaluated for both 512 and 1024 training samples. For 512 training samples, COMPOL consistently achieves the lowest prediction errors, outperforming baseline methods with improvements up to 72.41%. When increasing the training set size to 1024 samples, COMPOL maintains superior performance, achieving improvements up to 63.64% compared to baselines, notably excelling on the Lotka-Volterra and Belousov-Zhabotinsky datasets. Baseline neural operators adapted for multi-physics tasks exhibit significantly higher errors at both training scales. CMWNO, despite being explicitly designed for multi-physics modeling, consistently underperforms relative to COMPOL across all benchmarks. These results affirm the superior effectiveness and scalability of COMPOL in multi-physics modeling scenarios. Fig. 2 visualizes the element-wise absolute errors for the multiphase flow benchmark. COMPOL produces substantially lower errors across both phase pressure and saturation fields, while baselines such as DeepONet exhibit large localized errors.

B. Ablation Study of Coupling Mechanisms

a) Capacity vs. Coupling: Parameter-Matched FNO: COMPOL instantiates a separate neural operator per physical process and aggregates cross-process information in latent space; consequently, its parameter count scales roughly linearly with the number of processes M relative to a single FNO with naïve channel concatenation. To test whether our gains are merely due to increased capacity, we scale FNO to match COMPOL's parameter count by increasing Fourier modes, network width, and depth. On Belousov–Zhabotinsky (BZ; 3 processes) and Lotka–Volterra (LV; 2 processes) with 1024 samples, Table II reports relative L_2 errors. While parameter-matched FNO improves over its base configuration, further widening or deepening yields diminishing returns and does not close the gap to COMPOL, indicating that the advantage primarily stems from explicit cross-process coupling rather than model size.

b) Improvement over Single-Process Simulations: We further examine whether COMPOL's latent aggregation mechanisms can enhance predictive performance for single-process simulations. Specifically, we evaluate COMPOL on the single-

process Burgers' equation using training sets of 512 and 1024 examples, comparing results against baseline methods (Table III). Results show that COMPOL significantly improves predictive accuracy, even in single-process scenarios, by effectively leveraging advanced latent aggregation techniques (attention-based and recurrent methods). These mechanisms enrich latent feature representations, implicitly regularize learning, adaptively select relevant features, and better capture underlying temporal dynamics. Consequently, COMPOL demonstrates improved optimization stability, superior generalization, and increased overall accuracy.

c) Predictive Spectrum Analysis.: Energy spectra offer a physics-informed diagnostic of scale-wise fidelity in the Fourier domain. For each state variable, we compute the mean absolute error (MAE) between predicted and ground-truth spectral energy distributions (lower is better). As summarized in Table IV, COMPOL matches the ground-truth spectra most closely across both Belousov–Zhabotinsky (BZ; u, v, w) and Gray–Scott (GS; u, v) systems. Relative to FNO, COMPOL-ATN reduces spectral MAE on BZ by $\sim$5% for u and $\sim$28% for v, while COMPOL-RNN yields the best w with an $\sim$11% drop. On GS, COMPOL-RNN improves over FNO by $\sim$2.5% for u and $\sim$7.7% for v, whereas other baselines lag substantially (e.g., Transolver and DeepONet). These consistent gains indicate that coupling mechanisms in COMPOL help match the ground-truth spectral content across variables and systems.

V. Conclusion

We have introduced COMPOL, a novel coupled multi-physics neural operator learning framework that extends Fourier neural operators to effectively model interactions between multiple physical processes in complex systems. Our approach employs an attention-based feature aggregation mechanism to capture rich interdependencies among physical processes within the latent space, enabling explicit modeling of cross-process dynamics. Extensive experiments across diverse benchmarks, including reaction-diffusion systems, pattern-forming chemical reactions, multiphase geological flows, and thermo-hydro-mechanical processes, demonstrate that COMPOL achieves significant improvements in predictive accuracy compared to state-of-the-art methods. Ablation studies further confirm that these gains stem from the explicit cross-process coupling mechanism rather than increased model capacity. These results establish COMPOL as an effective and scalable framework for learning the complex dynamics of coupled multi-physics systems.

While COMPOL demonstrates strong performance across the benchmarks studied, several limitations remain. First, all experiments use fixed structured grids, leaving COMPOL's ability to generalize across spatial resolutions unexplored; integrating resolution-invariant architectures is a natural extension. Second, the current evaluation focuses on single-step predictions from initial to final states; assessing stability under long-term autoregressive rollouts, where error accumulation is a known challenge, is an important direction for future work.

VI. Support of Open Science

To support reproducibility, our complete codebase, including the COMPOL implementation, baseline comparisons, and all data generation scripts, is publicly available at https://github.com/AriaQJ/COMPOL.

Acknowledgment

This work was supported in part by startup funding from the Department of Computer Science at Florida State University and the Cockrell School of Engineering at the University of Texas at Austin. Portions of this manuscript were revised with the assistance of Claude (Anthropic, https://www.anthropic.com) for language editing. and polishing.

References

[1] L. C. Evans, *Partial differential equations*. American Mathematical Society, 2022, vol. 19.

[2] A. J. Chorin, J. E. Marsden, and J. E. Marsden, *A mathematical introduction to fluid mechanics*. Springer, 1990, vol. 3.

[3] T. L. Bergman, *Fundamentals of heat and mass transfer*. John Wiley & Sons, 2011.

[4] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical mathematics*. Springer Science & Business Media, 2010, vol. 37.

[5] T. J. Hughes, *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation, 2003.

[6] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Neural operator: Learning maps between function spaces with applications to pdes," *Journal of Machine Learning Research*, vol. 24, no. 89, pp. 1–97, 2023.

[7] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, "Learning nonlinear operators via deeponet based on the universal approximation theorem of operators," *Nature machine intelligence*, vol. 3, no. 3, pp. 218–229, 2021.

[8] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Fourier neural operator for parametric partial differential equations," *arXiv preprint arXiv:2010.08895*, 2020.

[9] D. E. Keyes, L. C. McInnes, C. Woodward, W. Gropp, E. Myra, M. Pernice, J. Bell, J. Brown, A. Clo, J. Connors *et al.*, "Multiphysics simulations: Challenges and opportunities," *The International Journal of High Performance Computing Applications*, vol. 27, no. 1, pp. 4–83, 2013.

[10] Y. Bazilevs, K. Takizawa, and T. E. Tezduyar, *Computational fluid-structure interaction: methods and applications*. John Wiley & Sons, 2013.

[11] M. Cross and H. Greenside, *Pattern formation and dynamics in nonequilibrium systems*. Cambridge University Press, 2009.

[12] A. Vaswani, "Attention is all you need," *Advances in Neural Information Processing Systems*, 2017.

[13] G. Wen, Z. Li, K. Azizzadenesheli, A. Anandkumar, and S. M. Benson, "U-fno—an enhanced fourier neural

operator-based deep-learning model for multiphase flow," *Advances in Water Resources*, vol. 163, p. 104180, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0309170822000562

[14] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Neural operator: Graph kernel network for partial differential equations," *arXiv preprint arXiv:2003.03485*, 2020.

[15] L. Lu, X. Meng, S. Cai, Z. Mao, S. Goswami, Z. Zhang, and G. E. Karniadakis, "A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data," *Computer Methods in Applied Mechanics and Engineering*, vol. 393, p. 114778, 2022.

[16] H. Wu, H. Luo, H. Wang, J. Wang, and M. Long, "Transolver: A fast transformer solver for pdes on general geometries," *arXiv preprint arXiv:2402.02366*, 2024.

[17] Y. Yin, M. Kirchmeyer, J.-Y. Franceschi, A. Rakotomamonjy, and P. Gallinari, "Continuous pde dynamics forecasting with implicit neural representations," *arXiv preprint arXiv:2209.14855*, 2022.

[18] O. Boussif, Y. Bengio, L. Benabbou, and D. Assouline, "Magnet: Mesh agnostic neural pde solver," *Advances in Neural Information Processing Systems*, vol. 35, pp. 31 972–31 985, 2022.

[19] H. Tang, Q. Kong, and J. P. Morris, "Multi-fidelity fourier neural operator for fast modeling of large-scale geological carbon storage," *Journal of Hydrology*, vol. 629, p. 130641, 2024.

[20] S. Li, Z. Wang, R. Kirby, and S. Zhe, "Deep multi-fidelity active learning of high-dimensional outputs," in *The 25th International Conference on Artificial Intelligence and Statistics*, 2022.

[21] S. Li, X. Yu, W. Xing, M. Kirby, A. Narayan, and S. Zhe, "Multi-resolution active learning of fourier neural operators," in *The 27th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2024.

[22] D. Higdon, J. Gattiker, B. Williams, and M. Rightley, "Computer model calibration using high-dimensional output," *Journal of the American Statistical Association*, vol. 103, no. 482, pp. 570–583, 2008.

[23] S. Zhe, W. Xing, and R. M. Kirby, "Scalable high-order gaussian process regression," in *The 22nd International Conference on Artificial Intelligence and Statistics*, 2019, pp. 2611–2620.

[24] M. McCabe, B. R.-S. Blancard, L. H. Parker, R. Ohana, M. Cranmer, A. Bietti, M. Eickenberg, S. Golkar, G. Krawezik, F. Lanusse *et al.*, "Multiple physics pretraining for physical surrogate models," *arXiv preprint arXiv:2310.02994*, 2023.

[25] M. A. Rahman, R. J. George, M. Elleithy, D. Leibovici, Z. Li, B. Bonev, C. White, J. Berner, R. A. Yeh, J. Kossaifi *et al.*, "Pretraining codomain attention neural operators for solving multiphysics pdes," *arXiv preprint arXiv:2403.12553*, 2024.

[26] Z. Hao, C. Su, S. Liu, J. Berner, C. Ying, H. Su, A. Anandkumar, J. Song, and J. Zhu, "Dpot: Auto-regressive denoising operator transformer for large-scale pde pre-training," *arXiv preprint arXiv:2403.03542*, 2024.

[27] X. Xiao, D. Cao, R. Yang, G. Gupta, G. Liu, C. Yin, R. Balan, and P. Bogdan, "Coupled multiwavelet neural operator learning for coupled partial differential equations," *arXiv preprint arXiv:2303.02304*, 2023.

[28] E. Weinan and B. Engquist, "Multiscale modeling and computation," *Notices of the AMS*, vol. 50, no. 9, pp. 1062–1070, 2003.

[29] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Mathematics of computation*, vol. 19, no. 90, pp. 297–301, 1965.

[30] J. D. Murray, *Mathematical biology: I. An introduction*. Springer Science & Business Media, 2007, vol. 17.

[31] V. Petrov, V. Gaspar, J. Masere, and K. Showalter, "Controlling chaos in the belousov—zhabotinsky reaction," *Nature*, vol. 361, no. 6409, pp. 240–243, 1993.

[32] J. E. Pearson, "Complex patterns in a simple system," *Science*, vol. 261, no. 5118, pp. 189–192, 1993.

[33] J. Bear and A. H.-D. Cheng, *Modeling groundwater flow and contaminant transport*. Springer, 2010, vol. 23.

[34] Q. Gao and A. Ghassemi, "Three-dimensional thermo-poroelastic modeling and analysis of flow, heat transport and deformation in fractured rock with applications to a lab-scale geothermal system," *Rock Mechanics and Rock Engineering*, vol. 53, pp. 1565–1586, 2020.

[35] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[36] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.