Supplementary Materials

1 Proof of Lemma 3.1

Lemma 3.1. In the proposed nonlinear coregionalization model, the marginal distribution of the output matrix $\mathbf{Y}$ is

$$p(\mathbf{Y}|\mathbf{X},\mathbf{B}) = \mathcal{N}\big(\mathrm{vec}(\mathbf{Y})|\mathbf{0}, (\mathbf{B}^\top\mathbf{K}_{BB}\mathbf{B})\otimes\mathbf{K}+\eta^{-1}\mathbf{I}\big).$$

Given two arbitrary outputs $y_m(\mathbf{x}_i)$ and $y_t(\mathbf{x}_j)$, *i.e.,* the m-th output for input $\mathbf{x}_i$ and t-th output for input $\mathbf{x}_j$, we have $\mathrm{cov}\big(y_m(\mathbf{x}_i), y_t(\mathbf{x}_j)\big) = k(\mathbf{x}_i,\mathbf{x}_j)\tilde{b}_m^\top\mathbf{K}_{BB}\tilde{b}_t + \eta^{-1}\cdot\mathbb{1}(\mathbf{x}_i=\mathbf{x}_j, m=t)$, where $\tilde{b}_m$ and $\tilde{b}_t$ are the m-th and t-th column of $\mathbf{B}$, respectively, and $\mathbb{1}(\cdot)$ is the indicator function.

Proof. First, from the likelihood $p(\mathbf{Y}|\mathbf{W},\mathbf{B}) = \mathcal{N}\big(\mathrm{vec}(\mathbf{Y})|\mathrm{vec}(\mathbf{WB}),\eta^{-1}\mathbf{I}\big)$, we can obtain that $\mathrm{vec}(\mathbf{Y}) = \mathrm{vec}(\mathbf{WB})+\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}\sim\mathcal{N}(\boldsymbol{\epsilon}|\mathbf{0},\eta^{-1}\mathbf{I})$. Using the property of vectorization (Minka, 2000), we can derive that $\mathrm{vec}(\mathbf{WB}) = (\mathbf{B}^\top\otimes\mathbf{I})\mathrm{vec}(\mathbf{W})$, and hence

$$\mathrm{cov}\big(\mathrm{vec}(\mathbf{WB})\big) = (\mathbf{B}^\top\otimes\mathbf{I})\mathrm{cov}\big(\mathrm{vec}(\mathbf{W})\big)(\mathbf{B}\otimes\mathbf{I}).$$

Since $p(\mathbf{W}|\mathbf{X},\mathbf{B}) = \mathcal{MN}(\mathbf{W}|\mathbf{0},\mathbf{K},\mathbf{K}_{BB}) = \mathcal{N}\big(\mathrm{vec}(\mathbf{W})|\mathbf{0},\mathbf{K}_{BB}\otimes\mathbf{K}\big)$, we have $\mathrm{cov}(\mathrm{vec}(\mathbf{W})) = \mathbf{K}_{BB}\otimes\mathbf{K}$. Therefore

$$\mathrm{cov}\big(\mathrm{vec}(\mathbf{WB})\big) = (\mathbf{B}^\top\otimes\mathbf{I})(\mathbf{K}_{BB}\otimes\mathbf{K})(\mathbf{B}\otimes\mathbf{I}) = (\mathbf{B}^\top\mathbf{K}_{BB}\mathbf{B})\otimes\mathbf{K}$$

and

$$\mathrm{cov}\big(\mathrm{vec}(\mathbf{Y})\big) = (\mathbf{B}^\top\mathbf{K}_{BB}\mathbf{B})\otimes\mathbf{K}+\eta^{-1}\mathbf{I}.$$

Finally, since $\mathrm{vec}(\mathbf{Y})$ is an affine transformation of $\mathrm{vec}(\mathbf{W})$ plus an independent Gaussian noise, it must follow a multivariate Gaussian distribution. Obviously, $\mathbb{E}\big(\mathrm{vec}(\mathbf{Y})\big) = \mathbf{0}$. Finally, we have

$$p(\mathbf{Y}|\mathbf{X},\mathbf{B}) = \mathcal{N}\big(\mathrm{vec}(\mathbf{Y})|\mathbf{0}, (\mathbf{B}^\top\mathbf{K}_{BB}\mathbf{B})\otimes\mathbf{K}+\eta^{-1}\mathbf{I}\big).$$

□

2 Prediction

Given a new input $\mathbf{x}^*$, we aim to obtain the predictive distribution of the d dimensional output $\mathbf{y}^*$ at the highest fidelity,

$$p(\mathbf{y}^*|\mathbf{x}^*,\{\mathbf{X}^{(i)},\mathbf{Y}^{(i)}\}_{i=1}^F) = \int p(\mathbf{y}^*|\mathbf{w}_*^{(F)},\mathbf{B}^{(F)},\{\eta_j\}_{j=1}^F)$$
$$\cdot\prod_{i=1}^F p(\mathbf{w}_*^{(i)}|\widehat{\mathbf{X}}^{(i)},\mathbf{w}_*^{(i-1)},\mathbf{x}^*,\mathbf{W}^{(i)},\mathbf{B}^{(i)})q(\mathbf{W}^{(i)})\mathrm{d}\{\mathbf{W}^{(i)},\mathbf{w}_*^{(i)}\}_{i=1}^F,$$

where $\mathbf{w}_*^{(i)}$ are the corresponding basis weight functions of $\mathbf{x}^*$ in the i-th fidelity,

$$p(\mathbf{w}_*^{(i)}|\widehat{\mathbf{X}}^{(i)},\mathbf{w}_*^{(i-1)},\mathbf{x}^*,\mathbf{W}^{(i)},\mathbf{B}^{(i)}) = \mathcal{MN}(\mathbf{w}_*^{(i)}|\mathbf{k}_{*n}^{(i)}\mathbf{K}^{(i)^{-1}}\mathbf{W}^{(i)},\mathbf{k}_{**}^{(i)}-\mathbf{k}_{*n}^{(i)}\mathbf{K}^{(i)^{-1}}\mathbf{k}_{n*}^{(i)},\mathbf{K}_{BB}^{(i)})$$

is a conditional matrix Gaussian distribution, $\mathbf{k}_{*n}^{(i)}$ is the cross covariance (or kernel) between $[\mathbf{x}^{*^\top},\mathbf{w}_*^{(i-1)^\top}]$ and $\widehat{\mathbf{X}}^{(i)}$, and $\mathbf{k}_{n*}^{(i)} = \mathbf{k}_{*n}^{(i)^\top}$. The exact predictive distribution is not analytical. Therefore, we can generate a collection of IID samples for $\mathbf{y}^*$, and compute the empirical distribution as an approximation. To generate one sample, we start from the first fidelity, and recursively sample each $\mathbf{W}^{(i)}$ from $q(\mathbf{W}^{(i)})$, and $\mathbf{w}_*^{(i)}$ from $p(\mathbf{w}_*^{(i)}|\widehat{\mathbf{X}}^{(i)},\mathbf{w}_*^{(i-1)},\mathbf{x}^*,\mathbf{W}^{(i)},\mathbf{B}^{(i)})$ until the highest fidelity is arrived. Finally, we sample $\mathbf{y}_*$ from $p(\mathbf{y}_*|\mathbf{w}_*^{(F)},\mathbf{B}^{(F)},\{\eta_j\}_{j=1}^F)$.

3 Training with Non-Overlapping Inputs

Although MFHoGP is developed for physical simulations where the inputs of the training examples are nested across fidelities, our learning algorithm can be trivially adjusted for non-overlapping inputs. Specifically, in each fidelity $i>1$, when $\mathbf{X}^{(i)}$ are no longer included in $\mathbf{X}^{(i-1)}$, to obtain the parameterized sample $\widetilde{\mathbf{W}}^{(i-1)}(\mathbf{X}^{(i)},\mathbf{B}^{(i-1)})$ (see Sec. 4.3 in the main paper), we cannot directly take part of the sample $\widetilde{\mathbf{W}}^{(i-1)}(\mathbf{X}^{(i-1)}, \mathbf{B}^{(i-1)})$. Instead, we sample $\widetilde{\mathbf{W}}^{(i-1)}(\mathbf{X}^{(i)}, \mathbf{B}^{(i-1)})$ conditioned on it, from a conditional matrix Gaussian distribution. This is similar to the sampling of the weights in prediction (see Sec. 2). Specifically, let us denote $\widetilde{\mathbf{W}}^{(i-1)}(\mathbf{X}^{(i)}, \mathbf{B}^{(i-1)})$ by $\widetilde{\mathbf{W}}_i^{(i-1)}$ and $\widetilde{\mathbf{W}}^{(i-1)}(\mathbf{X}^{(i-1)}, \mathbf{B}^{(i-1)})$ by $\widetilde{\mathbf{W}}_{i-1}^{(i-1)}$ for simplicity. We have

$$\begin{aligned}&p(\widetilde{\mathbf{W}}_i^{(i-1)} | \widetilde{\mathbf{W}}_{i-1}^{(i-1)}) \\ &= \mathcal{MN}(\widetilde{\mathbf{W}}_i^{(i-1)} | \mathbf{K}_{i,i-1}^{(i)} {\mathbf{K}^{(i-1)}}^{-1} \widetilde{\mathbf{W}}_{i-1}^{(i-1)}, \mathbf{K}^{(i)} - \mathbf{K}_{i,i-1}^{(i)} {\mathbf{K}^{(i-1)}}^{-1} \mathbf{K}_{i-1,i}^{(i)}, \mathbf{K}_{BB}^{(i-1)})\end{aligned}$$

where $\mathbf{K}^{(i-1)}$ is the covariance (kernel) matrix on $\mathbf{X}^{(i-1)}$, $\mathbf{K}^{(i)}$ on $\mathbf{X}^{(i)}$, $\mathbf{K}_{i,i-1}^{(i)}$ is the cross covariance between $\mathbf{X}^i$ and $\mathbf{X}^{i-1}$, and $\mathbf{K}_{i-1,i}^{(i)}$ between $\mathbf{X}^{i-1}$ and $\mathbf{X}^i$. All the other computation in learning and prediction are the same.

4 Experimental Details

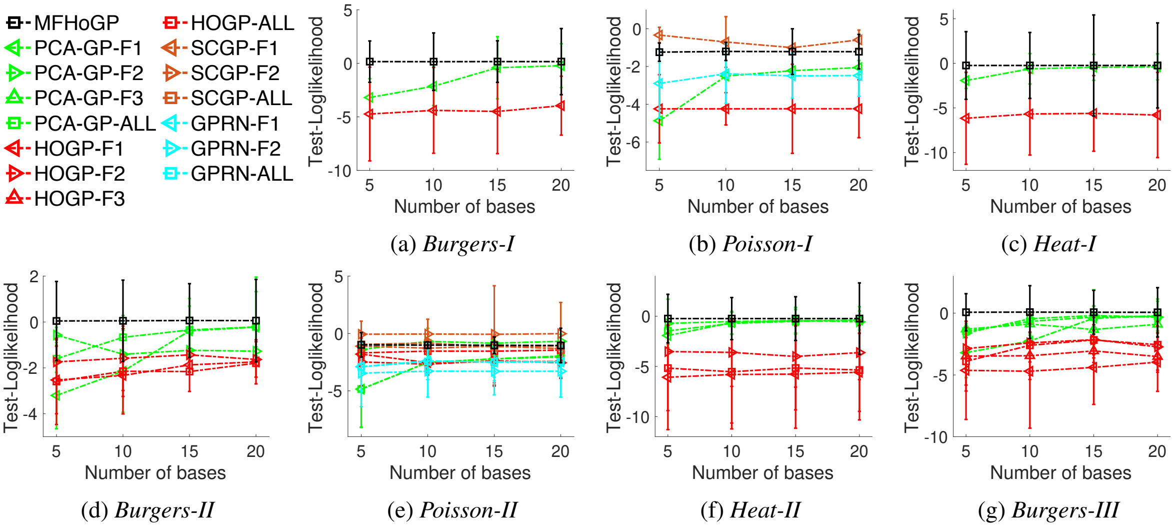
4.1 Data Preparation for Small Solution Fields

As mentioned in our main paper, the small datasets were collected from solving three fundamental partial differential equations (PDEs), Burgers', Poisson's and heat equations, in small spatial/temporal domains. Each equation plays an important role in scientific and engineering applications. The details of the PDEs and data generation are listed as follows.

Burgers' equation is considered as a canonical nonlinear hyperbolic PDE; it is widely used to describe various physical phenomena, such as fluid dynamics (Chung, 2010), nonlinear acoustics (Sugimoto, 1991) and traffic flows (Nagel, 1996). Because it can develop discontinuities (shock waves) based on a normal conservation equation, it also serves as a benchmark test case for many numerical solvers and surrogate models (Kutluay et al., 1999; Shah et al., 2017; Raissi et al., 2017b). The viscous version of this equation is given by $\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = v\frac{\partial^2 u}{\partial x^2}$, where u represents the volume, x indicates a spatial location, t denotes the time, and v represents the viscosity. We set $x \in [0, 1]$, $t \in [0, 3]$, and $u(x, 0) = \sin(x\pi/2)$ with homogeneous Dirichlet boundary conditions. We uniformly sampled viscosities $v \in [0.001, 0.1]$ as the input parameter to generate the solution field. The equation is solved using the finite element with hat functions in space and backward Euler in time domains. The spatial-temporal domain is discretized into a 16×16 regular rectangular mesh for the first (lowest) fidelity solver. The subsequent solvers of higher fidelities double the nodes in each mesh dimension, *e.g.,* 32×32 for the second fidelity and 64×64 for the third fidelity. The result fields (*i.e.,* outputs) are computed from a 128×128 spatial-temporal regular mesh.

Poisson's equation is an elliptic PDE commonly used in mechanical engineering and physics to describe potential fields, *e.g.,* gravitational and electrostatic fields (Chapra et al., 2010). It is a generalization of Laplace's equation (Persides, 1973) and written as $\Delta u = 0$, where Δ is the Laplace operator and u indicates the volume. Despite its simplicity, Poisson's equation is frequently seen in physics and often serves as a basic test case for surrogate models (Lagaris et al., 1998; Tuo et al., 2014). In our experiment, we set a 2D spatial domain $\mathbf{x} \in [0, 1] \times [0, 1]$ with Dirichlet boundary conditions. The constant values of the four boundaries and the centre of the rectangle domain are used as the input parameters, each of which ranges from 0.1 to 0.9. We uniformly sampled the input parameters to generate the corresponding potential fields as the outputs. The PDE is solved using the finite difference method with the first order centre differencing scheme and regular rectangle meshes. We used an 8×8 mesh for the coarsest level solver. The subsequent refined solver uses a finer mesh that doubles the node in each dimension. The result potential fields are computed with a 32×32 spatial-temporal regular grid.

Heat equation is a basic PDE that describes how heat flows evolve over time. Although originally introduced in 1822 to explain heat flows only, the heat equation is ubiquitous in many scientific fields, such as probability theory (Spitzer, 1964; Burdzy et al., 2004) and financial mathematics (Black and Scholes, 1973). Hence, it is also widely used as a surrogate model (Efe and Ozbay, 2003; Raissi et al., 2017a). The heat equation is given by $\frac{\partial u}{\partial t} + \alpha\Delta u = 0$, where u represents the heat, α the thermal conductivity, and Δ the Laplace operator. We set a 2D spatial-temporal domain $x \in [0, 1]$, $t \in [0, 5]$ with the Neumann boundary condition at $x = 0$ and $x = 1$, and $u(x, 0) = H(x - 0.25) - H(x - 0.75)$, where $H(\cdot)$ is the Heaviside step function. The input parameters include the flux rate of the left boundary at $x = 0$ (ranging from 0 to 1), the flux rate of the right boundary at $x = 1$ (ranging from -1 to 0), and the thermal conductivity (ranging from 0.01 to 0.1). The equation is solved using finite difference in space and backward Euler in time domains. The spatial-temporal domain is discretized into a 16×16 regular rectangular mesh for the first (lowest) fidelity solver. A refined solver uses a 32×32 mesh for the second fidelity. The result fields are computed on a 100×100 spatial-temporal grid.



(a) *Burgers-I* (b) *Poisson-I* (c) *Heat-I*

(d) *Burgers-II* (e) *Poisson-II* (f) *Heat-II* (g) *Burgers-III*

Figure 1: The test-loglikelihood (in log scale) on small datasets. The results are averaged from 5 runs. After the dash in each caption (*e.g.,* "*Burgers-II*") is how many fidelities across the training data. -F{1,2,3} indicates the model trained with a particular fidelity's examples and -F-ALL with all the examples.

4.2 Data Preparation for Large-Scale Simulations of Lid-Driven Cavity Flows

We also examined MFHoGP in lid-driven cavity flows (Bozeman and Dalton, 1973), a classic computational fluid dynamics problem. The problem describes how liquid inside a cavity is driven by the lids on the walls, making the pressures vary locally and eventually leading to laminar and turbulent flows inside the cavity. The simulation of lid driven cavity flows involves solving the incompressible Navier-Stokes (NS) equation (Chorin, 1968), $\rho(\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \mu\nabla^2\mathbf{u}$, where ρ is the density, p the pressure, $\mathbf{u}$ the velocity, and μ the dynamic viscosity. The PDE is well known to be challenging to solve due to their complicated behaviours under large Reynolds numbers. It is thus commonly used as a benchmark test case for numerical solvers (Bozeman and Dalton, 1973; Strang and Fix, 1973) and surrogate models (Terragni et al., 2011; Xing et al., 2016). In our experiments, we considered a square cavity $\mathbf{x} \in [0,1] \times [0,1]$ filled with liquid and the time $t \in [0,1]$. The top lid is given a tangential velocity to drive the fluid to flow while the other lids on the remaining walls stay steady. No-slip conditions are applied to all the lids. Given the Reynold number $Re \in [10, 500]$ and the top boundary velocity ranging from 0 to 1, the spatial-temporal pressure field is computed on a 100×100 regular mesh at 100 evenly spaced time points. Hence, we have one million outputs for each input setting. We used the SIMPLE algorithm (Caretto et al., 1973) with a stagger grid (Versteeg and Malalasekera, 2007), the up-wind scheme (Versteeg and Malalasekera, 2007) for the spatial difference, and the implicit time scheme with fixed time steps to solve the PDE. For the lowest fidelity solver, we used a 16×16 spatial mesh and $10,000$ time step to ensure the numerical stability. A subsequent finer solver (of the second fidelity) uses a 32×32 spatial mesh and the same number of time steps.

4.3 Test Log Likelihood

We report the test loglikelihood of PCA-GP, HOGP, SCGP, GPRN and MFHoGP on small datasets in Fig.1. Since KPCA-GP and IsoMap-GP are not standard probabilistic models (they do not have likelihood terms), their test loglikelihoods are unavailable. As we can see, similar to the RMSE results (in the main paper), MFHoGP consistently outperforms all the competing methods, except on *Poisson-I* and *Poisson-II*, SCGP obtains slightly higher likelihood. However, SCGP is inefficient and cannot handle a large number of outputs, *e.g.,* over ten thousands.

4.4 Local Output Recovery

We supplement the local prediction results in the settings of *Heat*-2, *Burgers*-2 and *Burgers*-3, which are shown in Fig. 2, 3 and 4, respectively. We also show the local prediction results of simulating lid-driven cavity flows in Fig. 5 *(F1=200, F2=20)*. Each column are the results of one method. The leftmost is MFHoGP. For each setting, we show the difference between the prediction and ground-truth in 10 randomly selected test fields.

As we can see, MFHoGP always obtains better predictions for individual outputs. This is implied by the fact that most output regions of MFHoGP are rendered by lighter colours. Hence, it confirms that our method not only yields a superior

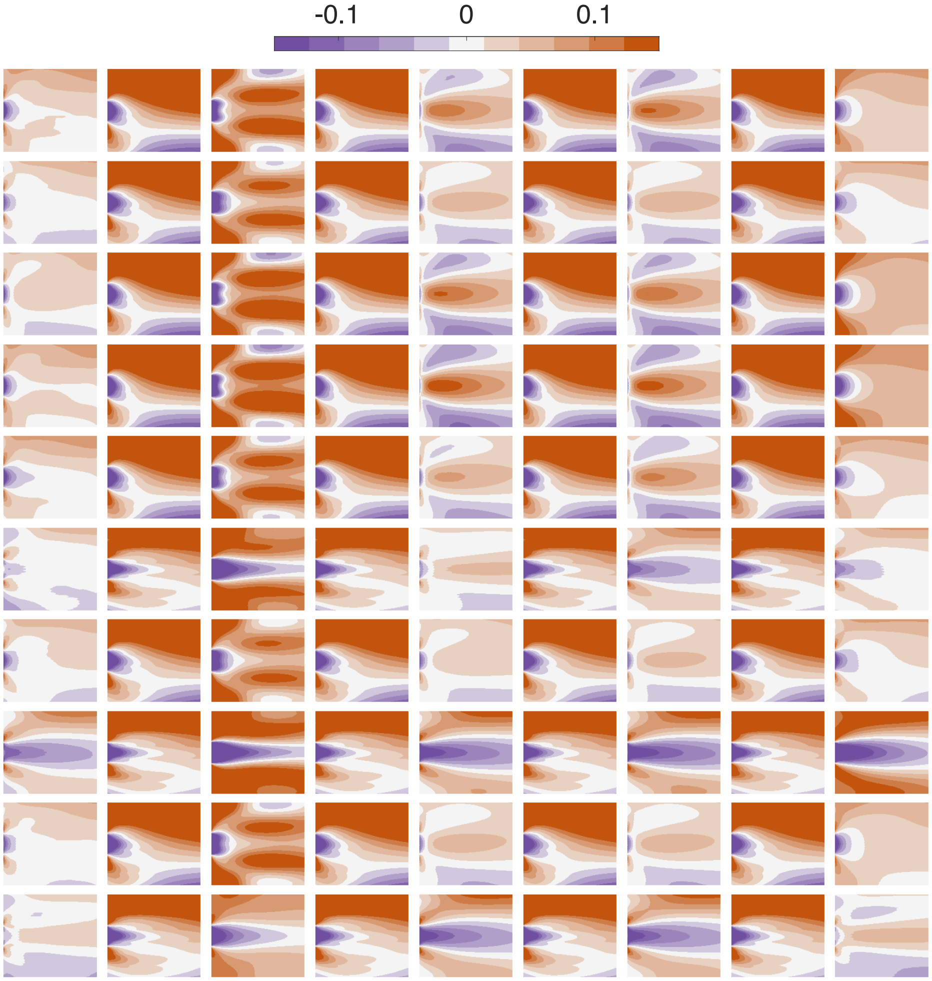


Figure 2: Visualization of local errors in *Heat*-2. Each image represents the difference between the prediction and ground-truth over individual outputs of a test example. From the left column to the right are the results of MFHoGP, PCA-GP-{1,2}, KPCA-GP-{1,2}, IsoMap-GP-{1,2} and HOGP-{1,2}.

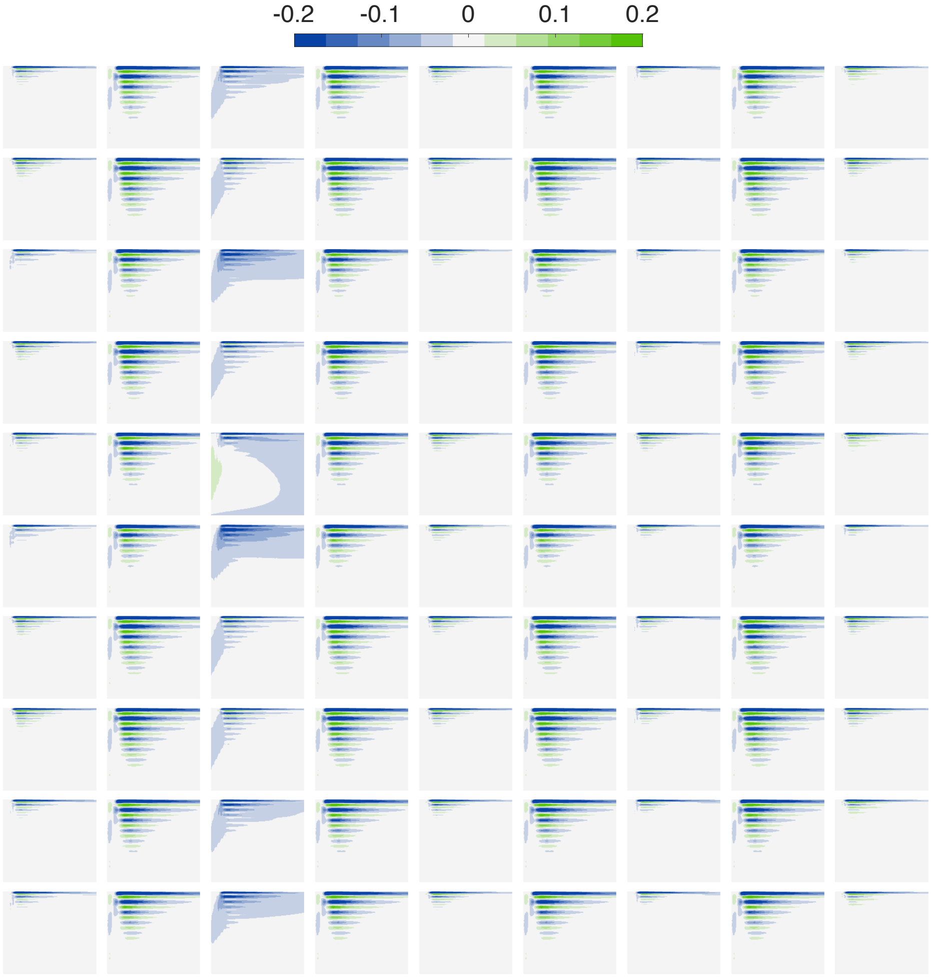


Figure 3: Visualization of local errors in *Burgers*-2. From the left column to the right are the results of MFHoGP, PCA-GP-{1,2}, KPCA-GP-{1,2}, IsoMap-GP-{1,2} and HOGP-{1,2}.

global prediction accuracy but also better recovers individual outputs locally.

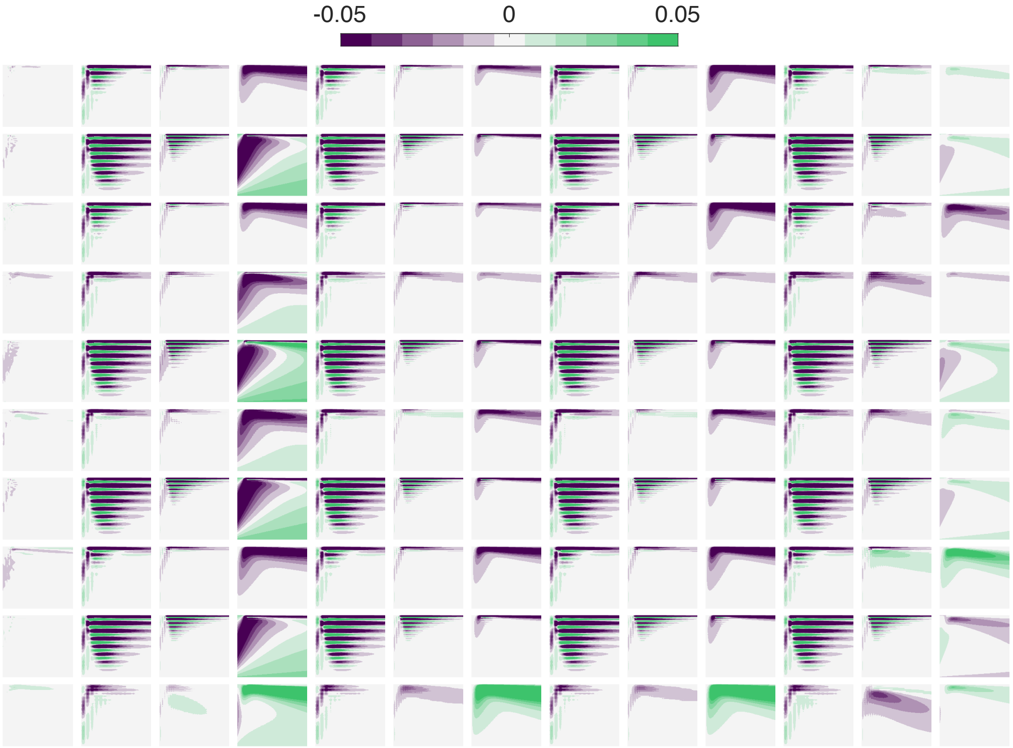


Figure 4: Visualization of local errors in *Burgers*-3. From the left column to the right are the results of MFHoGP, PCA-GP-{1,2,3}, KPCA-GP-{1,2,3}, IsoMap-GP-{1,2,3} and HOGP-{1,2,3}.

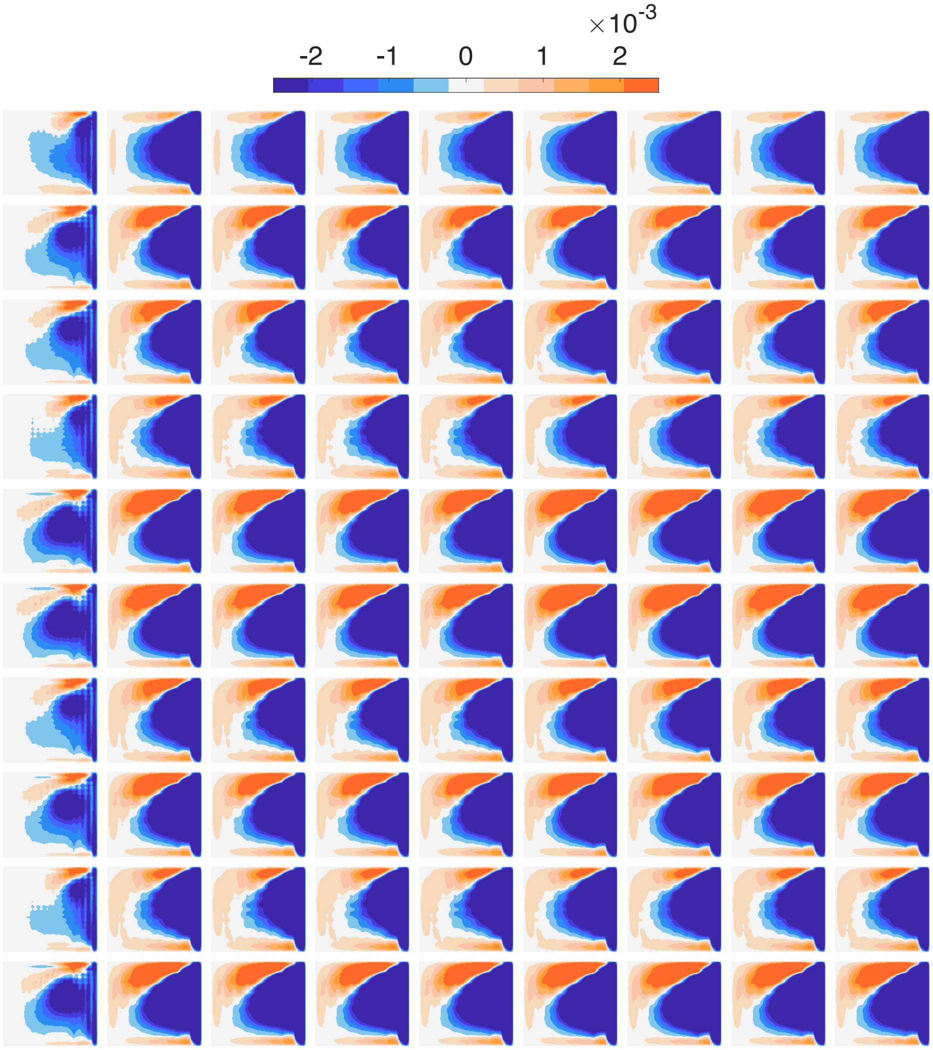


Figure 5: Visualization of local errors in the one-million pressure field prediction for lid-driven cavity flows. Each image represents the difference between the prediction and ground-truth over individual outputs of a test spatial field (100 × 100) at a randomly chosen time point, in the setting of *F1=200, F2=20*. From the left column to the right are the results of MFHoGP, PCA-GP-{1,2}, KPCA-GP-{1,2}, IsoMap-GP-{1,2} and HOGP-{1,2}.

References

Black, F. and Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of political economy*, 81(3):637–654.

Bozeman, J. D. and Dalton, C. (1973). Numerical study of viscous flow in a cavity. *Journal of Computational Physics*, 12(3):348–363.

Burdzy, K., Chen, Z.-Q., Sylvester, J., et al. (2004). The heat equation and reflected brownian motion in time-dependent domains. *The Annals of Probability*, 32(1B):775–804.

Caretto, L., Gosman, A., Patankar, S., and Spalding, D. (1973). Two calculation procedures for steady, three-dimensional flows with recirculation. In *Proceedings of the third international conference on numerical methods in fluid mechanics*, pages 60–68. Springer.

Chapra, S. C., Canale, R. P., et al. (2010). *Numerical methods for engineers*. Boston: McGraw-Hill Higher Education,.

Chorin, A. J. (1968). Numerical solution of the navier-stokes equations. *Mathematics of computation*, 22(104):745–762.

Chung, T. (2010). *Computational fluid dynamics*. Cambridge university press.

Efe, M. O. and Ozbay, H. (2003). Proper orthogonal decomposition for reduced order modeling: 2d heat flow. In *Proceedings of 2003 IEEE Conference on Control Applications, 2003. CCA 2003.*, volume 2, pages 1273–1277. IEEE.

Kutluay, S., Bahadir, A., and Özde**c**s, A. (1999). Numerical solution of one-dimensional burgers equation: explicit and exact-explicit finite difference methods. *Journal of Computational and Applied Mathematics*, 103(2):251–261.

Lagaris, I. E., Likas, A., and Fotiadis, D. I. (1998). Artificial neural networks for solving ordinary and partial differential equations. *IEEE transactions on neural networks*, 9(5):987–1000.

Minka, T. P. (2000). Old and new matrix algebra useful for statistics.

Nagel, K. (1996). Particle hopping models and traffic flow theory. *Physical review E*, 53(5):4655.

Persides, S. (1973). The laplace and poisson equations in schwarzschild's space-time. *Journal of Mathematical Analysis and Applications*, 43(3):571–578.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2017a). Machine learning of linear differential equations using gaussian processes. *Journal of Computational Physics*, 348:683–693.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2017b). Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. *arXiv preprint arXiv:1711.10561*.

Shah, A., Xing, W., and Triantafyllidis, V. (2017). Reduced-order modelling of parameter-dependent, linear and nonlinear dynamic partial differential equation models. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2200):20160809.

Spitzer, F. (1964). Electrostatic capacity, heat flow, and brownian motion. *Probability theory and related fields*, 3(2):110–121.

Strang, G. and Fix, G. J. (1973). *An analysis of the finite element method*, volume 212. Prentice-hall Englewood Cliffs, NJ.

Sugimoto, N. (1991). Burgers equation with a fractional derivative; hereditary effects on nonlinear acoustic waves. *Journal of fluid mechanics*, 225:631–653.

Terragni, F., Valero, E., and Vega, J. M. (2011). Local pod plus galerkin projection in the unsteady lid-driven cavity problem. *SIAM Journal on Scientific Computing*, 33(6):3538–3561.

Tuo, R., Wu, C. J., and Yu, D. (2014). Surrogate modeling of computer experiments with different mesh densities. *Technometrics*, 56(3):372–380.

Versteeg, H. K. and Malalasekera, W. (2007). *An introduction to computational fluid dynamics: the finite volume method*. Pearson education.

Xing, W., Triantafyllidis, V., Shah, A., Nair, P., and Zabaras, N. (2016). Manifold learning for the emulation of spatial fields from computational models. *Journal of Computational Physics*, 326:666–690.