Report on suitability and potential of ROM to fusion models

A Non-intrusive ROM for Solvers with High-dimensional Outputs

Deyu Ming and Serge Guillas
University College London
UKAEA Report: 2047352_2-TN-01 D1.1
Progress report with preliminary findings
April 27, 2021

1 Introduction

Many modern physical computer models involve solving PDEs with numerical solvers, such as finite element methods (FEM), which can be computationally expensive due to

- ever more complex and larger-scale models;
- high-dimensional input and output;
- large demands on computational resources.

These create challenges to efficient uncertainty quantification of computer models, such as the fusion models, as we often need to run the models in a large number of times for tasks such as sensitivity analysis, uncertainty propagation and model calibration. To tackle these challenges, reduced order models (ROM) are needed to

- serve as low-dimensional replacements with comparable accuracy;
- reduce evaluation time of original solvers;
- save storage, e.g., for high-dimensional output.

Traditional reduced order models, also known as intrusive reduced order models, often are constructed using reduced basis methods [16], among which the Proper Orthogonal Decomposition (POD) is perhaps the most popular technique. The intrusive reduced order models for original high-fidelity models with high-dimensional output are typically built using a two-phase procedure called offline-online decomposition:

- *offline phase*: high-fidelity solutions/outputs are obtained and reduced basis is calculated;
- *online phase*: the original problems are projected onto the reduced space for efficient computation of solutions at new inputs.

However, the online phase of the intrusive reduced order modelling is challenging in practice because:

- expertise and domain knowledge are required to project the equations and physics of the original high-fidelity problems to constructed reduced space;
- dimensionality reduction techniques are largely constrained by the problem formulation;
- uncertainty is not incorporated.

For these reasons, in this report we focus on non-intrusive reduced order models for problems with high dimensional outputs, utilising the family of Gaussian process (GP) surrogates. These have been successfully implemented for dimension reduction of either outputs or inputs. For instance:

- [9] used Functional Principal Components Analysis (FPCA) as an equivalent approach to POD for time series outputs of tsunami waves, and [2] used Spherical Harmonics and Gaussian Markov Random Fields for optimal reduction of surfaces outputs.
- For inputs, [13] employed a kernel-based approach to extract the few input field directions of most influence for the outputs in order to build GPs with few input dimensions (orders of magnitude gain in dimension).

The report is organised as follows. In Section 2, a non-intrusive ROM with GP surrogates and POD is described. The method is then applied in Section 3 for a anisotropic heat transport problem. Future directions are discussed in Section 4.

2 Non-intrusive ROM with Gaussian Process Surrogates

The non-intrusive reduced order modelling is a data-driven approach that uses a statistical surrogate model to mimic the functional relations between the model input and constructed reduced output space in the online phase of the offline-online decomposition. The utilisation of statistical surrogates alleviates the difficulties involved in reformulating the original high-fidelity problems under the intrusive reduced order modelling. In particular, with GP surrogates we are able to quantify uncertainty of the high-dimensional outputs predicted at unobserved input positions.

Let $\mathbf{X} \in \mathbb{R}^{N\times D}$ contain N sets of D dimensional input of a computer model, which produces N corresponding sets of K dimensional output $\mathbf{Y} \in \mathbb{R}^{N\times K}$ accordingly. Then, one can mimic the functional relationships between the input $\mathbf{X}$ and each output dimension $\mathbf{Y}_k \in \mathbb{R}^{N\times 1}$ by a GP surrogate $\mathcal{GP}_k$ independently for $k = 1, \ldots, K$ without considering the dependence between output dimensions [6]. Ignoring the potential cross-dependence does not pose a serious issue unless we are interested in the joint distribution of the output, and it can be shown [11] that the independently constructed GP surrogates correspond to the marginal GPs of a joint GP surrogate under certain dependence structures. The GP surrogate $\mathcal{GP}_k$ is formally defined as a multivariate normal distribution with respect to $\mathbf{Y}_k$:

$$\mathbf{Y}_k \sim \mathcal{N}(\boldsymbol{\mu}_k(\mathbf{X}), \sigma_k^2 \mathbf{R}_k(\mathbf{X})),$$

in which the i-th element of $\boldsymbol{\mu}_k(\mathbf{X}) \in \mathbb{R}^{N\times 1}$ is often specified by a trend function $f_k(\mathbf{X}_i)$ with $\mathbf{X}_i \in \mathbb{R}^{1\times D}$ being the i-th row of $\mathbf{X}$, and the ij-th element of $\mathbf{R}_k(\mathbf{X}) \in \mathbb{R}^{N\times N}$ is given by $c_k(\mathbf{X}_i, \mathbf{X}_j)$, where c_k is a given kernel function. The trend function f_k can be formulated as a linear combination of a set of basis functions of $\mathbf{X}_i$ and we assume a constant trend function $f_k(\mathbf{X}_i) = b_k$ in this report.

There are various choices for c_k (see [17]). In this report, we use the separable kernel function:

$$c_k(\mathbf{X}_i, \mathbf{X}_j) = \prod_{d=1}^{D} c_{k,d}(X_{id}, X_{jd}),$$

where $c_{k,d}$ is a one-dimensional kernel function. A typical choice for $c_{k,d}$ in computer model emulation is the squared exponential (SExp) kernel:

$$c_{k,d}(X_{id}, X_{jd}) = \exp\left\{-\frac{(X_{id} - X_{jd})^2}{\gamma_{k,d}^2}\right\},$$

where $\gamma_{k,d} > 0$ is the range parameter. However, the SExp kernel has been criticised for its over-smoothness [20] for physical problems as well as its associated ill-conditioned problems [3, 8]. Another popular kernel choice is the Matérn kernel [17]:

$$c_{k,d}(X_{id}, X_{jd}) = \exp\left(-\frac{\sqrt{2p+1}\, r_{ij,d}}{\gamma_{k,d}}\right) \frac{p!}{(2p)!} \sum_{i=0}^{p} \frac{(p+i)!}{i!(p-i)!} \left(\frac{2r_{ij,d}\sqrt{2p+1}}{\gamma_{k,d}}\right)^{p-i},$$

where $r_{ij,d} = X_{id} - X_{jd}$. The Matérn kernel is known to be less prone to ill-conditioning issues and provides a reasonably adequate smoothness to the GP surrogates. In particular, the Matérn-2.5 kernel, which is defined as the Matérn kernel with $p = 2$:

$$c_{k,d}(X_{id}, X_{jd}) = \left(1 + \frac{\sqrt{5}|X_{id} - X_{jd}|}{\gamma_{k,d}} + \frac{5(X_{id} - X_{jd})^2}{3\gamma_{k,d}^2}\right) \exp\left\{-\frac{\sqrt{5}|X_{id} - X_{jd}|}{\gamma_{k,d}}\right\},$$

is the default kernel choice for many computer model emulation packages, such as `DiceKriging` [19] and `RobustGaSP` [7]. Therefore, we employ the Matérn-2.5 kernel in this report.

The posterior predictive distribution $\mathcal{N}(\widehat{\mu}_k(\mathbf{x}^*), \widehat{\sigma}_k^2(\mathbf{x}^*))$ of $\mathcal{GP}_k$ with respect to the output $Y_k^*(\mathbf{x}^*)$ at an unobserved input position $\mathbf{x}^*$ is given in different analytical forms depending on how the model parameters b_k, σ_k^2 and $\{\gamma_{k,d}\}_{d=1,\ldots,D}$ are estimated. Different maximum-likelihood-based estimation approaches and the corresponding expressions for $\widehat{\mu}_k(\mathbf{x}^*)$ and $\widehat{\sigma}_k^2(\mathbf{x}^*)$ are discussed in [19, 8].

The main computational bottlenecks of the GP surrogate construction are the number of data points N and the dimension K of the output of a computer model. Since the inference of GP surrogates involve inversions of the correlation matrix $\mathbf{R}_k \in \mathbb{R}^{N\times N}$ with computational complexity of $\mathcal{O}(N^3)$, it soon becomes computationally prohibitive to build GP surrogates in practice when N is more than several thousands. In such a case, one may need sparse approximations [12] to the GP to reduce the computational complexity induced by the big data.

In computer model experiments, one often does not have big data (i.e., realisations from the underlying computer model) due to the limited computational budget. However, if the input dimension D is large, then small data are insufficient to explore adequately the whole input domain and thus the resulting GP surrogates can be inaccurate. High input dimension also causes challenges to the model estimation because a large number of range parameters $\{\gamma_{k,d}\}_{d=1,\dots,D}$ need to be estimated for each output dimension. To alleviate this issue, one can reduce the input dimension D to P such that $P \ll D$ by dimension reduction techniques such as POD, kernel dimension reduction [13], and active subspace [21].

A high output dimension K creates the issue that it can be computational burdensome to build K independent GP surrogates: without parallel implementation the training and validation of a huge amount of GP surrogates are practically infeasible. This report tackles the latter issue on high-dimensional outputs (e.g., a snapshot where each point on the snapshot represents a FE solution and contributes to the output dimensionality) produced by computer models. Perhaps the most straightforward approach to address the issue is to reduce the output dimension K to L such that $L \ll K$ by POD.

The POD of $\mathbf{Y} \in \mathbb{R}^{N\times K}$ can be done with following steps:

1. Compute the sample mean $\boldsymbol{\mu}_{\mathbf{Y}} \in \mathbb{R}^{1\times K}$ of $\mathbf{Y}$ and obtain the centred output matrix $\mathbf{Y}_c = \mathbf{Y} - \boldsymbol{\mu}_{\mathbf{Y}}$;

2. Implement the eigendecomposition of $\mathbf{G} = \frac{1}{N}\mathbf{Y}_c\mathbf{Y}_c^\top$ such that $\mathbf{G} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\top$, where the columns of $\mathbf{V} \in \mathbb{R}^{N\times N}$ contains the eigenvectors of $\mathbf{G}$ and the diagonal of $\boldsymbol{\Lambda} \in \mathbb{R}^{N\times N}$ contains the corresponding eigenvalues $(\lambda_1, \dots, \lambda_N)$ in descending order;

3. Compute $\tilde{\mathbf{V}} = \mathbf{Y}_c^\top\mathbf{V} \in \mathbb{R}^{K\times N}$, which contains the eigenvectors of sample covariance matrix $\mathbf{C} = \frac{1}{N}\mathbf{Y}_c^\top\mathbf{Y}_c$;

4. Choose $L \leq N$ and obtain the low dimensional output $\widehat{\mathbf{Y}} = \mathbf{Y}_c\tilde{\mathbf{V}}_L \in \mathbb{R}^{N\times L}$, where $\tilde{\mathbf{V}}_L \in \mathbb{R}^{K\times L}$ contains the first L eigenvectors included in $\tilde{\mathbf{V}}$.

One can also obtain $\tilde{\mathbf{V}}$ by performing the singular value decomposition (SVD) of $\mathbf{Y}_c$ that is implemented, e.g., in the PCA function of Python package `scikit-learn` [15]. After obtaining the low dimensional data $\widehat{\mathbf{Y}}$, we then construct L independent GP surrogates of each of L dimensions of $\widehat{\mathbf{Y}}$. Let $\mathcal{N}(\widehat{\mu}_l(\mathbf{x}^*), \widehat{\sigma}_l^2(\mathbf{x}^*))$ be the posterior predictive distribution of $\widehat{Y}_l^*(\mathbf{x}^*)$, the l-th dimension of the low dimensional output, predicted at an unobserved input position $\mathbf{x}^*$. Then the posterior predictive distribution of the corresponding high dimensional output $\mathbf{Y}^*(\mathbf{x}^*) \in \mathbb{R}^{1\times K}$ is given by

$$\mathcal{N}\left(\widehat{\boldsymbol{\mu}}(\mathbf{x}^*)\tilde{\mathbf{V}}_L^\top + \boldsymbol{\mu}_{\mathbf{Y}}, \tilde{\mathbf{V}}_L\widehat{\boldsymbol{\Sigma}}(\mathbf{x}^*)\tilde{\mathbf{V}}_L^\top\right),$$

where $\widehat{\boldsymbol{\mu}}(\mathbf{x}^*) = (\widehat{\mu}_1(\mathbf{x}^*), \dots, \widehat{\mu}_L(\mathbf{x}^*))$ and $\widehat{\boldsymbol{\Sigma}}(\mathbf{x}^*) = \mathrm{diag}(\widehat{\sigma}_1^2(\mathbf{x}^*), \dots, \widehat{\sigma}_L^2(\mathbf{x}^*))$.

Figure 1 demonstrates the procedure to build non-intrusive reduced order model with GP surrogates. In the offline phase, dimension-reduction techniques, e.g., POD, are applied to reduce the high-dimensional output to a low-dimensional space. Then in the online phase, GP surrogates are constructed independently on each reduced dimension. Using the constructed GP surrogate and reduced basis, one can obtain the predicted low-dimensional and in turn the high-dimensional output at new input positions with little computational efforts.

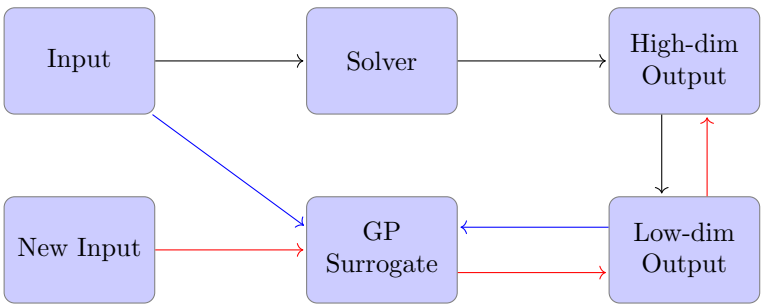

Figure 1: The workflow to construct non-intrusive ROM with GP. The black arrows represent the offline phase; the blue arrows represent the online phase; the red arrows represent the prediction procedure using the constructed non-intrusive ROM with GP.

3 2-D model of anisotropic heat transport

In this section, we explore the non-intrusive ROM with GP to mimic the FE solver to the 2-D problem "Open field lines with oscillating anisotropy directions" in [5]. The problem has two key inputs m and α that control the anisotropy of the solution field, i.e., the anisotropy direction is defined by

$$\mathbf{b} = \frac{\mathbf{B}}{|\mathbf{B}|}, \quad \mathbf{B} = \begin{pmatrix} \alpha(2y-1)\cos(m\pi x) + \pi \\ \pi\alpha m(y^2 - y)\sin(m\pi x) \end{pmatrix},$$

where $m/2$ is the number of oscillation periods in the computational domain and α is the amplitude. The output is a high-dimensional 2-D field defined on the square computational domain $[0,1] \times [0,1]$ and allows a closed form solution: .

3.1 Experimental Design

To construct the reduced basis via the POD and the GP surrogate, N=40 samples are arranged in a Latin hypercube over $m \in [0, 12]$ and $\alpha \in [0, 3]$ (see the left plot in Figure 2). We then run the FE solver (implemented in `Firedrake` [18]) of the toy problem to obtain the corresponding 2-D outputs, each of which contains FE solutions on $K = 78961$ nodes. These 40×78961 high-dimensional outputs are then reduced to 40 low-dimensional outputs (40×25) using POD by retaining the first 25 principal components out of the total 40 components, see the right plot in Figure 2, where the cumulative explained variance is defined as $\frac{\sum_{i=1}^{L} \lambda_i}{\sum_{i=1}^{N} \lambda_i}$ with L be the number of components.

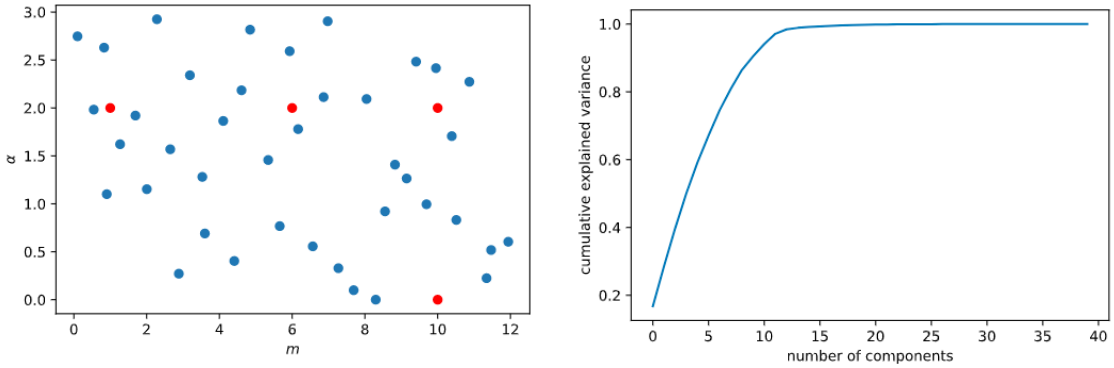

Figure 2: *(Left)*: Training and designing points generated for the inputs m and α. The blue points are design input locations generated from the Latin hypercube design and the red points are testing input locations; *(Right)*: cumulative explained variance given by the POD.

GP surrogates are then constructed independently for each of the 25 dimensions of the reduced order data. GP surrogates are trained with the Matérn-2.5 kernel using the `RobustGaSP` package in R.

3.2 Experimental Results

We test the constructed non-intrusive ROM at four testing input positions $(m, \alpha) = (6, 2)$, $(10, 2)$, $(1, 2)$ and $(10, 0)$ (see the left plot of Figure 2). The FE solutions (from the `Firedrake`) and the predicted solutions from the built ROM are compared in Figure 3. The normalised (to the range of FE solutions) errors between the FE solutions and the predicted solutions from the built ROM are shown in Figure 4. The coverage of the ROM (i.e., the instances that the FE solutions fall within the predictive bounds of GP-based ROM) are also given in Figure 5.

It can been seen from these results that the constructed ROM with GP could predict well the FE solutions of the anisotropic problem at input locations that are not realised. Among the four testing positions, the final case with $m = 10$ and $\alpha = 0$ presents the largest normalised errors up to 13%. This is not a surprising result because m has no effect on the FE solution of the problem when $\alpha = 0$. However, this information is not fully captured in the training data and thus not gained by the non-intrusive ROM with GP, which is pure data-driven method that only understands the functional relation between m, α and the solution field from the training set. As a result, we could observe 5 blurred oscillation periods in the predicted solutions from ROM in Figure 3. However, the predictive interval (whose upper and lower bounds are given at two standard deviations $2\widehat{\sigma}$ above and below the predictive mean $\widehat{\mu}$) of the

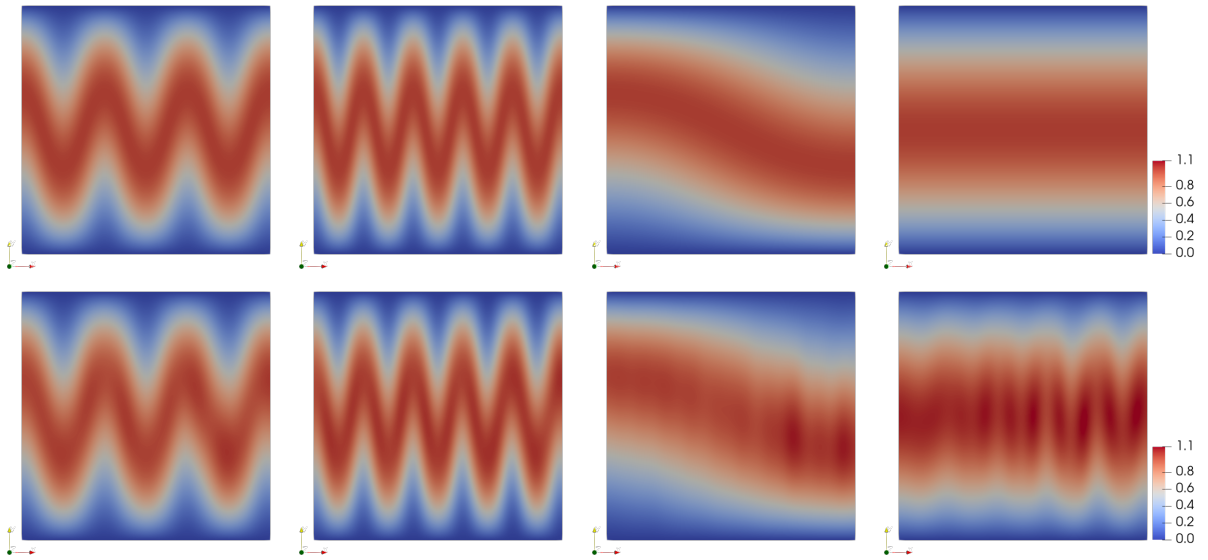

Figure 3: Comparisons of FE solutions to the predicted solutions given by the constructed GP-based ROM. The first row gives the FE solutions. The second row gives the predicted solutions from the GP-based ROM. The columns from left to right correspond to testing input positions $(m, \alpha) = (6, 2)$, $(10, 2)$, $(1, 2)$ and $(10, 0)$ respectively.

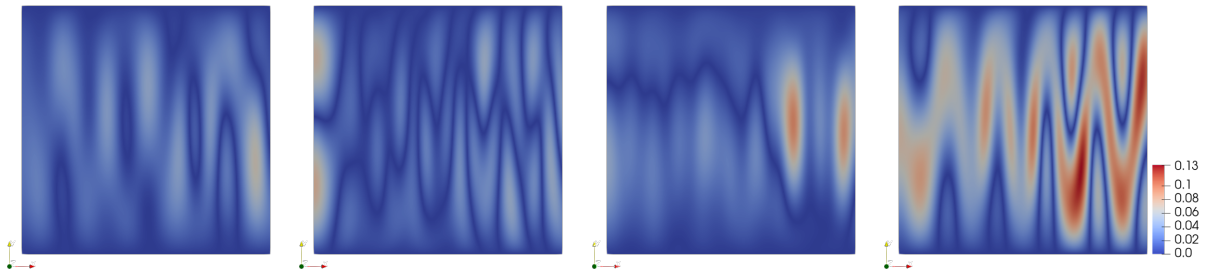

Figure 4: The normalised errors between FE solutions and the predicted solutions from the ROM with GP surrogate. The plots from left to right correspond to testing input positions $(m, \alpha) = (6, 2)$, $(10, 2)$, $(1, 2)$ and $(10, 0)$ respectively.

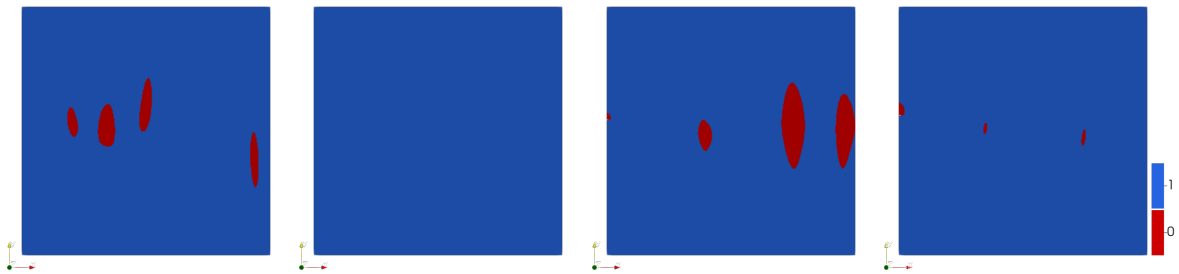

Figure 5: The coverage of constructed ROM with GP, giving the instances that FE solutions fall within the predictive bounds provided by the ROM with GP. 1 indicates that the FE solution is covered by the predictive interval (whose upper and lower bounds are given at two standard deviations $2\widehat{\sigma}$ above and below the predictive mean $\widehat{\mu}$) and 0 indicates otherwise. The plots from left to right correspond to testing input positions $(m, \alpha) = (6, 2)$, $(10, 2)$, $(1, 2)$ and $(10, 0)$ respectively.

GP-based ROM covers the FE solutions sufficiently in this case, demonstrating that one can benefit from the predictive uncertainty embedded in the non-intrusive ROM coupled with GP emulation.

4 Future Directions

We demonstrate in this report that a non-intrusive ROM with GP surrogate could be used to replace computationally expensive computer solvers for problems with high-dimensional output, in one of the building blocks of nuclear fusion modelling. However, the predictive performance of GP-based ROM relies on the information contained in the training data, i.e., the quality of computer experimental design. We use Latin hypercube sampling (LHS) in this experiment, but it is worth exploring the benefits provided by the sequential designs [1], especially in cases where FE solutions are changing rapidly in small regions of the input space. The rapid changes of FE solution field also indicates that the GP surrogate should incorporate the non-stationary features, giving rise to the more advanced Gaussian process models with deep hierarchies [4]. Furthermore, dimension reduction techniques such as POD lose information when the original data are projected onto a lower dimensional space, and thus some extreme but important events could be masked in the low dimensional data, a scenario called masking effect. As a result, if the GP surrogate is built on the low dimensional data one may not be able to recover these outlying events using the constructed non-intrusive ROM.

Although the non-intrusive ROM requires no domain knowledge and access to the source code of original problems, it ignores the physics implied by the underlying problem and thus may be inaccurate comparing to the its intrusive counter-party. Therefore, it would be worth exploring the trade-off between the speed and accuracy of intrusive and non-intrusive MOR, especially in context of UQ. It would also be interesting to find a middle ground where one could exploit the benefits (e.g., accuracy, speed and uncertainty) of both intrusive and non-intrusive ROM, producing a physics-informed non-intrusive ROM. Some relevant literature on physics-informed machine learning (say using a boundary condition or other approaches) include [22, 10, 23].

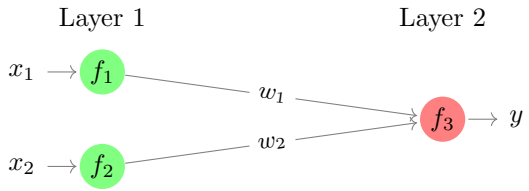

Figure 6: An illustrative example of a system of three computer models f_1, f_2 and f_3. Note this is only for illustration. Linked GP in [14] can work on any feed-forward computer systems.

Since fusion models are often multi-disciplinary and multi-physics, the recent advances on linked Gaussian process surrogates [14] could be considered and explored as a potential candidate to construct non-intrusive ROM for fusion systems by linking non-intrusive ROM of individual sub-models. For example, to construct the ROM of the two-layered system in Figure 6, one could first build GP-based non-intrusive ROM (as demonstrated above) for all individual sub-models (f_1, f_2 and f_3) and then construct the non-intrusive ROM of the whole system by linking the non-intrusive ROM of f_1 and f_2 to that of f_3 through the reduced space w_1 and w_2 analytically. One key benefit of this approach for system-wise reduced order modelling is that one only needs to do dimensionality reduction to the outputs of sub-models. Whereas, to build intrusive ROM, one has to make extra challenging efforts to reformulate the original high-fidelity model (e.g., f_3) under both reduced input (e.g., w_1 and w_2) and output (e.g., y).

References

[1] J. Beck and S. Guillas, *Sequential design with mutual information for computer experiments (MICE): emulation of a tsunami model*, SIAM/ASA Journal on Uncertainty Quantification, 4 (2016), pp. 739–766.

[2] K.-L. Chang, S. Guillas, et al., *Computer model calibration with large non-stationary spatial outputs: application to the calibration of a climate model*, Journal of the Royal Statistical Society Series C, 68 (2019), pp. 51–78.

[3] K. R. Dalbey, *Efficient and Robust Gradient Enhanced Kriging Emulators*, Tech. Rep. SAND2013–7022, Sandia National Laboratories: Albuquerque, NM, USA, 2013.

[4] A. Damianou and N. D. Lawrence, *Deep Gaussian processes*, in Artificial intelligence and statistics, PMLR, 2013, pp. 207–215.

[5] F. Deluzet and J. Narski, *A two field iterated asymptotic-preserving method for highly anisotropic elliptic equations*, Multiscale Modeling & Simulation, 17 (2019), pp. 434–459.

[6] M. Gu and J. O. Berger, *Parallel partial Gaussian process emulation for computer models with massive output*, The Annals of Applied Statistics, 10 (2016), pp. 1317–1347.

[7] M. Gu, J. Palomo, and J. O. Berger, *RobustGaSP: robust Gaussian stochastic process emulation in R*, arXiv:1801.01874, (2018).

[8] M. Gu, X. Wang, and J. O. Berger, *Robust Gaussian stochastic process emulation*, The Annals of Statistics, 46 (2018), pp. 3038–3066.

[9] S. Guillas, A. Sarri, S. J. Day, X. Liu, F. Dias, et al., *Functional emulation of high resolution tsunami modelling over Cascadia*, Annals of Applied Statistics, 12 (2018), pp. 2023–2053.

[10] K. Kashinath, M. Mustafa, A. Albert, J. Wu, C. Jiang, S. Esmaeilzadeh, K. Azizzadenesheli, R. Wang, A. Chattopadhyay, A. Singh, et al., *Physics-informed machine learning: case studies for weather and climate modelling*, Philosophical Transactions of the Royal Society A, 379 (2021), p. 20200093.

[11] K. N. Kyzyurova, *On uncertainty quantification for systems of computer models*, PhD thesis, Duke University, 2017.

[12] H. Liu, Y.-S. Ong, X. Shen, and J. Cai, *When Gaussian process meets big data: a review of scalable GPs*, IEEE transactions on neural networks and learning systems, 31 (2020), pp. 4405–4423.

[13] X. Liu and S. Guillas, *Dimension reduction for Gaussian process emulation: An application to the influence of bathymetry on tsunami heights*, SIAM/ASA Journal on Uncertainty Quantification, 5 (2017), pp. 787–812.

[14] D. Ming and S. Guillas, *Linked Gaussian process emulation for systems of computer models using Matérn kernels and adaptive design*, arXiv:1912.09468, (2021).

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, *Scikit-learn: Machine learning in Python*, Journal of Machine Learning Research, 12 (2011), pp. 2825–2830.

[16] A. Quarteroni, A. Manzoni, and F. Negri, *Reduced Basis Methods for Partial Differential Equations: An Introduction*, vol. 92, Springer, 2015.

[17] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, Cambridge, MA, 2006.

[18] F. Rathgeber, D. A. Ham, L. Mitchell, M. Lange, F. Luporini, A. T. McRae, G.-T. Bercea, G. R. Markall, and P. H. Kelly, *Firedrake: automating the finite element method by composing abstractions*, ACM Transactions on Mathematical Software (TOMS), 43 (2016), pp. 1–27.

[19] O. Roustant, D. Ginsbourger, and Y. Deville, *DiceKriging, DiceOptim: two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization*, Journal of Statistical Software, 51 (2012).

[20] M. L. Stein, *Interpolation of Spatial Data: Some Theory for Kriging*, Springer, New York, 1999.

[21] R. Tripathy, I. Bilionis, and M. Gonzalez, *Gaussian processes with built-in dimensionality reduction: applications to high-dimensional uncertainty propagation*, Journal of Computational Physics, 321 (2016), pp. 191–223.

[22] I. Vernon, S. E. Jackson, and J. A. Cumming, *Known boundary emulation of complex computer models*, SIAM/ASA Journal on Uncertainty Quantification, 7 (2019), pp. 838–876.

[23] D. Watson-Parris, *Machine learning for weather and climate are worlds apart*, Philosophical Transactions of the Royal Society A, 379 (2021), p. 20200098.