Statement of Research

Randy Price (rprice25@gmu.edu)

My main research areas are in the intersection of partial differential equations, data assimilation and machine learning. Most recently I am interested in a simplification of the Kalman filter called nudging and its applications in the areas of machine learning and data assimilation. Given a continuous dynamical system,

$$\frac{d}{dt}u = F(u), \quad u(0) = u_0,$$

the nudging algorithm entails solving an associated system,

$$\frac{d}{dt}w = F(w) - \mu(I_h w - y^{QoI}), \quad w(0) = w_0 \text{ (arbitrary)}.$$

Here I_h is a linear operator called the interpolant operator and y^{QoI} represents a quantity of interest. The quantity of interest will be either observations of the original system or a desired state. Nudging has a rich history which originates in meteorology decades ago. For more references we refer the reader to [3, 2] and references therein.

1. Machine Learning + Data Assimilation

My machine learning research is focused on capturing the dynamics of ODEs/PDEs with deep neural networks (DNNs) and applying the nudging methodology. The works in this statement use residual neural networks (ResNets) of the form,

$$\begin{aligned} y_1 &:= \sigma(W_0 y_0 + b_0), \\ y_{\ell+1} &:= y_\ell + \tau\sigma(W_\ell y_\ell + b_\ell), \quad \ell = 1, ..., L-2, \\ y_L &:= W_{L-1} y_{L-1}. \end{aligned} \tag{1}$$

Here $y_0 \in \mathbb{R}^d$ is the input vector, inner layer feature vectors $y_\ell \in \mathbb{R}^{n_\ell}$ and $y_L \in \mathbb{R}^{d^*}$ is the output vector. Then we have an activation function σ, weight matrices $W_\ell \in \mathbb{R}^{n_\ell \times n_{\ell+1}}$ and bias vectors $b_\ell \in \mathbb{R}^{n_\ell}$. The main applications of this work is in data assimilation but the same techniques can be applied to do control or aid neural network surrogate models.

Recent Work

NINNs: Nudging Induced Neural Networks (2022)
https://arxiv.org/abs/2203.07947. (submitted.)
In this paper [2] the methods to control neural networks via a nudging feedback control term are developed. Nudging Induced Neural Networks (NINNs) are introduced on residual neural networks (ResNets) as shown in 2.

$$\begin{aligned} y_1 &:= \sigma(W_0 y_0 + b_0) - \tau\mu g_1(y^{QoI}, y_1), \\ y_{\ell+1} &:= y_\ell + \tau\sigma(W_\ell y_\ell + b_\ell) - \tau\mu g_{\ell+1}(y^{QoI}, y_\ell), \quad \ell = 1, ..., L-2, \\ y_L &:= W_{L-1} y_{L-1} - \tau\mu g_L(y^{QoI}, y_{L-1}). \end{aligned} \tag{2}$$

In 2, each function g_ℓ is a feedback control term that is based on the original nudging algorithm. NINNs are applied to ResNets that have learned one time step of the Lorenz ODEs and the 1D Kuramoto-Sivashinsky PDE with comparable data assimilation results to the nudging algorithm. We prove convergence error estimates for NINNs when the algorithm is deployed for data assimilation and control. We denote by u the true solution and by w_{NINN} the proposed NINN solution. Then we have the following error estimate provided certain assumptions are met and μ is large enough,

$$\|u - w_{NINN}\| \lesssim \epsilon_{NN} + \epsilon_{\Delta t} + \epsilon_{NINN} + 2^{-k}|w_0 - u_0| := \text{I} + \text{II} + \text{III} + \text{IV}.$$

Here I is a neural network approximation error. Then II is a time discretization error such as forward Euler. Finally, III is an error arising from introducing NINNs and IV is the standard exponential decay term.

Data Assimilation with Deep Neural Nets Informed by Nudging (2021)
https://arxiv.org/abs/2111.11505. (revised for Computer Methods in Applied Mechanics and Engineering.)
In this paper [1] deep neural networks (DNNs) learn one time step of the nudging algorithm. The DNNs informed by nudging are able to simulate the nudging algorithm by incorporating the previous time step and available data into the input of the DNN. The output is the approximate nudging solution advanced Δt time units. To generate the training

data we generate N_s true solutions $\{u^i\}_{i=1}^{N_s}$ and their corresponding nudging solutions $\{w^i\}_{i=1}^{N_s}$ on the interval $[t_1\ t_{N+1}]$. Below, $u^i(t_k)$ represents the i-th true solution at time t_k and $w^i(t_k)$ represents the corresponding i-th nudging solution at time t_k. Finally, $I_h(u^i(t_k))$ are the observations from the true solutions. Then the training data is the collection,

$$\left\{\left\{\left(\begin{bmatrix} w^i(t_k) \\ I_h(u^i(t_k))\end{bmatrix}, w^i(t_{k+1})\right)\right\}_{k=1}^{N}\right\}_{i=1}^{N_s}.$$

We discovered this approach works well for ODEs but struggles for 1D PDEs. For PDEs, we found applying NINNs is more effective.

Current and Future Work

Data Assimilation for Neural Network Surrogate PDE Models (in progress)
In this paper we will explore performing data assimilation on high dimensional surrogate PDE models. The challenge here is the increased difficulty of training the neural network surrogate with thousands of degrees of freedom to learn. To circumvent the curse of dimensionality we use the Discrete Empirical Interpolation Method (DEIM) on the spatial degrees of freedom. After we learn the DEIM points, a subset of the spatial degrees of freedom, we will deploy NINNs for applications in data assimilation and control of DNNs.

Learning One Time Step of a Chemically Reacting Flow (in progress)
In this paper we discuss techniques for simulating chemically reacting flows with DNNs. Chemically reacting flows are common in engineering applications such as hypersonic flow, combustion, explosions, manufacturing process and environmental assessments. The number of reactions in combustion simulations can exceed 100 making a large number of flow and combustion problems beyond the capabilities of current supercomputers. Motivated by this, DNNs will be introduced to reduce the computational cost. DNNs are constructed which update the species u_n at the n-th timestep to u_{n+1} at the $n+1$-th timestep. Parallel DNNs are trained for each species, taking in u_n as input and outputting one component of u_{n+1}. These DNNs are applied to multiple species and reactions common in chemically reacting flows such as H2-O2 reactions. Experimental results show that the DNNs are able to accurately replicate the dynamics in various situations and in presence of errors. Furthermore, applying NINNs increases the accuracy in the presence of approximate data and allows us to consider variable equivalence ratios (ratio between air and fuel).

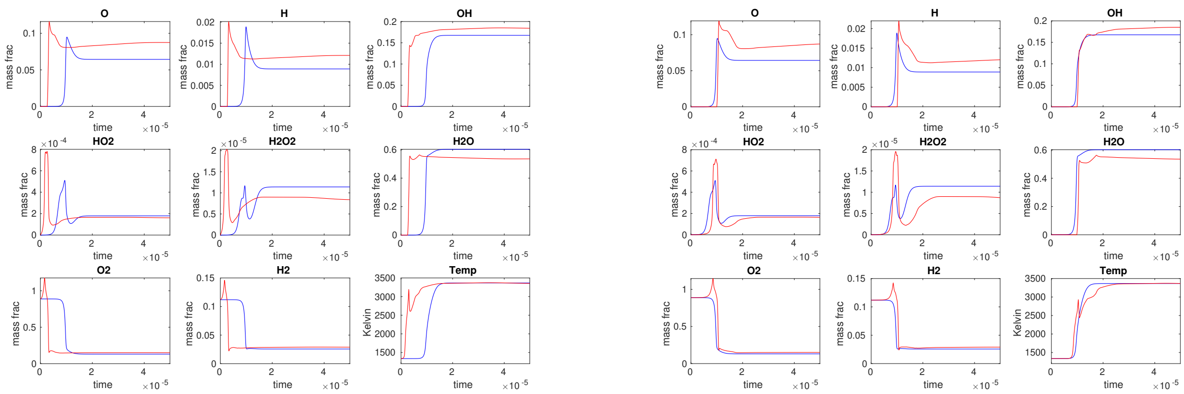

Figure 1: Left: Without NINNs. Right: Nudging towards initial data. Initial temperature 1700K. Blue is the CHEMKIN solution and red is the ResNet solution.

2. Data Assimilation

My data assimilation research (without machine learning) is focused on the nudging data assimilation algorithm for the Navier–Stokes equations. I am interested in global well-posedness, regularity, and convergence of the nudging solution. I am also interested in comparing the nudging algorithm against the ensemble Kalman filter (EnKF).

Recent Work

Continuous Data Assimilation for the Three Dimensional Navier-Stokes Equations (2020)
https://arxiv.org/abs/2003.01329. (SIAM J. Mathematical Analysis.)

In this paper [4] we show that with appropriate restraints on the observations and with general type 1 interpolants, the nudging algorithm is able to recover the full state exponentially in time in terms of the L^2 norm. More specifically, we show the global well-posedness, regularity, and convergence of the nudging algorithm for Leray-Hopf weak solutions of the three dimensional Navier-Stokes equations with certain conditions on the observed data of the following form. Here h is a measure of the quality of the observations. We have as $h \to 0$ then $\|I_h(u) - u\| \to 0$. Let ν represent viscosity, f represent the forcing and let h_0 be the largest appropriate positive real number (this will depend on the interpolant used) such that

$$\|f\|_{L^2(\Omega)}^2 \leq \frac{\nu^4 \lambda_1}{16h} \text{ and } 4\lambda_1 \leq \frac{1}{h^2},\ \forall h \leq h_0.$$

Let $h \leq h_0$ be such that

$$\sup_{t\in[0,T)} \|I_h(u)\|_{H^1(\Omega)}^2 \leq \frac{\nu^2}{16h}, \tag{3}$$

then provided we choose $\mu \sim \frac{\nu}{h^2}$, we have

$$\sup_{t\in[0,T)} \|w\|_{H^1(\Omega)} < \infty \text{ and } \|w-u\|_{L^2(\Omega)} \lesssim e^{-ct}\|u_0\|_{L^2(\Omega)}.$$

One sufficient condition for (3) to hold in the case of modal interpolants is $\sup_{t\in[t_0,T)} \|u\|_{H^s} < \infty$ for $s > \frac{1}{2}$. Furthermore, condition (3) is not needed in the two dimensional case because we have global regularity of solutions and well-defined bounds on the attractor for the required norms. In three dimensions we don't have this so we restrict the underlying true solution to be a Leray-Hopf weak solution which allows us to work with the energy inequalities provided we use the Galerkin construction.

Current and Future Work

Comparison of the Nudging and Ensemble Kalman Filter Data Assimilation Algorithms (in progress)
In this paper we will show with numerical experiments that the nudging data assimilation algorithm is comparable with the ensemble Kalman filter. The algorithms are tested under different observation patterns, different noise levels and different sparsity patterns. We also consider moving observation windows.

3. Uncertainty Quantification

My uncertainty quantification research is focused on the 2D Navier-Stokes equations with random viscosity. I am interested in the applications of polynomial chaos for solving stochastic Navier–Stokes problems.

Recent Work

On Surrogate Learning for Linear Stability Assessment of Navier-Stokes Equations with Stochastic Viscosity (2022) https://arxiv.org/abs/2103.00622. (Applications of Mathematics).
Surrogate learning for PDE's is an effort to decrease the overall computation cost when solving (stochastic) PDE's. The popular surrogates at this time are based on neural networks, generalized polynomial chaos (gPC) and Gaussian process regression. Taking a step towards this direction, this paper [6] is focused on a similar computationally extensive task, linear stability analysis of the Navier-Stokes equations with stochastic viscosity given in the form of a general polynomial chaos expansion. We show that three different surrogates are able to provide meaningful results when applied to the linear stability problem.

A Stochastic Galerkin Method for the Time-Dependent Navier-Stokes Equations (2022) https://arxiv.org/abs/2207.04513. (Journal of Computational Physics)
In this paper [7] we develop a stochastic Galerkin method for solving the 2D Navier-Stokes equations with random viscosity given in the form of a general polynomial chaos expansion. We extended the methodology from [5] into the stochastic framework. We were able to verify our solution by comparing it against a Monte-Carlo and a stochastic collocation method. We also explore the properties of the resulting stochastic solutions.

Current and Future Work

Future work includes comparing surrogate models for solving the Navier–Stokes equations with stochastic viscosity given in the form of a general polynomial chaos expansion.

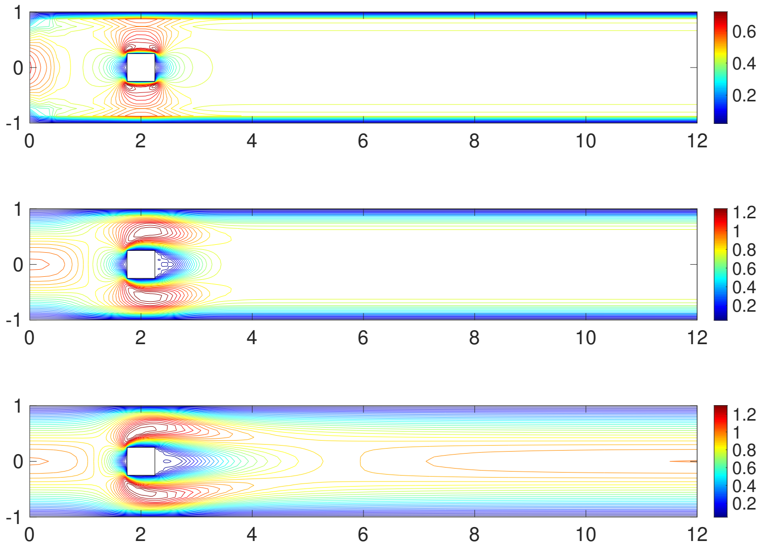

Figure 2: Mean horizontal velocity at times 0.1s (top), 1s (center) and 100s (bottom).

References

[1] H. Antil, R. Löhner, and R. Price, *Data assimilation with deep neural nets informed by nudging*, arXiv preprint arXiv:2111.11505, (2021).

[2] ———, *Ninns: Nudging induced neural networks*, arXiv preprint arXiv:2203.07947, (2022).

[3] A. Azouani, E. Olson, and E. S. Titi, *Continuous data assimilation using general interpolant observables*, Journal of Nonlinear Science, 24 (2014), pp. 277–304.

[4] A. Biswas and R. Price, *Continuous data assimilation for the three-dimensional navier–stokes equations*, SIAM Journal on Mathematical Analysis, 53 (2021), pp. 6697–6723.

[5] D. A. Kay, P. M. Gresho, D. F. Griffiths, and D. J. Silvester, *Adaptive time-stepping for incompressible flow part ii: Navier–stokes equations*, SIAM Journal on Scientific Computing, 32 (2010), pp. 111–128.

[6] B. Sousedík, H. C. Elman, K. Lee, and R. Price, *On surrogate learning for linear stability assessment of navier-stokes equations with stochastic viscosity*, Applications of Mathematics, (2022), pp. 1–23.

[7] B. Sousedík and R. Price, *A stochastic galerkin method with adaptive time-stepping for the navier–stokes equations*, Journal of Computational Physics, 468 (2022), p. 111456.