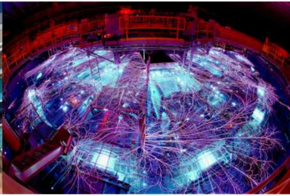
Laboratories

SAND2019-0998C



Machine Learning Closure Modeling for Reduced-Order Models of Dynamical Systems

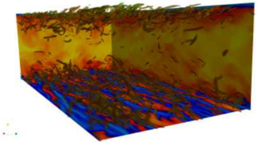
Eric J. Parish, SNL

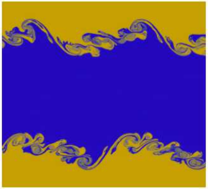
12/4/18

ratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia,
neywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract

Numerical simulation

Numerical simulation has evolved to be a powerful tool in science and engineering





- Contributed to new scientific discoveries
- Revolutionized the engineering design process
- Numerical simulation of multiscale systems is an outstanding challenge!

1.) NASA: `www.nas.nasa.gov/SC11/demos/demo20.html`

Properties of multiscale systems

Disparate length and time-scales

- Display many orders of length and time scales

Many systems do not have scale separation

- Very challenging to develop models
- Direct computations are expensive
- Often rely on reduced-complexity models

Reduced-complexity simulation is a pacing item in computational physics

Reduced-complexity numerical simulation

Sandia National Laboratories

- Many types of reduced-complexity modeling:
  1. **Projection-based reduced order models**
  2. **Coarse numerical simulation**

Reduced-complexity numerical simulation

- Many types of reduced-complexity modeling:
  1. **Projection-based reduced order models**
  2. **Coarse numerical simulation**

- Reduced-complexity methods for multiscale systems suffer from
  - Stability
  - Accuracy
  - **Issues stem from truncation ("closure problem")**

Reduced-complexity numerical simulation

- Many types of reduced-complexity modeling:
  1. **Projection-based reduced order models**
  2. **Coarse numerical simulation**

- Reduced-complexity methods for multiscale systems suffer from
  - Stability
  - Accuracy
  - **Issues stem from truncation ("closure problem")**

- Problems are amplified in complex multiscale/multiphysics problems
  - Developed and tuned for canonical systems
  - Often inaccurate in important regimes

Talk Overview

- **Main focus is on the "closure problem":**
  - Primarily address this in the context of Galerkin methods

1. Outline the MZ-VMS Framework
   - Subgrid-scale modeling framework for reduced-order methods

2. Develop a data-driven machine learning MZ-VMS model
   - Apply to advection diffusion equation

Galerkin Problem Statement (Global Case)

- Nonlinear initial value problem

$$\frac{\partial u}{\partial t} + \mathcal{R}(u) = f \qquad x \text{ in } \Omega$$

- The state variable is expressed as

$$u(x,t) = \sum_{j=1}^{N} w_j(x) a_j(t), \qquad w, u \in \mathcal{V}$$

- $\mathbf{a}$ **are the modal coefficients**
- Galerkin method leads to the weighted residual form

$$\big(w, u_t\big) + \big(w, \mathcal{R}(u) - f\big) = 0$$

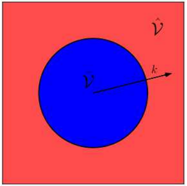
- $(\cdot,\cdot)$ is an inner product
- **Challenge in multiscale systems:**
  - For accurate answers N is often prohibitively large
    - *($N \approx \infty$ for continuum problems)*
  - How can we reduce N to M?

Variational Multiscale Method

- Projection-based multiscale splitting framework
- Developed by Hughes et al. for multiscale phenomena
- Relies on scale separation projectors
- Main idea: sum decomposition of the solution space

- Sum decomposition:

$$\mathcal{V} = \tilde{\mathcal{V}} \oplus \hat{\mathcal{V}}$$



- Leads to state decomposition:

$$u(x,t) = \sum_{j=1}^{M} \tilde{w}_j \tilde{a}_j + \sum_{j=M+1}^{N} w'_j a'_j$$

Hughes, T. J., Feijoo, G., Mazzei, L., and Qunicy, J., "The variational multiscale method - a paradigm for computational mechanics," Computer methods in applied mechanics and engineering, Vol. 166, 1998, pp. 173-189.

Variational Multiscale Method

- VMS decomposition of solution space $\mathcal{V} = \tilde{\mathcal{V}} \oplus \hat{\mathcal{V}}$

$$u(x,t) = \tilde{u}(x,t) + \hat{u}(x,t)$$

- Splitting leads to two sub-problems
  - M-dimensional coarse-scale equation:

$$\left(\tilde{w}, \tilde{u}\right) + \left(\tilde{w}, \mathcal{R}(\tilde{u}) - f\right) = -\left(\tilde{w}, \mathcal{R}(u) - \mathcal{R}(\tilde{u})\right)$$

  - $N - M$ dimensional fine-scale equation:

$$\left(w', u'\right) + \left(w', \mathcal{R}(u) - \mathcal{R}(\tilde{u})\right) = -\left(w', \mathcal{R}(\tilde{u}) - f\right)$$

- **Goal is to solve the coarse-scale problem**

Modeling Challenge

- Unclosed coarse-scale equation:

$$\left(\tilde{w}, \tilde{u}\right) + \left(\tilde{w}, \mathcal{R}(\tilde{u}) - f\right) = -\left(\tilde{w}, \mathcal{R}(u) - \mathcal{R}(\tilde{u})\right)$$

- Model for unresolved physics:

$$\downarrow$$

$$\mathcal{M}(\tilde{u}) \approx -\left(\tilde{w}, \mathcal{R}(u) - \mathcal{R}(\tilde{u})\right)$$

- Closed coarse-scale equation:

$$\downarrow$$

$$\boxed{\left(\tilde{w}, \tilde{u}\right) + \left(\tilde{w}, \mathcal{R}(\tilde{u}) - f\right) = \mathcal{M}(\tilde{u})}$$

- **How can we construct $\mathcal{M}$ in a systematic way?**
  - **We use the Mori-Zwanzig formalism**

Mori-Zwanzig: A Basic Example

Mori (1961), Zwanzig (1966)

- Basic linear system

$$\frac{dx}{dt} = A_{11}x + A_{12}y, \qquad \frac{dy}{dt} = A_{21}x + A_{22}y$$

- Seek ROM where y is unresolved

$$\frac{dx}{dt} = A_{11}x + \mathcal{M}(x)$$

- Solve y equation with integrating factors (superposition)

$$\frac{dx}{dt} = A_{11}x + \int_0^t A_{12}e^{A_{22}(t-s)}A_{21}x(s)\,ds + A_{12}y_0e^{A_{22}t}$$

- **Model reduction leads to memory effects!**
- **Mori-Zwanzig formalism generalizes to non-linear**

Mori-Zwanzig Formalism

- Unclosed coarse-scale equation:

$$\left(\tilde{w}, \tilde{u}\right) + \left(\tilde{w}, \mathcal{R}(\tilde{u}) - f\right) = -\left(\tilde{w}, \mathcal{R}(u) - \mathcal{R}(\tilde{u})\right)$$

- Mori-Zwanzig process for closed coarse-scale equation:

$$\downarrow$$

$$\left(\tilde{w}, \tilde{u}\right) + \left(\tilde{w}, \mathcal{R}(\tilde{u}) - f\right) = \left(\tilde{w}, \int_0^t K(\tilde{u}(t-s), s)\right)$$

- **Challenge: Memory term is not computable**
- **Memory term is a starting point to develop models**

Mori-Zwanzig Models

- t-model

$$\int_0^t \mathbf{K}(\tilde{u}(t-s), s)ds \approx t\mathbf{K}(\tilde{u}(t), 0)$$

  - Benefits: Model is complete (no parameters)
  - Drawbacks: Model can be inaccurate

- τ-model

$$\int_0^t \mathbf{K}(\tilde{u}(t-s), s)ds \approx \tau\mathbf{K}(\tilde{u}(t), 0)$$

  - Benefits: More accurate than the t-model
  - Drawbacks: Requires user defined parameters

- **Can we use machine learning to do better?**

Mori-Zwanzig Machine Learning Model

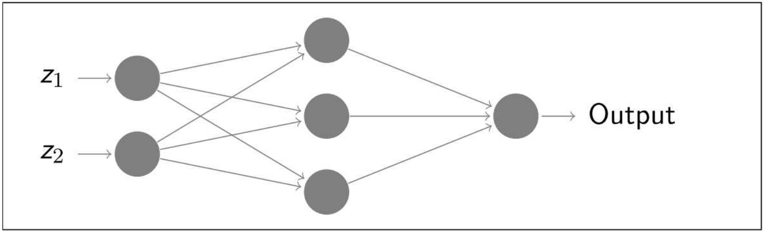
- Can we approximate the MZ memory intergral with a machine learning model?

$$\int_0^t \mathbf{K}(\tilde{u}(t-s), s)ds \approx \delta(\mathbf{z}(\tilde{u}(t)))$$

- **Important Questions**
  - What ML architecture should we use to construct δ?
  - What input features should we use?

Neural Network Model

- Neural networks are a popular ML model



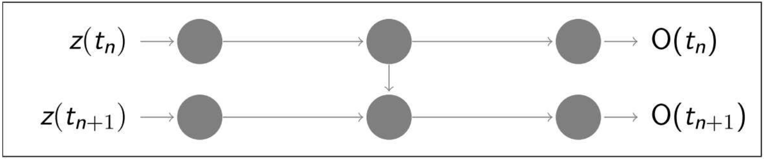
- Relies on function composition

$$\delta(\mathbf{z}(\tilde{\mathbf{a}}_A, \mu)) = \mathbf{g}_N(\cdot; \eta_N) \circ \mathbf{g}_{N-1}(\cdot; \eta_{N-1}) \circ \ldots \circ \mathbf{g}_0(\mathbf{z}(\tilde{\mathbf{a}}_A, \mu); \eta_0),$$

  - η_i : weights at the i_{th} layer
  - g_i: activation functions at the i_{th} layer
- **Can't capture non-Markovian effects**

Recurrent Neural Network Model

- Generalization of NNs for sequential problems
- Capable of capturing memory effects



Input Features

- Accuracy of machine learning algorithms depends on input features
  - Input feature selection is often an art
- MZ-VMS provides a promising feature

$$\mathbf{z} = \mathbf{K}(\tilde{u}(t), 0)$$

Training the Machine Learning Model

- Neural network is coupled to the forward model
  - Standard techniques (backprop) can't be used to train the neural network
  - Training needs to be coupled to the forward model
- Training is performed with the adjoint equations,

$$\frac{d}{dt}\lambda(t) = -\frac{\partial \tilde{\mathbf{F}}}{\partial \tilde{\mathbf{a}}}^{T}\lambda(t) - \frac{\partial \delta}{\partial \tilde{\mathbf{a}}}^{T}\lambda(t) + \mathbf{W}_{\epsilon_d \epsilon_d}(\mathbf{d} - \tilde{\mathbf{a}}).$$

$$\lambda(t_f) = 0$$

$$\eta = \eta_0 + \int \mathbf{C}_{\eta\eta}\frac{\partial \delta}{\partial \eta}^{T}\lambda dt.$$

Steepest decent update:

$$\eta^{n+1} = \eta^{n} + \epsilon \int \mathbf{C}_{\eta\eta}\frac{\partial \delta}{\partial \eta}^{T}\lambda dt$$

Numerical Example: Advection Diffusion ROM

- Examine the parameterized advection-diffusion equation

$$\frac{\partial}{\partial t}u(x,t) = \frac{\partial u}{\partial x} + \frac{1}{Re}\frac{\partial^2 u}{\partial x^2},$$

$$u(0,t) = u(2,t) = 0, \qquad u(x,0) = x(2-x)\exp(2x),$$

- Parameters:
  - $Re \in [5, 250]$: Reynolds number
- Truth model is a finite difference scheme

$$\frac{du_k}{dt} = p_0 \frac{u_{k+1} - u_k}{\Delta x} + p_1 \frac{u_{k+1} - 2u_k + u_{k-1}}{\Delta x^2},$$

  - Truth model is 100 dimensional

Numerical Example: Advection Diffusion ROM

- Generation of Reduced Model:
  - Solve truth model for $Re = [5, 85, 170, 250]$
  - Search solution snapshots for low dimensional basis
    - eg. POD, PCA, ...
  - Galerkin projection of truth model onto low dimensional basis
- Mathematically:

$$\frac{d\mathbf{u}}{dt} = \mathbf{A}\mathbf{u}, \qquad \mathbf{u} \in \mathbb{R}^{100}$$

$$\downarrow$$

$$\frac{d\tilde{\mathbf{a}}}{dt} = \tilde{\mathbf{A}}\tilde{\mathbf{a}}, \qquad \tilde{\mathbf{a}} \in \mathbb{R}^{3}$$

- Dimensionality of the system reduced by 30x
- However, ROM has error

ML Closure Model

- We augment our ROM with a closure term:

$$\frac{d\tilde{\mathbf{a}}}{dt} = \tilde{\mathbf{A}}\tilde{\mathbf{a}} + \delta(\mathbf{z}(\tilde{\mathbf{a}}))$$

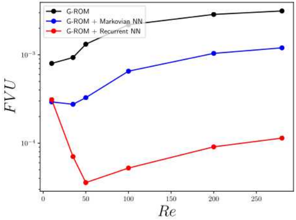
- Input features are:

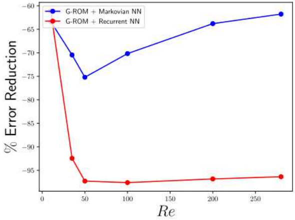
$$\mathbf{z}(t) = \{\mathbf{K}(\tilde{\mathbf{a}}(t), 0), Re\}$$

- Network details:
  - Employ a combined NN + RNN network
  - Recurrent activation function: linear
  - Markovian activation functions: ReLU
  - One hidden layer, one neuron in RNN
  - One hidden layer, eight neurons in NN

Training: Regression Results

- Markovian and recurrent networks are trained individually for various Reynolds numbers





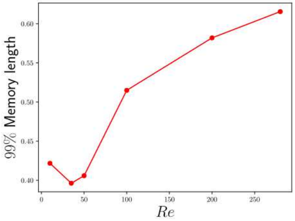
- **Recurrent network leads to $> 98\%$ error reduction for high Reynolds numbers**

Training: Physical insight

- Recurrent neural network has the recursion:

$$\mathbf{h}^{n+1} = \boxed{c_0}\,\mathbf{h}^n + c_1\mathbf{z}^{n+1}$$

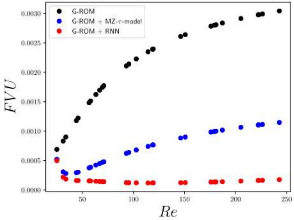
  - Parameter c_0 can be interpreted as the "forget" parameter
  - Defines the memory length

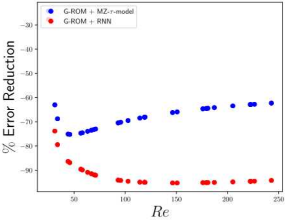


- **Memory length grows with increasing Reynolds numbers**

Testing

- RNN is tested for $Re \in [20, 250]$
- Compared to state of the art MZ model





- **Recurrent network leads to $> 90\%$ error reduction for high Reynolds numbers**
- **Model is predictive at new Reynolds numbers**
  - Low generalization error

Summary

- Quantifying and reducing errors in reduced-order models is of critical importance
- We outlined the MZ-VMS method for reduced-order models
  - VMS is used to isolate the "subgrid" errors
  - MZ is used a starting point to develop models
- Outlined a data-informed approach that combines MZ-VMS with machine learning
  - We use recurrent neural networks to model memory effects
  - MZ-VMS memory is used as an input into the RNN
- Demonstrated method on advection diffusion equation
  - $> 98\%$ error reduction on training data
  - $> 90\%$ error reduction on testing data

Thank you for your time!

Sandia National Laboratories

- Funding sources:
  - Sandia National Labs von Nemann Postdoctoral Fellowship

Reference Papers:

Parish E.J., Wentland C., and Duraisamy K., "A Petrov-Galerkin Reduced-Order Model with Memory Effects," *Submitted*, 2018.
Parish E.J. and Duraisamy K., "A Unified Framework for Multiscale Modeling using Mori-Zwanzig and the Variational Multiscale Method," *arXiv 1712.09669v1*, 2017.
Parish, E. J. and Duraisamy K., "A Dynamic Subgrid Scale Model for Large Eddy Simulations Based on the Mori-Zwanzig Formalism", *Journal of Computational Physics*, 2017.
Gouasmi, A., Parish, E.J., and Duraisamy, K., Characterizing Memory Effects in Coarse-Grained Nonlinear Systems Using the Mori-Zwanzig formalism," *Proc. Royal Society A*, 2017
Parish E.J. and Duraisamy K. "A Paradigm for data-driven modeling using Field Inversion and Machine Learning," *JCP*, 2016.