A New Approach to Solve Multiscale SPDEs based on Multiscale Basis functions and Graphical Model learning

Jiang Wan and Nicholas Zabaras

Materials Process Design and Control Laboratory
Sibley School of Mechanical and Aerospace Engineering
101 Frank H. T. Rhodes Hall
Cornell University
Ithaca, NY 14853-3801

Email: zabaras@cornell.edu
URL: http://mpdc.mae.cornell.edu/
Introduction

- All physical systems have inherent associated randomness. Uncertainties introduced across various length scales have a non-trivial interactions.

  Physical properties, structure follow a statistical description

  Use micro averaged models for resolving physical (macro) scale

- Conventional collocation based methods for SPDEs are based on large quantities of samples, which results in overwhelming computational cost especially for multiscale problems.
Introduction

Therefore, it is desirable to have fast tools that give accurate predictions given stochastic input.

Compared with expensive deterministic solvers, regression models are more efficient. However, due to the curse of dimensionality, a single classical regression model is not feasible for multiscale problems with high-dimensional stochastic input.

As a result, we use local regression models, generally low dimensional, to capture micro-scale features of the system.

Then a probabilistic model with a graph is applied to denote the conditional independence structure between local coarse features. We use the combination of such a graphical model and low dimensional local regression models to predict physical responses without solving any multiscale PDEs.
Problem definition

- Basic fluid flow equations defined in a complete probability space \((\Omega, \mathcal{F}, P)\)
  \[
  u(x, \omega) = -a(x, \omega) \nabla p(x, \omega) \quad \forall x \in D \\
  \nabla \cdot u(x, \omega) = f(x)
  \]
  with boundary conditions
  \[
  p = \bar{p} \quad \text{on } \partial D_p \quad , \quad u \cdot n = \bar{u} \quad \text{on } \partial D_u \quad \partial D = \partial D_p \cup \partial D_u
  \]

- Multiscale scales
  - length scale of the system, \(L\)
  - \(>>\) length scale of property variation, \(l\)

- We develop fast predictive models of coarse scale features \(u(x, \omega), p(x, \omega)\)
  given fine scale properties \(a(x, \omega)\) based on a local regression model
  and a global graphical model
Material property is defined as a cell-wise constant on a fine scale grid, \( T_h = \bigcup_{i=1}^{N_h} e_i \) as \( a(x, \omega) = \{ a_i(\omega) \}_{i=1}^{N_h} \)

A coarse-scale grid, \( T_c = \bigcup_{K=1}^{N_c} E_K \), conforming to the fine-scale one is defined to seek the coarse-scale solution. Material property of a coarse element is defined as

\[
a_K = \{ a_i(\omega) | x \in E_K, \forall x \in e_i \}
\]
Regression model

- Denote \( y(x, \omega) := (u(x, \omega), p(x, \omega))^T \). We model \( y(x, \omega) \) as
  \[
y(x, \omega) = \sum_i W_i(\omega) \psi_i(x, \omega)
  \]
  where \( \psi_i(x, \omega) \) are vector basis functions and \( W_i(\omega) \) are scalar weights.

- Here the basis functions are taken as functions of \( a(x, \omega) = \{a_i(\omega)\}_{i=1}^{N_h} \), i.e.
  \[
  \psi_i(x, \omega) = \psi_i(a(x, \omega))
  \]

- In this work, we also degrade the global regression problem into a set of local regression problems as
  \[
y_K(x, \omega) = \sum_i W_i(\omega) \psi_i^K(a_K) \quad \forall x \in E_K
  \]
  with multiscale basis functions which capture small-scale effects on the large scales [1].
Global graphical model

- Graphical representation of the regression model

![Diagram of a global graphical model with nodes and connections labeled with variables such as $E_K$, $a_K$, $\psi_i^K(a_K)$, $W_i(\omega)$, and $y_K(x, \omega)$]
Construction of the regression model is composed of the following steps:

1. Construct local regression models for basis functions
   \[ \psi_{K,i}(a_K) \]

2. Construct global regression models for weights \( W_i(\omega) \)

3. Predict coarse-scale response
   \[ y_K(x,\omega) = \sum_i W_i(\omega)\psi_i^K(a_K) \]
In the $K$-th coarse element, denote the vector basis functions for velocity $\mathbf{u}(x, \omega)$ by $\psi^K = \{\psi^K_1 \ldots \psi^K_4\}$

Basis function $V = \text{span}\{\psi^K_i\}$, $\psi^K_i = -a_K(x) \nabla \phi^K_i$

The basis function for pressure $p$ is taken as a piecewise constant, i.e.

$$\phi^K_p(x) = \begin{cases} 1, & \text{if } x \in K \\ 0, & \text{if } x \notin K \end{cases}$$
Local regression for basis functions

- The basis functions are defined locally on each coarse element and are mutually independent given the input \( a_K = \{ a_i(\omega) | x \in E_K \ \forall x \in e_i \} \)

- On each coarse element, we have a radial basis function (RBF) regression model

\[
p\left(\psi^K_i\right) = N\left(\psi^K_i \mid \sum_m c^K_m g^K_m (a_K), \sigma^2 \mathbb{I}\right)
\]

for a fixed location \( x \) with unnormalized Gaussian kernels

\[
g^K_m (a_K) = \exp\left(-\frac{\|a^K_K - a^K_m\|_2^2}{L^2}\right)
\]

where \( a_m \) is the center of the \( m \)-th kernel and \( L \) is the bandwidth.

- Given a set of \( N \) training data \( \left\{ a^{(n)}_{K}, \psi_{i}^{K,(n)} \right\}^{N}_{n=1} \), we get the constant coefficients \( c^K_m \) by minimized the error function

\[
\mathbb{E}_i = \frac{1}{2} \sum_{n=1}^{N} \left\{ \psi^K_i \left( a^{(n)}_{K} \right) - \psi^K_{i,(n)} \right\}^2
\]
Learning of weights

Construct local regression models for basis functions
\( \psi_{K,i}(a_K) \)

Construct global regression models for weights \( W_i(\omega) \)

Predict coarse-scale response
\[
y_K(x, \omega) = \sum_i W_i(\omega) \psi^K_i(a_K)
\]
A hierarchical graphical model for weights (add a layer of coarse scale features)

Fine scale features $a_k$

Coarse scale features $\xi_K$

Weights $W_i(\omega)$
Learning of weights

- The multiscale basis function values can serve as coarse-scale features that include fine-scale information.

- We collect multiscale basis function values at four regularly distributed locations on each coarse element to predict the weights.

Local regression model

\[ \psi^K_i (x, a_K) \approx \sum_m c^K_m g^K_m (x, a_K) \]

\[ \xi_K := \{ \psi^K_i (x_1) \cdots \psi^K_i (x_4) \} \]
Potential functions

- Probabilistic model of weights
  
  \[ p(w | \xi) \propto \prod_i \varphi_i(w_i | \xi) \prod_{(i,j)} \varphi_{ij}(w_i, w_j | \xi) \]

  where \( \xi = \{\xi_K\} \) and \( \varphi(\cdot) \) are called “potential functions” which represent the correlation between random variables.

  - self potential: \( \phi_i(w_i) := \exp \left( b_i w_i - \frac{1}{2} \sum_K \sum_{r=1}^4 d_r a_{K,r} \left( \xi_{K,r}^i \right)^2 \right) w_i^2 \)
    
    \( \xi_{K,r}^i = \psi_{i}^{K}(x_r) \in \xi_K \)

  - edge potential: \( \varphi_{ij}(w_i, w_j) = \exp \left( -w_i w_j \sum_K \sum_{r=1}^4 d_r a_{K,r} \xi_{K,r}^i \xi_{K,r}^j \right) \)
    
    \( a_{K,r} = a_{K}(x_r) \)
Learning of model parameters

- The unknown parameters in this model are $\{d_1 \cdots d_4\}, \{b_1 \cdots b_p\}$ with priors:
  
  $p(d_r) \sim \text{Uniform}(0,1)$ \quad $p(b_i) \sim \text{N}(0,1)$

- The posterior of weights

  \[ p(b, d | w, \xi) \propto \prod_i p(b_i) \prod_r p(d_r) \prod_i \phi_i(w_i | \xi) \prod_{i,j} \phi_{ij}(w_i, w_j | \xi) \]

- Since $\xi = \xi(a_K)$, given a set of training data $\{w^{(i)}, a_K^{(i)}\}_{i=1}^M$, we use Markov chain Monte Carlo to get samples of unknown parameters and take the posterior mean as the values in prediction.
Belief propagation prediction

- Given fine scale properties $a(x, \omega) = \{a_K\}_{K=1}^{N_c}$, we predict the distribution of weights from the graphical model.

Local regression model

$$\psi^K_i(x_j, a_K) \approx \sum_m c^K_m g^K_m(x_j, a_K)$$

$$\xi_K := \{\psi^K_i(x_1) \cdots \psi^K_i(x_4)\}$$

$$p(w | \xi) \propto \prod_i \varphi_i(w_i | \xi) \prod_{(i,j)} \varphi_{ij}(w_i, w_j | \xi)$$

Find the marginal distributions

$$p(w_i | \xi) = \int p(w | \xi) \, dw_{-i}$$
Belief propagation prediction

- Belief propagation (BP) is an efficient way to solve such inference problem based on passing local messages [2].

- The definition of message from node $i$ to node $j$ in the factor graph is

$$m_{ij}(w_j) \propto \int \varphi_{ij}(w_i, w_j) \varphi_i(w_i) \prod_{k \in N(i) \setminus j} m_{ki}(w_i) dw_i$$

marginal distribution

$$p(w_i) \propto \varphi_i(w_i) \prod_{k \in N(i)} m_{ki}(w_i)$$
According to the exponential form of self potential functions, the marginal distribution of each weights is a Gaussian

$$\phi_i (w_i) \propto N\left(\mu_i, P_i^{-1}\right)$$

which makes the computation of messages tractable.

$$\phi_i (w_i) \prod_{k \in N(i) \setminus j} m_{ki}(w_i) \propto N\left(\mu_{i\setminus j}, P_{i\setminus j}\right)$$

$$P_{i\setminus j} = P_{ii} + \sum_{k \in N(i) \setminus j} P_{ki}$$

$$\mu_{i\setminus j} = P_{i\setminus j}^{-1}\left(P_{ii}\mu_{ii} + \sum_{k \in N(i) \setminus j} P_{ki}\mu_{ki}\right)$$

$$m_{ij}(w_j) \propto \int_{w_i} \varphi_{ij}(w_i, w_j) \phi_i (w_i) \prod_{k \in N(i) \setminus j} m_{ki}(w_i)dw_i$$

$$\propto N\left(\mu_{ij}, P_{ij}\right)$$

$$P_{ij} = -A_{ij}^2 P_{i\setminus j}^{-1}$$

$$\mu_{ij} = -P_{ij}^{-1} A_{ij} \mu_{i\setminus j}$$

$$A_{ij} = \sum_{K=1}^{4} \sum_{r=1}^{4} d_r a_{K,r} \xi_{K,r}^i \xi_{K,r}^j$$
Belief Propagation Prediction

The Gaussian BP algorithm

- Initialize:
  - For each node $w_i$, find its neighbors $N(i)$
  - Define the self potential function for each node
    \[
    \varphi_i(w_i | \xi) \sim N(\mu_i, P_i^{-1})
    \]
  - Set the initial $i \rightarrow N(i)$ broadcast messages $\tilde{P}_i = 0, \tilde{\mu}_i = 0$
  - Set the initial message between nodes $k \rightarrow i$
    \[P_{ki} = 0, \mu_{ki} = 0\]

- Iterate:
  - Broadcast the aggregated sum messages
    \[
    \tilde{P}_i = P_{ii} + \sum_{k \in N(i)} P_{ki}
    \]
    \[
    \tilde{\mu}_i = \tilde{P}_i^{-1} \left( P_{ii} \mu_{ii} + \sum_{k \in N(i)} P_{ki} \mu_{ki} \right)
    \]
Belief Propagation Prediction

- Compute the $i \rightarrow j$ message

$$P_{ij} = -A_{ij}^2 P_{i\setminus j}^{-1}$$

$$\mu_{ij} = -P_{ij}^{-1} A_{ij} \mu_{i\setminus j}$$

- If the estimated message $\left(\mu_{ij}, P_{ij}\right)$ does not converge according to a predefined threshold, repeat iteration.

> Infer: compute the marginal means and variance of each node

$$P_i = P_{ii} + \sum_{k \in N(i)} P_{ki}$$

$$\mu_i = P_i^{-1} \left( P_{ii} \mu_{ii} + \sum_{k \in N(i)} P_{ki} \mu_{ki} \right)$$
Numerical examples

- The log-permeability is taken as a Gaussian Markov random field with the covariance kernel

\[
\text{Cov}(x, y) = \sigma^2 \exp\left(-\frac{|x_1 - y_1|}{L_1} - \frac{|x_2 - y_2|}{L_2}\right)
\]

and zero mean.

- In this example, we take \(\sigma=1.0\), \(L_1=0.1\), \(L_2=0.1\). The fine scale permeability is defined on a 64*64 grid. In the computation, a 8*8 coarse grid is used.

- 2,000 data are generated to train the local regression model for multiscale basis functions.

- 5,000 data are generated to train the graphical model of weights.
Example 1

Suppose the following fluid flow problem in random heterogeneous porous media

\[ u(x, \omega) = -a(x, \omega) \nabla p(x, \omega) \quad \forall x \in [0, 1]^2 \]

\[ \nabla \cdot u(x, \omega) = 0 \]

Flow is induced from left bottom corner to right up corner with Dirichlet boundary conditions. Neumann conditions are assumed everywhere else.
Sample results 1

- Log-permeability
- Pressure
- X-velocity
- Y-velocity

**Log-permeability**

**X-velocity**

**Y-velocity**

**Pressure**
Prediction 1

log-permeability

pressure

x - velocity

y - velocity

x

y

u

v

p

0.8
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0.2

0

0.1

0.2

0.3

0.4

0.5

0.6

0.7

0.8

0.9

1

1.1

1.2

1.3

1.4

1.5

1.6

1.7

1.8

1.9

2

2.5

0.5

-0.5

-1

-1.5

-2

-2.5
Sample results 2

1.6 1.4 1.2 1.0 0.8 0.6 0.4 0.2 0.0 -0.2 -0.4 -0.6 -0.8 -1.0 -1.2 -1.4 -1.6 -1.8

0.9 0.85 0.8 0.75 0.7 0.65 0.6 0.55 0.5 0.45 0.4 0.35 0.3 0.25 0.2 0.15 0.1 0.05
Prediction 2

log-permeability

pressure

x - velocity

y - velocity

u

v

p
Mean (5,000 samples)
Variance (5,000 samples)
Predicted variance
empirical PDF at point (0.1875,0.5625)
empirical PDF at point (0.5625, 0.5625)

- Pressure PDF
- Reference
- Prediction

- x-velocity PDF
- Reference
- Prediction

- y-velocity PDF
- Reference
- Prediction
Example 2

- We now use different boundary conditions
  - Flow is induced from left side to right side with Dirichlet boundary conditions. Neumann conditions are assumed on the other two sides.

  - The same Gaussian random field is used as stochastic input (log-permeability)

\[ v \cdot n = 0 \]

\[ p\bigg|_{x=0} = 2 \]

\[ p\bigg|_{x=1} = 0 \]

\[ v \cdot n = 0 \]
Sample results 1

log-permeability

pressure

y - velocity

y - velocity

x - velocity

x - velocity
Prediction 1

- **log-permeability**
  - **x - velocity**
  - **y - velocity**

- **pressure**
  - **y - velocity**
  - **y - velocity**
Prediction 2

**log-permeability**

- **x-velocity**
- **y-velocity**

**pressure**

- **x**
- **y**

**x**-velocity:
- **u**
- **v**

**y**-velocity:
- **u**
- **v**

---

**Materials Process Design and Control Laboratory**

Cornell University
Mean (5,000 samples)

- Graph showing the mean of 5,000 samples with x and y axes ranging from 0 to 1.
- Color scale indicating values from 0 to 1.

---

**Materials Process Design and Control Laboratory**

**Cornell University**

---

38
Predicted mean
Variance (5,000 samples)
Predicted variance
empirical PDF at point (0.1875, 0.5625)
empirical PDF at point (0.5625,0.5625)
Conclusion

- Based on the hierarchical graphical model that links local regression models for basis functions, we develop a fast regression model for multiscale stochastic PDEs without solving any PDEs.

- The local regression problem is in a low-dimensionality and the global graphical model is of sparse structure. The parameters can be fixed after a one-time training process.

- The comparison of predicted results and deterministic solver results shows that both the prediction of a single sample and its statistics are accurate.