High Performance Embedded Hybrid Methodology for Uncertainty Quantification in Multi-physics Problems

Charles Tong - LLNL
Barry Lee - PNNL
Akshay Mittal, Gianluca Iaccarino - Stanford

Motivations and Project Objectives

Project Objective

To develop a hybrid UQ methodology, and associated UQ tools, for multi-physics applications

- The vision of this project is to develop technology for building practical UQ analysis of multi-physics systems founded on sound mathematics and having good parallel efficiency. This technology is developed around a hybrid UQ approach, which employs both intrusive and non-intrusive methods in a flexible manner within a simulation model.

Benefits

Advancing the predictability of multi-physics phenomena through UQ and large-scale simulations

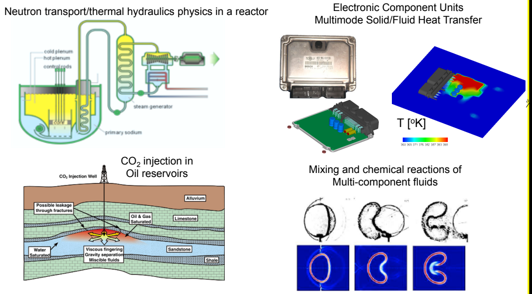
- Computational advances in algorithms and computer architectures are making it possible to analyze deterministic multi-physics models through simulation. To further advance the predictability in multi-physics simulations, UQ capability is needed. This capability will require carefully designed UQ approaches and algorithms.

Challenges

Involving both mathematics and CS issues

- Stable and accurate propagation and representations of uncertainties
- Scalable algorithms for core computational kernels (e.g. solvers)
- Cache-aware data storage schemes to achieve good cache-hit ratios
- Computation-to-processor topology mapping schemes to minimize complex processor communication

Examples of multi-physics applications



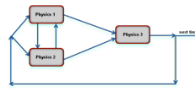
Mathematical Framework

Background

- Simulations of multi-physics problems are typically based on coupling of single-physics software modules
- UQ capabilities can be introduced in the single-physics modules (intrinsic uncertainties) but the overall uncertainties are coupled

A Mathematical Description

Consider the simple linear multi-physics feedback scenario



feed-back system

$$\begin{pmatrix} L_{11}(\omega_1) & L_{12} & L_{13} \\ L_{21} & L_{22}(\omega_2) & L_{23} \\ L_{31} & L_{32} & L_{33}(\omega_3) \end{pmatrix}\begin{pmatrix} u_1(\omega_1,\omega_2,\omega_3) \\ u_2(\omega_1,\omega_2,\omega_3) \\ u_3(\omega_1,\omega_2,\omega_3) \end{pmatrix} = \begin{pmatrix} f_1(\omega_1) \\ f_2(\omega_2) \\ f_3(\omega_3) \end{pmatrix}$$

Assumptions:
- uncertainties arise only in the intra-physics modules
- interactions are linear
- controllable/observable forcing terms have the same aleatory uncertainties as the intra-physics operator

Mathematical structures of the system can be used to accurately, stably, and efficiently perform UQ analysis.

Example: Processing of modules (i.e. propagation of uncertainties) based on relative strengths of the component operators

- Process ordering for more stable propagation and better parallelism

$$\begin{pmatrix} L_{11}(\omega_1) & L_{12} & L_{13} \\ L_{21} & L_{22}(\omega_2) & L_{23} \\ L_{31} & L_{32} & L_{33}(\omega_3) \end{pmatrix} \approx \begin{pmatrix} L_{11}(\omega_1) & \varepsilon & L_{13} \\ \varepsilon & L_{22}(\omega_2) & \varepsilon \\ L_{31} & \varepsilon & L_{33}(\omega_3) \end{pmatrix} \sim \begin{pmatrix} L_{22}(\omega_2) & 0 & 0 \\ 0 & L_{11}(\omega_1) & L_{13} \\ 0 & L_{31} & L_{33}(\omega_3) \end{pmatrix}$$

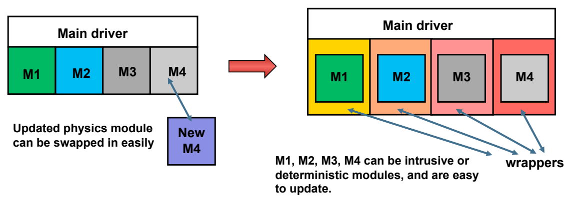
- can lead to reduced error propagation
- can permit loosely synchronized UQ analysis of subsystems

$$\left\|(L_{22})^{-1}\varepsilon\right\| << \left\|(L_{22})^{-1}\right\| \quad \text{and} \quad \left\|\begin{pmatrix} L_{11} & L_{13} \\ L_{31} & L_{33} \end{pmatrix}^{-1}\begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}\right\| << \left\|\begin{pmatrix} L_{11} & L_{13} \\ L_{31} & L_{33} \end{pmatrix}^{-1}\right\|$$

$$\Rightarrow \begin{bmatrix} L_{22}(\omega_2) & \epsilon & \epsilon \\ \epsilon & L_{11}(\omega_1) & L_{13} \\ \epsilon & L_{31} & L_{33}(\omega_3) \end{bmatrix}\begin{pmatrix} \tilde{u}_2(\omega_2,\epsilon) \\ \tilde{u}_1(\omega_1,\epsilon,\omega_3) \\ \tilde{u}_3(\omega_1,\epsilon,\omega_3) \end{pmatrix} = \begin{pmatrix} f_2(\omega_2) \\ f_1(\omega_1) \\ f_3(\omega_3) \end{pmatrix}$$

- Solution components do not depend on intrinsic uncertainties of weakly coupled modules

Computational Infrastructure & Software



- **Desirable features in a multi-physics simulation development framework:**
  - **plug-and-play modules**
  - **rich set of common services (e.g. solvers, parallelization)**
  - **well-defined and easy-to-use interfaces**
  - **allow rapid multi-physics code development with state-of-the art physics modules**
- **QUESTION: is such a framework feasible when uncertainties are also embedded in the simulation.**
- **One objective in the project is to design and implement such a software infrastructure.**
- **Functions of the software infrastructure**
  - **support multiple UQ methods (e.g. PCE, sampling)**
  - **manage uncertainties within individual modules**
  - **bridge uncertainty information across modules**
  - **track uncertainty flow during simulation**
  - **provide tools for non-intrusive UQ methods**
  - **facilitate task allocation/scheduling on HPCs**
  - **perform on-the-fly UQ, e.g. dimension reduction**
  - **provide check-pointing for fault tolerance**

Intrusive/Non-Intrusive Coupling - Example

We use the following test problem to illustrate a forward uncertainty propagation method.

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} - Ku;\ 0 \le x \le 1;\ 0 \le t \le T$$

$$u(0,t) = u(1,t) = 0, t \ge 0$$

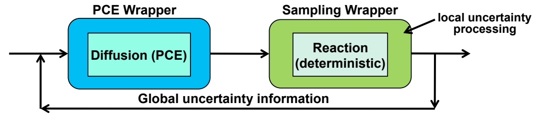
$$u(x,0) = f(x), 0 \le x \le 1$$

- **D and K are second order random variables**
- **D = [0.0001, 0.001] with uniform distribution**
- **K = [0.1, 0.5] with uniform distribution**
- **time-step = 0.01**
- **Track root-mean-squared errors at T=2**

- **Operator splitting scheme: (n = time step)**
  - **diffusion module: PCE (polynomial chaos)**
  - **reaction module: sampling**

$$\begin{bmatrix} \frac{\partial u}{\partial t} - D\frac{\partial^2}{\partial x^2} & 0 \\ \text{I.C.} & \frac{\partial u}{\partial t} + K \end{bmatrix}\begin{bmatrix} u_D^{n+1/2} \\ u_R^{n+1} \end{bmatrix}$$

- **Global uncertainty representation: PCE**
- **Global uncertainty managed by software framework**

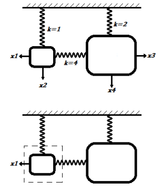


PCE Order	Pure PCE rms error	Pure PCE Time (s)	Hybrid rms error	Hybrid Time (s)
1	4.9e-4	8.0	7.8e-3	10.5
2	5.7e-4	10.5	4.8e-5	14.5
3	5.7e-4	13.5	4.0e-5	23.5
4	5.7e-4	16.5	8.9e-5	29.5
5	5.7e-4	20.5	8.6e-5	40.0
6	5.7e-4	25.5	8.7e-5	46.0

Discussions:
- **The hybrid method attains smaller errors than the pure PCE method when p is sufficiently large. This may be due to the use of analytic solution.**
- **The higher accuracy can be attained for pure PCE with smaller time step.**
- **Hybrid methods take longer, but**
  - **Pure PCE may need smaller time step**
  - **Multi-species problem will need special ODE solvers (the reason for operator splitting)**
  - **Some more code optimization may be possible (in the non-intrusive module)**

Iterative Coupling Under Uncertainty - Example

We use a simple mechanics problem to illustrate an algorithm to study coupled problems solving iteratively decoupled systems



1) Assuming the masses are "uncertain", the governing equations read

$$M\frac{d^2}{dt^2}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + K\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0$$

with $M = M(s)$ and s a uniform r.v.

2) When considering a P-order PCE expansion, the solution of a coupled *4x(P+1)* system is required $x_\ell(s,t) = \sum_{j=0}^{P} X_\ell^{(j)}(t)\pi_j(s)$

3) In order to express the coupling explicitly we write the stiffness as

$$M\frac{d^2}{dt^2}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + (\Lambda + C)\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0$$

with Λ a diagonal matrix

4) We can build an iterative solution procedure, defining $x_\ell(s,t) = \hat{x}_\ell(s,t) + \Delta x_\ell(s,t)$ with the m-th update as

$$(M\frac{d^2}{dt^2} + \Lambda)\begin{bmatrix} \Delta\hat{x}_1 \\ \Delta\hat{x}_2 \\ \Delta\hat{x}_3 \\ \Delta\hat{x}_4 \end{bmatrix}^m = -C\begin{bmatrix} \Delta\hat{x}_1 \\ \Delta\hat{x}_2 \\ \Delta\hat{x}_3 \\ \Delta\hat{x}_4 \end{bmatrix}^{m-1} - C\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \end{bmatrix}$$

5) A PCE expansion, equivalent to (2) requires "only" the solution or 4 uncoupled problem iteratively

6) For the case in the figure, we assume:

$$M(s) = \begin{pmatrix} g(s) & & & \\ & g(s) & & \\ & & 1 & \\ & & & 1 \end{pmatrix} \Lambda = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 4 & \\ & & & 4 \end{pmatrix} C = \begin{pmatrix} 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 2 \\ -2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \end{pmatrix}$$

PCE Order	Coupled Time (s)	Uncoupled Time (s)	Uncoupled time per Iteration
4	6.36	56.2	0.884
6	12.7	112.5	0.887
8	19.0	167.9	0.884
10	25.2	223.7	0.889
12	31.6	280.9	0.827

Conclusions

Uncertainty Representation

- We studied the feasibility of using of wrappers around single-physics codes with embedded UQ methods to represent globally the uncertainties using PC expansions

Iterative solution of coupled UQ problems

- We investigated the use of an iterative coupling algorithm to propagate uncertainty through uncoupled solutions of single-physics UQ problems.

Towards more realistic problems

- The initial steps have focused on simple model problems, we are currently considering more realistic PDE-based multi-physics problems

ASCR PI Meeting 2011 – Washington DC