SAND2019-2005C

Large-Scale Uncertainty Propagation via Overlapping Domain Decomposition

K. T. Carlberg[1] S. Guzzetti[2] M. Khalil[1,†] K. Sargsyan[1]

[1] Sandia National Laboratories [2] Emory University

[†] mkhalil@sandia.gov

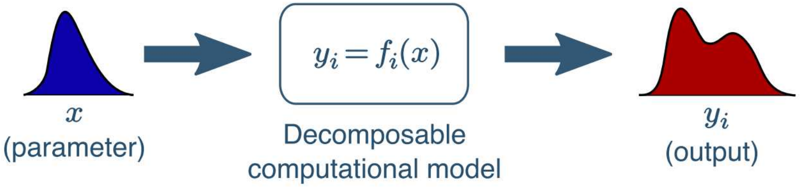
SIAM Conference on Computational Science and Engineering
Spokane, Washington, USA
February 25 - March 1, 2019

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC.,a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Overview

UQ in Decomposable Systems

- Objective: UQ Solvers for decomposable systems at extreme scale



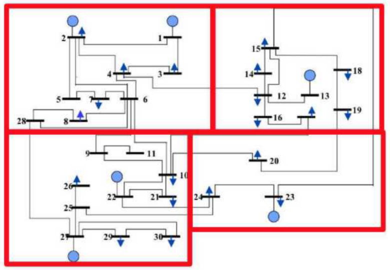
$y_i = f_i(x)$

x
(parameter)

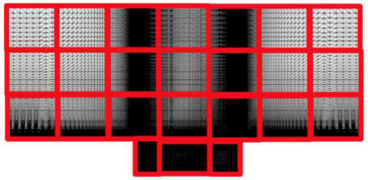
Decomposable
computational model

y_i
(output)

- Decomposable systems characterize many complex scientific/engineering applications
  - Multiple subsystems
  - Subsystems:
    - Characterized by different physics or scales
    - Different discretization schemes and deterministic solution techniques
    - Tailored uncertainty analysis schemes



Source: Lesieutre, Pinar, and Roy. "Power system extreme event detection:The vulnerability frontier." Hawaii International Conference on System Sciences, Proceedings of the 41st Annual. IEEE, 2008.



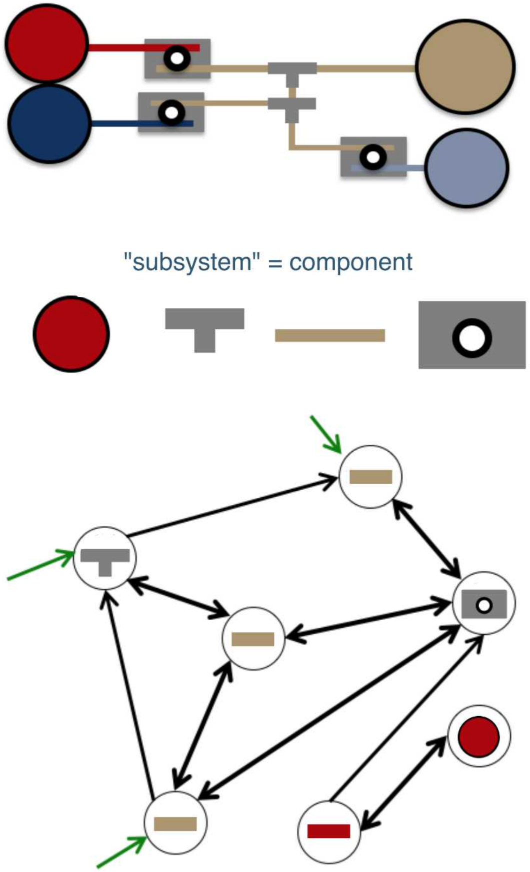
Source: nnsa.energy.gov

Challenges for UQ in Decomposable Systems

- For extreme-scale decomposable systems, high-fidelity simulation and UQ are typically achievable at the subsystem level, but intractable for the full system:
    - Long full-system simulation times
    - Limits on the problem size that can be reliability simulated
    - Integrating subsystem models possessing different physics or scales
- Full-system surrogate models, which are typically employed to make UQ tractable, may not be possible to construct, as they rely on a training set of these (possibly infeasible) simulations
- Proposed approach performs extensive UQ at the (tractable) subsystem level while propagating coupling information using new techniques inspired by domain decomposition

UQ Solvers for Decomposable Systems

- Idea: Model as subsystem network
  - ✦ UQ at subsystem level
  - ✦ System-level coordination for compatibility
  - \+ Promotes subsystem independence
  - \+ Focuses computational effort locally
  - \+ Tailored uncertainty analysis, discretizations, and solvers can be applied to each subsystem separately
- Method: DDUQ
  - ✦ Assigns each subsystem to a node in a directed graph, and coupling random variables to edges
  - ✦ New domain decomposition methods for UQ to enforce statistical compatibility
  - \+ Full-system UQ from subsystem analyses



Propagating Uncertainties in Networks

- Goal: propagate uncertainties from subsystems to the full system
- Current state of the art:
  - Simplified probability representations [Gu, Renaud, Penninger ,2006; Du and Chen, 2005; Chiralaksanakul and Mahadevan, 2005]
  - Simplifying assumptions on network topology: Feed-forward systems [Amaral, Allaire, Willcox, 2014]
  - Two-component systems [Arnst, Ghanem, Phipps, Red-Horse, 2014; Parks, Bochev, Lehoucq, 2008; Olson, D., Bochev, P., Luskin, M., Shapeev, 2014; Constantine, Phipps, Wildey, 2014]
  - Enforce compatibility for deterministic solves/samples [Liao and Willcox, 2015]
  - Enforce compatibility (weakly) in distributions [Liao and Willcox, 2015; Ghoreishi, Allaire, 2017]
  - Domain Decomposition for SPDEs [Sarkar, Benabbou, Ghanem, 2009; Desai, Khalil, Pettit, Poirel, Sarkar, 2018; Zhang, Babaee, Karniadakis, 2018]
- Proposed framework:
  - Rigorous: Enforces strong compatibility (matching random variables)
  - Scalable: Handles multiple components
  - General: Components as black-boxes (beyond stochastic PDEs)
  - Flexible: Tackles arbitrary networks with general inter-connectivity
  - Parallelizable: exhibits excellent weak parallel scalability

Representing Uncertainties

- We represent uncertainties probabilistically using random variables (RVs)
- Use functional representations to preserve coupling of RVs across the network
- Strategy: Use well-established Polynomial Chaos Expansion
  - ✦ Represent model inputs/outputs as RVs
  - ✦ Construct PCEs for uncertain parameters
  - ✦ Obtain PCEs for model outputs using *Non-intrusive Spectral Projection*:
    - ■ *Sampling*-based
    - \+ Relies on black-box utilization of the subcomponent computational model
    - ■ Evaluate projection integrals *numerically* using
      - – Quadrature/Sparse-Quadrature methods
        - \+ Fast convergence
  - ✦ Given a *germ* $\boldsymbol{\xi}(\omega) = \{\xi_1, \cdots, \xi_n\}$ – a set of *i.i.d.* RVs
  - ✦ Any RV in $L^2(\Omega, \mathfrak{S}(\boldsymbol{\xi}), P)$ can be written as a PCE:

$$u(\omega) \simeq \sum_{k=0}^{P} u_k \Psi_k(\boldsymbol{\xi}(\omega))$$

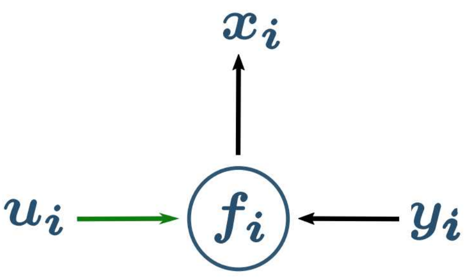
- – u_k are coefficients (deterministic)
- – $\Psi_k(\cdot)$ are functions orthogonal w.r.t. $p(\boldsymbol{\xi})$

DDUQ Formulation

- Subsystem uncertainty propagation of PCE coefficients

$$\boldsymbol{x_i} = \boldsymbol{f_i}\left(\boldsymbol{y_i}, \boldsymbol{u_i}\right)$$

  - Exogenous inputs: $\boldsymbol{u_i}$ (Vector of PCE coefficients)
  - Endogenous inputs: $\boldsymbol{y_i}$ (Vector of PCE coefficients)
  - Outputs: $\boldsymbol{x_i}$ (Vector of PCE coefficients)
  - Uncertainty propagation operator: $\boldsymbol{f_i}$



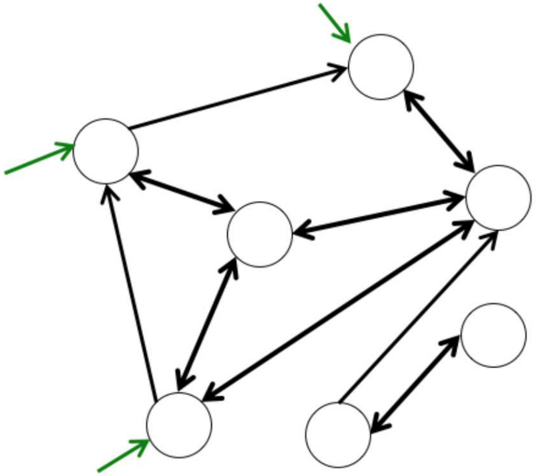
- Full-system uncertainty propagation

$$\boldsymbol{x} = \boldsymbol{f}\left(\boldsymbol{y}, \boldsymbol{u}\right)$$

  - Adjacency matrix satisfies $\boldsymbol{y} = \boldsymbol{I_x^y}\,\boldsymbol{x}$

$$\boldsymbol{x} = \boldsymbol{f}\left(\boldsymbol{I_x^y}\,\boldsymbol{x}, \boldsymbol{u}\right)$$

fixed-point problem



DDUQ Solvers

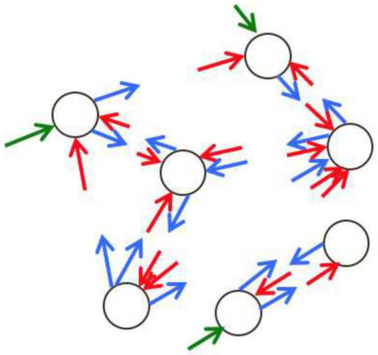
$$\boxed{\boldsymbol{x} = \boldsymbol{f}\left(\boldsymbol{I}^{\boldsymbol{y}}_{\boldsymbol{x}}\,\boldsymbol{x}, \boldsymbol{u}\right)}$$

- Jacobi iterations with relaxation

$$\bar{\boldsymbol{x}} = \boldsymbol{f}\left(\boldsymbol{I}^{\boldsymbol{y}}_{\boldsymbol{x}}\,\boldsymbol{x}^{(k)}, \boldsymbol{u}\right)$$

$$\boldsymbol{x}^{(k+1)} = \omega\bar{\boldsymbol{x}} + (1-\omega)\,\boldsymbol{x}^{(k)}$$

  - ✦ Independent subsystems at each iteration

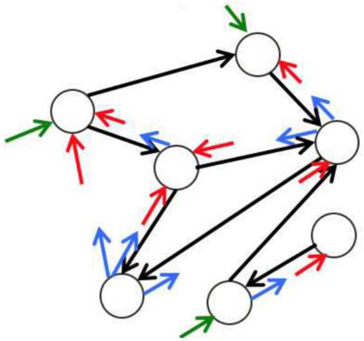


- Gauss–Seidel iterations with relaxation

$$\bar{\boldsymbol{x}} = \boldsymbol{f}\left(\omega\left[\boldsymbol{P}^{(y_i)}_{(p_i)}\right]^T\boldsymbol{L}\left[\boldsymbol{P}^{(x_i)}_{(p_i)}\right]\bar{\boldsymbol{x}} + \left[\boldsymbol{P}^{(y_i)}_{(p_i)}\right]^T\left((1-\omega)\,\boldsymbol{L} + \boldsymbol{D} + \boldsymbol{U}\right)\left[\boldsymbol{P}^{(x_i)}_{(p_i)}\right]\boldsymbol{x}^{(k)}, \boldsymbol{u}\right)$$

$$\boldsymbol{x}^{(k+1)} = \omega\bar{\boldsymbol{x}} + (1-\omega)\,\boldsymbol{x}^{(k)}$$

  - ✦ Feed-forward network (directed acyclic graph) at each iteration
  - ✦ Permutation matrix $\left[\boldsymbol{P}^{(y_i)}_{(p_i)}\right]$ determines DAG



A priori error bound

- PCE truncation within the network may lead to errors in the computed solution
- Let's define $\boldsymbol{g} : (\boldsymbol{\chi}, \boldsymbol{v}) \mapsto \boldsymbol{f}(\boldsymbol{I}_{\boldsymbol{x}}^{\boldsymbol{y}} \boldsymbol{\chi}, \boldsymbol{v})$ such that overall problem becomes

$$\boldsymbol{x} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u})$$

- The 'truth' model associated with a high-fidelity representation of the network (e.g. high-order PCE) leads to the following fixed-point problem

$$\boldsymbol{x}_\star = \boldsymbol{g}_\star(\boldsymbol{x}_\star, \boldsymbol{u}).$$

  - $\boldsymbol{x} \in \mathbb{R}^{\mathsf{x}}$, $\boldsymbol{x}_\star \in \mathbb{R}^{\mathsf{x}_\star}$, with $\mathsf{x}_\star > \mathsf{x}$
  - Prolongation operator $\boldsymbol{I}_H^h : \mathbb{R}^{\mathsf{x}} \to \mathbb{R}^{\mathsf{x}_\star}$
  - Restriction operator $\boldsymbol{I}_h^H : \mathbb{R}^{\mathsf{x}_\star} \to \mathbb{R}^{\mathsf{x}}$ such that $\boldsymbol{I}_h^H \boldsymbol{I}_H^h = \boldsymbol{I}$
- Under mild assumptions, the error can be bounded as

$$\|\boldsymbol{I}_h^H \boldsymbol{x}_\star - \boldsymbol{x}\| \leq \frac{1}{1 - L_{\boldsymbol{g}_\star}^H} \|\boldsymbol{I}_h^H \boldsymbol{g}_\star(\boldsymbol{I}_H^h \boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{x}\|$$

  - $L_{\boldsymbol{g}_\star}^H < 1$ is the restricted Lipschitz constant of the fixed-point mapping $\boldsymbol{g}_\star$
- *Self-consistent* UQ methods: The right-hand side of above inequality will be zero
  - Non-intrusive Spectral Projection with fixed quadrature rule

2D stationary stochastic nonlinear heat equation

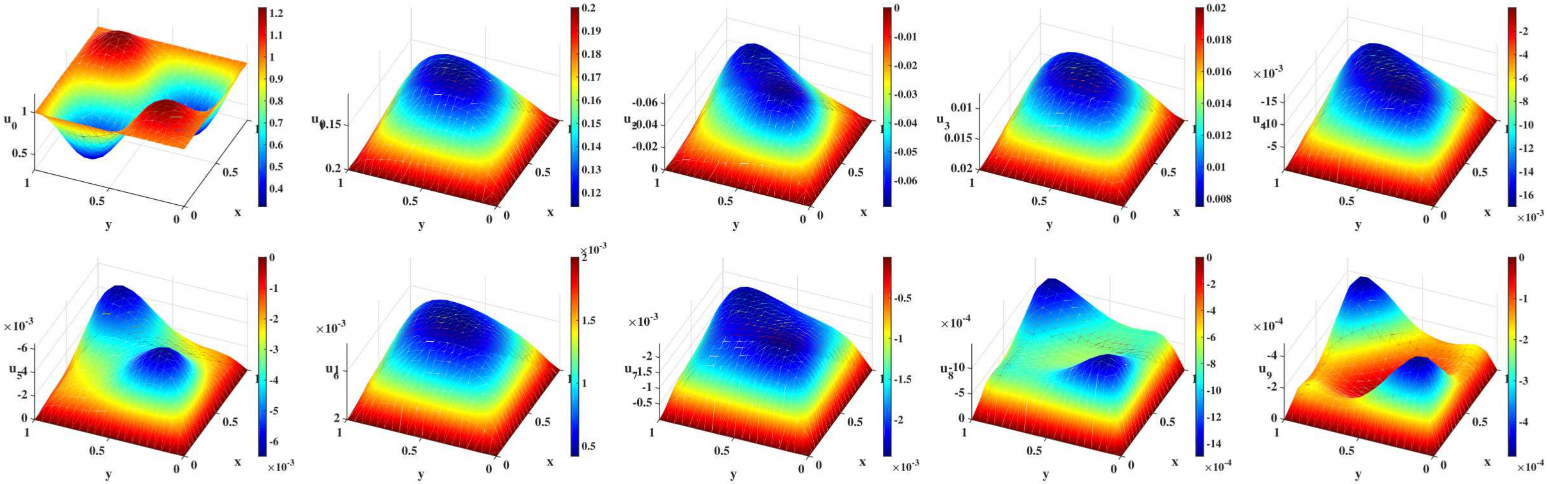
- Two-dimensional stationary nonlinear heat equation with uncertain diffusion and boundary condition

$$-\nabla u + \left(e^{\mu u} - 1\right) = 10\sin\left(2\pi x_1\right)\sin\left(2\pi x_2\right)\ , \qquad x \in \Omega := (0,1)^2$$

$$u_\Gamma = T_1(\xi) = 1.0 + 0.2\xi_1 + 0.02\left(\xi_1^2 - 1\right) + 0.002\left(\xi_1^3 - 3\xi_1\right)$$
$$\mu = T_2(\xi) = 1.0 + 0.2\xi_2 + 0.02\left(\xi_2^2 - 1\right) + 0.002\left(\xi_2^3 - 3\xi_2\right)$$

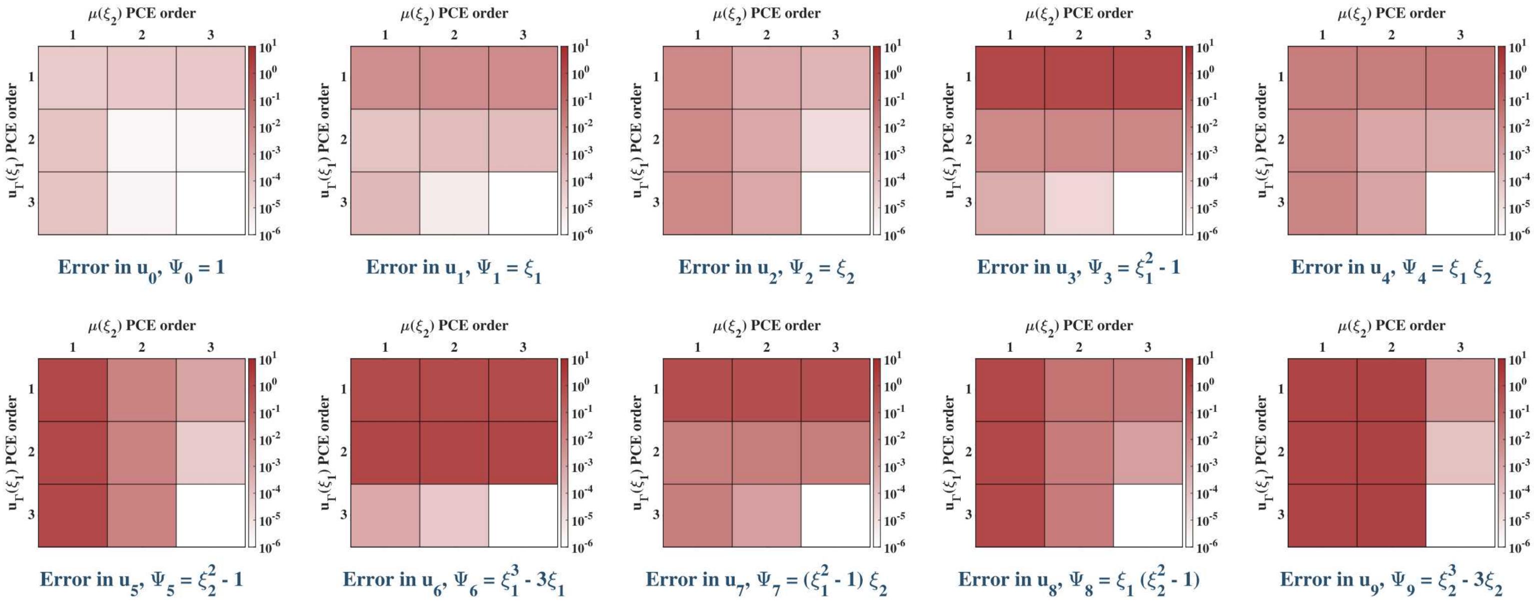
  - ✦ $\xi = (\xi_1, \xi_2)$ iid standard normal random variables
  - ✦ The basis functions $\{\Psi_k\}$ are 2D Hermite polynomials of order at most $p = 3$

- Output PCE coefficients (10 coefficients for a 2nd-order 2D PCE)):



Error analysis

- Examine the effect of using a lower fidelity input on the system response
- Utilize lower-order (truncated) input parameter PCEs
- Normalized L_2 error norms for the 3rd-order output PCE coefficients:



Error in u_0, $\Psi_0 = 1$

Error in u_1, $\Psi_1 = \xi_1$

Error in u_2, $\Psi_2 = \xi_2$

Error in u_3, $\Psi_3 = \xi_1^2 - 1$

Error in u_4, $\Psi_4 = \xi_1 \xi_2$

Error in u_5, $\Psi_5 = \xi_2^2 - 1$

Error in u_6, $\Psi_6 = \xi_1^3 - 3\xi_1$

Error in u_7, $\Psi_7 = (\xi_1^2 - 1)\, \xi_2$

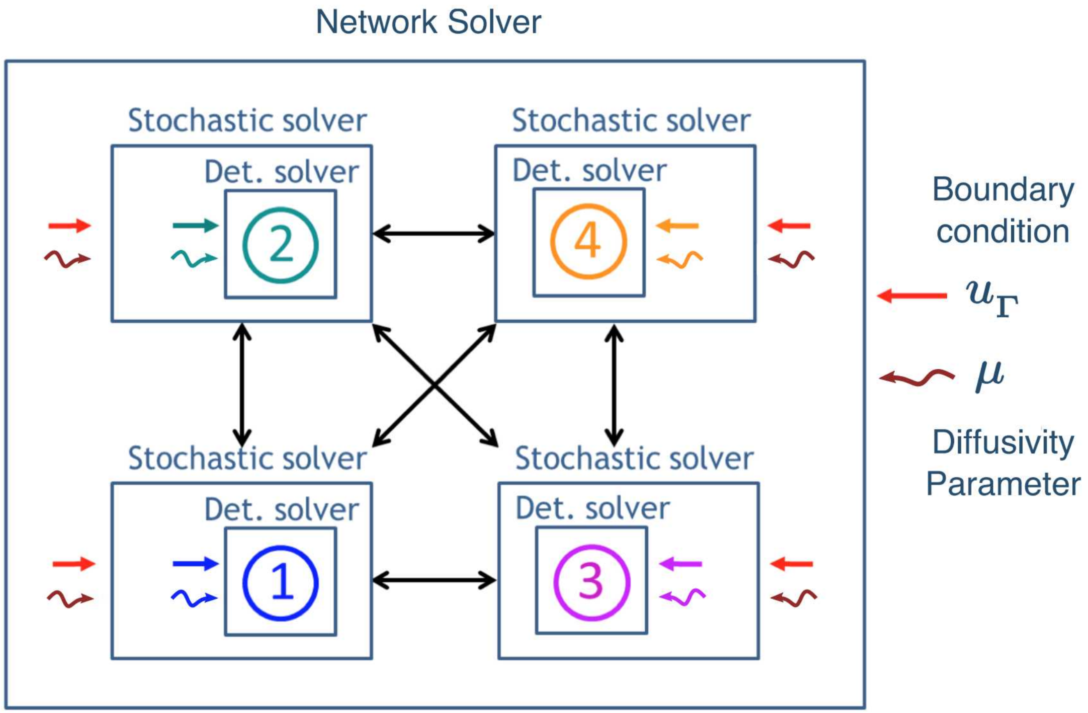
Error in u_8, $\Psi_8 = \xi_1\, (\xi_2^2 - 1)$

Error in u_9, $\Psi_9 = \xi_2^3 - 3\xi_2$

- Observations:
  - ✦ The mean response (u_0 coefficient) converges rapidly: mean response of a system normally depends on lower order moments of the uncertain inputs.
  - ✦ Higher-order output PCE coefficients: strong correlation between input order and error in the output PCE coefficient multiplying polynomials of corresponding seed

Workflow in Network Solver

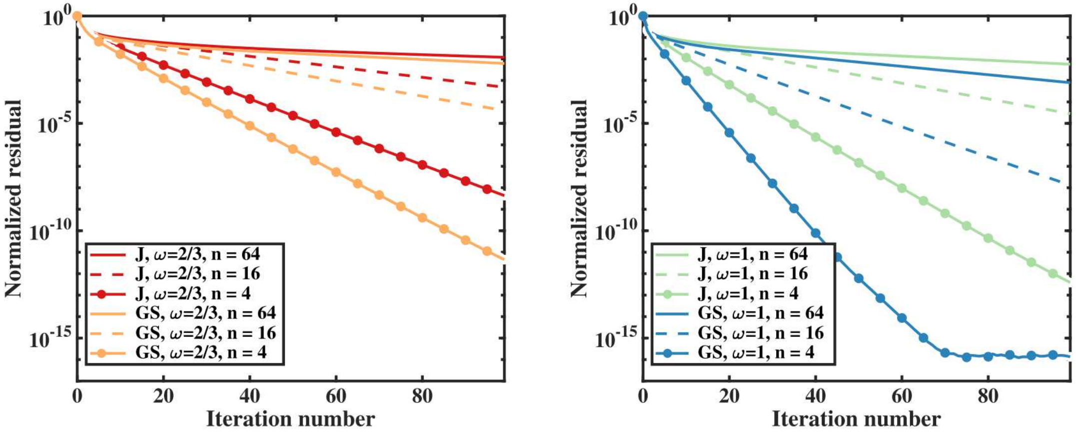
- Schematic of workflow for network solver in tackling the 2D heat equation: 4 subdomains (subcomponents)



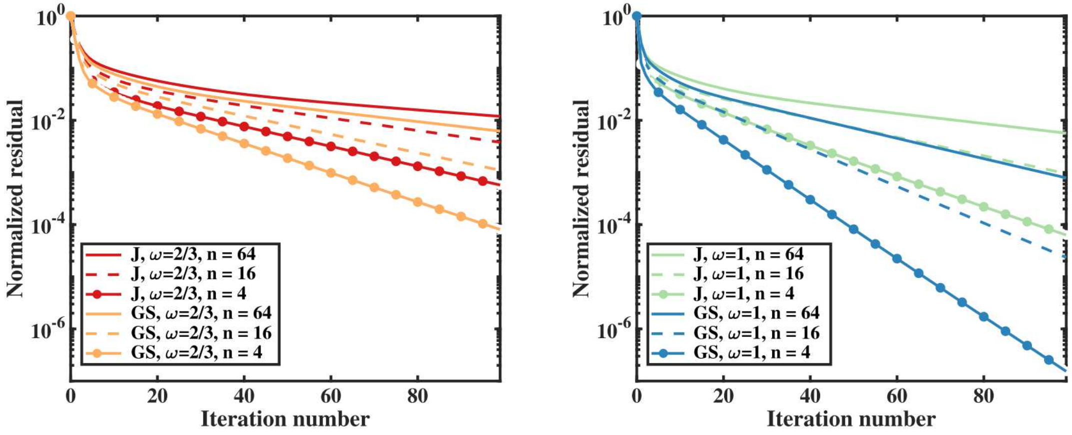
- Modularity: network, stochastic, deterministic solvers independent
- Flexibility: Tailored deterministic solvers for each subsystem
- Extensiblity: new application requires new deterministic solver only

Convergence results

- Examine convergence of PCE coefficients (unknowns)
- Weak scalability: The subdomain problem size (i.e. number of subdomain-level finite-element nodes) fixed while increasing the number of subdomains

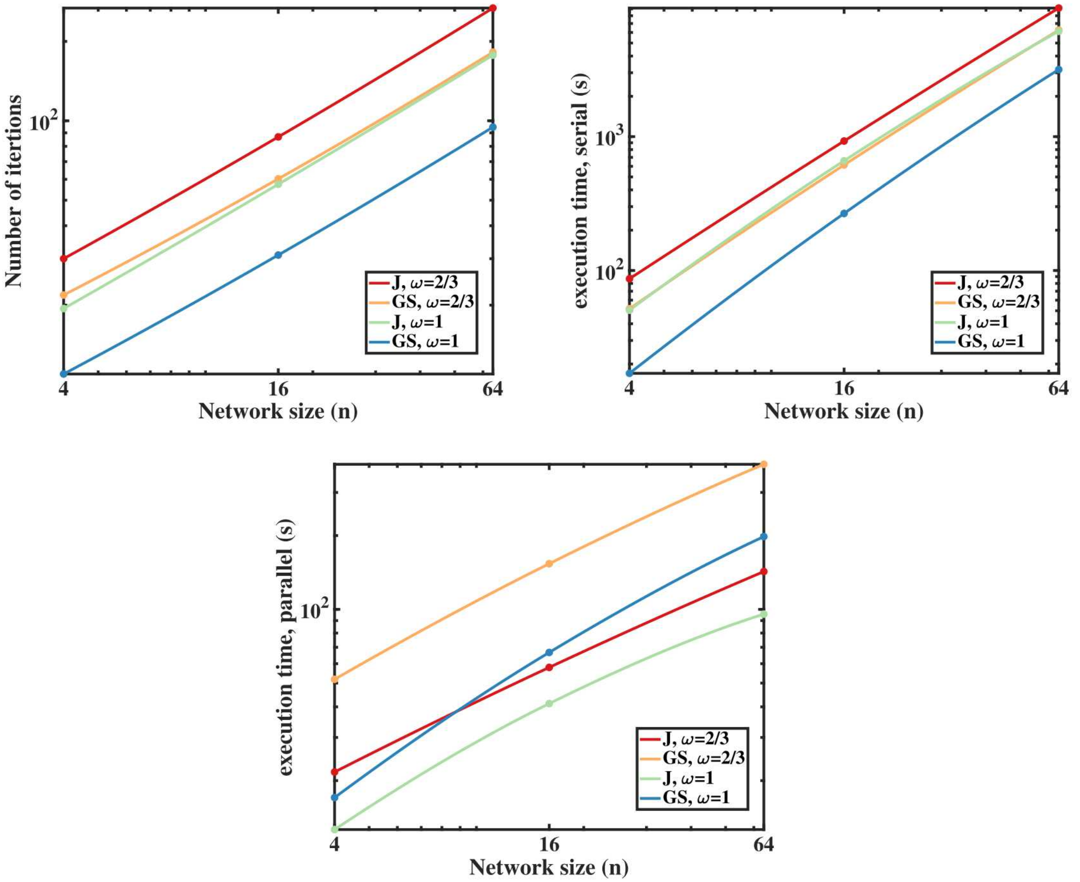


- Strong scalability: The global problem size (i.e. number of global finite-element nodes) fixed while increasing the number of subdomains



Weak scalability results

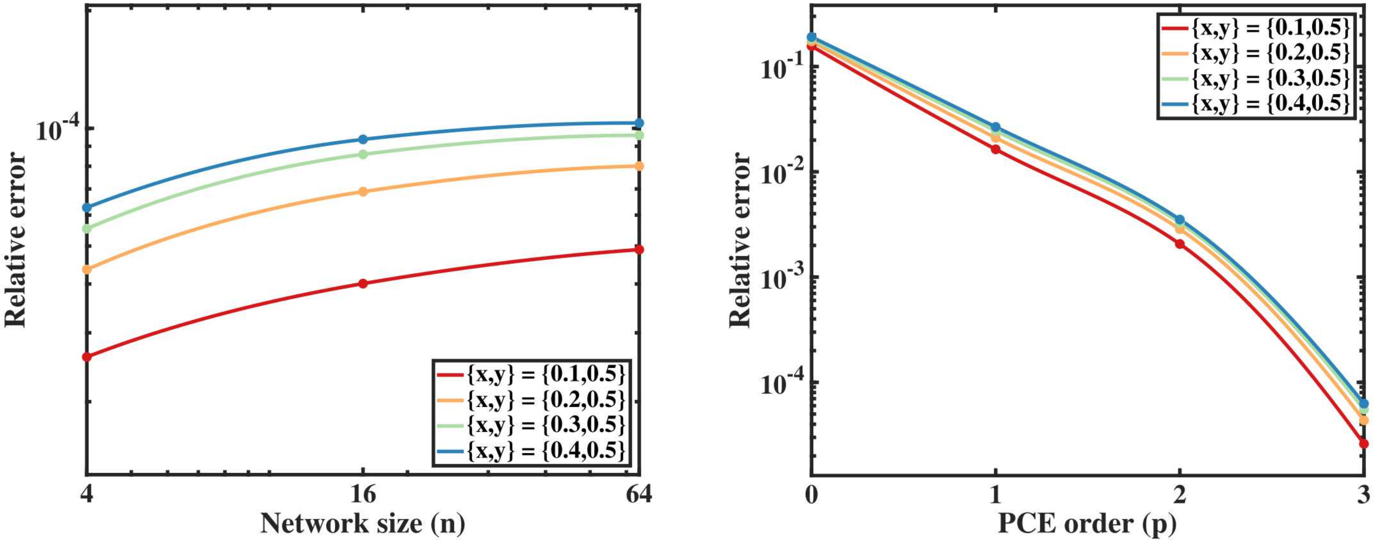
- All timings are obtained by performing calculations on an Intel(R) Xeon(R) Core(TM) i7-5557U CPU @ 3.10GHz with 16 GB RAM



- Observations:
  - \+ Iteration count *weakly* dependent on network size
  - \+ Jacobi faster (for parallel implementations) than Gauss–Seidel

Error Analysis

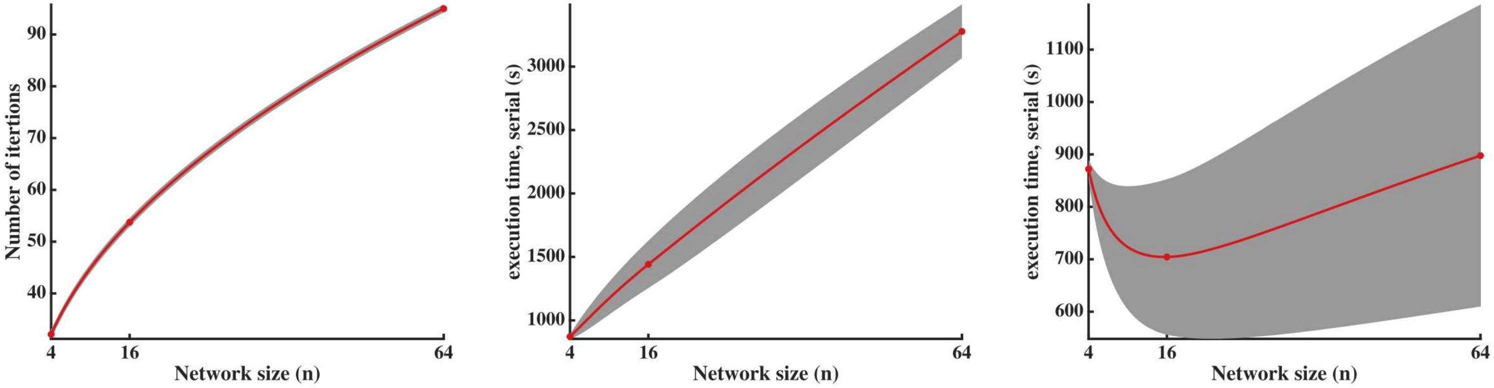
- Normalized error for solution (PC coefficients) at isolated physical locations for
  - ✦ varying network size (fixed PCE-order of unknowns)
  - ✦ varying PCE-order representation of unknowns (fixed network size of 4)



- Observations:
  - + Errors relatively small
  - ✦ Errors increase with (1) distance from boundary, (2) network size, (3) lower-fidelity PCE representation

Gauss-Seidel Iterations: Effect of Permutation Matrix

- Strong scalability results for Gauss-Seidel iterations with random permutation matrix (10 Monte Carlo samples)



- Observations:
  - \+ Minimal impact on the total number of iterations
  - \+ Weak impact on the serial execution time
  - \- Strong impact on the parallel execution time
    - \+ Extreme gains to be had in using optimal permutation matrix

Summary

- Presented a framework for propagation of uncertainties through decomposable systems
- Network-based UQ with coupling information propagated using new techniques inspired by domain decomposition
- Proposed methodology extends existing suite of techniques:
    - Rigorous: Enforces strong compatibility (matching random variables)
    - Scalable: Handles multiple components
    - General: Components as black-boxes (beyond stochastic PDEs)
    - Flexible: Tackles arbitrary networks with general inter-connectivity
    - Parallelizable: exhibits excellent weak parallel scalability
- Application: 2D nonlinear stochastic static heat equation with uncertain parameters and boundary conditions