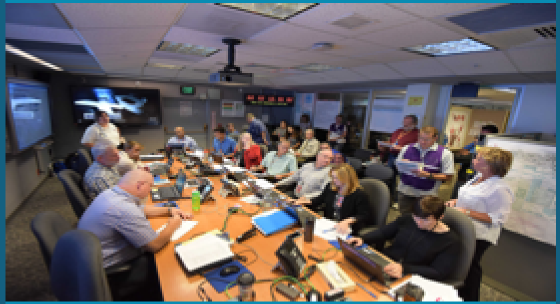
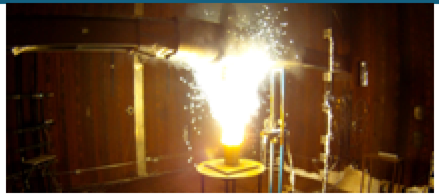
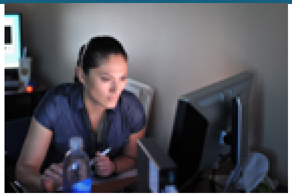
Sandia National Laboratories

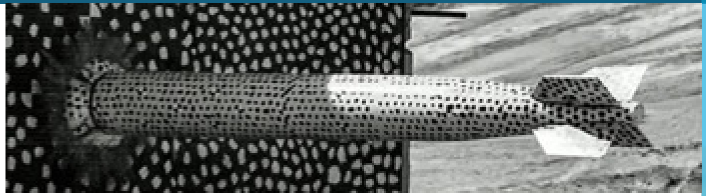
SAND2020-13306C

Comprehensive Uncertainty Quantification by Solving PDF Equations using Lagrangian Methods









Hemanth Kolla, Reese Jones, Julia Plews, Ross Wagnild
Sandia National Laboratories

WCCM-ECCOMAS 2020

MS187: Improving Predictive Capabilities Through Model Error Quantification.

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.

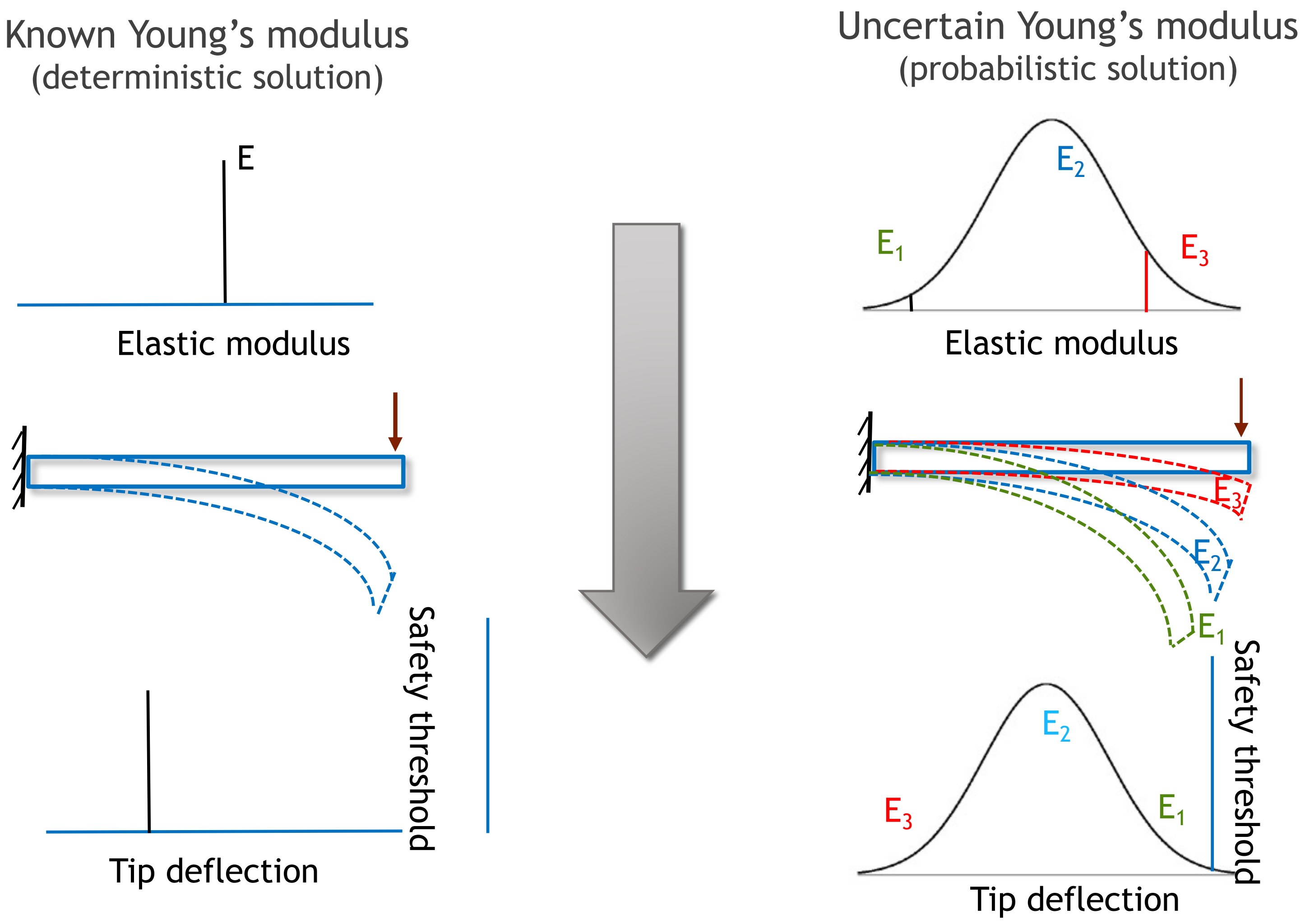
Acknowledgments

- Funding from Laboratory Directed Research & Development (LDRD) program at Sandia National Labs.

- Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & 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.

- This presentation describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

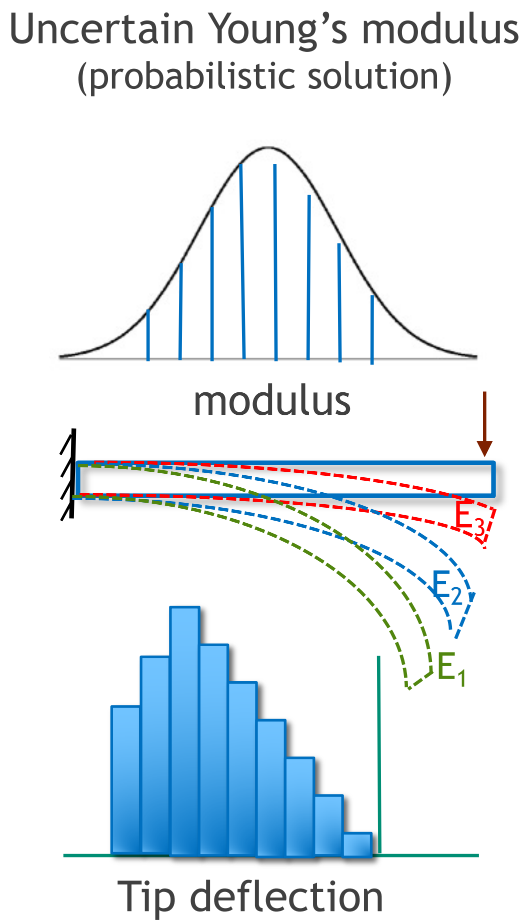
Forward UQ for Computational Mechanics



- Model parameter uncertainty represented by a distribution, $p(E)$. Presumed known (inverse UQ, expert knowledge, published data).
- **Forward UQ** is quantify statistics of outcomes of a physical system due to input parameter uncertainties, $p(E)$.
- **Challenging** when systems are non-linear, and/or the number of uncertain parameters is large
- **Prohibitive** for high-fidelity/expensive computational systems but this is where it is needed most.

Problem: Forward UQ State-of-art

- Focus only on global observables of interest (e.g. “tip displacement of the beam”).
- Get a **simplified/reduced representation** through surrogates or reduced order models of the computational system that is affordable for many samples but is **typically inaccurate**.
- Intrusive Polynomial Chaos: expand the PDE system with new dimensions representing uncertainty – **prohibitive coding burden**, closure issues and stiff equations.
- In general, most methods <u>require numerous forward solves</u> (on the order of 10^3-10^5 of either the reduced system or full system).

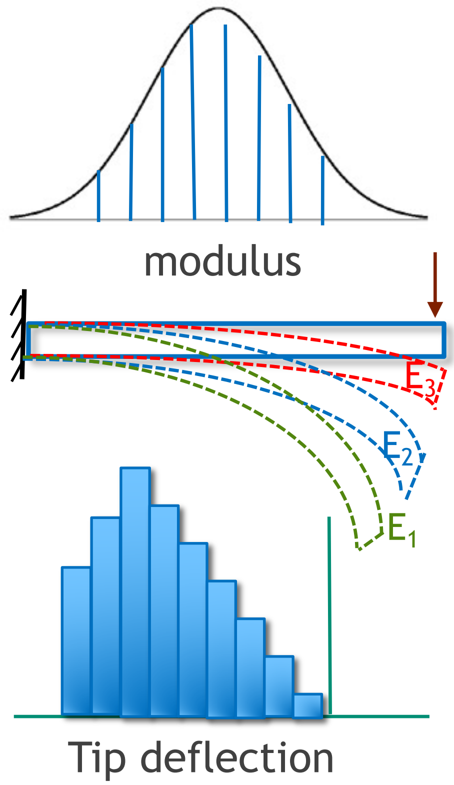


Existing methods are reductive, computationally cumbersome and do not integrate well with mature codes. Forward UQ is not wide spread as it should be.

Problem: Forward UQ State-of-art

- Focus only on global observables of interest (e.g. "tip displacement of the beam").
- Get a **simplified/reduced representation** through surrogates or reduced order models of the computational system that is affordable for many samples but is **typically inaccurate**.
- Intrusive Polynomial Chaos: expand the PDE system with new dimensions representing uncertainty – **prohibitive coding burden**, closure issues and stiff equations.
- In general, most methods require numerous forward solves (on the order of 10^3-10^5 of either the reduced system or full system).

Uncertain Young's modulus
(probabilistic solution)



Existing methods are reductive, computationally cumbersome and do not integrate well with mature codes. Forward UQ is not wide spread as it should be.

Our proposal: Comprehensive yet computationally efficient UQ deployed directly in HPC codes.

Proposed Solution: Solve PDF equations

Comprehensive UQ (obtain full probabilistic field information), computationally efficient (without requiring numerous forward solves).

What: Solve directly for joint PDF evolution equations:

- The solution variables (e.g. displacement, velocity) have a distribution purely due to physics parameter uncertainty (e.g. plasticity model, diffusivity).
- The governing equations (PDEs) determine evolution equations (exact but unclosed) for the joint PDF.

How:

- The PDF equations can be solved efficiently using the **original** PDE machinery/code with marginal cost.
- Allows retaining details of numerics, domain/geometry, boundary conditions.

Why:

- Not seek surrogates which can be reductive/inaccurate (especially at the response extremes) and case/configuration specific.
- More detailed UQ information (beyond the basic statistics of global QOIs) since a distribution is associated with every location and time in the system.
- Handle non-linearities better than intrusive methods (e.g. PCE).

Background: Governing equations with uncertain parameters

- **Solid Mechanics:** Consider transient visco-elastic problem (small strain limit):
  - Computational model: $\rho\ddot{\boldsymbol{u}}^p = \nabla\cdot\boldsymbol{\sigma} = \boldsymbol{\mathcal{R}}(\nabla\,\boldsymbol{u}^p, \nabla\,\boldsymbol{u}^e, \lambda)$
  - Solution vector $[\,\boldsymbol{u}^p,\ \boldsymbol{u}^e]$ has a distribution at any (x_i, t) due to uncertainty in inelastic material response properties λ (e.g., initial yield, hardening coeff, saturation modulus).
- **Fluid Mechanics:** Consider multi-species reacting flow problem:
  - Computational model: $D_t(\rho Y^i) = -\nabla\cdot J + \dot{\omega}^i = \boldsymbol{\mathcal{R}}(Y^i, \nabla Y^i, \lambda)$ D_t - substantial derivative

$$D_t(\rho c_P T) = -\nabla\cdot q + \dot{\omega}^T = \boldsymbol{\mathcal{R}}(T, \nabla T, \lambda)$$

  - The solution vector $[Y^i, T]$ has a distribution at any (x_j, t) due to uncertainty in chemical rate parameters λ (activation energy, pre-exponential factor, temperature exponent).
- In general, governing PDEs of a vector of solution variables $\boldsymbol{\phi}(\mathbf{x}, t)$ can be written in a form $\dfrac{\mathrm{D}\boldsymbol{\phi}}{\mathrm{D}t} = \boldsymbol{\mathcal{R}}(\boldsymbol{\phi}, \boldsymbol{\lambda})$
  - Right hand side, $\boldsymbol{\mathcal{R}}$, contains all physics; micro-scale properties parametrized by $\boldsymbol{\lambda}$ which are sources of uncertainty.

Background: PDF equations

- The PDF, $\boldsymbol{p}_{\boldsymbol{\phi}}(\boldsymbol{\Phi}; \mathbf{x}, t)$, has full field information of “statistics of outcomes”.
- A governing equation for the PDF can be derived directly from the underlying PDE:

$$\frac{\mathrm{D}\boldsymbol{\phi}}{\mathrm{D}t} = \boldsymbol{\mathcal{R}}(\boldsymbol{\phi}, \boldsymbol{\lambda}) \qquad \Longrightarrow \qquad \frac{\mathrm{D}\boldsymbol{p}_{\phi}}{\mathrm{D}t} = -\frac{\partial}{\partial \phi_i}\left[\boldsymbol{p}_{\boldsymbol{\phi}} \langle \boldsymbol{\mathcal{R}} | \phi_i\,, \Lambda; \boldsymbol{x}, t\rangle\right]$$

  - The derivation is exact [1,2], contains no approximations or assumptions.
  - Side detail: The derivation involves the joint PDF, $\boldsymbol{p}_{\boldsymbol{\phi\lambda}}(\boldsymbol{\Phi}, \Lambda; \mathbf{x}, t)$, that encapsulates the parameter uncertainty, $\boldsymbol{p}_{\lambda}(\boldsymbol{\Lambda})$.
- The PDF equation is:
  - “Unclosed”, even though its derivation is exact. Closure can be understood by thinking of the PDF equation as not a differential equation, but an integro-differential equation i.e. numerical solution requires information from **far away**.
  - Has large dimensionality. At each (x_j, t), the PDF is a multidimensional function (#dimensions = #$\boldsymbol{\phi}$ components).

[1] S.B. Pope, 1985, *Prog. Energy Combust Sci.*, vol.11, pp: 119-192.
[2] D. Venturi, D.M. Tartakovsky, A.M. Tartakovsky, G.E. Karniadakis, 2013, *J. Comp. Phys.*, vol.243, pp:323-343.

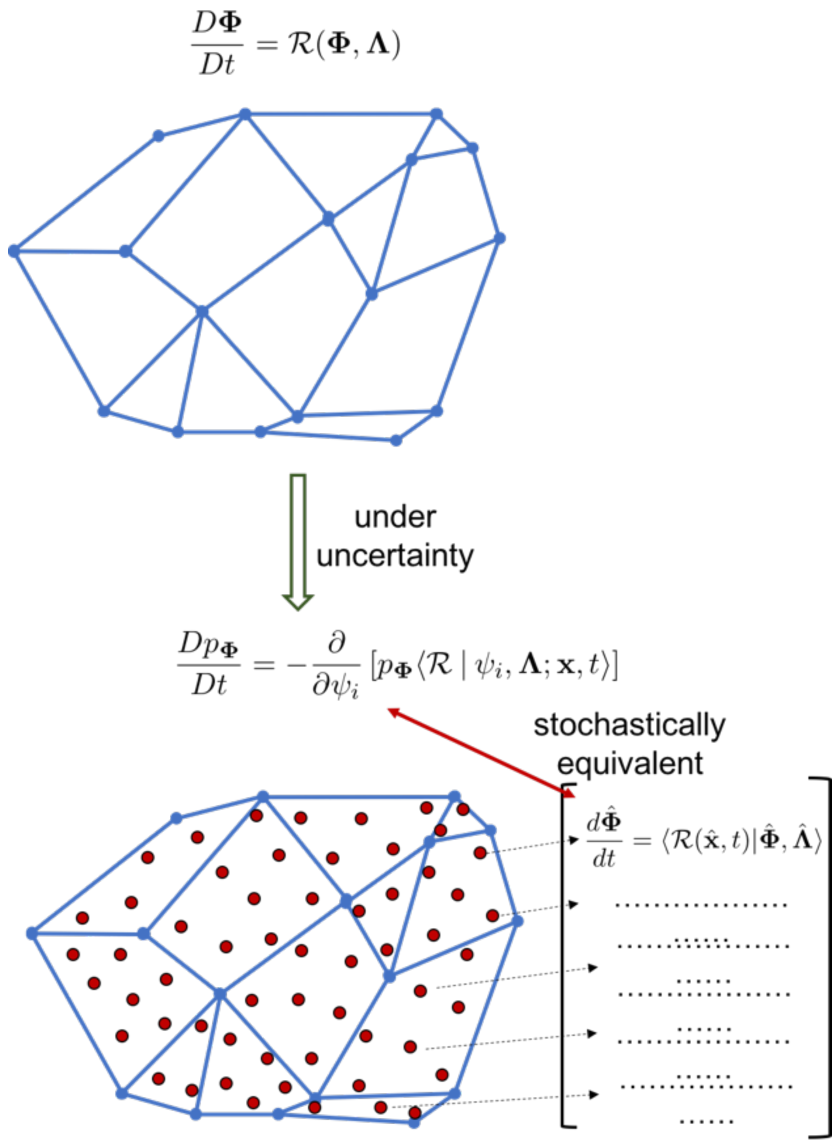
Challenge 1: Solving PDF equations efficiently

- The PDF equation needs to be solved in $\boldsymbol{\Phi}$-$\boldsymbol{\Lambda}$-$\boldsymbol{x}$-$\boldsymbol{t}$ space. Leverage Eulerian-Lagrangian equivalence of continuum mechanics:
  - Use existing $\boldsymbol{x}$-$\boldsymbol{t}$ discretizations i.e. spatial mesh, time stepping of existing solvers.
  - Sample the $\boldsymbol{\Phi}$-$\boldsymbol{\Lambda}$ space using notional Lagrangian samples (e.g. Monte-Carlo). Numerically efficient for large-dimensional spaces.
- The PDF equation (itself a PDE) is solved in a ***stochastically equivalent*** manner by solving Lagrangian ODEs:
  - Each sample has state $(\widehat{\boldsymbol{\phi}}, \hat{\lambda}, \widehat{\boldsymbol{x}})$ that is evolved by:

$$\mathbf{d}\widehat{\boldsymbol{\phi}}/\mathbf{dt} = \langle \boldsymbol{\mathcal{R}}(\widehat{\boldsymbol{x}}, \mathbf{t}) | \widehat{\boldsymbol{\phi}}, \hat{\boldsymbol{\lambda}} \rangle$$

  - PDF constructed from samples ensemble satisfies the PDF equation.

$$\frac{\mathrm{D}\boldsymbol{p}_{\boldsymbol{\phi}}}{\mathrm{Dt}} = -\frac{\partial}{\partial \phi_i}\left[\boldsymbol{p}_{\boldsymbol{\phi}} \langle \boldsymbol{\mathcal{R}} | \phi_i, \Lambda; \boldsymbol{x}, t \rangle\right] \quad \equiv \quad \frac{\mathbf{d}\widehat{\boldsymbol{\phi}}}{\mathbf{dt}} = \langle \boldsymbol{\mathcal{R}}(\widehat{\boldsymbol{x}}, \mathbf{t}) | \widehat{\boldsymbol{\phi}}, \hat{\boldsymbol{\lambda}} \rangle$$



Challenge 2: Solving PDF equations accurately

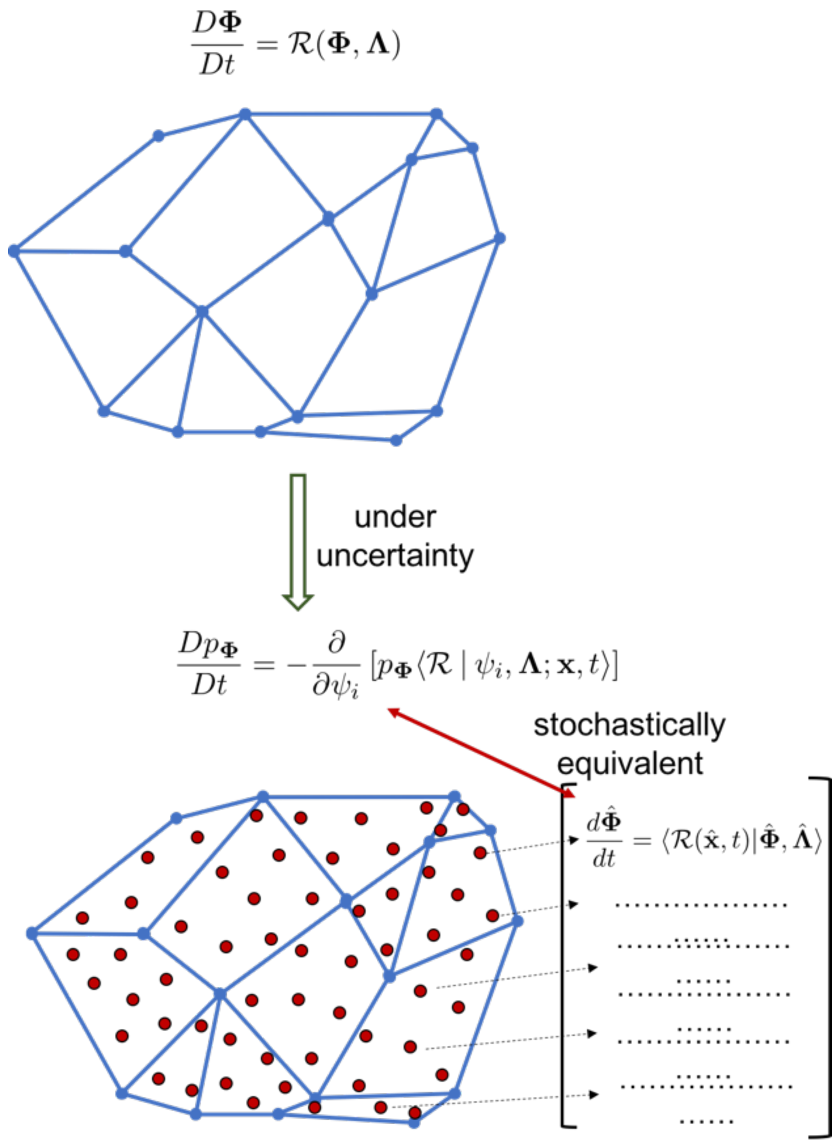
- Accuracy determined by closure; not circumvented in the Lagrangian system:
  - Each sample state update needs the unclosed **rate-of-change**

$$\mathbf{d}\hat{\boldsymbol{\phi}}/_{\mathbf{d}\boldsymbol{t}} = \langle \boldsymbol{\mathcal{R}}(\hat{\boldsymbol{x}}, \mathbf{t}) | \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\lambda}} \rangle$$

  - Problem for terms involving derivatives w.r.t. $\boldsymbol{x}$ or $\boldsymbol{t}$ (e.g., Laplace op)
- Use independent information, guided by physics/domain intuition:
  - Have an independent **"nominal mesh solution" guide closure**

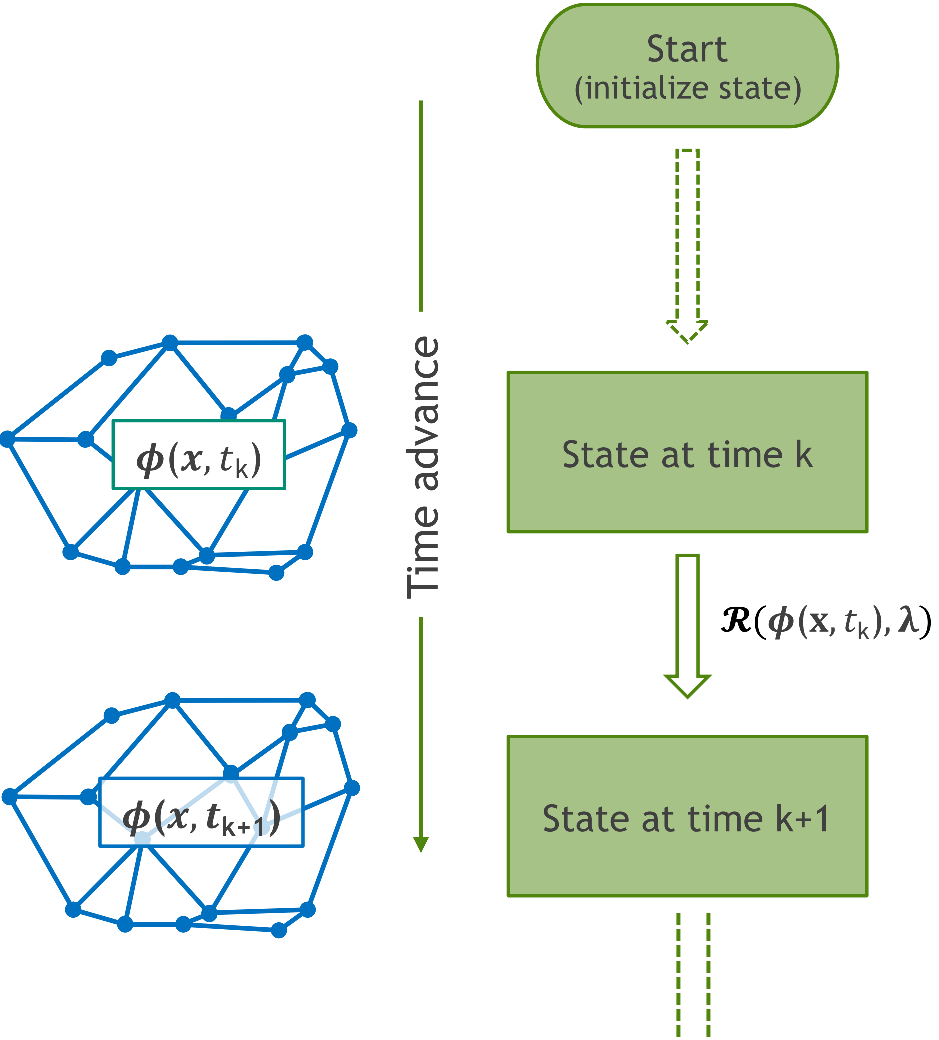
$$\left.\begin{array}{l} \boldsymbol{\phi}(\boldsymbol{x}, t, \lambda = \Lambda_n) \\ \boldsymbol{\mathcal{R}}(\boldsymbol{x}, t, \lambda = \Lambda_n) \end{array}\right\} \Longrightarrow \quad \langle \boldsymbol{\mathcal{R}}(\hat{\boldsymbol{x}}, \mathbf{t}) | \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\lambda}} \rangle$$

  - Physics-informed scaling laws can guide such closure models.
  - Also available, unconditional averages, $\langle \boldsymbol{\phi}(\boldsymbol{x}, \mathbf{t}) \rangle$ and $\langle \boldsymbol{\mathcal{R}}(\boldsymbol{x}, \mathbf{t}) \rangle$, from average over particles-in-cell. These can constrain the closure.



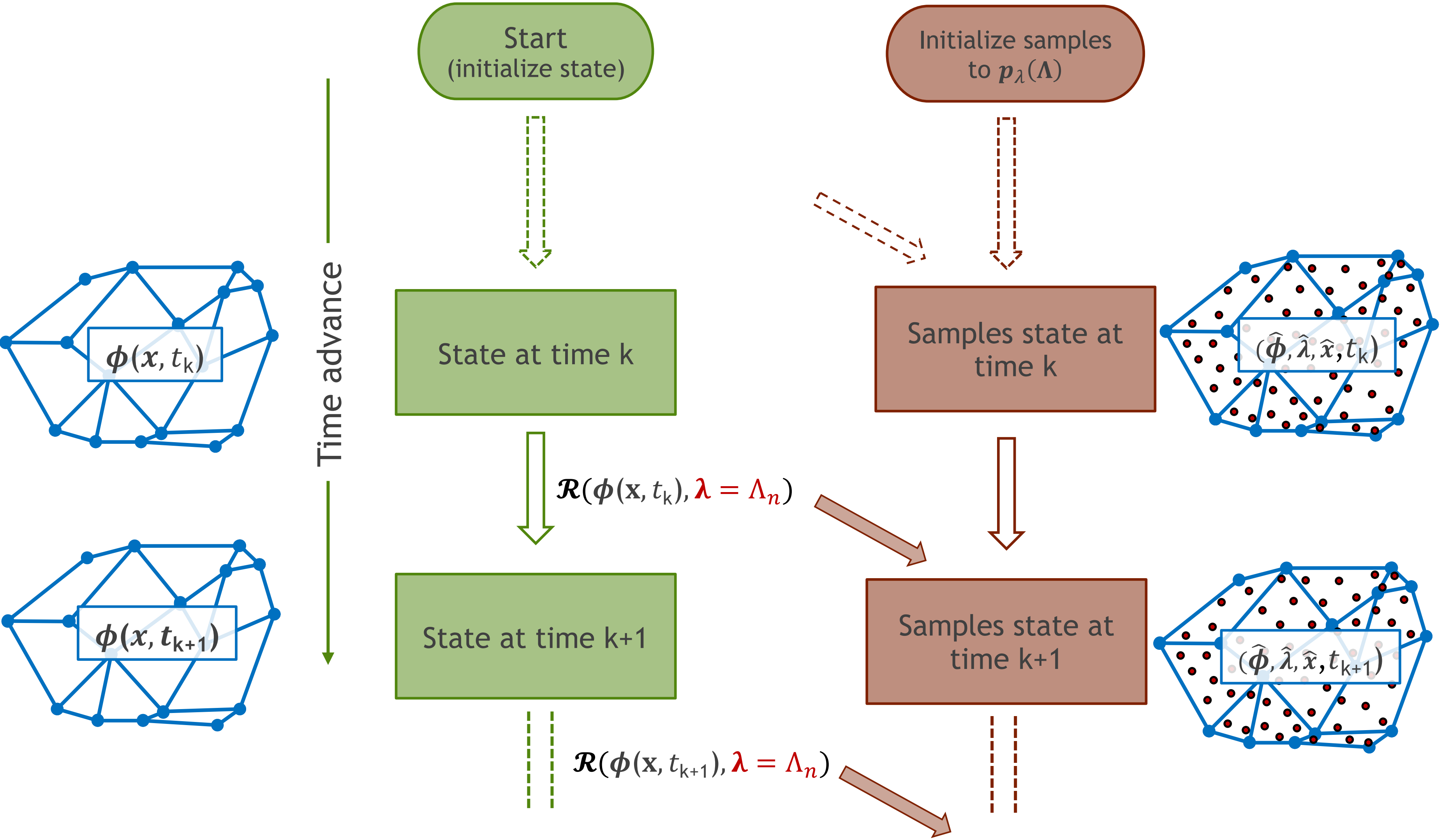
Implementation in existing HPC codes

Existing codes: Construct mesh, initialize, time-advance



Implementation in existing HPC codes

Existing codes: Construct mesh, initialize, time-advance ✚ **Augment with Lagrangian PDF samples**

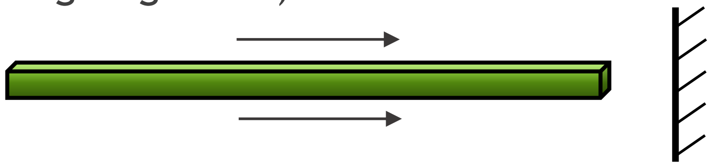


Samples initial state corresponds to uncertain parameter distribution and initial field $\phi(\mathbf{x}, t{=}0)$.

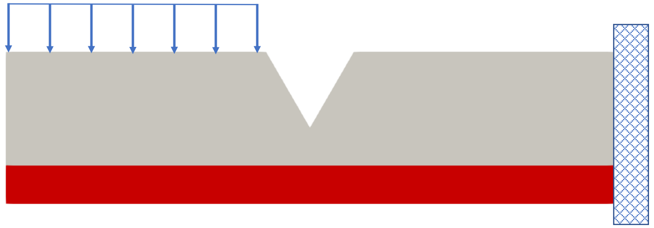
- Samples state advance by ODE, requires nominal solution $\mathcal{R}$ to close $\langle \mathcal{R}(\hat{x}, \mathrm{t}) | \hat{\phi}, \hat{\lambda} \rangle$
- Coupling is one-way, i.e., no feedback from samples state to mesh state.

Demonstration of the PDF method

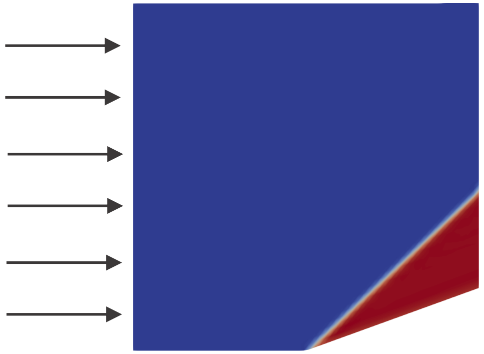
- Solid mechanics explicit dynamics (total Lagrangian FE):
  - Elastic bar with impact



  - Notched plate with impact



- Computational fluid mechanics (Eulerian FV):
  - Oblique shock in flow past ramp



The rest of the talk will focus exclusively on the solid mechanics problems.

PDF method for Computational Solid Mechanics

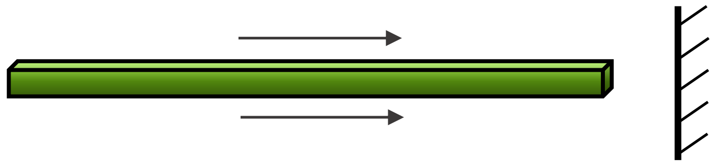
- Computational solid mechanics (mostly) adopts Lagrangian formulation:
  - Displacement itself is one of the solution variables.
- Our "Lagrangian samples" approach becomes an "intrusive multi-sampled" approach.
  - Multiple solution states, at each node, each corresponding to a different parameter realization.
- Basic strategy: solve only few samples exactly, rest approximately.
- For explicit dynamics:
  - Time explicit solution of linear momentum $\rho_0 \ddot{u} = \nabla . \mathbf{P} = \mathbf{R}(\nabla u, \lambda \,; t)$
  - For exact solution samples all quantities (def gradient, ∇ue, rate-of-change, $\mathbf{R}_{\boldsymbol{e}}(\nabla ue, \lambda \,; t)$) are known exactly.
  - For other samples, approximate (close) their rate of change from exact quantities ∇ue, $\mathbf{R}_{\boldsymbol{e}}(\nabla ue, \lambda \,; t)$, etc.
- Ratio of exact to approximate samples, and closure scheme, represent accuracy vs cost tradeoff.

Closure schemes under investigation

- Taylor series with one exact solution:
  - Approximate ∇u from ∇ue, evaluate $\mathbf{R}(\nabla u, \lambda\,; t)$ using constitutive model for stress.
  - Approximate $\mathbf{R}(\nabla u, \lambda\,; t)$ directly from $\mathbf{R}_e(\nabla ue, \lambda\,; t)$.
- "Structured" grid approximations (exact samples form a structured grid in parameter space):
  - Finite difference extrapolation with Taylor series
  - Polynomial (global, piecewise) tensor product bases interpolation.
  - Sparse grid interpolation.
- "Unstructured" grid approximations (exact samples can be chosen freely and not at specific locations in parameter space):
  - Simplicial tessellation.
  - Reproducing kernel approximation.
  - Maximum entropy approximation.

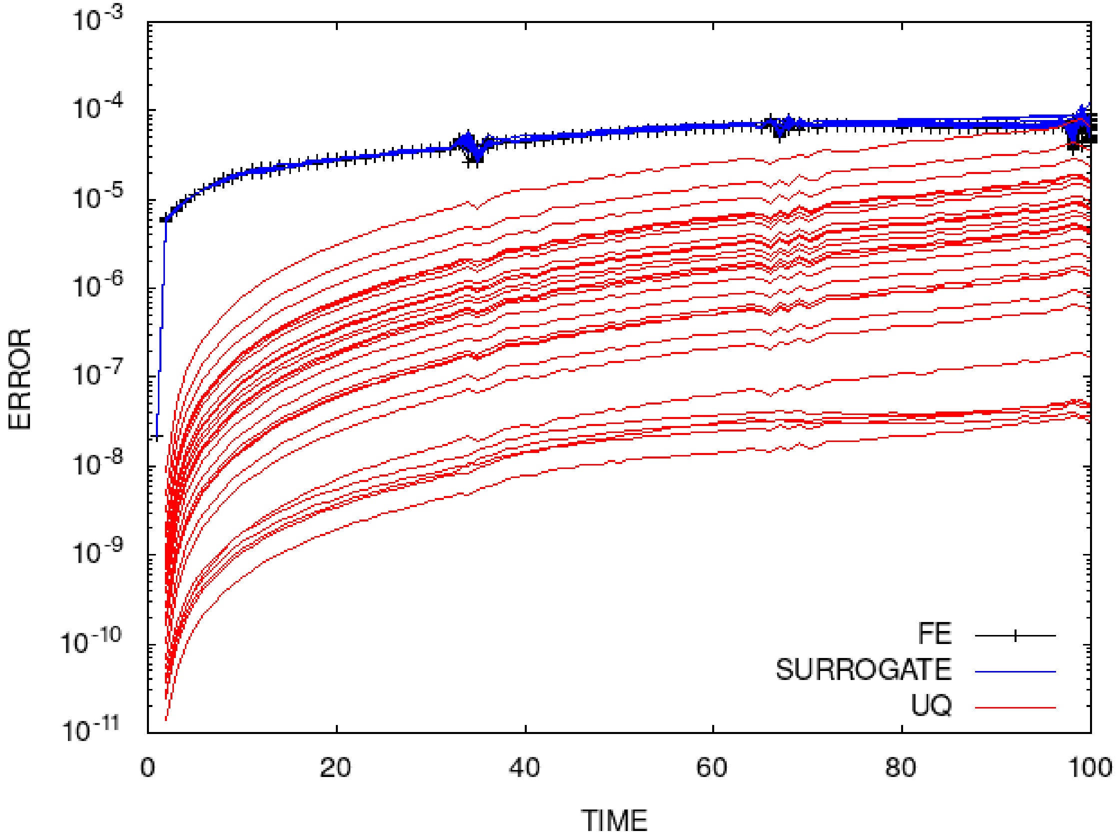
Results: Taylor Series approximation

- Method implemented in mini-app NimbleSM (https://github.com/NimbleSM).
- Demo Problem: bar impact, elastic, 1 uncertain parameter (shear modulus).
- Has exact analytical solution.



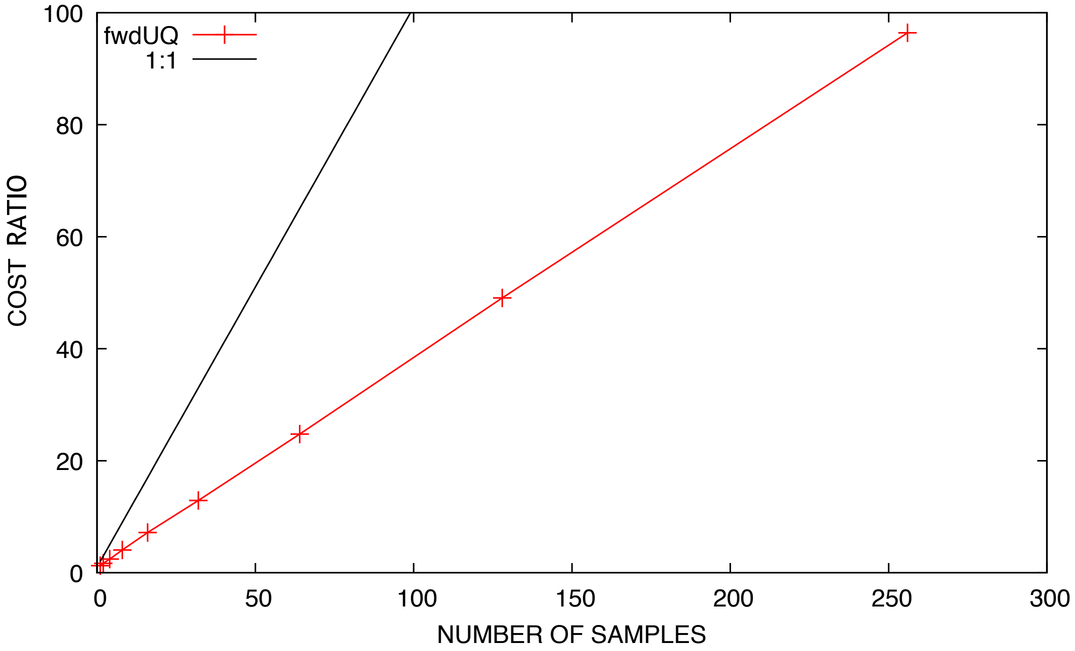
ACCURACY

Black: discretization error, **Blue:** Total error
Red: approximation error = total – discretization



COST

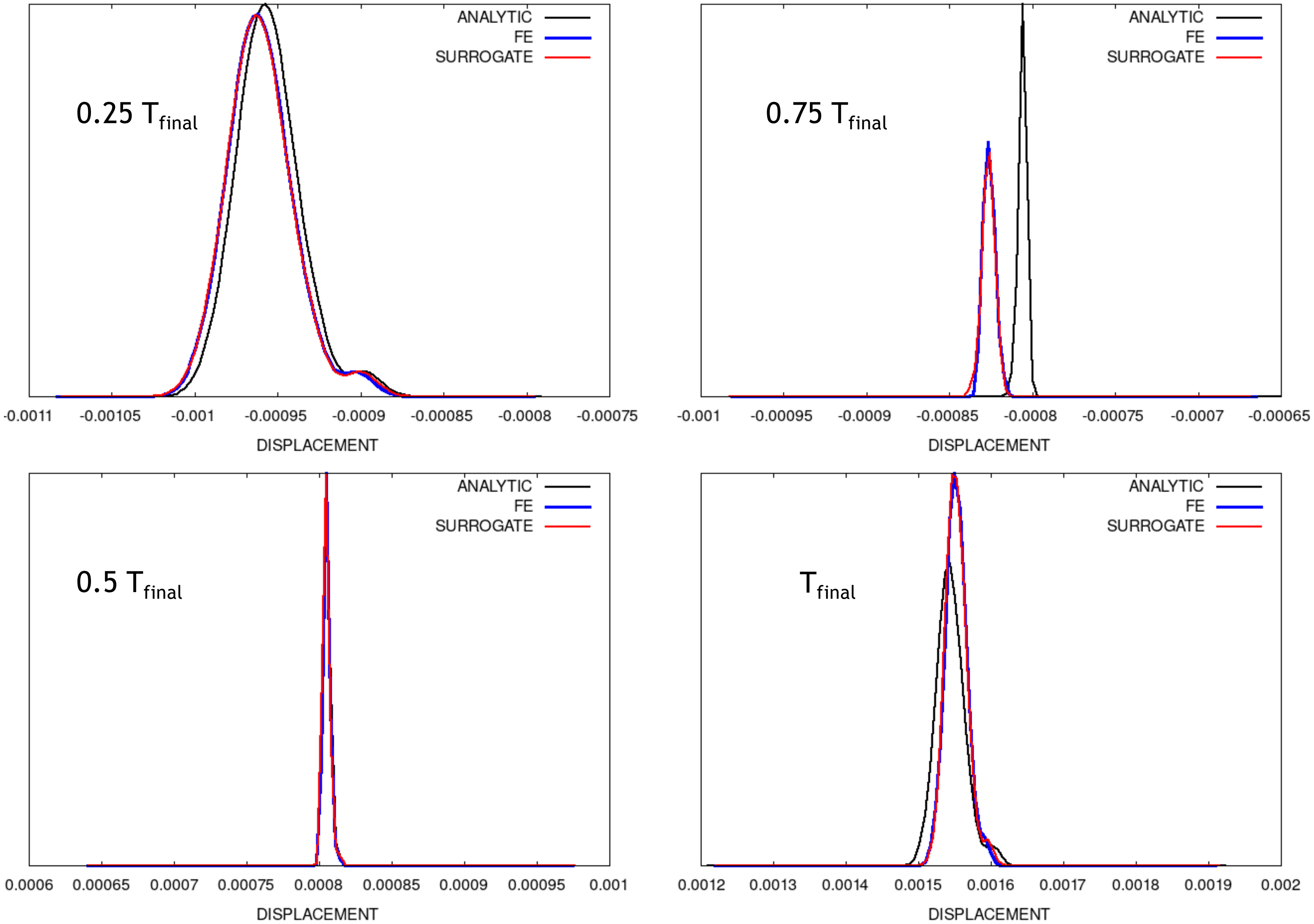
- Cost is linear with #samples; less than a full system cost by a **factor of 2.5.**
- We expect more gains in parallel implementation.



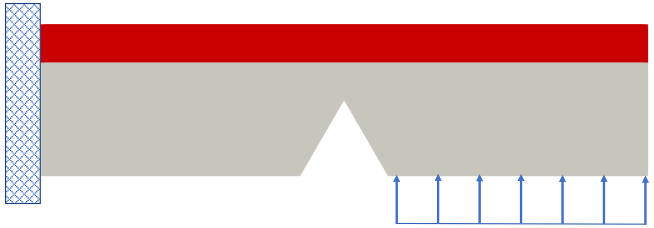
Results: Taylor Series approximation

Comprehensive Information

- PDFs of displacements, and other derivable quantities, available at **all nodes and times**
- Four times, bar mid-point, shown for comparison, 30 sample run:
  - **Analytical**
  - **Brute Force FE (many query)**
  - **PDF method**
- Comprehensive information at a small fraction of the cost.
- Leading order error is due to FE discretization



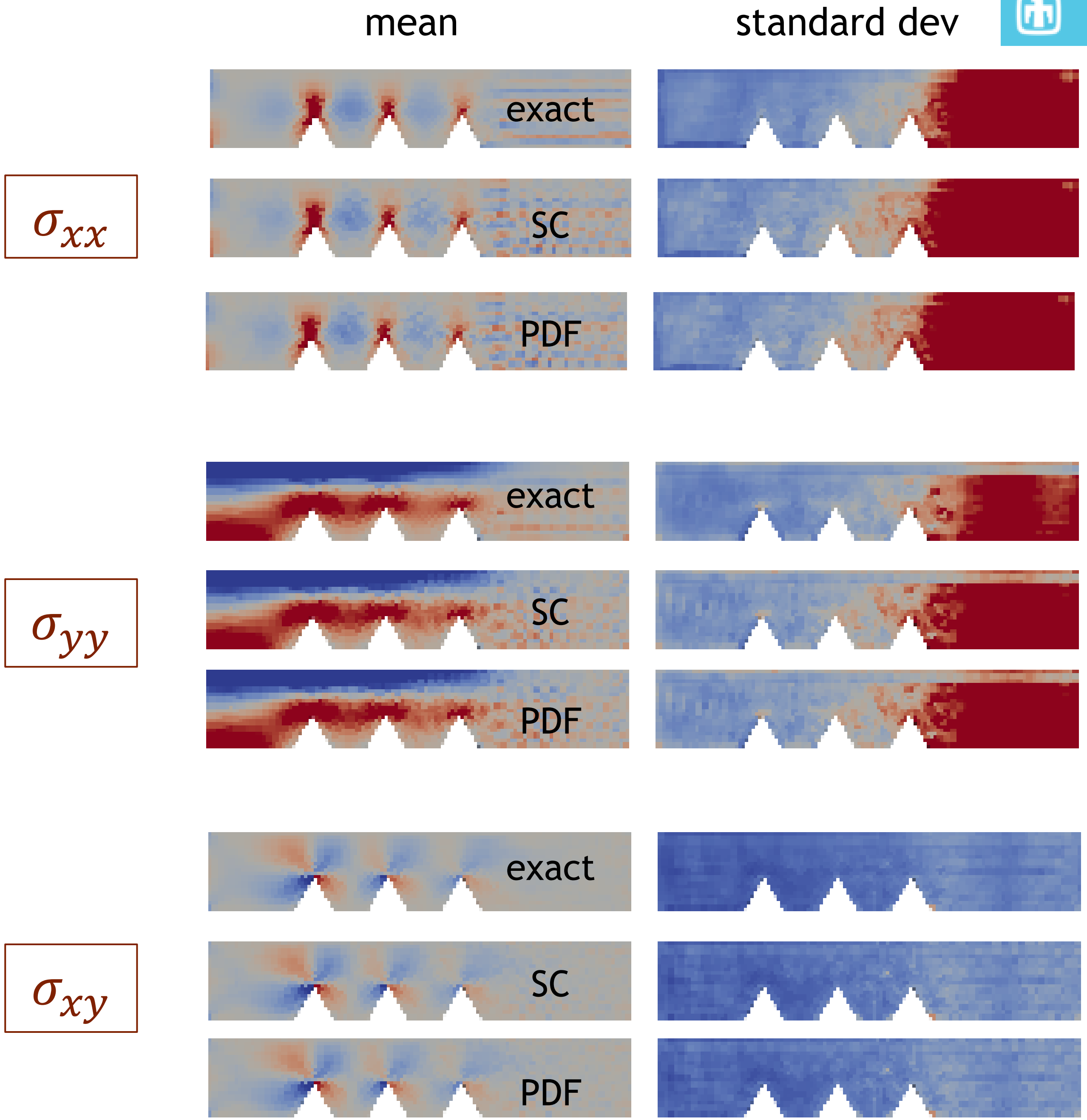
Results: Sparse Grid Interpolation



- Demo Problem: notched plate with impact, neo-Hookean materials aluminum (grey) and glass (red), 4 uncertain parameters.
- Simulations run with 512 approximate, 8 exact samples.
- Brute force "many query" runs (all 512 samples treated exactly) for accuracy comparisons.
- Note: The PDF method is similar in spirit to sparse grid stochastic collocation (SC):
  - Key difference: SC directly interpolates solution states u. Our method interpolates rate-of-change, $\mathbf{R}$, and time advances u.
  - Mathematical analysis shows error between the two different by a pre-factor
- We compare mean/stdev of individual stress components, and distributions of 'failure stress', an important QoI.
- Comparisons between PDF method and exact (brute force), SC and exact (brute force).

Results: Sparse Grid Interpolation

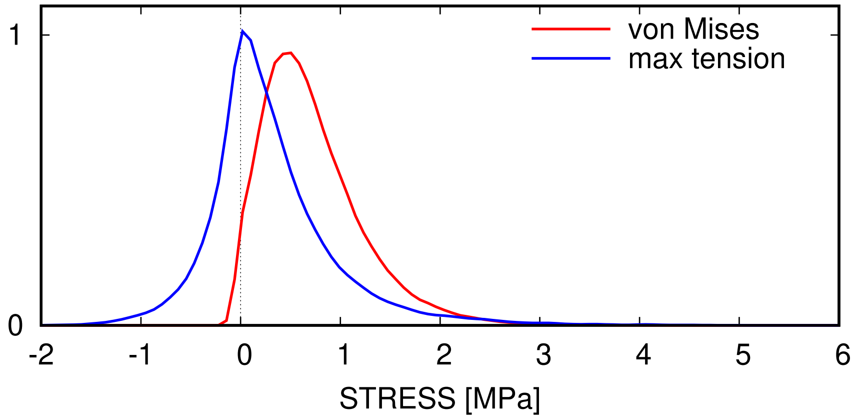
- Mean (left), stdev (right) of three stress components: σ_{xx}, σ_{yy}, σ_{xy}.
- Exact (top), Stochastic Collocation (middle), PDF method (bottom).
- Reasonably good agreement. Small differences compared to exact solution.
- Stochastic collocation and PDF solutions are almost identical.



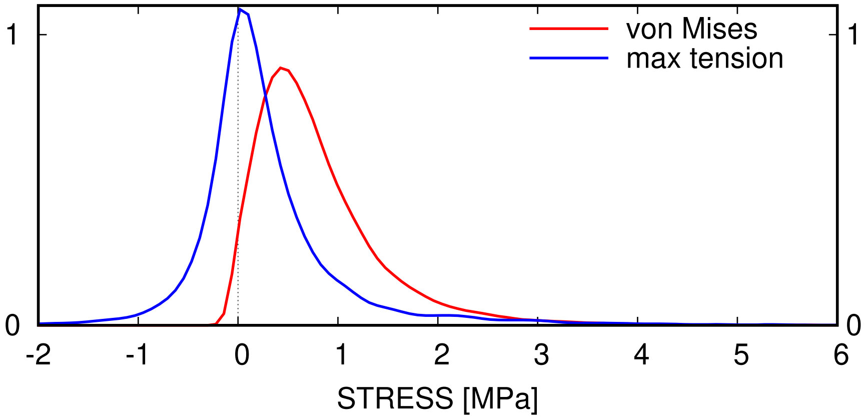
Results: Sparse Grid Interpolation

- Distributions of ‘failure stresses’.
- Space-time maximum of Von Mises and tensile stresses compared against failure limit.
- Reasonable agreement, small differences.

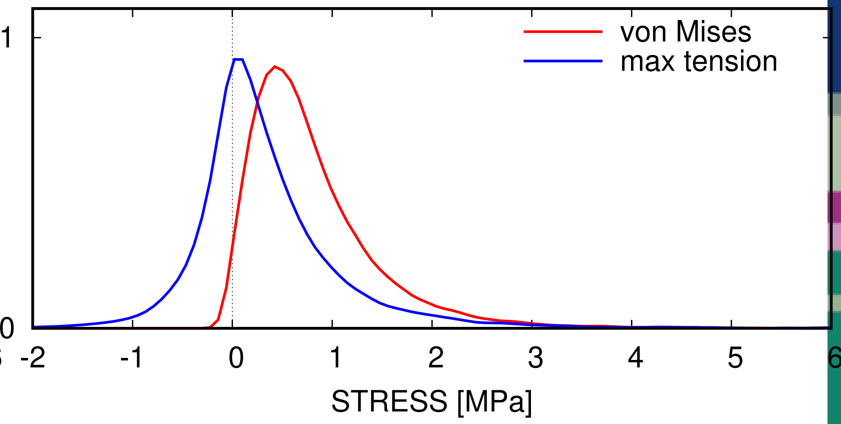
exact



SC



PDF



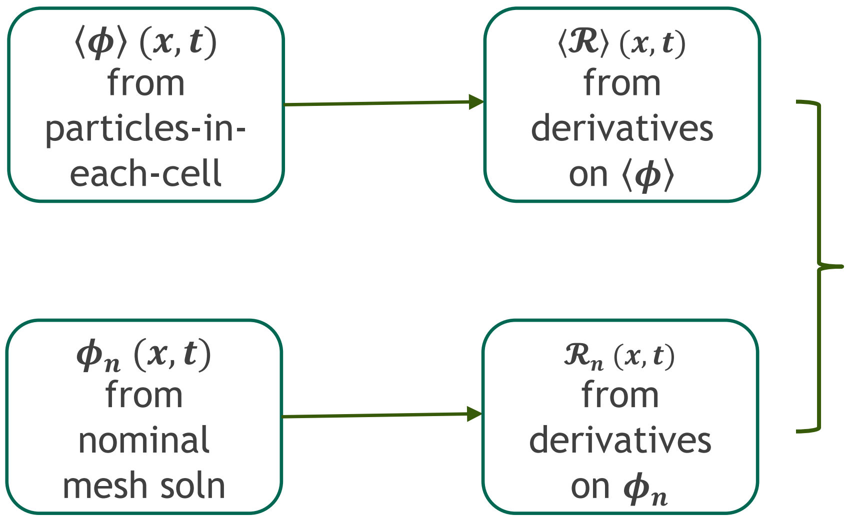
Conclusions and Future Work

- We propose solving PDF equations as an inexpensive approach to propagate parameter uncertainty.
- The PDF equations are derived exactly from governing PDEs, subject to uncertain parameter distribution.
- PDF equations can be solved efficiently by leveraging Eulerian-Lagrangian equivalence:
  - Sample the uncertain parameter(s), approximately evolve solution corresponding to the samples.
- For computational solid mechanics, the method is equivalent to “intrusive many-samples” approach.
- Accuracy of method dependent on approximation (closure) schemes.
- Various approaches, with varying accuracy vs cost tradeoff are being investigated.
- Results from simple target problems (elastic bar with impact, dual material notch with impact) are promising.
- Extending method for quasi-static and fully implicit methods is part of future work.

Backup Slides

Closure loop: Information flow from mesh to Lagrangian samples

- Hat quantities, $\hat{\phi}$, are sample states, which is what we are primarily trying to advance.
- What we need is RHS for each sample ODE, which is a conditional expectation: $\langle \mathcal{R}(\hat{x}, t) | \hat{\phi}, \hat{\lambda} \rangle$.
- What we have are:
  - the nominal solution on the mesh, i.e., $\phi_n(x,t)$, at each cell in the mesh. Also gives nominal RHS, $\mathcal{R}(\phi(\mathbf{x}, t_k), \lambda = \Lambda_n)$
  - the unconditional average, $\langle \phi \rangle (x,t)$, in each mesh cell. This average is obtained by simply averaging over all particles-in-cell. This also gives the unconditional average of the RHS, $\langle \mathcal{R} \rangle (x,t)$ (since averaging and derivatives commute).



+

$$\langle \mathcal{R} \rangle = \int \langle \mathcal{R} \mid \hat{\lambda} \rangle \, p_\lambda(\Lambda) d\Lambda$$

$$\langle \mathcal{R} \mid \hat{\lambda} \rangle = \underbrace{f\left(\hat{\lambda} / \Lambda_n \right)} * \mathcal{R}_n$$

Closure statement
From constitutive model

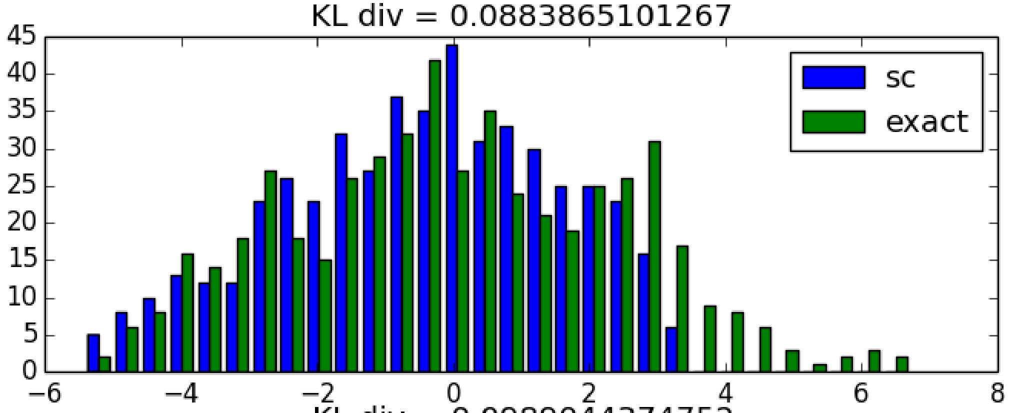
Give $\langle \mathcal{R}(\hat{x}, t) | \hat{\phi}, \hat{\lambda} \rangle$

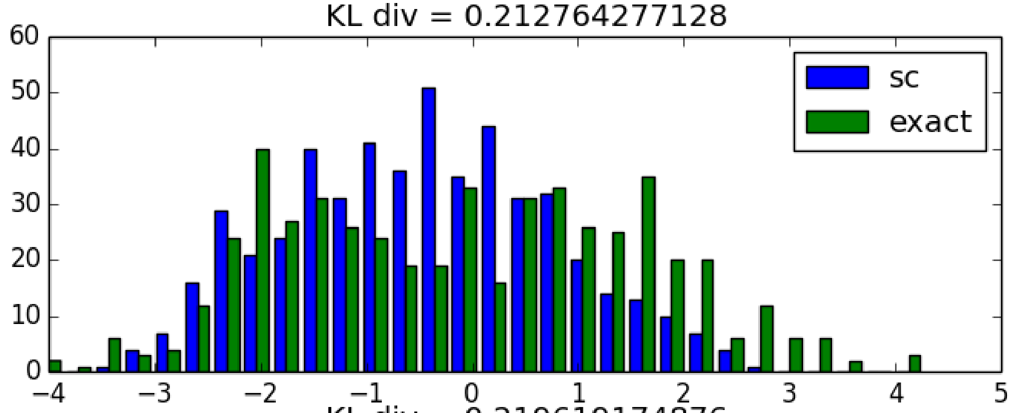
You are essentially translating two quantities that apply across the cell, $\langle \mathcal{R} \rangle$ and $\mathcal{R}_n$, to a quantity specific to each individual particle, $\langle \mathcal{R} \mid \hat{\lambda} \rangle$. In the process you're embedding physics knowledge in the form of $f\left(\hat{\lambda} / \Lambda_n \right)$.

Results: Sparse Grid Interpolation

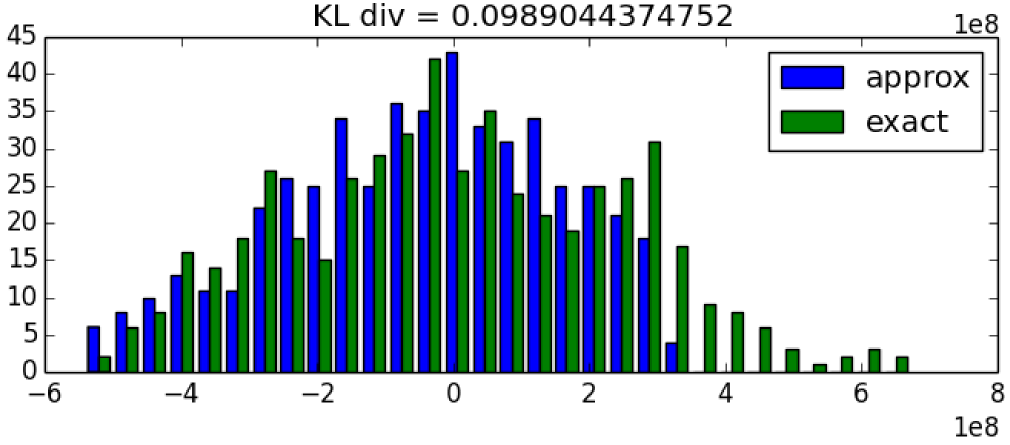
- Histograms of stress components (σ_{xx}, σ_{xy}) in element of interest
- Comparing Stochastic Collocation vs exact (Top), PDF method vs exact (Bottom)

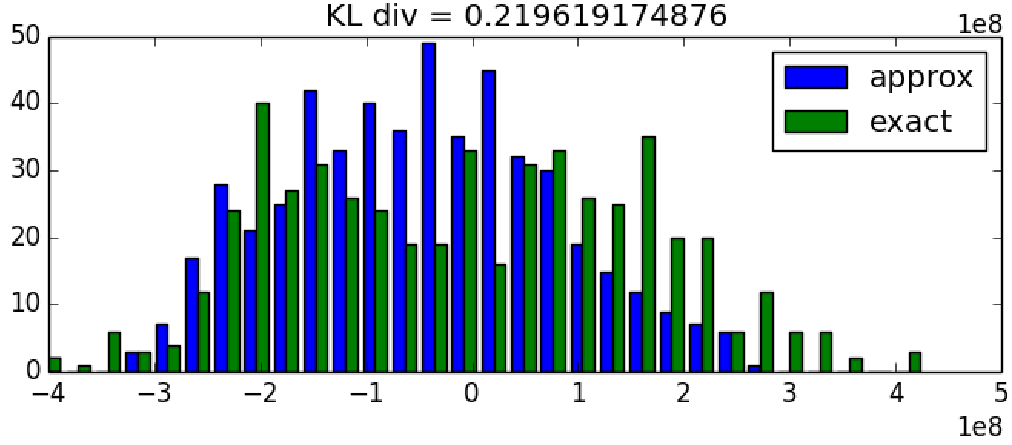
σ_{xx} σ_{xy}





SC vs exact





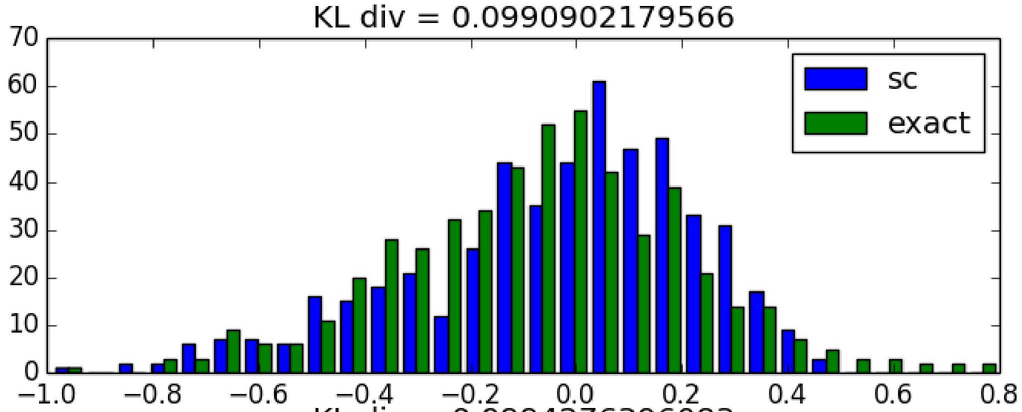
PDF vs exact

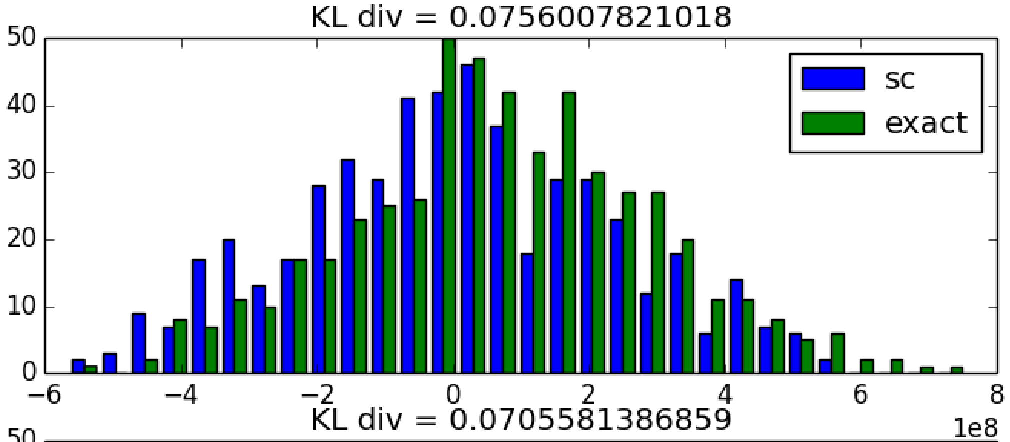
Results: Sparse Grid Interpolation

- Comparisons at later time.
- Both methods have comparable accuracy.
- Error (KL divergence) increases for σ_{xx}, decreases for σ_{xy}.

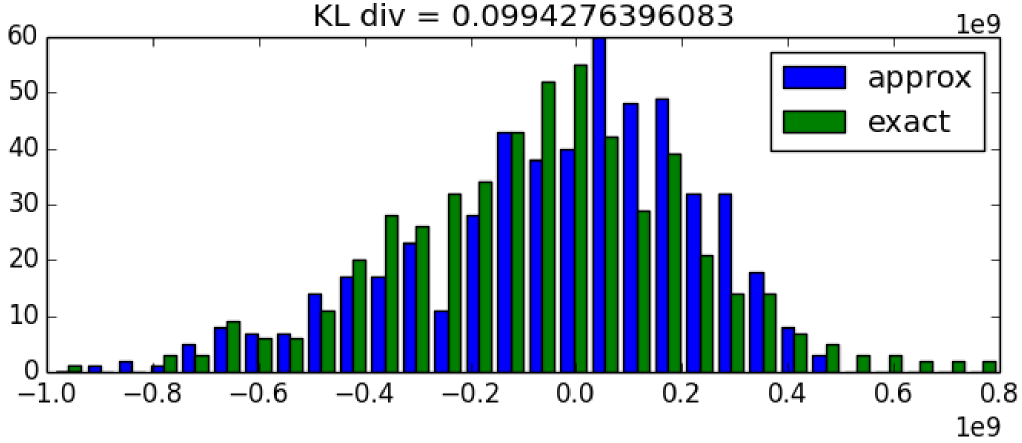
σ_{xx}

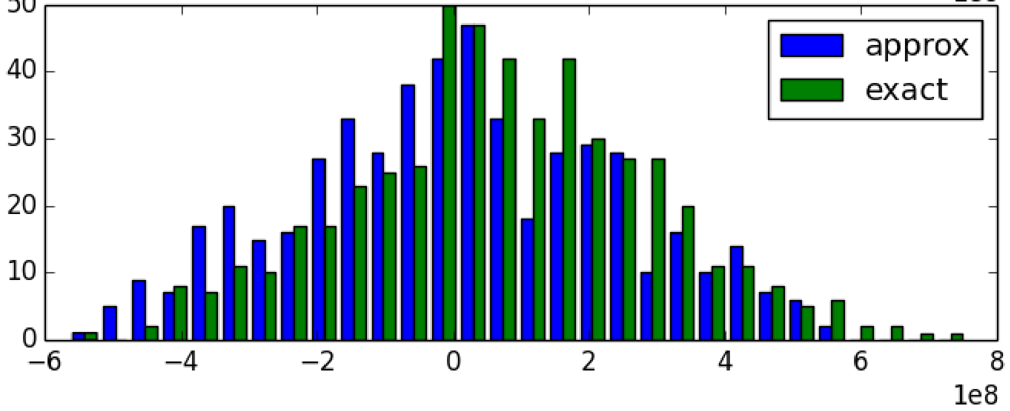
σ_{xy}





SC vs exact





PDF vs exact