Numerical Methods and UQ Analysis for Phase Field Equations

Tao Tang

Southern University of Science and Technology, China

April 27, 2018

Outline

1. Diffuse Interface / Phase Field Model
2. Uncertainty Quantification (UQ)
3. UQ and stability
   - Stochastic Galerkin methods
4. Sample-based methods and stability
5. Conclusions and references

Phase field model

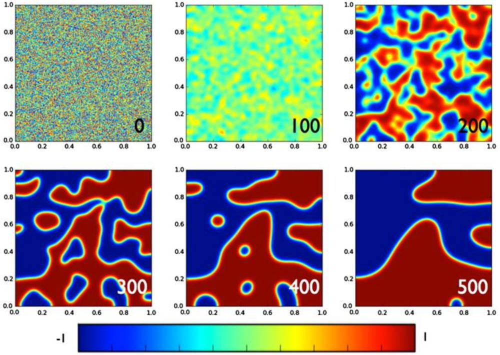
- A very simple/popular tool

Phase field model

- A very simple/popular tool
- Long history, extensive literature

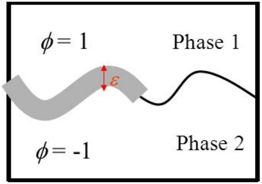
Phase field model

- A very simple/popular tool
- Long history, extensive literature
- An approximation to the sharp interface

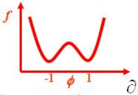


Phase-Field Models

- **Approach to moving boundary problems**
  - Phases associated with value of ϕ
    - Interface implies $\phi = 0$
  - Diffuse interface
    - Original problem obtained when $\varepsilon \to 0$

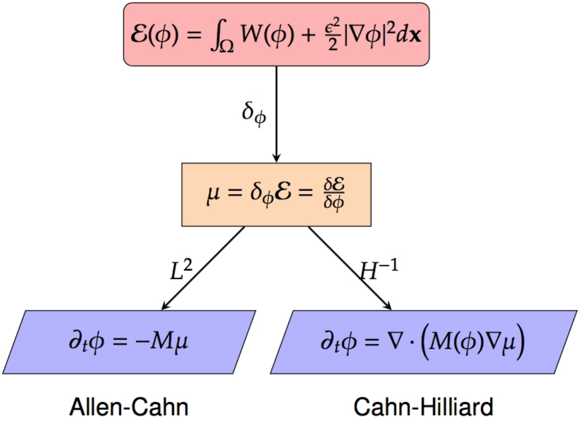


- **Dynamics of ϕ**
  - Can be derived from a free energy $F[\phi, \varepsilon]$



  - Non-conserved order parameter: Allen-Cahn equation $\dfrac{\partial \phi}{\partial t} = -\dfrac{\delta F}{\delta \phi}$
  - Conserved order parameter: Cahn-Hilliard equation $\dfrac{\partial \phi}{\partial t} = \nabla^2 \dfrac{\delta F}{\delta \phi}$

Allen-Cahn and Cahn-Hilliard Equation



ϕ : concentration or temperature
W : double-well function
μ : chemical potential
M : diffusion mobility

Allen-Can/ Cahn-Hilliard Eqns

Allen-Can/ Cahn-Hilliard Eqns

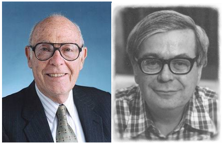
Allen-Cahn Eqn: $u_t = \delta\Delta u + f(u)$

Cahn-Hilliard Eqn: $u_t = \Delta(-\delta\Delta u + f(u))$

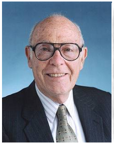
Allen-Can/ Cahn-Hilliard Eqns

Allen-Cahn Eqn: $u_t = \delta\Delta u + f(u)$

Cahn-Hilliard Eqn: $u_t = \Delta(-\delta\Delta u + f(u))$



Allen-Can/ Cahn-Hilliard Eqns



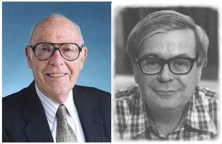

Allen-Cahn Eqn: $u_t = \delta \Delta u + f(u)$

Cahn-Hilliard Eqn: $u_t = \Delta(-\delta \Delta u + f(u))$

- Molecular beam epitaxy (MBE) eqn

$$u_t = -\delta \Delta^2 u + \nabla \cdot f(\nabla u),$$

Allen-Can/ Cahn-Hilliard Eqns



Allen-Cahn Eqn: $u_t = \delta\Delta u + f(u)$

Cahn-Hilliard Eqn: $u_t = \Delta(-\delta\Delta u + f(u))$

- Molecular beam epitaxy (MBE) eqn

$$u_t = -\delta\Delta^2 u + \nabla \cdot f(\nabla u),$$

- Typical f: $f(\phi) = \phi|\phi|^2 - \phi$. An important feature is that they can be viewed as the gradient flow of the following energy functionals:

$$E_{MBE}(u) = \int_\Omega \left[\frac{\delta}{2}|\Delta u|^2 + \frac{1}{4}(|\nabla u|^2 - 1)^2\right] dx$$

$$E_{CH}(u) = \int_\Omega \left[\frac{\delta}{2}|\nabla u|^2 + \frac{1}{4}(|u|^2 - 1)^2\right] dx$$

Energy decay (the key for numerical stability)

- For the energy functionals of phase field problems

$$E(u(t) \leq E(u(s)), \quad \forall t \geq s.$$

Energy decay (the key for numerical stability)

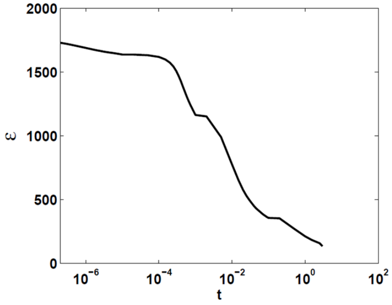
- For the energy functionals of phase field problems

$$E(u(t) \leq E(u(s)), \quad \forall t \geq s.$$

- Example: Cahn-Hilliard impainting

$$u_t = \Delta\left(-\delta\Delta u - \frac{1}{\delta}F'(u)\right) + \lambda(f - u)$$

[Bertozzi etc. IEEE Tran. Imag. Proc. 2007, Commun. Math. Sci, 2011]



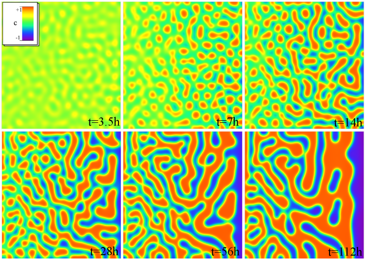
Numerical Challenge for Phase Field Computations

- Difficulties:

  Catch dynamics (small Δt) & steady state (large Δt)

  Higher order methods vs. efficiency

  Long-Time Integration

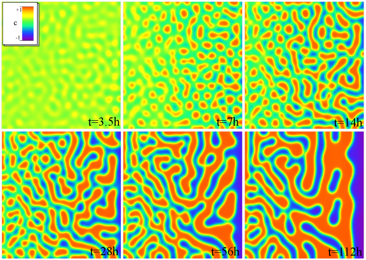


Numerical Challenge for Phase Field Computations

- Difficulties:
  Catch dynamics (small Δt) & steady state (large Δt)

  Higher order methods vs. efficiency

  Long-Time Integration



- Different Approaches:
  Energy decay methods (numerous efforts)

  Adaptivity in time/space; Moving mesh spectral method etc

start from Allen-Cahn eqn

- To demonstrate the main idea, we consider

$$
\begin{aligned}
&u_t = u_{xx} + u(1-u^2), \quad x \in (-1,1), \\
&u(\pm 1, t) = 0, \\
&u(x,0) = f(x).
\end{aligned}
$$

start from Allen-Cahn eqn

- To demonstrate the main idea, we consider

$$
\begin{aligned}
& u_t = u_{xx} + u(1 - u^2), \quad x \in (-1, 1), \\
& u(\pm 1, t) = 0, \\
& u(x, 0) = f(x).
\end{aligned}
$$

- Define the energy function in L^2- space

$$
E(u) = \int_\Omega (\frac{1}{2}|\nabla u|^2 + F(u))dx
$$

where $F(u) = \frac{1}{4}(1 - u^2)^2$.

start from Allen-Cahn eqn

- To demonstrate the main idea, we consider

$$\begin{aligned} u_t &= u_{xx} + u(1-u^2), \quad x \in (-1,1), \\ u(\pm 1, t) &= 0, \\ u(x,0) &= f(x). \end{aligned}$$

- Define the energy function in L^2- space

$$E(u) = \int_\Omega (\frac{1}{2}|\nabla u|^2 + F(u))dx$$

where $F(u) = \frac{1}{4}(1-u^2)^2$.

- Multiplying u_t on both sides of the AC eqn, and then use integration by parts gives

$$\frac{d}{dt}E(u) \leq 0, \quad \forall t > 0.$$

Implicit scheme: Crank-Nicholson

- The CN scheme

$$\frac{u_j^{n+1}-u_j^n}{\Delta t}=\frac{1}{2\Delta x^2}\left(\delta_-\delta_+u_j^{n+1}+\delta_-\delta_+u_j^n\right)+\frac{u_j^{n+1}+u_j^n}{2}\left(1-\frac{(u_j^{n+1})^2+(u_j^n)^2}{2}\right).$$

where

$$\delta_+u_j=u_{j+1}-u_j,\quad \delta_-u_j=u_j-u_{j-1}.$$

Implicit scheme: Crank-Nicholson

- The CN scheme

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{2\Delta x^2}\left(\delta_-\delta_+ u_j^{n+1} + \delta_-\delta_+ u_j^n\right) + \frac{u_j^{n+1} + u_j^n}{2}\left(1 - \frac{(u_j^{n+1})^2 + (u_j^n)^2}{2}\right).$$

where

$$\delta_+ u_j = u_{j+1} - u_j, \quad \delta_- u_j = u_j - u_{j-1}.$$

- For the CN scheme, the following gradient flow property is satisfied

$$\hat{E}(u^{n+1}) \leq \hat{E}(u^n),$$

where

$$\hat{E}(u^{n+1}) := \sum_j \left(\frac{u_{j+1}^{n+1} - u_j^n}{\Delta x}\right)^2 \Delta x + \sum_j \frac{1}{4}\left(1 - (u_j^{n+1})^2\right)^2$$

simple proof

- Multiplying $u_j^{n+1} - u_j^n$ on both sides of the CN scheme gives

$$\left(u_j^{n+1} - u_j^n\right)^2 = \frac{\lambda}{2}\left(u_j^{n+1} - u_j^n\right)\left(\delta_-\delta_+ u_j^{n+1} + \delta_-\delta_+ u_j^n\right) + \frac{\Delta t}{2}\left((u_j^{n+1})^2 - (u_j^n)^2\right)\left(1 - \frac{(u_j^{n+1})^2 + (u_j^n)^2}{2}\right),$$

where $\lambda = \Delta t/\Delta x^2$.

simple proof

- Multiplying $u_j^{n+1} - u_j^n$ on both sides of the CN scheme gives

$$\left(u_j^{n+1} - u_j^n\right)^2 = \frac{\lambda}{2}\left(u_j^{n+1} - u_j^n\right)\left(\delta_-\delta_+ u_j^{n+1} + \delta_-\delta_+ u_j^n\right) + \frac{\Delta t}{2}\left((u_j^{n+1})^2 - (u_j^n)^2\right)\left(1 - \frac{(u_j^{n+1})^2 + (u_j^n)^2}{2}\right),$$

where $\lambda = \Delta t/\Delta x^2$.

- Summing over j and using integration by parts give

$$\begin{aligned}
&\sum_j \left(u_j^{n+1} - u_j^n\right)^2 \Delta x \\
= \quad & -\frac{\lambda}{2}\sum_j \delta_+\left(u_j^{n+1} - u_j^n\right)\left(\delta_+ u_j^{n+1} + \delta_+ u_j^n\right)\Delta x \\
& + \frac{\Delta t}{2}\sum_j \left[(u_j^{n+1})^2 - (u_j^n)^2 - \frac{1}{2}\left((u_j^{n+1})^4 - (u_j^n)^4\right)\right]\Delta x \\
= \quad & -\frac{\lambda}{2}\sum_j \left[\left(\delta_+ u_j^{n+1}\right)^2 - \left(\delta_+ u_j^n\right)^2\right]\Delta x - \frac{\Delta t}{4}\sum_j \left[\left(1 - (u_j^{n+1})^2\right)^2 - \left(1 - (u_j^n)^2\right)^2\right]\Delta x.
\end{aligned} \tag{1}$$

simple proof

- Multiplying $u_j^{n+1} - u_j^n$ on both sides of the CN scheme gives

$$\left(u_j^{n+1} - u_j^n\right)^2 = \frac{\lambda}{2}\left(u_j^{n+1} - u_j^n\right)\left(\delta_-\delta_+ u_j^{n+1} + \delta_-\delta_+ u_j^n\right) + \frac{\Delta t}{2}\left((u_j^{n+1})^2 - (u_j^n)^2\right)\left(1 - \frac{(u_j^{n+1})^2 + (u_j^n)^2}{2}\right),$$

  where $\lambda = \Delta t/\Delta x^2$.
- Summing over j and using integration by parts give

$$\begin{aligned} &\sum_j \left(u_j^{n+1} - u_j^n\right)^2 \Delta x \\ = \quad & -\frac{\lambda}{2}\sum_j \delta_+\left(u_j^{n+1} - u_j^n\right)\left(\delta_+ u_j^{n+1} + \delta_+ u_j^n\right)\Delta x \\ & + \frac{\Delta t}{2}\sum_j \left[(u_j^{n+1})^2 - (u_j^n)^2 - \frac{1}{2}\left((u_j^{n+1})^4 - (u_j^n)^4\right)\right]\Delta x \\ = \quad & -\frac{\lambda}{2}\sum_j \left[\left(\delta_+ u_j^{n+1}\right)^2 - \left(\delta_+ u_j^n\right)^2\right]\Delta x - \frac{\Delta t}{4}\sum_j \left[\left(1 - (u_j^{n+1})^2\right)^2 - \left(1 - (u_j^n)^2\right)^2\right]\Delta x. \end{aligned} \tag{1}$$

- As the left-hand side of (1) is non-negative, we obtain

$$\frac{1}{2}\sum_j \left[\left(\frac{\delta_+ u_j^{n+1}}{\Delta x}\right)^2 - \left(\frac{\delta_+ u_j^n}{\Delta x}\right)^2\right]\Delta x + \frac{1}{4}\sum_j \left[\left(1 - (u_j^{n+1})^2\right)^2 - \left(1 - (u_j^n)^2\right)^2\right]\Delta x \le 0.$$

Linear approximation

- Note that CN scheme is nonlinear. Consider *linear* scheme:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{\Delta x^2}\delta_-\delta_+ u_j^{n+1} + u_j^{n+1}\Big(1 - (u_j^n)^2\Big), \quad 1 \le j \le J - 1$$

Linear approximation

- Note that CN scheme is nonlinear. Consider *linear* scheme:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{\Delta x^2}\delta_-\delta_+ u_j^{n+1} + u_j^{n+1}\left(1 - (u_j^n)^2\right), \quad 1 \le j \le J-1$$

- If $\Delta t < 1$, then the following stability results hold: Assume the initial data satisfies

Linear approximation

- Note that CN scheme is nonlinear. Consider *linear* scheme:

$$\frac{u_j^{n+1}-u_j^n}{\Delta t} = \frac{1}{\Delta x^2}\delta_-\delta_+ u_j^{n+1} + u_j^{n+1}\Big(1-(u_j^n)^2\Big), \quad 1 \le j \le J-1$$

- If $\Delta t < 1$, then the following stability results hold: Assume the initial data satisfies
  - L^∞-stability:

$$\|u^n\|_\infty \le e^{2t_n}\|u^0\|_\infty, \quad n \ge 0;$$

Linear approximation

- Note that CN scheme is nonlinear. Consider *linear* scheme:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{\Delta x^2}\delta_-\delta_+ u_j^{n+1} + u_j^{n+1}\left(1 - (u_j^n)^2\right), \quad 1 \leq j \leq J-1$$

- If $\Delta t < 1$, then the following stability results hold: Assume the initial data satisfies
  - L^∞-stability:
  $$\|u^n\|_\infty \leq e^{2t_n}\|u^0\|_\infty, \quad n \geq 0;$$
  - L^1-stability:
  $$\|u^n\|_1 \leq e^{2t_n}\|u^0\|_1, \quad n \geq 0;$$

Linear approximation

- Note that CN scheme is nonlinear. Consider *linear* scheme:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{\Delta x^2}\delta_-\delta_+ u_j^{n+1} + u_j^{n+1}\Big(1 - (u_j^n)^2\Big), \quad 1 \le j \le J-1$$

- If $\Delta t < 1$, then the following stability results hold: Assume the initial data satisfies
  - L^∞-stability:
  $$\|u^n\|_\infty \le e^{2t_n}\|u^0\|_\infty, \quad n \ge 0;$$
  - L^1-stability:
  $$\|u^n\|_1 \le e^{2t_n}\|u^0\|_1, \quad n \ge 0;$$
  - L^2-stability:
  $$\|u^n\|_2 \le e^{2t_n}\|u^0\|_2, \quad n \ge 0.$$

Linear scheme satisfying gradient flow property

- consider

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{\Delta x^2}\delta_-\delta_+ u_j^{n+1} + \frac{u_j^{n+1} + u_j^n}{2}\left(1 - (u_j^n)^2\right), \quad 1 \le j \le J - 1$$

Linear scheme satisfying gradient flow property

- consider

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{\Delta x^2}\delta_-\delta_+ u_j^{n+1} + \frac{u_j^{n+1} + u_j^n}{2}\left(1 - (u_j^n)^2\right), \quad 1 \le j \le J - 1$$

- If $\Delta t < 1$ and

$$\Delta t \|u_0\|_\infty e^{cT} < 1,$$

then the following gradient flow property is satisfied

$$\hat{E}(u^{n+1}) \le \hat{E}(u^n),$$

where

$$\hat{E}(u^{n+1}) := \sum_j \left(\frac{u_{j+1}^{n+1} - u_j^n}{\Delta x}\right)^2 \Delta x + \sum_j \frac{1}{4}\left(1 - (u_j^{n+1})^2\right)^2$$

Convex splitting for Cahn-Hillard eqn

- CH eqn:

$$\frac{\partial u}{\partial t} = \Delta(-\delta\Delta u + f(u)), \quad \mathbf{x} \in \Omega, \; t \in (0, T],$$

Convex splitting for Cahn-Hillard eqn

- CH eqn:
$$\frac{\partial u}{\partial t} = \Delta(-\delta \Delta u + f(u)), \quad \mathbf{x} \in \Omega, \; t \in (0, T],$$
- Eyre's splitting scheme for $u_t = -\nabla E(u)$. We need energy stable scheme $dE(u)/dt \leq 0$.

Convex splitting for Cahn-Hillard eqn

- CH eqn:

$$\frac{\partial u}{\partial t} = \Delta(-\delta\Delta u + f(u)), \quad \mathbf{x} \in \Omega,\ t \in (0, T],$$

- Eyre's splitting scheme for $u_t = -\nabla E(u)$. We need energy stable scheme $dE(u)/dt \leq 0$.
- Convexity splitting

$$E(u) = E_c(u) - E_e(u)$$

where $E_c, E_e \in C^2$ and are strictly convex.

Convex splitting for Cahn-Hillard eqn

- CH eqn:

$$\frac{\partial u}{\partial t} = \Delta(-\delta\Delta u + f(u)), \quad \mathbf{x} \in \Omega, \ t \in (0, T],$$

- Eyre's splitting scheme for $u_t = -\nabla E(u)$. We need energy stable scheme $dE(u)/dt \leq 0$.
- Convexity splitting

$$E(u) = E_c(u) - E_e(u)$$

where $E_c, E_e \in C^2$ and are strictly convex.
- The semi-implicit discretization is given by

$$\frac{U^{n+1} - U^n}{\Delta t} = -\Big(\nabla E_c(U^{n+1}) - \nabla E_e(U^n)\Big).$$

Various Eyre’s type or various extension

Convex splitting for Cahn-Hillard eqn

- Using the splitting form for CHE

$$E_c(u) = \int_\Omega \left(\frac{\delta}{2}|\nabla u|^2 + \frac{\beta}{2}u^2\right)dx, \quad E_e(u) = \int_\Omega \left(\frac{\beta}{2}u^2 - F(u)\right)dx,$$

Convex splitting for Cahn-Hillard eqn

- Using the splitting form for CHE

$$E_c(u) = \int_\Omega \left(\frac{\delta}{2}|\nabla u|^2 + \frac{\beta}{2}u^2\right)dx, \quad E_e(u) = \int_\Omega \left(\frac{\beta}{2}u^2 - F(u)\right)dx,$$

- we have

$$\begin{aligned}\frac{u^{n+1}-u^n}{\Delta t} &= \Delta\left(\frac{\delta E_c(u^{n+1})}{\delta u} - \frac{\delta E_e(u^n)}{\delta u}\right) \\ &= -\delta\Delta^2 u^{n+1} + \beta\Delta u^{n+1} - \beta\Delta u^n + \Delta f(u^n).\end{aligned}$$

Convex splitting for Cahn-Hillard eqn

- Using the splitting form for CHE

$$E_c(u) = \int_\Omega \left(\frac{\delta}{2}|\nabla u|^2 + \frac{\beta}{2}u^2\right)dx, \quad E_e(u) = \int_\Omega \left(\frac{\beta}{2}u^2 - F(u)\right)dx,$$

- we have

$$\begin{aligned} \frac{u^{n+1} - u^n}{\Delta t} &= \Delta\left(\frac{\delta E_c(u^{n+1})}{\delta u} - \frac{\delta E_e(u^n)}{\delta u}\right) \\ &= -\delta\Delta^2 u^{n+1} + \beta\Delta u^{n+1} - \beta\Delta u^n + \Delta f(u^n). \end{aligned}$$

- If the constant β is sufficiently large, then

$$E(u^{n+1}) \leq E(u^n), \quad n = 0, 1, \cdots .$$

Lower order with p-adaptivity

- In each time interval, compute $|E_h(U^{n+1}) - E_h(U^n)$. If the difference is small, then move to next time level;

Lower order with p-adaptivity

- In each time interval, compute $|E_h(U^{n+1}) - E_h(U^n)$. If the difference is small, then move to next time level;
- If the difference is large, judge how many order-enhancements are needed.

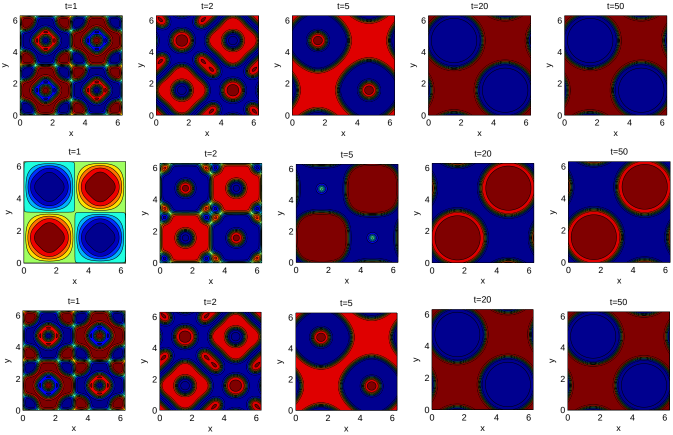
Lower order with p-adaptivity

- In each time interval, compute $|E_h(U^{n+1}) - E_h(U^n)$. If the difference is small, then move to next time level;
- If the difference is large, judge how many order-enhancements are needed.

- (a)

(b)

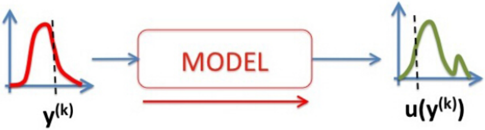
(c)



Uncertainty Quantification (UQ)

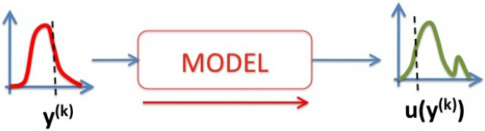
Uncertainty propagation: non-intrusive

- Non-intrusive methods: only require (multiple) solutions of the original (deterministic) model



Uncertainty propagation: non-intrusive

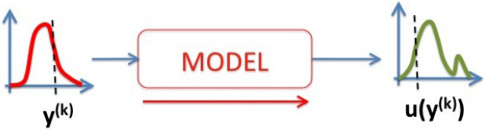
- Non-intrusive methods: only require (multiple) solutions of the original (deterministic) model



  - Simple extension of the ”conventional” simulation paradigm

Uncertainty propagation: non-intrusive

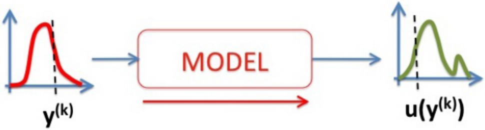
- Non-intrusive methods: only require (multiple) solutions of the original (deterministic) model



  - Simple extension of the ”conventional” simulation paradigm
  - Embarrassingly parallel: solutions are independent

Uncertainty propagation: non-intrusive

- Non-intrusive methods: only require (multiple) solutions of the original (deterministic) model

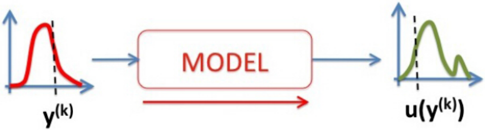


  - Simple extension of the "conventional" simulation paradigm
  - Embarrassingly parallel: solutions are independent

- Monte Carlo (low order), stochastic collocation (High order), ect.

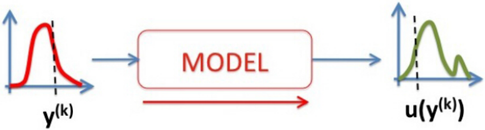
Non-intrusive approach: Monte Carlo

- If you know how to sample $\left\{\mathbf{y}^{(\mathbf{k})}\right\}_{k=1}^{M}$ ... it's done



Non-intrusive approach: Monte Carlo

- If you know how to sample $\left\{\mathbf{y}^{(\mathbf{k})}\right\}_{k=1}^{M}$ ... it's done

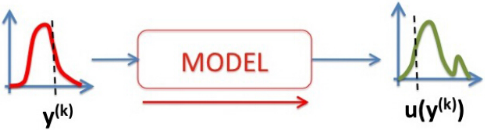


- Approximation of statistic moments

$$\mathbb{E}[u] \approx \frac{1}{M}\sum_{k=1}^{M} u\left(\mathbf{y}^{(\mathbf{k})}\right).$$

Non-intrusive approach: Monte Carlo

- If you know how to sample $\left\{\mathbf{y}^{(\mathbf{k})}\right\}_{k=1}^{M}$ ... it's done



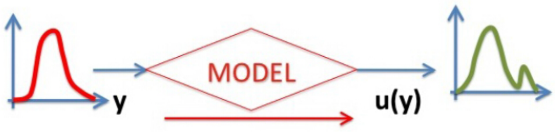
- Approximation of statistic moments

$$\mathbb{E}[u] \approx \frac{1}{M} \sum_{k=1}^{M} u\left(\mathbf{y}^{(\mathbf{k})}\right).$$

- Slow convergence rate $M^{-\frac{1}{2}}$, but independent of the dimension d.

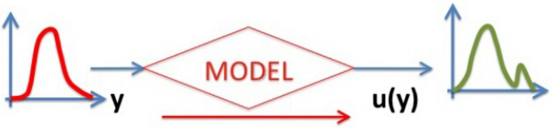
Uncertainty propagation: intrusive

- Intrusive methods: require the formulation and solution of a stochastic version of the original model



Uncertainty propagation: intrusive

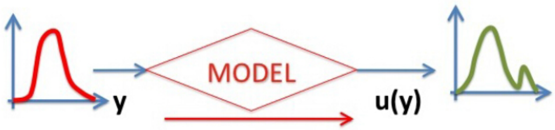
- Intrusive methods: require the formulation and solution of a stochastic version of the original model



  - Exploit the mathematical structure of the problem

Uncertainty propagation: intrusive

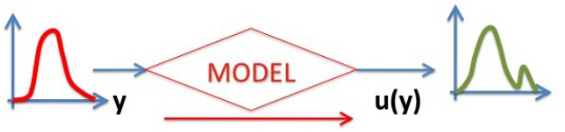
- Intrusive methods: require the formulation and solution of a stochastic version of the original model



  - Exploit the mathematical structure of the problem
  - Leverage theoretical & algorithmic advancements

Uncertainty propagation: intrusive

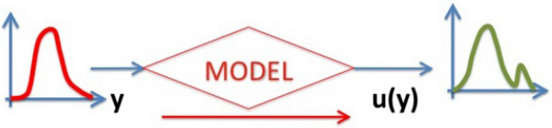
- Intrusive methods: require the formulation and solution of a stochastic version of the original model



  - Exploit the mathematical structure of the problem
  - Leverage theoretical & algorithmic advancements
  - New codes are needed

Uncertainty propagation: intrusive

- Intrusive methods: require the formulation and solution of a stochastic version of the original model

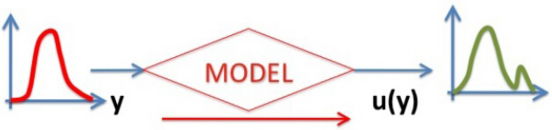


  - Exploit the mathematical structure of the problem
  - Leverage theoretical & algorithmic advancements
  - New codes are needed

- Perturbation methods (low order), polynomial chaos (High order), ect.

Uncertainty propagation: intrusive

- Intrusive methods: require the formulation and solution of a stochastic version of the original model



  - Exploit the mathematical structure of the problem
  - Leverage theoretical & algorithmic advancements
  - New codes are needed

- Perturbation methods (low order), polynomial chaos (High order), ect.

- Computational cost can be high for large-scale problems.

Numerical Study for Uncertainties

- Different Approaches:

Numerical Study for Uncertainties

- Different Approaches:
  - Generalized Polynomial chaos expansions

Numerical Study for Uncertainties

- Different Approaches:
  - Generalized Polynomial chaos expansions
  - Multi-level Monte Carlo method

Numerical Study for Uncertainties

- Different Approaches:
  - Generalized Polynomial chaos expansions
  - Multi-level Monte Carlo method
  - Bayesian approach to SPDEs

Numerical Study for Uncertainties

- Different Approaches:
  - Generalized Polynomial chaos expansions
  - Multi-level Monte Carlo method
  - Bayesian approach to SPDEs
- Open Issues:

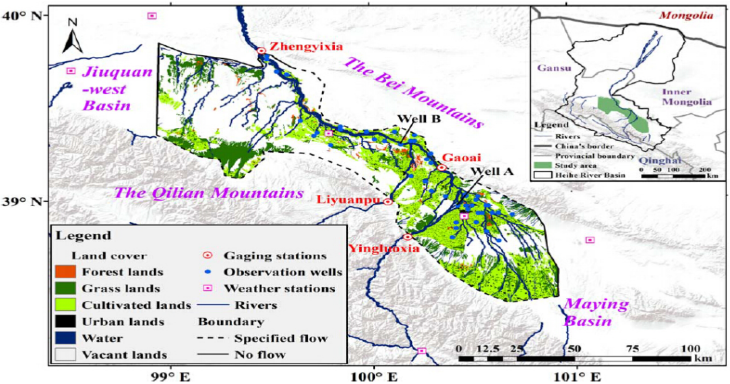
Numerical Study for Uncertainties

- Different Approaches:
  - Generalized Polynomial chaos expansions
  - Multi-level Monte Carlo method
  - Bayesian approach to SPDEs
- Open Issues:
  - High Dimensions, curse-of-dimensionality

Numerical Study for Uncertainties

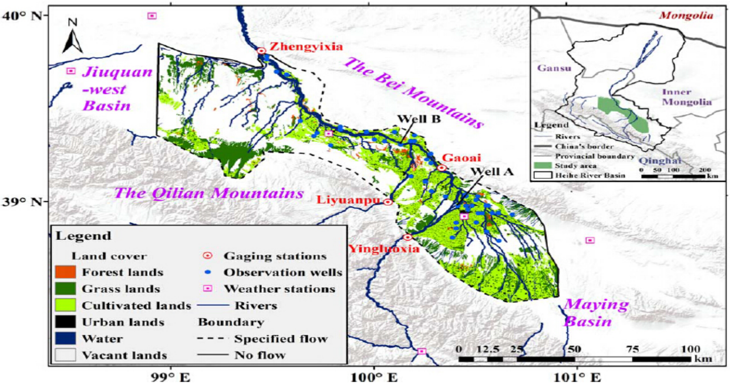
- Different Approaches:
  - Generalized Polynomial chaos expansions
  - Multi-level Monte Carlo method
  - Bayesian approach to SPDEs
- Open Issues:
  - High Dimensions, curse-of-dimensionality
  - Parametric Discontinuities

Subsurface flow in random media (Dagan '89, Zhang '02)



Subsurface flow in random media (Dagan '89, Zhang '02)

$$\begin{cases} \nabla \cdot (K(x,\omega)\nabla h(x)) + g(x) = 0, & x \in \Gamma, \\ h(x) = H(x), & x \in \Gamma_D, \\ K(x,\omega)\nabla h(x) \cdot \mathbf{n}(x) = -Q(x), & x \in \Gamma_n. \end{cases}$$



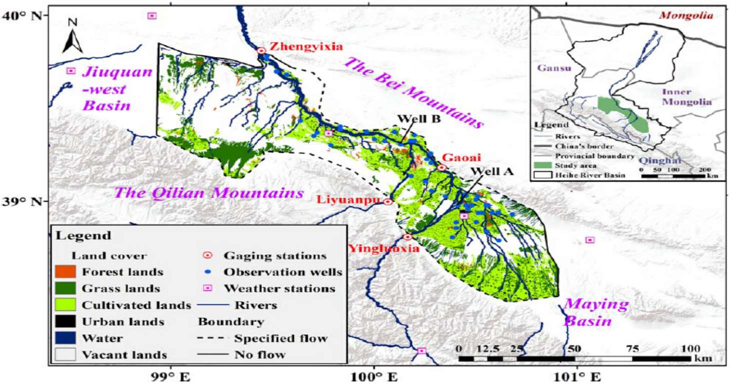
Subsurface flow in random media (Dagan '89, Zhang '02)

$$\begin{cases} \nabla\cdot(K(x,\omega)\nabla h(x)) + g(x) = 0, & x\in\Gamma, \\ h(x) = H(x), & x\in\Gamma_D, \\ K(x,\omega)\nabla h(x)\cdot\mathbf{n}(x) = -Q(x), & x\in\Gamma_n. \end{cases}$$

Log-normal conductivity:

$Y(x,\omega) = \ln K(x,\omega)$,

$Y(x,\omega) \sim$ Gaussian random field



The elliptic model problem

- Elliptic PDEs with random input:

$$-\nabla \cdot (\kappa(x, \mathbf{y})\nabla u) = f(x, \mathbf{y}), \quad u|_{\partial D} = 0$$

The elliptic model problem

- Elliptic PDEs with random input:

$$-\nabla \cdot (\kappa(x, \mathbf{y})\nabla u) = f(x, \mathbf{y}), \quad u|_{\partial D} = 0$$

- $\mathbf{y} = (y_1, y_2, ..., y_d)$: d independent parameters, $d \gg 1.$

The elliptic model problem

- Elliptic PDEs with random input:

$$-\nabla \cdot (\kappa(x, \mathbf{y})\nabla u) = f(x, \mathbf{y}), \quad u|_{\partial D} = 0$$

- $\mathbf{y} = (y_1, y_2, ..., y_d)$: d independent parameters, $d \gg 1.$

- Each parameter admits known density : $y_k \in \Gamma_k \subset \mathbb{R}, \quad y_k \sim \rho_k.$

The elliptic model problem

- Elliptic PDEs with random input:

$$-\nabla \cdot (\kappa(x, \mathbf{y}) \nabla u) = f(x, \mathbf{y}), \quad u|_{\partial D} = 0$$

- $\mathbf{y} = (y_1, y_2, ..., y_d)$: d independent parameters, $d \gg 1.$

- Each parameter admits known density : $y_k \in \Gamma_k \subset \mathbb{R}, \quad y_k \sim \rho_k.$

- $\mathbf{y} \in \Gamma := \otimes_{k=1}^{d} \Gamma_k \subset \mathbb{R}^d, \quad \mathbf{y} \sim \rho(\mathbf{y}) = \Pi_{k=1}^{d} \rho_k(y_k).$

The elliptic model problem

- Elliptic PDEs with random input:

$$-\nabla \cdot (\kappa(x, \mathbf{y})\nabla u) = f(x, \mathbf{y}), \quad u|_{\partial D} = 0$$

- $\mathbf{y} = (y_1, y_2, ..., y_d)$: d independent parameters, $d \gg 1.$
- Each parameter admits known density : $y_k \in \Gamma_k \subset \mathbb{R}, \quad y_k \sim \rho_k.$
- $\mathbf{y} \in \Gamma := \otimes_{k=1}^{d} \Gamma_k \subset \mathbb{R}^d, \quad \mathbf{y} \sim \rho(\mathbf{y}) = \Pi_{k=1}^{d} \rho_k(y_k).$
- The input-output operator (with respect to $\mathbf{y}$) is smooth:

The elliptic model problem

- Elliptic PDEs with random input:

$$-\nabla \cdot (\kappa(x, \mathbf{y})\nabla u) = f(x, \mathbf{y}), \quad u|_{\partial D} = 0$$

- $\mathbf{y} = (y_1, y_2, ..., y_d)$: d independent parameters, $d \gg 1$.
- Each parameter admits known density : $y_k \in \Gamma_k \subset \mathbb{R}, \quad y_k \sim \rho_k$.
- $\mathbf{y} \in \Gamma := \otimes_{k=1}^{d} \Gamma_k \subset \mathbb{R}^d, \quad \mathbf{y} \sim \rho(\mathbf{y}) = \Pi_{k=1}^{d} \rho_k(y_k)$.
- The input-output operator (with respect to $\mathbf{y}$) is smooth:
  - Babuska-Nobile et al '07, Cohen-DeVore-Schwab '10

Generalized Polynomial Chaos (Xiu & Karniadakis '03)

- Multivariate polynomial expansions:

$$u(x, \mathbf{y}) \approx \sum_{\alpha \in \mathcal{I}} \widehat{c}_\alpha(x) \phi_\alpha(\mathbf{y}), \quad \text{with} \quad \int \rho(\mathbf{y}) \phi_\alpha(\mathbf{y}) \phi_\beta(\mathbf{y}) d\mathbf{y} = \delta_{\alpha\beta}.$$

Input	Polynomial	Density	Support
Normal	Hermite $He_n(x)$	$e^{-\frac{x^2}{2}}$	$[-\infty, +\infty]$
Uniform	Legendre $L_n(x)$	$\frac{1}{2}$	$[-1, 1]$

Generalized Polynomial Chaos (Xiu & Karniadakis '03)

- Multivariate polynomial expansions:

$$u(x,\mathbf{y}) \approx \sum_{\alpha\in\mathcal{I}} \widehat{c}_\alpha(x)\phi_\alpha(\mathbf{y}), \quad \text{with} \quad \int \rho(\mathbf{y})\phi_\alpha(\mathbf{y})\phi_\beta(\mathbf{y})d\mathbf{y} = \delta_{\alpha\beta}.$$

Input	Polynomial	Density	Support
Normal	Hermite $He_n(x)$	$e^{-\frac{x^2}{2}}$	$[-\infty, +\infty]$
Uniform	Legendre $L_n(x)$	$\frac{1}{2}$	$[-1, 1]$

- Multi-index $\alpha = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}_0^d$. Assign a single index:

$$u_N(x,\mathbf{y}) = \sum_{\alpha\in\mathcal{I}} \widehat{c}_\alpha(x)\phi_\alpha(\mathbf{y}) = \sum_{n=1}^{N} \widehat{c}_n(x)\phi_n(\mathbf{y}), \quad N = \#\{\mathcal{I}\}.$$

Generalized Polynomial Chaos (Xiu & Karniadakis '03)

- Multivariate polynomial expansions:

$$u(x,\mathbf{y}) \approx \sum_{\alpha\in\mathcal{I}} \widehat{c}_\alpha(x)\phi_\alpha(\mathbf{y}), \quad \text{with} \quad \int \rho(\mathbf{y})\phi_\alpha(\mathbf{y})\phi_\beta(\mathbf{y})d\mathbf{y} = \delta_{\alpha\beta}.$$

Input	Polynomial	Density	Support
Normal	Hermite $He_n(x)$	$e^{-\frac{x^2}{2}}$	$[-\infty,+\infty]$
Uniform	Legendre $L_n(x)$	$\frac{1}{2}$	$[-1,1]$

- Multi-index $\alpha = (\alpha_1,\ldots,\alpha_d) \in \mathbb{N}_0^d$. Assign a single index:

$$u_N(x,\mathbf{y}) = \sum_{\alpha\in\mathcal{I}} \widehat{c}_\alpha(x)\phi_\alpha(\mathbf{y}) = \sum_{n=1}^{N} \widehat{c}_n(x)\phi_n(\mathbf{y}), \quad N = \#\{\mathcal{I}\}.$$

- Total degree polynomial spaces with degree q:

$$\mathcal{I} := \{\alpha : \|\alpha\|_1 \le q\} \quad \Rightarrow \quad N = \binom{q+d}{d} = \frac{(q+d)!}{q!d!}.$$

Advantages of gPC

- Computation of statistic moments:

$$\mathbb{E}[u] \approx \mathbb{E}[u_N] = \widehat{c}_1(x), \quad \mathbb{V}ar[u] \approx \mathbb{V}ar[u_N] = \sum_{n=2}^{N} \widehat{c}_n^2(x)$$

Advantages of gPC

- Computation of statistic moments:

$$\mathbb{E}[u] \approx \mathbb{E}[u_N] = \widehat{c}_1(x), \quad \mathbb{V}ar[u] \approx \mathbb{V}ar[u_N] = \sum_{n=2}^{N} \widehat{c}_n^2(x)$$

- High order rate of convergence for smooth problems

Advantages of gPC

- Computation of statistic moments:

$$\mathbb{E}[u] \approx \mathbb{E}[u_N] = \widehat{c}_1(x), \quad \mathbb{V}ar[u] \approx \mathbb{V}ar[u_N] = \sum_{n=2}^{N} \widehat{c}_n^2(x)$$

- High order rate of convergence for smooth problems
- The goal: efficient recover the unknown coefficients $\{\widehat{c}_n(x)\}$

Intrusive Approach: Stochastic Galerkin

$$\left\langle \phi_n(\mathbf{y}), -\nabla \cdot (\kappa(x, \mathbf{y})\nabla u) \right\rangle_{\rho(\mathbf{y})} = \left\langle \phi_n(\mathbf{y}), f(x, \mathbf{y}) \right\rangle_{\rho(\mathbf{y})}, \quad n = 1, ..., N.$$

$$\Downarrow$$

$$-\nabla \cdot [\mathbf{A}(x)\nabla \mathbf{c}] = \mathbf{f} \quad \text{Deterministic system}$$

- $\mathbf{c} = (\widehat{c}_1(x), ..., \widehat{c}_N(x))^\top, \ \mathbf{f} = (\widehat{f}_1(x), ..., \widehat{f}_N(x))^\top, \ \mathbf{A}(x) = [\mathbf{A}_{n,m}(x)]$

Intrusive Approach: Stochastic Galerkin

$$\left\langle \phi_n(\mathbf{y}), -\nabla\cdot(\kappa(x,\mathbf{y})\nabla u)\right\rangle_{\rho(\mathbf{y})} = \left\langle \phi_n(\mathbf{y}), f(x,\mathbf{y})\right\rangle_{\rho(\mathbf{y})}, \quad n = 1,...,N.$$

$$\Downarrow$$

$$-\nabla\cdot[\mathbf{A}(x)\nabla\mathbf{c}] = \mathbf{f} \quad \text{Deterministic system}$$

- $\mathbf{c} = (\widehat{c}_1(x),...,\widehat{c}_N(x))^\top$, $\mathbf{f} = (\widehat{f}_1(x),...,\widehat{f}_N(x))^\top$, $\mathbf{A}(x) = [\mathbf{A}_{n,m}(x)]$
- The corresponding components:

$$\widehat{f}_n(x) = \int_\Gamma \rho(\mathbf{y})f(x,\mathbf{y})\phi_n(\mathbf{y})d\mathbf{y} \quad n = 1,...,N.$$

$$\mathbf{A}_{n,m}(x) = \int_\Gamma \rho(\mathbf{y})\kappa(x,\mathbf{y})\phi_n(\mathbf{y})\phi_m(\mathbf{y})d\mathbf{y}, \quad 1 \le n,m \le N.$$

Intrusive Approach: Stochastic Galerkin

$$\left\langle \phi_n(\mathbf{y}), -\nabla\cdot(\kappa(x,\mathbf{y})\nabla u)\right\rangle_{\rho(\mathbf{y})} = \left\langle \phi_n(\mathbf{y}), f(x,\mathbf{y})\right\rangle_{\rho(\mathbf{y})}, \quad n = 1, ..., N.$$

$$\Downarrow$$

$$-\nabla\cdot[\mathbf{A}(x)\nabla\mathbf{c}] = \mathbf{f} \quad \text{Deterministic system}$$

- $\mathbf{c} = (\widehat{c}_1(x), ..., \widehat{c}_N(x))^\top$, $\mathbf{f} = (\widehat{f}_1(x), ..., \widehat{f}_N(x))^\top$, $\mathbf{A}(x) = [\mathbf{A}_{n,m}(x)]$
- The corresponding components:

$$\widehat{f}_n(x) = \int_\Gamma \rho(\mathbf{y}) f(x,\mathbf{y})\phi_n(\mathbf{y})d\mathbf{y} \quad n = 1, ..., N.$$

$$\mathbf{A}_{n,m}(x) = \int_\Gamma \rho(\mathbf{y})\kappa(x,\mathbf{y})\phi_n(\mathbf{y})\phi_m(\mathbf{y})d\mathbf{y}, \quad 1 \le n, m \le N.$$

- Drawbacks: coupled system, hard to solve in general.

Non-intrusive Approach: Stochastic Collocation

- Sampling the parametric space:

$$\left\{\mathbf{y}^{(\mathbf{m})}\right\}_{m=1}^{M} \xrightarrow[\text{PDE Solver}]{} \left\{u\left(x, \mathbf{y}^{(\mathbf{m})}\right)\right\}_{m=1}^{M}$$

Non-intrusive Approach: Stochastic Collocation

- Sampling the parametric space:

$$\left\{\mathbf{y}^{(\mathbf{m})}\right\}_{m=1}^{M} \xrightarrow[\text{PDE Solver}]{} \left\{u\left(x, \mathbf{y}^{(\mathbf{m})}\right)\right\}_{m=1}^{M}$$

- Impose the collocation condition:

$$u_N\left(x, \mathbf{y}^{(\mathbf{m})}\right) = \sum_{n=1}^{N} \phi_n\left(\mathbf{y}^{(\mathbf{m})}\right) \widehat{c}_n(x) \approx u\left(x, \mathbf{y}^{(\mathbf{m})}\right), \quad m = 1, ..., M.$$

Non-intrusive Approach: Stochastic Collocation

- Sampling the parametric space:

$$\left\{\mathbf{y}^{(\mathbf{m})}\right\}_{m=1}^{M} \xrightarrow[\text{PDE Solver}]{} \left\{u\left(x, \mathbf{y}^{(\mathbf{m})}\right)\right\}_{m=1}^{M}$$

- Impose the collocation condition:

$$u_N\left(x, \mathbf{y}^{(\mathbf{m})}\right) = \sum_{n=1}^{N} \phi_n\left(\mathbf{y}^{(\mathbf{m})}\right) \widehat{c}_n(x) \approx u\left(x, \mathbf{y}^{(\mathbf{m})}\right), \quad m = 1, ..., M.$$

- Consider the scalar case (drop x): approximation problem

$$\mathbf{A}\mathbf{c} \approx \mathbf{u}, \quad \mathbf{A} \in \mathbb{R}^{M\times N}, \quad \mathbf{A}_{m,n} = \phi_n\left(\mathbf{y}^{(\mathbf{m})}\right)$$

Non-intrusive Approach: Stochastic Collocation

- Sampling the parametric space:

$$\left\{\mathbf{y}^{(\mathbf{m})}\right\}_{m=1}^{M} \xrightarrow[\text{PDE Solver}]{} \left\{u\left(x,\mathbf{y}^{(\mathbf{m})}\right)\right\}_{m=1}^{M}$$

- Impose the collocation condition:

$$u_N\left(x,\mathbf{y}^{(\mathbf{m})}\right) = \sum_{n=1}^{N} \phi_n\left(\mathbf{y}^{(\mathbf{m})}\right) \widehat{c}_n(x) \approx u\left(x,\mathbf{y}^{(\mathbf{m})}\right), \quad m = 1,\dots,M.$$

- Consider the scalar case (drop x): approximation problem

$$\mathbf{Ac} \approx \mathbf{u}, \quad \mathbf{A} \in \mathbb{R}^{M\times N}, \quad \mathbf{A}_{m,n} = \phi_n\left(\mathbf{y}^{(\mathbf{m})}\right)$$

- The approximation "$\approx$" will be explained later.

of Sampling points M vs. polynomial degree N

- Sparse approximation via ℓ^1 minimization

of Sampling points M vs. polynomial degree N

- Sparse approximation via ℓ^1 minimization
  - If less information is given $M < N$, we expect a sparse approximation

$$\operatorname*{argmin}_{\mathbf{c}\in\mathbb{R}^N} \|\mathbf{c}\|_1 \quad \text{subject to} \quad \mathbf{A}\mathbf{c} = \mathbf{u}.$$

of Sampling points M vs. polynomial degree N

- Sparse approximation via ℓ^1 minimization
  - If less information is given $M < N$, we expect a sparse approximation

$$\operatorname*{argmin}_{\mathbf{c}\in\mathbb{R}^N} \|\mathbf{c}\|_1 \quad \text{subject to} \quad \mathbf{Ac} = \mathbf{u}.$$

  - The idea comes from the compressed sensing community ; care should made for choosing samples: stability & efficiency

of Sampling points M vs. polynomial degree N

- Sparse approximation via ℓ^1 minimization
  - If less information is given $M < N$, we expect a sparse approximation

$$\operatorname*{argmin}_{\mathbf{c}\in\mathbb{R}^N} ||\mathbf{c}||_1 \quad \text{subject to} \quad \mathbf{A}\mathbf{c} = \mathbf{u}.$$

  - The idea comes from the compressed sensing community ; care should made for choosing samples: stability & efficiency
- Discrete least-squares (e.g. Burkardt-Eldred '2009)

of Sampling points M vs. polynomial degree N

- Sparse approximation via ℓ^1 minimization
  - If less information is given $M < N$, we expect a sparse approximation

$$\operatorname*{argmin}_{\mathbf{c}\in\mathbb{R}^N} \|\mathbf{c}\|_1 \quad \text{subject to} \quad \mathbf{Ac} = \mathbf{u}.$$

  - The idea comes from the compressed sensing community ; care should made for choosing samples: stability & efficiency
- Discrete least-squares (e.g. Burkardt-Eldred '2009)
  - If more information is available, i.e., $M > N$, we can consider the least-squares approach

$$\mathbf{c} = \min_{\mathbf{z}\in\mathbb{R}^N} \|\mathbf{Az} - \mathbf{u}\|_2.$$

of Sampling points M vs. polynomial degree N

- Sparse approximation via ℓ^1 minimization
  - If less information is given $M < N$, we expect a sparse approximation

$$\underset{\mathbf{c}\in\mathbb{R}^N}{\operatorname{argmin}} \|\mathbf{c}\|_1 \quad \text{subject to} \quad \mathbf{A}\mathbf{c} = \mathbf{u}.$$

  - The idea comes from the compressed sensing community ; care should made for choosing samples: stability & efficiency
- Discrete least-squares (e.g. Burkardt-Eldred '2009)
  - If more information is available, i.e., $M > N$, we can consider the least-squares approach

$$\mathbf{c} = \min_{\mathbf{z}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{z} - \mathbf{u}\|_2.$$

  - Least-squares is preferred when data are polluted by noise

of Sampling points M vs. polynomial degree N

- Sparse approximation via ℓ^1 minimization
  - If less information is given $M < N$, we expect a sparse approximation

$$\operatorname*{argmin}_{\mathbf{c}\in\mathbb{R}^N} \|\mathbf{c}\|_1 \quad \text{subject to} \quad \mathbf{A}\mathbf{c} = \mathbf{u}.$$

  - The idea comes from the compressed sensing community ; care should made for choosing samples: stability & efficiency
- Discrete least-squares (e.g. Burkardt-Eldred '2009)
  - If more information is available, i.e., $M > N$, we can consider the least-squares approach

$$\mathbf{c} = \min_{\mathbf{z}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{z} - \mathbf{u}\|_2.$$

  - Least-squares is preferred when data are polluted by noise
  - Various sampling methods can be adopted, again, stability & efficiency are important

Stability results for unbounded domain (Tang-Zhou, 'SISC14)

- Stable with high probability:

$$\Pr\left\{ |||\hat{\mathbf{A}} - \mathbf{I}||| \geq \frac{5}{8} \right\} \leq M^{-\gamma} \text{ provided that } \frac{M}{\log M} > \gamma N,\ L > 5\sqrt{N}.$$

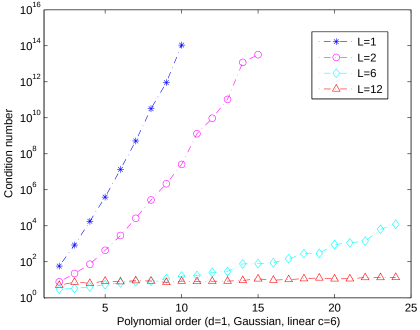


Figure: Condition number against polynomial order in 1D (Gaussian)

Allen-Cahn Eqn

Consider a simple stochastic ACE:

$$u_t(x,t,\mathbf{z}) = \delta(\mathbf{z})u_{xx} + u(1-u^2), \quad x \in (-1,1),$$
$$u(\pm 1, t, \mathbf{z}) = 0, \quad u(x,0) = u_0(x,\mathbf{z}).$$

- ... the input random vector $\mathbf{z} = (z_1, \cdots, z_d) \in \mathbb{R}^d$, where $\{\mathbf{z}_k\}_{k=1}^d$ are independent random parameters.

Allen-Cahn Eqn

Consider a simple stochastic ACE:

$$u_t(x,t,\mathbf{z}) = \delta(\mathbf{z})u_{xx} + u(1-u^2), \quad x \in (-1,1),$$
$$u(\pm 1,t,\mathbf{z}) = 0, \quad u(x,0) = u_0(x,\mathbf{z}).$$

- ... the input random vector $\mathbf{z} = (z_1, \cdots, z_d) \in \mathbb{R}^d$, where $\{\mathbf{z}_k\}_{k=1}^d$ are independent random parameters.
- ... assume each $z_k \in \Gamma_k \subset \mathbb{R}$ has an associated probability density function $\rho_k(z_k)$. Due to the independence, $\mathbf{z} \in \Gamma := \otimes_k \Gamma_k$ satisfies $\rho(\mathbf{z}) = \prod_{k=1}^d \rho_k$.

Allen-Cahn Eqn

Consider a simple stochastic ACE:

$$u_t(x,t,\mathbf{z}) = \delta(\mathbf{z})u_{xx} + u(1-u^2), \quad x \in (-1,1),$$
$$u(\pm 1, t, \mathbf{z}) = 0, \quad u(x,0) = u_0(x,\mathbf{z}).$$

- ... the input random vector $\mathbf{z} = (z_1, \cdots, z_d) \in \mathbb{R}^d$, where $\{\mathbf{z}_k\}_{k=1}^d$ are independent random parameters.
- ... assume each $z_k \in \Gamma_k \subset \mathbb{R}$ has an associated probability density function $\rho_k(z_k)$. Due to the independence, $\mathbf{z} \in \Gamma := \otimes_k \Gamma_k$ satisfies $\rho(\mathbf{z}) = \prod_{k=1}^d \rho_k$.
- Mean value and variance function:

$$\mathbb{E}[u](x,t) = \int_\Gamma \rho(\mathbf{z})u(x,t,\mathbf{z})d\mathbf{z}, \quad \mathbb{V}\text{ar}[u](x,t) = \int_\Gamma \rho(\mathbf{z})(u-\mathbb{E}[u])^2 d\mathbf{z}.$$

Free energy for ACE

- Consider a new free energy in the expectation sense, i.e.

$$\widehat{E}(u) := \mathbb{E}\left[\int_\Omega \left(\frac{\delta}{2}|\nabla u|^2 + F(u)\right) dx\right] = \int_\Gamma \int_\Omega \rho(\mathbf{z}) \left(\frac{\delta}{2}|\nabla u|^2 + F(u)\right) dx d\mathbf{z}.$$

Free energy for ACE

- Consider a new free energy in the expectation sense, i.e.

$$\widehat{E}(u) := \mathbb{E}\left[\int_{\Omega}\left(\frac{\delta}{2}|\nabla u|^2 + F(u)\right)dx\right] = \int_{\Gamma}\int_{\Omega}\rho(\mathbf{z})\left(\frac{\delta}{2}|\nabla u|^2 + F(u)\right)dxd\mathbf{z}.$$

- It can be shown that

$$\frac{d}{dt}\widehat{E}(u) \leq 0.$$

... a new guide for designing numerical schemes

Free energy for ACE

- Consider a new free energy in the expectation sense, i.e.

$$\widehat{E}(u) := \mathbb{E}\left[\int_\Omega \left(\frac{\delta}{2}|\nabla u|^2 + F(u)\right) dx\right] = \int_\Gamma \int_\Omega \rho(\mathbf{z}) \left(\frac{\delta}{2}|\nabla u|^2 + F(u)\right) dx d\mathbf{z}.$$

- It can be shown that

$$\frac{d}{dt}\widehat{E}(u) \leq 0.$$

... a new guide for designing numerical schemes

- Another interesting problem is to investigate the following free energy:

$$\overline{E}(u) := \int_\Omega \left(\frac{\delta}{2}|\nabla \bar{u}|^2 + F(\bar{u})\right) \mathrm{d}x \quad \text{with} \quad \bar{u} = \mathbb{E}[u].$$

i.e.,consider the free energy with respect to the mean

Stochastic Galerkin methods

- Expand the solution in the parametric space by polynomials

$$u(x,t,\mathbf{z}) \approx u_M = \sum_{k=1}^{M} v_k(x,t)\phi_k(\mathbf{z}), \quad \int_\Gamma \rho(\mathbf{z})\phi_k(\mathbf{z})\Phi_j(\mathbf{z})d\mathbf{z} = \delta_{kj}.$$

... $\mathbf{v} = \{v_1, \ldots, v_M\}$ are the expansion coefficients to be determined, and $\{\phi_k\}_{k=1}^M$ are orthogonal polynomials in Γ.

Stochastic Galerkin methods

- Expand the solution in the parametric space by polynomials

$$u(x,t,\mathbf{z}) \approx u_M = \sum_{k=1}^{M} v_k(x,t)\phi_k(\mathbf{z}), \quad \int_\Gamma \rho(\mathbf{z})\phi_k(\mathbf{z})\Phi_j(\mathbf{z})d\mathbf{z} = \delta_{kj}.$$

... $\mathbf{v} = \{v_1, \ldots, v_M\}$ are the expansion coefficients to be determined, and $\{\phi_k\}_{k=1}^M$ are orthogonal polynomials in Γ.

- Note

$$\mathbb{E}[u] \approx \mathbb{E}[u_M] = v_1(x,t), \qquad \mathbb{V}\text{ar}[u] \approx \mathbb{V}\text{ar}[u_M] = \sum_{k=2}^{M} v_k^2.$$

Stochastic Galerkin methods

- Expand the solution in the parametric space by polynomials

$$u(x,t,\mathbf{z}) \approx u_M = \sum_{k=1}^{M} v_k(x,t)\phi_k(\mathbf{z}), \qquad \int_\Gamma \rho(\mathbf{z})\phi_k(\mathbf{z})\Phi_j(\mathbf{z})d\mathbf{z} = \delta_{kj}.$$

  ... $\mathbf{v} = \{v_1, \ldots, v_M\}$ are the expansion coefficients to be determined, and $\{\phi_k\}_{k=1}^M$ are orthogonal polynomials in Γ.
- Note

$$\mathbb{E}[u] \approx \mathbb{E}[u_M] = v_1(x,t), \qquad \mathbb{V}\mathrm{ar}[u] \approx \mathbb{V}\mathrm{ar}[u_M] = \sum_{k=2}^{M} v_k^2.$$

- To compute **v**, perform the stochastic Galerkin projection:

$$\left\langle \frac{\partial u_M}{\partial t}, \phi_k \right\rangle_{\rho(\mathbf{z})} = \left\langle \delta(\mathbf{z})\frac{\partial^2 u_M}{\partial x^2}, \phi_k \right\rangle_{\rho(\mathbf{z})} + \left\langle u_M(1-u_M^2), \phi_k \right\rangle_{\rho(\mathbf{z})}, \quad k = 1, ..., M.$$

Stochastic Galerkin methods

- Expand the solution in the parametric space by polynomials

$$u(x,t,\mathbf{z}) \approx u_M = \sum_{k=1}^{M} v_k(x,t)\phi_k(\mathbf{z}), \quad \int_\Gamma \rho(\mathbf{z})\phi_k(\mathbf{z})\Phi_j(\mathbf{z})d\mathbf{z} = \delta_{kj}.$$

... $\mathbf{v} = \{v_1, \ldots, v_M\}$ are the expansion coefficients to be determined, and $\{\phi_k\}_{k=1}^M$ are orthogonal polynomials in Γ.

- Note

$$\mathbb{E}[u] \approx \mathbb{E}[u_M] = v_1(x,t), \qquad \mathbb{V}\mathrm{ar}[u] \approx \mathbb{V}\mathrm{ar}[u_M] = \sum_{k=2}^{M} v_k^2.$$

- To compute **v**, perform the stochastic Galerkin projection:

$$\left\langle \frac{\partial u_M}{\partial t}, \phi_k \right\rangle_{\rho(\mathbf{z})} = \left\langle \delta(\mathbf{z})\frac{\partial^2 u_M}{\partial x^2}, \phi_k \right\rangle_{\rho(\mathbf{z})} + \left\langle u_M(1-u_M^2), \phi_k \right\rangle_{\rho(\mathbf{z})}, \quad k = 1, ..., M.$$

-
$$\|u_M\|_{L^2(\Gamma\otimes D)} \leq \mathrm{e}^{2t}\|u_{0,M}\|_{L^2(\Gamma\otimes D)},$$

Stochastic Galerkin methods and stability

- $$\left\langle \frac{\partial u_M}{\partial t}, \phi_k \right\rangle_{\rho(\mathbf{z})} = \left\langle \delta(\mathbf{z}) \frac{\partial^2 u_M}{\partial x^2}, \phi_k \right\rangle_{\rho(\mathbf{z})} + \left\langle u_M(1-u_M^2), \phi_k \right\rangle_{\rho(\mathbf{z})}, \quad 1 \le k \le M,$$

  gives

  $$\mathbf{v}_t = \mathbf{A}\Delta\mathbf{v} - \mathbf{f}(\mathbf{v}),$$

  where

  $$A_{kj} = \int_\Gamma \rho(\mathbf{z})\delta(\mathbf{z})\phi_k(\mathbf{z})\phi_j(\mathbf{z})d\mathbf{z}, \quad f_k = \int_\Gamma \rho(\mathbf{z}) f(\mathbf{v}^\top \Phi)\phi_k(\mathbf{z})d\mathbf{z}, \quad 1 \le k \le M.$$

  $f(u) = u(1-u^2)$, $u_M = \mathbf{v}^\top\Phi$ with $\Phi = (\phi_1, ..., \phi_N)^\top$.

Stochastic Galerkin methods and stability

- $$\left\langle \frac{\partial u_M}{\partial t}, \phi_k \right\rangle_{\rho(\mathbf{z})} = \left\langle \delta(\mathbf{z}) \frac{\partial^2 u_M}{\partial x^2}, \phi_k \right\rangle_{\rho(\mathbf{z})} + \left\langle u_M(1-u_M^2), \phi_k \right\rangle_{\rho(\mathbf{z})}, \quad 1 \le k \le M,$$

  gives

  $$\mathbf{v}_t = \mathbf{A}\Delta\mathbf{v} - \mathbf{f}(\mathbf{v}),$$

  where

  $$A_{kj} = \int_\Gamma \rho(\mathbf{z})\delta(\mathbf{z})\phi_k(\mathbf{z})\phi_j(\mathbf{z})d\mathbf{z}, \quad f_k = \int_\Gamma \rho(\mathbf{z}) f(\mathbf{v}^\top \Phi)\phi_k(\mathbf{z})d\mathbf{z}, \quad 1 \le k \le M.$$

  $f(u) = u(1-u^2)$, $u_M = \mathbf{v}^\top \Phi$ with $\Phi = (\phi_1, ..., \phi_N)^\top$.

- $\mathbf{A}$ is positive definite

Stochastic Galerkin methods and stability

- $$\left\langle \frac{\partial u_M}{\partial t}, \phi_k \right\rangle_{\rho(\mathbf{z})} = \left\langle \delta(\mathbf{z}) \frac{\partial^2 u_M}{\partial x^2}, \phi_k \right\rangle_{\rho(\mathbf{z})} + \left\langle u_M(1-u_M^2), \phi_k \right\rangle_{\rho(\mathbf{z})}, \quad 1 \le k \le M,$$

  gives

  $$\mathbf{v}_t = \mathbf{A}\Delta\mathbf{v} - \mathbf{f}(\mathbf{v}),$$

  where

  $$A_{kj} = \int_\Gamma \rho(\mathbf{z})\delta(\mathbf{z})\phi_k(\mathbf{z})\phi_j(\mathbf{z})d\mathbf{z}, \quad f_k = \int_\Gamma \rho(\mathbf{z}) f(\mathbf{v}^\top \Phi)\phi_k(\mathbf{z})d\mathbf{z}, \quad 1 \le k \le M.$$

  $f(u) = u(1-u^2)$, $u_M = \mathbf{v}^\top \Phi$ with $\Phi = (\phi_1, ..., \phi_N)^\top$.
- $\mathbf{A}$ is positive definite
- IC and BC are given by

$$\mathbf{v}(x,0) = \mathbf{v}_0(x), \qquad \mathbf{v}(\pm 1, t, \mathbf{z}) = 0,$$

The energy law for the Galerkin system

- Consider the Galerkin system

$$\mathbf{v}_t = \mathbf{A}\mathbf{v}_{xx} - \mathbf{f}(\mathbf{v}). \tag{2}$$

We define the associated free energy as

$$\widehat{E}_{\mathbb{P}}(\mathbf{v}) := \int_{\Omega} \mathbf{v}_x^{\top}\mathbf{A}\mathbf{v}_x + \mathbb{E}\left[F(\mathbf{v}^{\top}\Phi)\right] dx$$

The energy law for the Galerkin system

- Consider the Galerkin system

$$\mathbf{v}_t = \mathbf{A}\mathbf{v}_{xx} - \mathbf{f}(\mathbf{v}). \tag{2}$$

We define the associated free energy as

$$\widehat{E}_{\mathbb{P}}(\mathbf{v}) := \int_{\Omega} \mathbf{v}_x^\top \mathbf{A}\mathbf{v}_x + \mathbb{E}\left[F(\mathbf{v}^\top \Phi)\right] dx$$

- It can be shown that $\frac{d}{dt}\widehat{E}_{\mathbb{P}}(\mathbf{v}) \leq 0.$

The energy law for the Galerkin system

- Consider the Galerkin system

$$\mathbf{v}_t = \mathbf{A}\mathbf{v}_{xx} - \mathbf{f}(\mathbf{v}). \tag{2}$$

We define the associated free energy as

$$\widehat{E}_{\mathbb{P}}(\mathbf{v}) := \int_\Omega \mathbf{v}_x^\top \mathbf{A}\mathbf{v}_x + \mathbb{E}\left[F(\mathbf{v}^\top \Phi)\right] dx$$

- It can be shown that $\frac{d}{dt}\widehat{E}_{\mathbb{P}}(\mathbf{v}) \le 0$.
- Multiplying both sides of (2) by $-\mathbf{v}_t^\top$ to obtain $-\mathbf{v}_t^\top \mathbf{A}\mathbf{v}_{xx} + \mathbf{v}_t^\top \mathbf{f}(\mathbf{v}) \le 0$. Then

$$\begin{aligned}
&\int_\Omega \left(-\mathbf{v}_t^\top \mathbf{A}\mathbf{v}_{xx} + \mathbf{v}_t^\top \mathbf{f}(\mathbf{v})\right) dx \\
&= \int_\Omega \left(\mathbf{v}_{tx}^\top \mathbf{A}\mathbf{v}_x + \sum_j \frac{\partial v_j}{\partial t} \int_\Gamma \rho(\mathbf{z}) f(u_M)\phi_j(\mathbf{z}) d\mathbf{z}\right) dx \\
&= \int_\Omega \left(\mathbf{v}_{tx}^\top \mathbf{A}\mathbf{v}_x + \sum_j \frac{\partial v_j}{\partial t} \int_\Gamma \rho(\mathbf{z}) f(u_M)\phi_j(\mathbf{z}) d\mathbf{z}\right) dx \\
&= \int_\Omega \left(\mathbf{v}_{tx}^\top \mathbf{A}\mathbf{v}_x + \int_\Gamma \rho(\mathbf{z}) f(u_M)\frac{\partial u_M}{\partial t} d\mathbf{z}\right) dx \\
&= \frac{d}{dt}\int_\Omega \left(\mathbf{v}_x^\top \mathbf{A}\mathbf{v}_x + \int_\Gamma \rho(\mathbf{z}) F(u_M) d\mathbf{z}\right) dx.
\end{aligned}$$

Monte Carlo method

- First generates randomly a sample set $\{\mathbf{z}_l\}_{l=1}^{L}$ according to $\rho(\mathbf{z})$, and then solve the random ACE for each $\mathbf{z}_l$ to obtain $u_l = u(x, t, \mathbf{z}_l)$.

Monte Carlo method

- First generates randomly a sample set $\{\mathbf{z}_l\}_{l=1}^{L}$ according to $\rho(\mathbf{z})$, and then solve the random ACE for each $\mathbf{z}_l$ to obtain $u_l = u(x, t, \mathbf{z}_l)$.
- Then any energy stable scheme can be used as the deterministic solver. Computing the sample solutions can be done in a parallel way.

Monte Carlo method

- First generates randomly a sample set $\{\mathbf{z}_l\}_{l=1}^{L}$ according to $\rho(\mathbf{z})$, and then solve the random ACE for each $\mathbf{z}_l$ to obtain $u_l = u(x, t, \mathbf{z}_l)$.
- Then any energy stable scheme can be used as the deterministic solver. Computing the sample solutions can be done in a parallel way.
- The MC approach gives approximation for the mean function

$$\mathbb{E}[u] \approx \frac{1}{L} \sum_{l=1}^{L} u_l.$$

Monte Carlo method

- First generates randomly a sample set $\{\mathbf{z}_l\}_{l=1}^{L}$ according to $\rho(\mathbf{z})$, and then solve the random ACE for each $\mathbf{z}_l$ to obtain $u_l = u(x, t, \mathbf{z}_l)$.
- Then any energy stable scheme can be used as the deterministic solver. Computing the sample solutions can be done in a parallel way.
- The MC approach gives approximation for the mean function

$$\mathbb{E}[u] \approx \frac{1}{L}\sum_{l=1}^{L} u_l.$$

- Good: the energy law is preserved

$$\frac{d}{dt}\widehat{E}_L(u) =: \frac{1}{L}\sum_{l=1}^{L}\frac{d}{dt}E(u_l) \leq 0,$$

Bad: the associated convergence rate is only one half :$1/\sqrt{L}$.

Stochastic collocation methods

- In stochastic collocation methods, one generate special samples to obtain a higher order convergence rate ... the points are chosen as the root of the associated polynomials.

Stochastic collocation methods

- In stochastic collocation methods, one generate special samples to obtain a higher order convergence rate ... the points are chosen as the root of the associated polynomials.
- For example, if uniform density is considered, then one choose the Legendre Gaussian points as samples; while if normal distribution is considered, the Hermite Gaussian points will be used as samples.

Stochastic collocation methods

- In stochastic collocation methods, one generate special samples to obtain a higher order convergence rate ... the points are chosen as the root of the associated polynomials.
- For example, if uniform density is considered, then one choose the Legendre Gaussian points as samples; while if normal distribution is considered, the Hermite Gaussian points will be used as samples.
- Suppose the $\{\mathbf{z}_k\}_{k=1}^K$ are those samples (i.e., the tensor product of Gaussian-type points), we solve the random ACE for each point $\mathbf{z}_k$

$$
\begin{aligned}
&u_t(x,t,\mathbf{z}_k) = \delta(\mathbf{z}_k)u_{xx} + u(1-u^2), \quad x \in (-1,1), \\
&u(\pm 1, t, \mathbf{z}_k) = 0, \quad u(x,0) = u_0(x,\mathbf{z}_k).
\end{aligned}
$$

Stochastic collocation methods

- Again, any energy scheme for the above equation can be used.

Stochastic collocation methods

- Again, any energy scheme for the above equation can be used.
- The Lagrange interpolation can be used to construct a global approximation

$$u_K(x,t,\mathbf{z}) = \sum_{k=1}^{K} u_k(x,t,\mathbf{z}_k)\mathbf{T}_k(\mathbf{z}).$$

Notice that the Lagrange bases $\{\mathbf{T}_k\}_{k=1}^{K}$ are of tensor-product type. the Lagrange interpolation is constructed by tensorize the one-dimensional interpolation.

Stochastic collocation methods

- Again, any energy scheme for the above equation can be used.
- The Lagrange interpolation can be used to construct a global approximation

$$u_K(x,t,\mathbf{z}) = \sum_{k=1}^{K} u_k(x,t,\mathbf{z}_k)\mathbf{T}_k(\mathbf{z}).$$

Notice that the Lagrange bases $\{\mathbf{T}_k\}_{k=1}^{K}$ are of tensor-product type. the Lagrange interpolation is constructed by tensorize the one-dimensional interpolation.

- We have

$$\frac{d}{dt}\widehat{E}_K(u(x,t,\mathbf{z})) =: \sum_{k=1}^{K} \mathbf{w}_k \frac{d}{dt} E(u(x,t,\mathbf{z}_k)) \le 0.$$

Here $\{\mathbf{w}_k\}$ are the quadrature weights associated with the Gaussian-type points

$$\mathbf{w}_k = \int_\Gamma \rho(\mathbf{z})\mathbf{T}_k(\mathbf{z})d\mathbf{z} \ge 0, \quad k = 1, ..., K.$$

Some remarks

- We can expect a high order convergence rate provided that the solution is smooth in the parametric space.

Some remarks

- We can expect a high order convergence rate provided that the solution is smooth in the parametric space.
- However, for high dimensional parametric problems, the tensor-product rule will results in a huge number of samples: suppose we have K points in each dimension, then we shall have K^d points for the d-dimensional problem.

Some remarks

- We can expect a high order convergence rate provided that the solution is smooth in the parametric space.
- However, for high dimensional parametric problems, the tensor-product rule will results in a huge number of samples: suppose we have K points in each dimension, then we shall have K^d points for the d-dimensional problem.
- This number is huge when d is large (known as the curse of dimensionality).

Some remarks

- We can expect a high order convergence rate provided that the solution is smooth in the parametric space.
- However, for high dimensional parametric problems, the tensor-product rule will results in a huge number of samples: suppose we have K points in each dimension, then we shall have K^d points for the d-dimensional problem.
- This number is huge when d is large (known as the curse of dimensionality).
- To overcome this, one may resort to the so called sparse grid rule. However, in sparse grid approach, the positivity of the weights are no longer guaranteed, and the energy stability may not hold.

Conclusions

- UQ brings new research directions for phase field models (Cahn-Hilliard, MBE, Thin Film etc)

Conclusions

- UQ brings new research directions for phase field models (Cahn-Hilliard, MBE, Thin Film etc)
- Stochastic Galerkin/collocation methods, energy law;

Conclusions

- UQ brings new research directions for phase field models (Cahn-Hilliard, MBE, Thin Film etc)
- Stochastic Galerkin/collocation methods, energy law;
- UQ brings new challenges, e.g., high dimensionality

Conclusions

- UQ brings new research directions for phase field models (Cahn-Hilliard, MBE, Thin Film etc)
- Stochastic Galerkin/collocation methods, energy law;
- UQ brings new challenges, e.g., high dimensionality
- UQ introduces new analysis, e.g., probabilistic based analysis

Conclusions

- UQ brings new research directions for phase field models (Cahn-Hilliard, MBE, Thin Film etc)
- Stochastic Galerkin/collocation methods, energy law;
- UQ brings new challenges, e.g., high dimensionality
- UQ introduces new analysis, e.g., probabilistic based analysis
- UQ can be very interdisciplinary, and often involves subjects such as scientific computing, approximation theory, probability, random matrix, compressed sensing ect

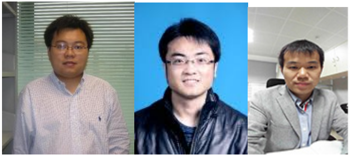
D. Li, Z. Qiao, and T. Tang, Characterizing the stabilization size for semi-implicit Fourier-spectral method to phase field equations, SIAM J. Numer. Anal., 2016

Y. Cheng, A. Kurganov, Z. Qu and T. Tang, Fast and stable explicit operator splitting methods for phase-field models, J. Comput. Phys. (2015)

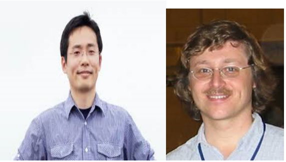
X. Feng, T. Tang and J. Yang, Long time numerical simulations for phase-field problems using p-adaptive spectral deferred correction methods, SIAM J. Sci. Comput. 37 (2015)

T. Tang and T. Zhou, Recent developments in high order numerical methods for UQ (Invited Review), Science in China, (45)2015

T. Tang, T. Zhou, Discrete least-squares in unbounded domain using random samples with applications to UQ, SISC, (36)2014.



Zhonghua Qiao (乔中华), Tao Zhou (周涛),Jiang Yang, (杨将)



Dong Li (李栋),Alex Kurganov (SUSTech)

SUSTech, Shenzhen