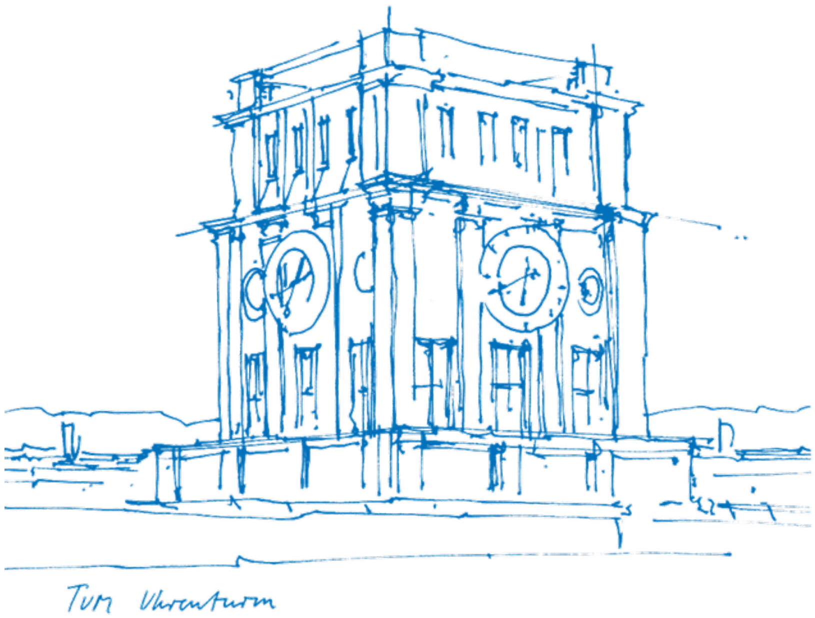
Algorithms for Uncertainty Quantification

Tobias Neckel

Lehrstuhl Informatik V

Summer Semester 2017



Lecture 7:

Polynomial chaos approximation 2: the stochastic Galerkin approach

Repetition

- polynomial chaos expansion
  - approximate quantity of interest by polynomial series
  - $f(t,\omega) \approx \sum_{n=0}^{N-1} \hat{f}_n(t)\,\phi_n(\omega)$
- orthogonal polynomials and polynomial chaos
  - inner product 0 for orthogonal polynomials
  - $< \phi_i(\omega), \phi_j(\omega) >_\rho = \delta_{ij}$
  - choose polynomial type according to input distribution
- pseudo-spectral approach
  - use quadrature rule to compute coefficients
  - $\hat{f}_n(t) = \sum_{k=0}^{K-1} f(t, x_k)\,\phi_n(x_k)\,w_k$
- model problem: damped oscillator
- multivariate case

Today's lecture

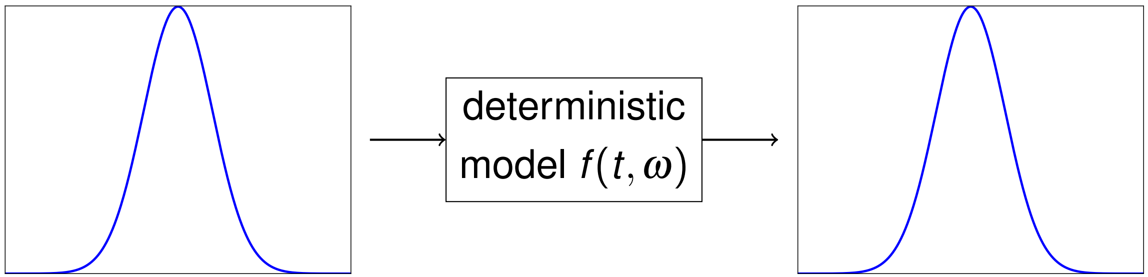
Topic

Stochastic Galerkin method

Content

- forward propagation of uncertainty
- idea of stochastic Galerkin method
- Galerkin projection
- example: damped linear oscillator
- comparison with non-intrusive methods

Forward propagation of uncertainty

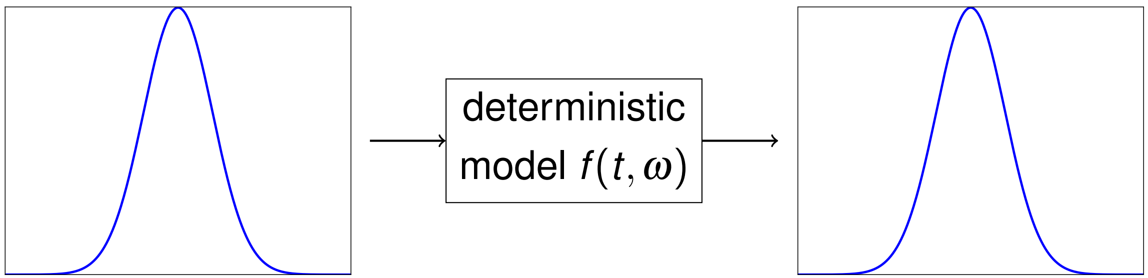


stochastic input Ω stochastic output Y

What we have

- deterministic model with solution $f(t,\omega)$
- random input variable $\Omega \sim \rho(\omega)$
- corresponding orthogonal polynomials $\phi_i(\omega)$

Forward propagation of uncertainty



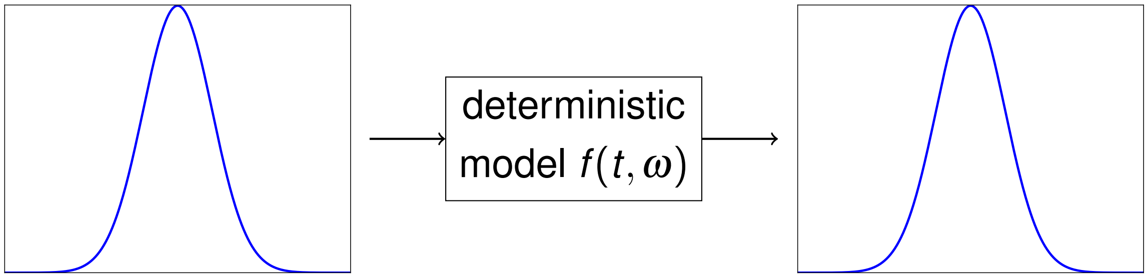
What we have

- deterministic model with solution $f(t,\omega)$
- random input variable $\Omega \sim \rho(\omega)$
- corresponding orthogonal polynomials $\phi_i(\omega)$

What we want

- stochastic output $f(t,\omega) = Y \sim p(Y)$
- quantities of interest: e.g. $\mathbb{E}[Y]$, $Var[Y]$

Forward propagation of uncertainty



stochastic input Ω stochastic output Y

Which method to use?

- remember: pseudo-spectral approach
  - write $f(t,\omega)$ as gPC expansion
  - use quadrature rule to compute coefficients
- quadrature introduces error

Stochastic Galerkin method

remember: polynomial chaos expansion

$$f(t,\omega) \approx \sum_{n=0}^{N-1} \hat{f}_n(t)\,\phi_n(\omega)$$

Idea

- do not rely on quadrature
- requires the polynomial chaos expansion of the uncertain inputs
- modify solver implementation to compute coefficients $\hat{f}_n(t)$

Properties

- faster convergence than the pseudo-spectral approach
- requires access to model/equations/code
- time-consuming modifications necessary

Galerkin projection

Analogy: Finite Elements

- formulate problem in weak form + discretize in space
- assumption: solution u is weighted sum of base of shape functions N_n

$$u(x) = \sum_n \hat{u}_n N_n(x)$$

- find best approximation to real solution
  $\rightarrow$ solve for coefficients $\hat{u}_n$

Galerkin projection

Analogy: Finite Elements

- formulate problem in weak form + discretize in space
- assumption: solution u is weighted sum of base of shape functions N_n

$$u(x) = \sum_n \hat{u}_n N_n(x)$$

- find best approximation to real solution
  $\rightarrow$ solve for coefficients $\hat{u}_n$

Stochastic Galerkin method

- solution: displacement $u(x) \rightarrow$ stochastic model output $f(t, \omega)$
- local shape functions $N_n(x) \rightarrow$ global orthogonal polynomials $\phi_n(\omega)$
- coefficients $\hat{u}_n \rightarrow$ coefficients $\hat{f}_n(t)$

stochastic Galerkin method

Steps

1. determine the polynomial chaos expansion of the uncertain inputs (this expansion is exact!)
2. write the underlying model's solution as an N^{th} order polynomial chaos expansion

$$\Omega = \sum_{m=0}^{M-1} \hat{c}_n \phi_m(\omega)$$

$$f(t,\omega) = \sum_{n=0}^{N-1} \hat{f}_n(t)\, \phi_n(\omega)$$

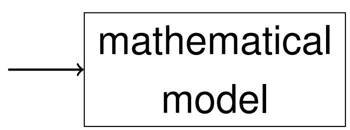
stochastic Galerkin method

Steps

1. determine the polynomial chaos expansion of the uncertain inputs (this expansion is exact!)
2. write the underlying model's solution as an N^{th} order polynomial chaos expansion
3. insert both expansions into model equations

$$\Omega = \sum_{m=0}^{M-1} \hat{c}_n \, \phi_m(\omega)$$

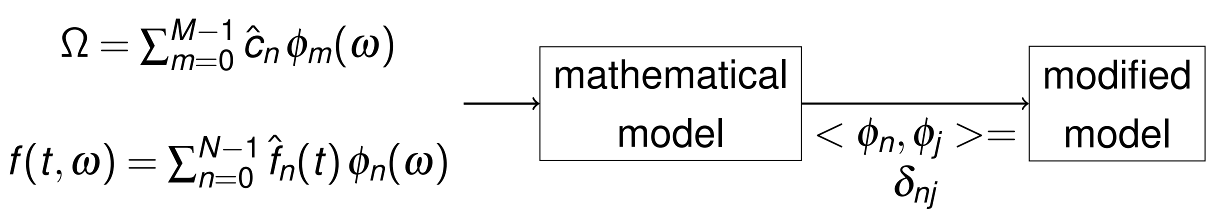
$$f(t,\omega) = \sum_{n=0}^{N-1} \hat{f}_n(t) \, \phi_n(\omega)$$



stochastic Galerkin method

Steps

1. determine the polynomial chaos expansion of the uncertain inputs (this expansion is exact!)
2. write the underlying model's solution as an N^{th} order polynomial chaos expansion
3. insert both expansions into model equations
4. use orthogonality to get a system of equations with N unknown coefficients



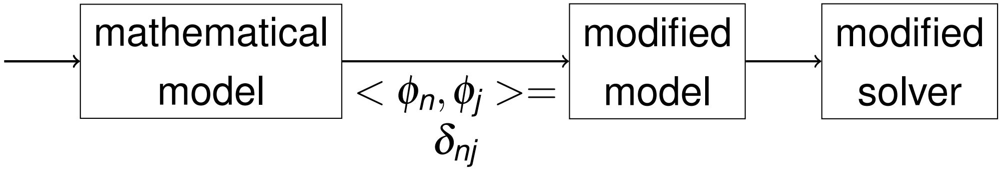
stochastic Galerkin method

Steps

1. determine the polynomial chaos expansion of the uncertain inputs (this expansion is exact!)
2. write the underlying model's solution as an N^{th} order polynomial chaos expansion
3. insert both expansions into model equations
4. use orthogonality to get a system of equations with N unknown coefficients
5. modify solver to solve new (coupled) system of equations

$$\Omega = \sum_{m=0}^{M-1} \hat{c}_n \phi_m(\omega)$$

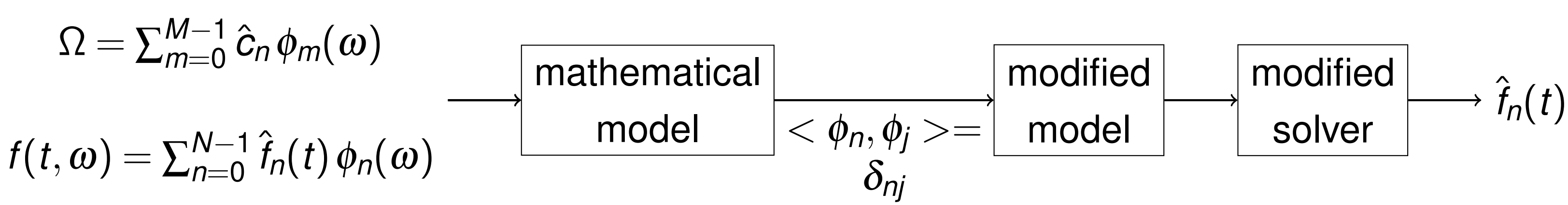
$$f(t,\omega) = \sum_{n=0}^{N-1} \hat{f}_n(t)\, \phi_n(\omega)$$



stochastic Galerkin method

Steps

1. determine the polynomial chaos expansion of the uncertain inputs (this expansion is exact!)
2. write the underlying model's solution as an N^{th} order polynomial chaos expansion
3. insert both expansions into model equations
4. use orthogonality to get a system of equations with N unknown coefficients
5. modify solver to solve new (coupled) system of equations
6. compute statistical properties from coefficients



Model problem – damped linear oscillator

System of first order ODEs

$$\begin{cases} \frac{dx}{dt}(t) = v(t) \\ \frac{dv}{dt}(t) = f\cos(\omega_O t) - cv(t) - kx(t) \\ x(0) = x_0 \\ v(0) = v_0 \end{cases}$$

- $x(t)$: position, x_0: initial position
- $v(t)$: velocity, v_0: initial velocity
- c – damping coefficient
- k – spring constant
- f – forcing amplitude
- ω_O – forcing frequency

Model problem – uncertainty input

Uncertain parameter: damping constant c

- assume c now as RV $C \sim \mathscr{U}(a,b)$
- linear transformation with $\Omega \sim \mathscr{U}(-1,1)$

$$c(\omega) = \underbrace{\frac{a+b}{2}}_{c_\mu} + \underbrace{\frac{b-a}{2}}_{c_\sigma}\omega$$

- polynomial chaos basis: legendre polynomials $\phi_i(\omega)$
- polynomial chaos expansion:

$$\begin{aligned} c &= c_\mu + c_\sigma \omega \\ &= c_\mu \phi_0(\omega) + c_\sigma \phi_1(\omega) \end{aligned}$$

Model problem – polynomial chaos

Polynomial chaos expansions

$$x(t,\omega) = \sum_{n=0}^{N-1} \hat{x}_n(t)\,\phi_n(\omega)$$
$$v(t,\omega) = \sum_{n=0}^{N-1} \hat{v}_n(t)\,\phi_n(\omega)$$

- note: coefficients depend on t, polynomials on ω
- notation from now on: $\phi_n(\omega) \to \phi_n, \qquad \hat{x}_n(t) \to \hat{x}_n, \qquad \hat{v}_n(t) \to \hat{v}_n$
- 2 steps:
  1. insert expansions into ODEs and IC
  2. transform system of equations via Galerkin ansatz and orthogonality

Model problem – initial conditions

1. insert expansions into IC

$$x(0) = x_0$$

$$\sum_{n=0}^{N-1} \hat{x}_n(0)\, \phi_n = x_0$$

Model problem – initial conditions

1. insert expansions into IC

$$x(0) = x_0$$

$$\sum_{n=0}^{N-1} \hat{x}_n(0)\,\phi_n = x_0$$

2. use Galerkin + orthogonality: inner product with $< \cdot,\ \phi_j >$

$$< \sum_{n=0}^{N-1} \hat{x}_n(0)\,\phi_n\,,\,\phi_j > \,=\, < x_0\,,\,\phi_j >$$

$$\sum_{n=0}^{N-1} \hat{x}_n(0) \underbrace{< \phi_n\,,\,\phi_j >}_{\delta_{nj}\gamma_j} = x_0 \underbrace{< \phi_0\,,\,\phi_j >}_{\delta_{0j}}$$

$$\hat{x}_j(0) = \delta_{0j}\,x_0 \quad \forall j = 0, \dots, N-1$$

Model problem – ODE component 1 (x)

1. insert expansions into ODE

$$\frac{d}{dt}x = v$$
$$\frac{d}{dt}\sum_{n=0}^{N-1}\hat{x}_n\phi_n = \sum_{n=0}^{N-1}\hat{v}_n\phi_n$$

Model problem – ODE component 1 (x)

1. insert expansions into ODE

$$\frac{d}{dt}x = v$$

$$\frac{d}{dt}\sum_{n=0}^{N-1}\hat{x}_n\phi_n = \sum_{n=0}^{N-1}\hat{v}_n\phi_n$$

2. use Galerkin + orthogonality: inner product with $< \ldots, \phi_j >$

$$< \frac{d}{dt}\sum_{n=0}^{N-1}\hat{x}_n\phi_n, \phi_j > = < \sum_{n=0}^{N-1}\hat{v}_n\phi_n, \phi_j >$$

$$\frac{d}{dt}\sum_{n=0}^{N-1}\hat{x}_n\underbrace{< \phi_n, \phi_j >}_{\delta_{nj}\gamma_j} = \sum_{n=0}^{N-1}\hat{v}_n\underbrace{< \phi_n, \phi_j >}_{\delta_{nj}\gamma_j}$$

$$\frac{d}{dt}\hat{x}_j = \hat{v}_j \quad \forall j = 0, \ldots, N-1$$

Model problem – ODE component 2 (v)

1. insert expansions into ODE

$$
\begin{aligned}
\frac{d}{dt} v &= f \cos(\omega_O t) - c\, v - k\, x \\
\frac{d}{dt} \sum_{n=0}^{N-1} \hat{v}_n \phi_n &= f \cos(\omega_O t) - \left(c_\mu \phi_0 + c_\sigma \phi_1\right) \sum_{n=0}^{N-1} \hat{v}_n \phi_n - k \sum_{n=0}^{N-1} \hat{x}_n \phi_n \\
&= f \cos(\omega_O t) - c_\mu \underbrace{\phi_0}_{=1} \sum_{n=0}^{N-1} \hat{v}_n \phi_n - c_\sigma \sum_{n=0}^{N-1} \hat{v}_n \phi_1 \phi_n - k \sum_{n=0}^{N-1} \hat{x}_n \phi_n
\end{aligned}
$$

Model problem – ODE component 2 (v)

2. use orthogonality: inner product with $< \cdot, \phi_j >$

$$< \frac{d}{dt} \sum_{n=0}^{N-1} \hat{v}_n \phi_n , \phi_j > = < f \cos(\omega_O t) , \phi_j > - < c_\mu \sum_{n=0}^{N-1} \hat{v}_n \phi_n , \phi_j >$$

$$- < c_\sigma \sum_{n=0}^{N-1} \hat{v}_n \phi_1 \phi_n , \phi_j > - < k \sum_{n=0}^{N-1} \hat{x}_n \phi_n , \phi_j >$$

Model problem – ODE component 2 (v)

2. use orthogonality: inner product with $< \cdot, \, \phi_j >$

$$< \frac{d}{dt} \sum_{n=0}^{N-1} \hat{v}_n \phi_n \, , \, \phi_j > = < f \cos(\omega_O t) \, , \, \phi_j > - < c_\mu \sum_{n=0}^{N-1} \hat{v}_n \phi_n \, , \, \phi_j >$$
$$- < c_\sigma \sum_{n=0}^{N-1} \hat{v}_n \phi_1 \phi_n \, , \, \phi_j > - < k \sum_{n=0}^{N-1} \hat{x}_n \phi_n \, , \, \phi_j >$$

$$\frac{d}{dt} \sum_{n=0}^{N-1} \hat{v}_n < \phi_n \, , \, \phi_j > = f \cos(\omega_O t) < \phi_0 , \, \phi_j > - c_\mu \sum_{n=0}^{N-1} \hat{v}_n < \phi_n \, , \, \phi_j >$$
$$- c_\sigma \sum_{n=0}^{N-1} \hat{v}_n < \phi_1 \phi_n \, , \, \phi_j > - k \sum_{n=0}^{N-1} \hat{x}_n < \phi_n \, , \, \phi_j >$$

Model problem – ODE component 2 (v) (cont'd)

2. use orthogonality: inner product with $< \cdot,\ \phi_j >$

$$\frac{d}{dt}\sum_{n=0}^{N-1}\hat{v}_n \underbrace{<\phi_n\,,\,\phi_j>}_{\delta_{nj}\gamma_j} = f\cos(\omega_O t)\underbrace{<\phi_0\,,\,\phi_j>}_{\delta_{0j}} - c_\mu \sum_{n=0}^{N-1}\hat{v}_n \underbrace{<\phi_n\,,\,\phi_j>}_{\delta_{nj}\gamma_j}$$

$$- c_\sigma \sum_{n=0}^{N-1}\hat{v}_n <\phi_1\,\phi_n\,,\,\phi_j> - k\sum_{n=0}^{N-1}\hat{x}_n \underbrace{<\phi_n\,,\,\phi_j>}_{\delta_{nj}\gamma_j}$$

Model problem – ODE component 2 (v) (cont'd)

2. use orthogonality: inner product with $< \cdot,\ \phi_j >$

$$\frac{d}{dt}\sum_{n=0}^{N-1}\hat{v}_n \underbrace{<\phi_n\,,\,\phi_j>}_{\delta_{nj}\gamma_j} = f\cos(\omega_O t)\underbrace{<\phi_0\,,\,\phi_j>}_{\delta_{0j}} - c_\mu\sum_{n=0}^{N-1}\hat{v}_n\underbrace{<\phi_n\,,\,\phi_j>}_{\delta_{nj}\gamma_j}$$

$$-\,c_\sigma\sum_{n=0}^{N-1}\hat{v}_n <\phi_1\,\phi_n\,,\,\phi_j> - k\sum_{n=0}^{N-1}\hat{x}_n\underbrace{<\phi_n\,,\,\phi_j>}_{\delta_{nj}\gamma_j}$$

$\Rightarrow$

$$\frac{d}{dt}\hat{v}_j\,\gamma_j = f\cos(\omega_O t)\,\delta_{0j} - c_\mu\hat{v}_j\,\gamma_j - c_\sigma\sum_{n=0}^{N-1}\hat{v}_n <\phi_1\,\phi_n\,,\,\phi_j> - k\,\hat{x}_j\,\gamma_j$$

$$\forall j = 0,\ldots,N-1$$

Model problem – stochastic Galerkin

Final IVP

- modifications leads to new IVP
- similar to original IVP
- 2 coupled ODEs $\rightarrow$ $2N$ coupled ODEs
- modified model solver can solve for $\hat{x}_j$, $\hat{v}_j$

$$\begin{cases} \frac{d}{dt}\hat{x}_j = \hat{v}_j \\ \frac{d}{dt}\hat{v}_j = \delta_{0j}\frac{1}{\gamma_j} f\cos(\omega_O t) - c_\mu \hat{v}_j - k\,\hat{x}_j - c_\sigma \sum_{n=0}^{N-1} \hat{v}_n \frac{<\phi_1\,\phi_n\,,\phi_j>}{\gamma_j} \\ \hat{x}_j(0) = \delta_{0j}\,x_0 \\ \hat{v}_j(0) = \delta_{0j}\,v_0 \qquad \forall j = 0,\ldots,N-1 \end{cases}$$

- expectation and variance computed as in pseudo-spectral approach

Model problem – stochastic Galerkin

Results

- $C \sim \mathscr{U}(0.08, 0.12)$
- $T = 15$
- deterministic result: $x(T) = -1.51e-01$
- stochastic Galerkin method, 3 coefficients:
  $E[x(T)] = -1.52e-01$, $\text{Var}[x(T)] = 7.80e-04$
- pseudo-spectral approach 5 nodes:
  $E[x(T)] = -1.52e-01$, $\text{Var}[x(T)] = 7.80e-04$
- Monte Carlo sampling, 100000 samples:
  $E[x(T)] = -1.53e-01$, $\text{Var}[x(T)] = 7.83e-04$

Model problem – stochastic Galerkin

Results

- $C \sim \mathcal{U}(0.08, 0.12)$
- $T = 15$
- deterministic result: $x(T) = -1.51e-01$
- stochastic Galerkin method, 3 coefficients:
  $E[x(T)] = -1.52e-01$, $\text{Var}[x(T)] = 7.80e-04$
- pseudo-spectral approach 5 nodes:
  $E[x(T)] = -1.52e-01$, $\text{Var}[x(T)] = 7.80e-04$
- Monte Carlo sampling, 100000 samples:
  $E[x(T)] = -1.53e-01$, $\text{Var}[x(T)] = 7.83e-04$

Comparison with pseudo-spectral approach

- difference in $E[x(T)]$: $2e-10$
- difference in $\text{Var}[x(T)]$: $1e-9$

Comparison with pseudo-spectral approach

stochastic Galerkin

- intrusive: need to modify model
  $\rightarrow$ model access required
  $\rightarrow$ redo for each model
- coefficients computed exactly

- modeling error:
  - series truncation

$\Rightarrow$ **more accurate**

pseudo-spectral approach

- non-intrusive: model treated as black box
  $\rightarrow$ only model output required
  $\rightarrow$ can reuse code
- coefficients approximatied via quadrature

- modeling error
  - series truncation
  - quadrature

$\Rightarrow$ **easier to use**

Comparison with pseudo-spectral approach

stochastic Galerkin

- intrusive: need to modify model
  $\rightarrow$ model access required
  $\rightarrow$ redo for each model
- coefficients computed exactly

- modeling error:
  - series truncation

$\Rightarrow$ **more accurate**

pseudo-spectral approach

- non-intrusive: model treated as black box
  $\rightarrow$ only model output required
  $\rightarrow$ can reuse code
- coefficients approximatied via quadrature

- modeling error
  - series truncation
  - quadrature

$\Rightarrow$ **easier to use**

Conclusion

- stochastic Galerkin method requires much more work
- accuracy gain must be "worth it"

Literature

- R. Ghanem, P. Spanos: *Stochastic Finite Elements: A Spectral Approach*, Springer New York, 1991
- Chapter 10 of R. C. Smith: *Uncertainty Quantification – Theory, Implementation, and Applications*, SIAM, 2014

Summary

Stochastic Galerkin method

- idea
  - insert polynomial expansions into model
  - modify model to compute coefficients
- Galerkin projection like in FEM
- comparison with non intrusive methods
  - needs model modifications
  - good convergence properties
- example: damped linear oscillator