Solving the steady state diffusion equation with uncertainty

Virginia Forstall
vhfors@gmail.com
Advisor: Howard Elman

elman@cs.umd.edu
Department of Computer Science

October 14, 2011

Abstract

The goal of this project is to efficiently solve a steady state diffusion equation with a random coefficient. Although, such equations can be solved using Monte-Carlo methods, the lengthy computation time can be constraining. Using a Karhunen-Loéve expansion allows the random coefficient to be approximated with a finite sum of random variables. This expansion combined with a Galerkin method or stochastic collocation method reduces computation time.

1 Introduction

The problem to solve is the steady state diffusion equation with a random field as the diffusion coefficient. The formal problem is

$$-\nabla \cdot (c(x,\omega)\nabla u) = f(x) \tag{1}$$

The boundary conditions are chosen to be deterministic. Specifically let $u(x,\omega) = g(x)$ on ∂D_D and $\frac{\partial u}{\partial n} = 0$ on ∂D_n. The diffusion coefficient, $c(x,\omega)$, is a random field of the form, $c = e^{a(x,\omega)}$ where $a(x,\omega)$ is also a random field. This assumption ensures c is positive for all x and thus guarantees existence and uniqueness of the solution,u. Assume the spatial domain $D \subset \mathbb{R}^2$ is bounded.

The solution, $u(x, \omega)$ is a function of the sample space. It will be the goal of this project to calculate the moments of this solution. Another meaningful result would be the cumulative distribution function of the solution.

An application of this include modeling groundwater flow through a porous medium. To do this the permiability of the medium is needed at every location in the spatial domain. However, knowing this at every point is infeasible. Instead, the permeability can be treated as a random variable at each point in the domain. Since the permeability at one point is related to the points around it, these points will be correlated and thus modeling this permeability with a random field makes sense.

In general, little is known about the structure of the diffusion coefficient. Previous work has been done where $\log(c(x, \omega)) = a(x, \omega)$ was written as an infinite series expansion of random variables [4]. This expansion, known as the Karhunen-Loéve expansion, can be truncated to provide an approximation of this field. In this project, the approximation will be found by taking the Karhunen-Loéve expansion of $c(x, \omega)$ instead.

In section 2 the algorithm and implementation are discussed. Section 4 discusses the Monte-Carlo method which will be used to verify the results. The remaining section lists the schedule for the project including milestones and deliverables.

2 Approach

The algorithm consists of two important parts. The first is to numerically approximate the random field. This will be done using a Karhunen-Loéve expansion. The second part is a stochastic finite element algorithm.

2.1 Approximating the random field

The first step is to discretize the spatial domain D. To perform the principal component analysis, the covariance at each pair of points on the spatial domain is needed. The covariance for points x and y is defined as

$$C(x, y) = \int_{\Omega} (c(x, \omega) - \mu(x))(c(y, \omega) - \mu(y)) dP(\omega) \ , \tag{2}$$

where

$$\mu(x) = \int_{\Omega} c(x, \omega) dP(\omega) \ . \tag{3}$$

A random field is wide sense stationary if its first and second moments are constant at each point on the spatial domain. Since a stationary random field will be assumed, the mean $\mu(x) = \mu \ \ \forall x$.

Since both the spatial domain and the probability space are discretized, the covariance matrix and mean are

$$C_{ij} = \frac{1}{n}\sum_{k=1}^{n}(c(x_i,\omega_k) - \hat{\mu}_i)(c(x_j,\omega_k) - \hat{\mu}_j) \tag{4}$$

$$\hat{\mu}_i = \frac{1}{n}\sum_{k=1}^{n} a(x_i,\omega_k) \tag{5}$$

where n is the number of samples.

Find the eigenpairs of the operator,

$$\mathcal{C}c(x) = \int_D C(x,y)c(y)dy = \lambda c(x)\ . \tag{6}$$

The eigenpairs can then be used to write the random field as an expansion of uncorrelated random variables, ξ,

$$c(x,\omega) = \mu(x) + \sum_{s=1}^{\infty}\sqrt{\lambda_s}c_s(x)\xi_s(\omega)\ . \tag{7}$$

This expression is known as the Karhunen-Loéve expansion. Using the first M terms of this expansion produces the approximation $c^{(M)}$ of the random field.

2.2 Stochastic finite element method

First the stochastic weak formulation of the problem is needed. The function space, $L^2(D)$ is defined as

$$L^2(D) = \{v : D \to \mathbb{R} | \int_D v(x)^2 dx < \infty\}\ . \tag{8}$$

For a probability space, Ω, $L^2(\Omega)$ is the set of random variables which have finite variance. Considering $H^1(D) = \{v(x,y) : D \to \mathbb{R} | v, \partial v/\partial x, \partial v/\partial y \in L^2(D)\}$, define the following subspaces:

$$H_0^1(D) = \{u \in H^1(D) : u = 0 \ \text{ on } \ \partial D_d\} \tag{9}$$

$$H_g^1(D) = \{u \in H^1(D) : u = g(x) \ \text{ on } \ \partial D_d\}\ . \tag{10}$$

The weak formulation is to find $u \in H_g^1(D) \times L^2(\Omega)$ such that

$$b(u,v) = l(v), \quad \forall v \in H_0^1(D) \times L^2(\Omega) \tag{11}$$

where

$$b(u,v) = \int_\Omega \int_D c(x,\omega) \nabla u(x,\omega) \cdot \nabla v(x,\omega) dx \, dP(\omega) \tag{12}$$

$$l(v) = \int_\Omega \int_D f(x) v(x,\omega) dx \, dP(\omega) \tag{13}$$

2.2.1 Stochastic Collocation method

The random field has been approximated by a finite number of random varaibles using the KL expansion. Therefore, the solution, $u(x,\omega)$ can be written as $u(x,\vec{\xi})$ where $\vec{\xi}$ is a vector of length M of the uncorrelated random variables. Thus the probability space becomes $L^2(\Gamma) = \{X : \Gamma \rightarrow \mathbb{R} | E[X^2] < \infty\}$. A set of points, $Z_{c_p} \subset \Gamma$, are known as collocation points. Because of the construction of Γ, each point is described by a vector of length M. The number of these points in this set, c_p, is a parameter of the method. It is the maximum degree of the polynomials that form the basis of the finite basis of $L^2(\Gamma)$. A greater number of points increases the accuracy of the results, but naturally adds computation time. Keeping this number significantly below the number of realizations in the Monte-Carlo method yields the savings in computation time. At each one of these points, we have a single value for the field, c. Thus, at each point the stochastic weak formulation becomes the deterministic weak formulation. That is find $u \in H_g^1$ s.t.

$$\int_D c_i^{(M)}(x) \nabla u_i(x) \cdot \nabla v_i(x) dx = \int_D f(x) v_i(x) dx \;\; \forall v_i \in H_0^1(D) \; . \tag{14}$$

Thus there are c_p problems that can each be solved with the deterministic Galerkin method. The last step is to use Lagrange interpolation to find an approximation of u at points in the sample space that are not in the set of collocation points. For particular sets of collocation points Lagrange interpolation can be used. Consider the basis of the probability space, $\{\pi_1(\vec{\xi}), ..., \pi_{c_p}(\vec{\xi})\}$ where π are polynomials which satisfy

$$\pi_j(\vec{z_i}) = \delta_{ij} \text{for} 0 \leq i,j \leq c_p \tag{15}$$

where $\vec{z_i}$ is a collocation point.

For the Lagrange interpolation

$$\pi_k(\vec{\xi}) = \prod_{i=1}^{m} L_k^i(\xi_i) \tag{16}$$

where

$$L_k^i(\xi_i) = \prod_{j \neq k} \frac{\xi_i - z_j}{z_k - z_j} \tag{17}$$

where z_j are the interpolation points, whose choice is related to the choice of collocation points. The ξ_i are the M random variables used in the KL expansion.

The final approximation to the solution according to this method is

$$u_{h,c_p}(x, \vec{\xi}) = \sum_{k=1}^{c_p} u_k(x)\pi_k(\vec{\xi}) \; . \tag{18}$$

2.2.2 Stochastic Galerkin method

The stochastic Galerkin method finds the stochastic discretization using a basis formed with polynomials of the M random variables kept in the KL expansion. Increasing the degree of the polynomials improves the approximation. This produces a larger matrix that can be solved using the deterministic Galerkin finite element method. Even thought the matrix is larger, computation time can still be managed by taking advantage of structure and/or sparsity. For examples, see [2],[4].

3 Implementation

Several things have to be addressed in using this algorithm. The first concern is determining the best way to discretize the spatial domain. When it comes to the expansion of infinite random variables, it is unclear what these random variables will look like. This will obviously depend on the structure of the random field, $c(x, \omega)$. Also, how does the number of terms kept in the series affect the accuracy of the solution? Decisions about the method for solving the equation also need to be made, including which of the two methods outlined above to use and whether or not preconditioning algorithms will be used.

The code for this algorithm and the Monte-Carlo algorithm will be written using Matlab. Both scripts will be run on a desktop computer with Unix operating system and 1.9 GB RAM.

4 Testing

Validation of the method will be performed using Monte-Carlo simiulations. The coefficient, $c(x, \omega)$, will be sampled q times. The number of samples q will be greater than c_p in the stochastic collocation method. These samples c_i for $i = 1, .., q$ each produce a finite element problem and will require many more solves of the discrete problem. Denote the solution of this problem be $u_h^i(x)$. The moments of these solutions can then be found. For example, the mean

$$E_q[u_h] = \frac{1}{q} \sum_{i=1}^{q} u_h^i(x) \ . \tag{19}$$

We can write error from the acutal solution as $e = E[u] - E_q[u_h] = E[u] - E[u_h] + E[u_h] - E_q[u_h]$. The first difference, $E[u] - E[u_h]$ approaches 0 as $h \to 0$ assuming that finite basis of the Galerkin method follows the usual requirements. The second difference, $E[u_h] - E_q[u_h] \to 0$ as $q \to \infty$. We also know that $E[u]$ is the solution to the deterministic problem when $E[c]$ is used as the diffusion coefficient, thus we can verify these results. This approach will likely take many iterations. One example, in [1], took between $q = 10^4$ and $q = 10^6$ to converge.

5 Validation

Validation of the Monte Carlo code will be by confirming that the Monte-Carlo solution converges to the solution of the deterministic equation which uses the mean of the random field $c(x, \omega)$. The validation of the method of this project will then be obtained by comparing the first and second moments with those of the Monte-Carlo solutions.

6 Schedule

The proposed schedule for the project is listed below.

Stage 1: October-Late November

- Clearly define the problem (what assumptions will be made?)
- Build the covariance matrix
- Compute the eigennodes

- Write code which generates Monte-Carlo solutions

Stage 2: Late November-December

- Test Monte Carlo code and run simulations
- Begin construction of the principal components analysis

Stage 3: December- late February

- Complete construction of PCA
- Write solution method

Stage 4: March - April

- Run numerical method
- Analyze accuracy and validity of the method
- Draw conclusions

Deliverables include a code that calculates the first and second moments of the solution to Equation 1 using Monte-Carlo method, a code that calculates the moments using the KL expansion technique outlined in section 2, and a comparison of the computational cost of the two methods. Mid year and end of the year progress reports will also be included.

7 References

References

[1] A. Gordon, Solving stochastic elliptic partial differential equations via stochastic sampling methods , M.S Thesis, University of Manchester, 2008.

[2] C.E. Powell and H.E. Elman, Block-diagonal preconditioning for spectral stochastic finite-element systems, IMA Journal of Numerical Analysis, 29, (2009), 350-375.

[3] C. Schwab and R. Todor, Karhunen-Loéve approximation of random fields by generalized fast multipole methods, Journal of Computational Physics, 217, (2006), 100-122.

[4] E. Ullmann, H. C. Elman, and O. G. Ernst, Efficient iterative solvers for stochastic Galerkin discretization of log-transformed random diffusion problems, 2011.

[5] X. Wan and G. Karniadakis, Solving elliptic problems with non-Gaussian spatially-dependent random coefficients, Computational Methods in Applied Mechanical Engineering, 198, (2009), 1985-1995.

[6] D. Xiu, Numerical Methods for Stochastic Computations, Princeton University Press, New Jersey, 2010.

[7] D. X and J. Hesthaven, High-order collocation methods for differential equations with random inputs, SIAM Jounral on Scientific Computing, 27, (2005), 1118-1139.