Test

Trieu Nhat Thanh Nguyen

February 2024

1 1D Stochastic Bar Problem

Consider a 1D stochastic bar problem where we aim to find the displacements along the bar.

1.1 Problem Setup

The problem parameters are as follows:

$$L = 1.0 \quad \text{(Length of the bar)}$$
$$E = 1.0 \quad \text{(Young's modulus)}$$
$$A = 1.0 \quad \text{(Cross-sectional area)}$$

We discretize the bar into 10 elements.

1.2 Polynomial Chaos Expansion

We employ a Polynomial Chaos Expansion (PCE) with 5 terms. To define the multivariate Hermite polynomials, we use:

$$\text{hermite\_polynomial}(n, x) = H_n(x\sqrt{2})$$

where $H_n(x)$ denotes the n-th order Hermite polynomial.

1.3 Bilinear Form

The bilinear form for the stochastic Galerkin method is given by:

$$\text{bilinear\_form}(\xi, \eta) = E \cdot A \cdot (1 + \xi) \cdot (1 + \eta)$$

where ξ and η are the local coordinates within the element.

1.4 Quadrature Integration

We perform quadrature using Gaussian quadrature to compute the weights for the integrals.

1.5 Action Matrix-Vector Product

The action matrix-vector product with Dirichlet boundary condition is computed as follows:

$$\text{action\_matrix}(u) = \sum_{i=1}^{N} \sum_{j=1}^{M} h \cdot \text{bilinear\_form}(\xi_i, \eta_j) \cdot u_j \cdot \text{weight}$$

The action matrix-vector product is computed using the following procedure:

For each element i:

Calculate the element length: $h = \dfrac{L}{\text{num\_elements}}$

Calculate local coordinate: $\xi = \left(\dfrac{2i+1}{2 \cdot \text{num\_elements}} \right) - \dfrac{1}{\sqrt{3}}$

For each term j in the expansion:

Compute $\eta = \sqrt{2} \cdot \xi \cdot \sqrt{j+1}$

Compute weight using quadrature: $\text{weight} = \displaystyle\int_{-1}^{1} \text{hermite\_polynomial}(j, x)^2 \, dx$

Update the action matrix: $\text{result}[j] += h \cdot \text{bilinear\_form}(\xi, \eta) \cdot u[j] \cdot weight$

Block-Matrix Structure of A

The action matrix A can be represented as a block-matrix structure where each block A_{ij} corresponds to the contribution of element i to term j in the polynomial chaos expansion.

Let's denote the number of elements as N and the number of terms in the polynomial chaos expansion as M. Then A can be represented as:

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1M} \\ A_{21} & A_{22} & \cdots & A_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \cdots & A_{NM} \end{bmatrix}$$

Each block A_{ij} is a square matrix representing the contribution of element i to term j in the polynomial chaos expansion.

Linear System Representation

The linear system $Au = b$ can then be represented as:

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1M} \\ A_{21} & A_{22} & \cdots & A_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \cdots & A_{NM} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{NM} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{NM} \end{bmatrix}$$

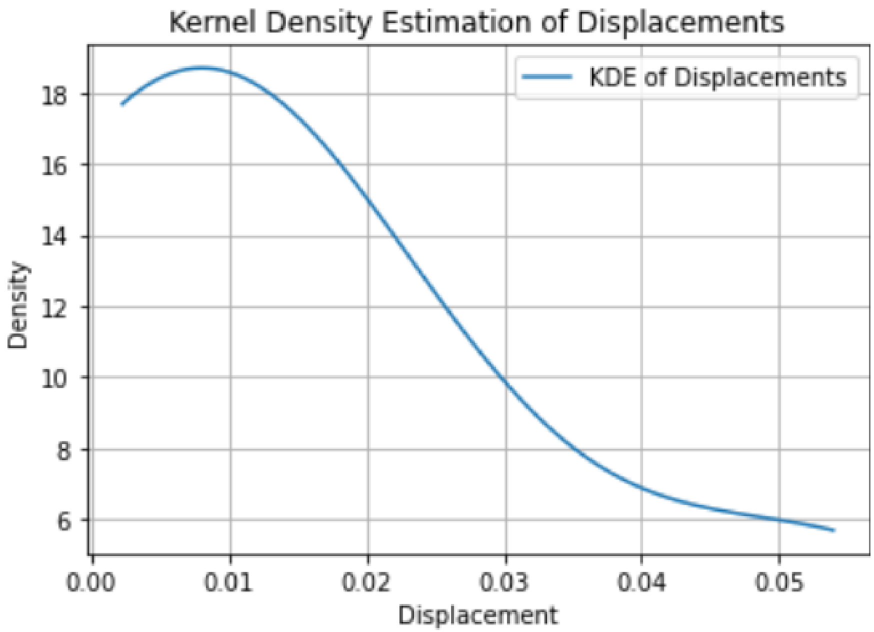

Figure 1: Kernel Density Estimation of Displacements

where:

- u_i represents the displacement at element i for each term in the polynomial chaos expansion.
- b_i represents the external force acting on element i for each term in the polynomial chaos expansion.

1.6 Conjugate Gradient Method

We employ the Conjugate Gradient method to solve the linear system.

1.7 Results

The displacements are calculated as follows:

$$\text{Displacements (U): } U = \text{conjugate\_gradient}(f, \mathbf{0})$$

1.8 Visualization

We visualize the Kernel Density Estimation (KDE) of displacements using matplotlib(Figure 1).