SFEM for Linear Static Problems

Dr. Nebojša Mojsilović
Institute of Structural Engineering
ETH Zurich

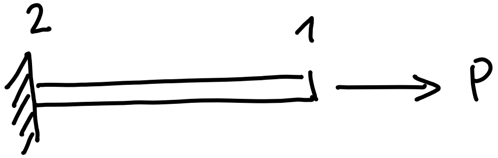
Introduction

- Implement reliability analysis for problems with implicit performance functions by the sensitivity based approach
- Sensitivity based approach requires the gradient of the performance function
- Classical perturbation method (chain rule) is applied for that task

Introduction

- Value of the performance function at each iteration – from deterministic structural analysis (FEM)
- Gradient of the performance function at each iteration – from sensitivity analysis

Introductory Example



$$K\,F = U$$

$$F = \begin{Bmatrix} P \\ 0 \end{Bmatrix}$$

$$U = \begin{Bmatrix} U_1 \\ U_2 \end{Bmatrix}$$

$$\begin{bmatrix} \frac{AE}{L} & 0 \\ 0 & \frac{AE}{L} \end{bmatrix} \begin{Bmatrix} U_1 \\ U_2 \end{Bmatrix} = \begin{Bmatrix} P \\ 0 \end{Bmatrix} \Rightarrow U = \begin{Bmatrix} PL/AE \\ 0 \end{Bmatrix}$$

SFEM-based Reliability Analysis

- FORM (First Order Reliability Method) Method 2 – search for the design point
- Performance function G(Y) and its gradient ∇G(Y) are needed at each iteration point in the standard normal space Y to search for minimum distance point on the limit state
- So, first we need to calculate G(Y)

Calculation Steps for G(Y)

- Assembling of matrices K and F
- Solving the equation system for displacements U
- Calculating the vector of desired response quantities S using

$$S=Q^T U+S_0$$

- Q^T relates U with S and S_0 is S for U=0
- Calculate the response function g(X) as

$$g(X)=g\{R(X),S(X)\}$$

- R is the vector of resistance variables and S is the vector of response quantities X is the vector of original random variables

Calculation Steps for G(Y)

- Original random variables X are transformed to equivalent uncorrelated reduced normal variables Y

$$Y=T(X)$$

$$\begin{Bmatrix} Y_1 \\ Y_2 \end{Bmatrix} = \begin{Bmatrix} T(x_1) \\ T(x_2) \end{Bmatrix} = \begin{Bmatrix} \dfrac{x_1 - \mu_1^N}{\sigma_1^N} \\ \\ \dfrac{x_2 - \mu_2^N}{\sigma_2^N} \end{Bmatrix}$$

- Computing of ∇G(Y) in order to implement FORM algorithm (classical perturbation – chain rule)

Classical Perturbation for ∇G(Y)

$$\frac{\partial G}{\partial Y_1} = \frac{\partial g}{\partial X_1} \frac{\partial X_1}{\partial Y_1} \qquad \frac{\partial X_1}{\partial Y_1} = \sigma_1^N$$

$$g(x) = g\{R(x), S(x)\}$$

$$\frac{\partial G}{\partial Y_1} = \frac{\partial g}{\partial R} \frac{\partial R}{\partial x_1} \frac{\partial x_1}{\partial Y_1} + \frac{\partial g}{\partial S} \frac{\partial S}{\partial x_1} \frac{\partial x_1}{\partial Y_1}$$

Classical Perturbation for ∇G(Y)

$\varepsilon_x > \varepsilon_0$ (probability)

ε_0 – limiting strain
P – nodal force
E – modulus of elasticity

$\Big\}$ uncorrelated random variables → we search for $\nabla G(Y)$

performance function $g(R,S) = \varepsilon_0 - \varepsilon_x$ → $R = \varepsilon_0$, $S = \varepsilon_x$

S depends on <u>P and E</u>

failure for $g < 0$

Classical Perturbation for ∇G(Y)

$$X = \begin{Bmatrix} X_1 \\ X_2 \\ X_3 \end{Bmatrix} = \begin{Bmatrix} \varepsilon_0 \\ P \\ E \end{Bmatrix} \qquad Y = \begin{Bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{Bmatrix} \qquad \Delta G(Y) = \begin{Bmatrix} \partial G / \partial Y_1 \\ \partial G / \partial Y_2 \\ \partial G / \partial Y_3 \end{Bmatrix}$$

$$\frac{\partial G}{\partial Y_1} = \frac{\partial g}{\partial R}\frac{\partial R}{\partial X_1}\frac{\partial X_1}{\partial Y_1} = \frac{\partial g}{\partial \varepsilon_0}\frac{\partial \varepsilon_0}{\partial \varepsilon_0}\sigma_1^N = \sigma_1^N \quad \text{since} \quad \frac{\partial S}{\partial X_1} = \frac{\partial S}{\partial \varepsilon_0} \equiv 0$$

$$\frac{\partial G}{\partial Y_2} = \frac{\partial g}{\partial S}\frac{\partial S}{\partial X_2}\frac{\partial X_2}{\partial Y_2} = \frac{\partial g}{\partial \varepsilon_x}\frac{\partial \varepsilon_x}{\partial P}\sigma_2^N = -1 \cdot \frac{\partial \varepsilon_x}{\partial P}\sigma_2^N \quad \text{since} \quad \frac{\partial R}{\partial X_2} = \frac{\partial \varepsilon_0}{\partial P} \equiv 0$$

$$\frac{\partial G}{\partial Y_3} = \frac{\partial g}{\partial S}\frac{\partial S}{\partial X_3}\frac{\partial X_3}{\partial Y_3} = \frac{\partial g}{\partial \varepsilon_x}\frac{\partial \varepsilon_x}{\partial E}\sigma_3^N = -1 \cdot \frac{\partial \varepsilon_x}{\partial E}\sigma_3^N \quad \text{since} \quad \frac{\partial R}{\partial X_3} = \frac{\partial \varepsilon_0}{\partial E} \equiv 0$$

Classical Perturbation for ∇G(Y)

$$\varepsilon_x = \frac{u_1 - u_2}{L} = P/AE \quad \text{thus}$$

$$\frac{\partial G}{\partial Y_2} = -1 \cdot \frac{\partial}{\partial P}\left(\frac{P}{AE}\right) \cdot \sigma_2^N = -\frac{1}{AE}\sigma_2^N$$

$$\frac{\partial G}{\partial Y_3} = -1 \cdot \frac{\partial}{\partial E}\left(\frac{P}{AE}\right) \cdot \sigma_3^N = -\frac{-P}{AE^2}\sigma_3^N = \frac{P}{AE^2}\sigma_3^N$$

$$\Rightarrow \nabla G(Y) = \begin{Bmatrix} \sigma_1^N \\ -\frac{1}{AE}\sigma_2^N \\ \frac{P}{AE^2}\sigma_3^N \end{Bmatrix}$$

Matrix Formulation for Multiple Variables

$Y = T(X) = A + B(X)$

$A = \begin{Bmatrix} -\mu_1^N / \sigma_1^N \\ -\mu_2^N / \sigma_2^N \end{Bmatrix}$ $B = \begin{bmatrix} 1/\sigma_1^N & 0 \\ 0 & 1/\sigma_2^N \end{bmatrix}$ $X = \begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix}$

FORM 2 $\rightarrow$ A and B change from iteration to iteration

$\nabla G(Y) = (B^{-1})^T \nabla g_x (R,S) =$ $J \rightarrow$ Jacobian

$= (B^{-1})^T \left[J_r \nabla g_r (R,S) + J_s \nabla g_s (R,S) \right]$

Matrix Formulation for Multiple Variables

∇g_x, ∇g_r, ∇g_s $\rightarrow$ gradient vectors with respect to X, R and S

$J_r = \dfrac{\partial R}{\partial X}$ $\quad J_s = \dfrac{\partial S}{\partial X}$ $\rightarrow$ Jacobian matrices of the transformations $R = R(X)$ and $S = S(X)$

$\nabla g_r(R,S)$, $\nabla g_s(R,S)$ and $J_R \left(\partial r_i / \partial x_j \right)$ $\rightarrow$ easily calculated

since $g(R,S)$ is expressed in terms of R and S and

since R are related to X

Matrix Formulation for Multiple Variables

not so easy to calculate dc $J_s = \partial s / \partial x$

$$\frac{\partial s}{\partial x_j} = \frac{\partial Q^T}{\partial x_j} U + Q^T \frac{\partial U}{\partial x_j} + \frac{\partial s_0}{\partial x_j} \qquad \text{since} \quad s = Q^T U + s_0$$

$$\frac{\partial U}{\partial x_j} = K^{-1} \frac{\partial F}{\partial x_j} + \frac{\partial K^{-1}}{\partial x_j} F \qquad \text{since} \quad U = K^{-1} F$$

$$\frac{\partial K^{-1}}{\partial x_j} = -K^{-1} \frac{\partial K}{\partial x_j} K^{-1} \qquad \text{since} \quad K \cdot K^{-1} = 1$$

Matrix Formulation for Multiple Variables

thus $\frac{\partial U}{\partial x_j} = K^{-1}\frac{\partial F}{\partial x_j} - K^{-1}\frac{\partial K}{\partial x_j}\underbrace{K^{-1}F}_{U} = K^{-1}\frac{\partial F}{\partial x_j} - K^{-1}\frac{\partial K}{\partial x_j}U$

further on $\frac{\partial S}{\partial x_j} = \frac{\partial Q^T}{\partial x_j}U + Q^T K^{-1}\left\{\frac{\partial F}{\partial x_j} - \frac{\partial K}{\partial x_j}U\right\} + \frac{\partial S_0}{\partial x_j}$.

now $\nabla G(Y)$ can be calculated!

Implementation of SFEM

K, F, Q and S_0 → constructed by assembling

$\frac{\partial K}{\partial x_j}, \frac{\partial F}{\partial x_j}, \frac{\partial Q}{\partial x_j}$ and $\frac{\partial S_0}{\partial x_j}$ → similarly by assembling

Implementation of SFEM

- Partial derivative stiffness matrices
- Partial derivative load vectors
- Routines for above differentiation and for the effective storage are needed
- Storage space itself is also needed
- Memory needed is increasing rapidly with increasing number of basic random variables considered

Conclusion

Steps in the process of reliability analysis:

- Formulation of limit states
- Definition of random variables
- Identification of load-type and resistance-type variables
- Computation of performance function gradients
- Implementation of FORM