CE394M: 1D-Finite Element Method

Krishna Kumar

University of Texas at Austin

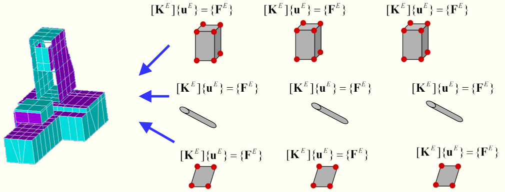
krishnak@utexas.edu

February 7, 2019

Overview

1. FEM workflow
2. 1D FEM
3. Assembly
4. Errors in the FEM
5. A case-study of a FE failure

Finite Element Analysis



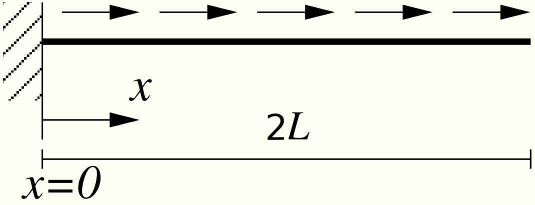
$$[\mathbf{K}]\{\mathbf{u}\} = \{\mathbf{F}\} \longrightarrow \{\mathbf{u}\} = [\mathbf{K}]^{-1}\{\mathbf{F}\}$$

Finite Element Analysis

FEM is a systematic procedure for approximating differential equations.
For any problem in any spatial dimension it follows the same steps:

1. Identify the equation of interest
2. Cast the equation of interest in a weak form
3. Select a finite element type
4. Construct the element matrix and vector
5. Assemble the global matrix and vector and apply boundary conditions
6. Solve the system of linear equations

1D Finite Element Analysis of a cantilever beam

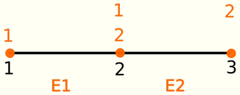


1D cantilever beam

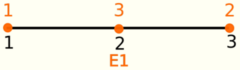
Assume L as unit length $L = 1$. Unit force $f = 1$.

1D Finite Element Analysis of a cantilever beam

What element should be used?



Linear elements



Qudrilateral element

1D discretization of a cantilever beam

1D FEM: Shape functions and derivatives

Shape function **N**:

$$\mathbf{N} = \begin{bmatrix} 1 - \frac{x}{L} & \frac{x}{L} \end{bmatrix}$$

$$\mathbf{N} = \begin{bmatrix} 1 - x & x \end{bmatrix}$$

B is the derivatives of the shape functions:

$$\mathbf{B} = \begin{bmatrix} \frac{dN_1(x)}{dx} & \frac{dN_2(x)}{dx} \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \frac{-1}{L} & \frac{1}{L} \end{bmatrix} = \begin{bmatrix} -1 & 1 \end{bmatrix}$$

In matrix format: $u_h = \mathbf{N}\mathbf{a_e}$ and $\epsilon_h = \mathbf{B}\mathbf{a_e}$.

1D FEM: Stiffness and force

Element stiffness k_e:

$$k_e = \int \mathbf{B}^T EA\, \mathbf{B}\, \mathrm{d}x = \int_0^L \begin{bmatrix} \frac{dN_1}{dx} \\ \frac{dN_2}{dx} \end{bmatrix} EA \begin{bmatrix} \frac{dN_1}{dx} & \frac{dN_2}{dx} \end{bmatrix} dx$$

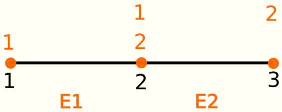
$$k_e = EA \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Right-hand side vector b_e is:

$$b_e = \int \mathbf{N}^T f\, \mathrm{d}x = \int_0^L \begin{bmatrix} -\frac{x}{L} + 1 \\ \frac{x}{L} \end{bmatrix} = \begin{bmatrix} -\frac{x^2}{2L} + x \\ \frac{x^2}{2L} \end{bmatrix}_0^L = \begin{bmatrix} -\frac{x^2}{2} + x \\ \frac{x^2}{2} \end{bmatrix}_0^1$$

$$b_e = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix}$$

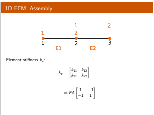
1D FEM: Assembly



Element stiffness k_e:

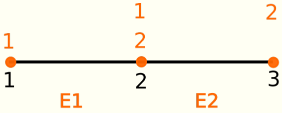
$$k_e = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}$$

$$= EA \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$



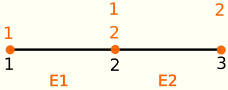
To perform a FE analysis, the element matrices and vectors are computed for each element and inserted into the global stiffness matrix **K** and global right-hand side vector **b**. The system $\mathrm{Ka} = \mathrm{b}$ is then solved to yield **a**. The process of inserting the element matrices and vectors into their global counterparts is known as assembly. A local (element) degree of freedom corresponds to a global degree of freedom, and an entry in a local matrix or vector is copied to its corresponding position in the global matrix.

1D FEM: Assembly



	element 1		element 2	
Local dof	1	2	1	2
Global dof	1	2	2	3

1D FEM: Global stiffness matrix



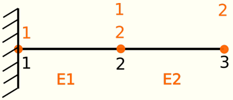
Global stiffness **K**:

$$k_e = \begin{bmatrix} k_{11}^1 & k_{12}^1 & 0 \\ k_{21}^1 & k_{22}^1 + k_{11}^2 & k_{12}^2 \\ 0 & k_{21}^2 & k_{22}^2 \end{bmatrix}$$

$$= EA \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1+1 & -1 \\ 0 & -1 & 1 \end{bmatrix} = EA \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

Where superscripts in k_{ij}^e denotes element numbering.

Boundary conditions



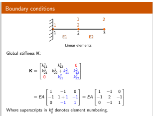
Linear elements

Global stiffness **K**:

$$\mathbf{K} = \begin{bmatrix} k_{11}^1 & k_{12}^1 & 0 \\ k_{21}^1 & k_{22}^1 + k_{11}^2 & k_{12}^2 \\ 0 & k_{21}^2 & k_{22}^2 \end{bmatrix}$$

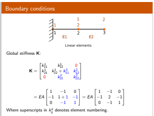
$$= EA \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1+1 & -1 \\ 0 & -1 & 1 \end{bmatrix} = EA \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

Where superscripts in k_{ij}^e denotes element numbering.



Neumann (force) boundary conditions appear naturally in the weak form of the equilibrium equation, and therefore do not require special consideration. In the context of variational methods, they are sometimes referred to as '*natural boundary conditions*'.

However, **Dirichlet (displacement) boundary conditions** still need to be applied. If we fail to apply proper displacement boundary conditions, the stiffness matrix will be singular, meaning that there is no unique solution. In the context of variational methods, displacement boundary conditions are often referred to as '*essential boundary conditions*'.



There are two approaches that are commonly used to apply displacement boundary conditions. One is to modify terms in the global stiffness matrix and place the displacement boundary condition in the right-hand side vector, and the other is to eliminate the degrees of freedom associated with a displacement boundary condition from the global system of equations.

Applying boundary conditions: Approach I

Consider a global system that has already been assembled by adding the contribution of each element:

$$\begin{bmatrix} K_{11} & K_{12} & 0 & 0 & 0 \\ K_{21} & K_{22} & K_{23} & 0 & 0 \\ 0 & K_{32} & K_{33} & K_{34} & 0 \\ 0 & 0 & K_{43} & K_{44} & K_{45} \\ 0 & 0 & 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

To apply a displacement of g at node one ($a_1 = g$):
we can zero the first row of the matrix, place a 'one' on the diagonal and set $b_1 = g$,

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ K_{21} & K_{22} & K_{23} & 0 & 0 \\ 0 & K_{32} & K_{33} & K_{34} & 0 \\ 0 & 0 & K_{43} & K_{44} & K_{45} \\ 0 & 0 & 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} g \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

Applying boundary conditions: Approach I

Consider a global system that has already been assembled by adding the contribution of each element:

$$\begin{bmatrix} K_{11} & K_{12} & 0 & 0 & 0 \\ K_{21} & K_{22} & K_{23} & 0 & 0 \\ 0 & K_{32} & K_{33} & K_{34} & 0 \\ 0 & 0 & K_{43} & K_{44} & K_{45} \\ 0 & 0 & 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

To apply a displacement of g at node one ($a_1 = g$):
we can zero the first row of the matrix, place a 'one' on the diagonal and set $b_1 = g$,

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ K_{21} & K_{22} & K_{23} & 0 & 0 \\ 0 & K_{32} & K_{33} & K_{34} & 0 \\ 0 & 0 & K_{43} & K_{44} & K_{45} \\ 0 & 0 & 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} g \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

Multiplying the top row of the matrix by the vector **a**, we can see that solving this system of equations will lead to $a_1 = g$. There is however a significant drawback to this technique: if the matrix was originally symmetric, we have destroyed the symmetry. We will not to be able to use any specialised linear solvers that exploit symmetry of a matrix, which will possibly double the solution time. For $g = 0$ we could also zero the first column and preserve symmetry, but this is not possible for $g \neq 0$.

Applying boundary conditions: Approach II

Prescribe the displacement at nodes 1 & 2, all entries in rows one and two in **K** and **b** will be equal to zero:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & K_{32} & K_{33} & K_{34} & 0 \\ 0 & 0 & K_{43} & K_{44} & K_{45} \\ 0 & 0 & 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

Since the g terms are known, we can take them to the right-hand side of the equation:

$$\begin{bmatrix} K_{33} & K_{34} & 0 \\ K_{43} & K_{44} & K_{45} \\ 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} b_3 \\ b_4 \\ b_5 \end{bmatrix} - \begin{bmatrix} 0 & K_{32} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}$$

Applying boundary conditions: Approach II

Prescribe the displacement at nodes 1 & 2, all entries in rows one and two in **K** and **b** will be equal to zero:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & K_{32} & K_{33} & K_{34} & 0 \\ 0 & 0 & K_{43} & K_{44} & K_{45} \\ 0 & 0 & 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

Since the g terms are known, we can take them to the right-hand side of the equation:

$$\begin{bmatrix} K_{33} & K_{34} & 0 \\ K_{43} & K_{44} & K_{45} \\ 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} b_3 \\ b_4 \\ b_5 \end{bmatrix} - \begin{bmatrix} 0 & K_{32} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}$$

The second approach is to eliminate the degrees of freedom where the displacement is prescribed. Firstly, recall that wherever a displacement condition is applied, the weight function must be equal to zero.

Now we have smaller matrix and any symmetry is preserved. This technique can be applied element-wise during assembly of the global matrix and it is used by finite element programs aimed at solid and structural mechanics. It is however more complicated to program than the first approach.

Global system of equations

Assemble the global system of equations:

$$\mathbf{K} = \begin{bmatrix} k_{11}^1 & k_{12}^1 & 0 \\ k_{21}^1 & k_{22}^1 + k_{11}^2 & k_{12}^2 \\ 0 & k_{21}^2 & k_{22}^2 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$\mathbf{K} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 1 \\ 1/2 \end{bmatrix}$$

Applying boundary conditions

Assemble the global system of equations:

$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 1 \\ 1/2 \end{bmatrix}$$

We need to apply the boundary condition $u(0) = 0$, which requires that $a_1 = 0$. The simplest way to impose this condition is to delete the first row and column of the stiffness matrix:

$$\begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/2 \end{bmatrix}$$

Solving this system we have: $a^T = (1/EA)\,[0 \quad 1.5 \quad 2.0]$.

Analytical solution of a 1D cantilever beam

The Euler–Bernoulli equation describes the relationship between the beam's deflection and the applied load:

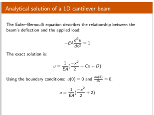
$$-EA\frac{d^2u}{dx^2} = 1$$

The exact solution is:

$$u = \frac{1}{EA}\left(\frac{-x^2}{2} + Cx + D\right)$$

Using the boundary conditions: $u(0) = 0$ and $\frac{du(2)}{dx} = 0$.

$$u = \frac{1}{EA}\left(\frac{-x^2}{2} + 2\right)$$



Analytical solution of a 1D cantilever beam

The Euler–Bernoulli equation describes the relationship between the beam's deflection and the applied load:

$$-EA\frac{d^2u}{dx^2} = 1$$

The exact solution is:

$$u = \frac{1}{EA}(\frac{-x^2}{2} + Cx + D)$$

Using the boundary conditions: $u(0) = 0$ and $\frac{du(2)}{dx} = 0$.

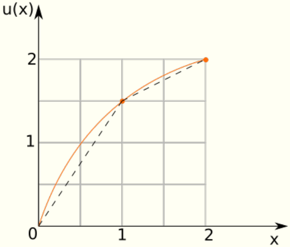
$$u = \frac{1}{EA}(\frac{-x^2}{2} + 2)$$

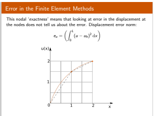
The finite element solution is exact at the nodes, but it is not exact between the nodes. This '*exactness*' is a feature of finite element methods in one dimension, but it does not carry over to higher dimensions.

Error in the Finite Element Methods

This nodal '*exactness*' means that looking at error in the displacement at the nodes does not tell us about the error. Displacement error norm:

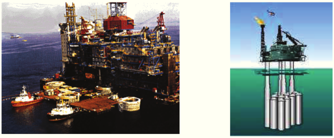
$$e_u = \left(\int_0^L (u - u_h)^2 \, \mathrm{d}x\right)$$



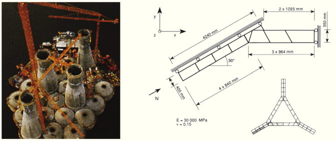


The finite element solution is exact at the nodes, but it is not exact between the nodes. This '*exactness*' is a feature of finite element methods in one dimension, but it does not carry over to higher dimensions.
Since the exact solution is quadratic, the finite element solution would be exact if you use elements which are quadratic or higher.

Sleipner A offshore platform sprang leak

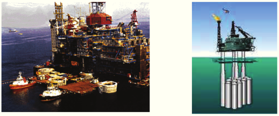


Platform with a reinforced concrete base structure.

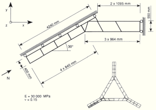


Concrete substructure during manufacturing on shore. Finite element mesh of a tricell detail. A tricell is a triangular concrete frame placed where three cells meet.

Sleipner A offshore platform sprang leak



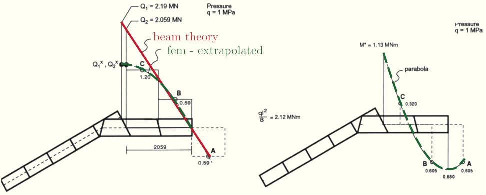
Platform with a reinforced concrete base structure.



Concrete substructure during manufacturing on shore. Finite element mesh of a tricell detail. A tricell is a triangular concrete frame placed where three cells meet.

Using the FE program NASTRAN, the shear stresses in the tricells were under-estimated by 47%. First, the chosen finite element mesh was exceedingly coarse so that the finite element shear stress was significantly too small.

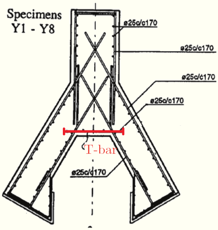
Sleipner A offshore platform sprang leak

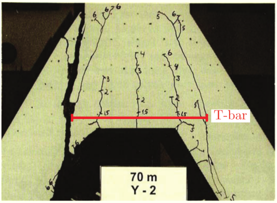


Comparison of the finite element shear forces (left) and moments (right) with the beam theory solution.

Second, the shear stresses at the boundary have been quadratically extrapolated using the shear forces at points A, B and C. We know however from beam theory that the shear force distribution is linear so that the shear force at the boundary is underestimated by $\approxeq$ 40%

Sleipner A offshore platform sprang leak





Experimental investigation of the tricell failure. As can be concluded from the failure mode the T-bar length is too short.

To make matters worse, the necessary reinforcement was automatically dimensioned based on the FE results without any checking by an engineer.