Numerical modeling of rock deformation:

06 FEM Introduction

Stefan Schmalholz
schmalholz@erdw.ethz.ch
NO E 61

AS 2009, Thursday 10-12, NO D 11
Introduction

• We have learned that geodynamic processes can be described with partial differential equations (PDE’s) derived from the concepts of continuum mechanics.
• These PDE’s describing geodynamic processes are often complicated and non-linear and closed-form analytical solutions do not exist.
• Therefore, the PDE’s are transformed into a system of linear equations which can be solved by a computer.
• The finite element method (FEM) is one method to transform systems of PDE’s into systems of linear equations. Other methods are for example the finite difference method, the finite volume method, the spectral method or the boundary element method.
• In this lecture the basic concept of the FEM is introduced by transforming a simple ordinary equation into a system of linear equations.
The model equation

We consider the equation

\[ A \frac{\partial^2 u(x)}{\partial x^2} + B = 0 \]

This equation is a second order, inhomogeneous ordinary differential equation.

Physically it describes for example:

- A) The deflection of a stretched wire under a lateral load
  \( u = \text{deflection}, \ A = \text{tension}, \ B = \text{lateral load} \)
- B) Steady state heat conduction with radiogenic heat production
  \( u = \text{temperature}, \ A = \text{thermal diffusivity}, \ B = \text{heat production} \)
- C) Viscous fluid flow between parallel plates
  \( u = \text{velocity}, \ A = \text{viscosity}, \ B = \text{pressure drop} \)
The analytical solution

We find the general analytical solution by simple integration

\[
\frac{\partial^2 u(x)}{\partial x^2} = -\frac{B}{A} \quad \text{integrate } \int dx
\]

\[
\frac{\partial u(x)}{\partial x} = -\frac{B}{A} x + C_1 \quad \text{integrate } \int dx
\]

\[
u(x) = -\frac{B}{2A} x^2 + C_1 x + C_2
\]

The two integration constants are found by two boundary conditions

\[
u(x = 0) = 0 \Rightarrow C_2 = 0
\]

\[
u(x = L) = 0 \Rightarrow C_1 = \frac{BL}{2A}
\]

The special solution is

\[
u(x) = -\frac{B}{2A} x^2 + \frac{BL}{2A} x
\]

The special solution is

\[
u(x) = -\frac{B}{2A} x^2 + \frac{BL}{2A} x
\]
The weak formulation - 1

The original equation

\[ A \frac{\partial^2 u(x)}{\partial x^2} + B = 0 \]

Multiply with test function

\[ N(x) \left[ A \frac{\partial^2 u(x)}{\partial x^2} + B \right] = 0 \]

Integrate spatially

\[ \int_0^L N(x) \left[ A \frac{\partial^2 u(x)}{\partial x^2} + B \right] dx = 0 \]

\[ \int_0^L \left( N(x)A \frac{\partial u(x)}{\partial x} \right) dx - \int_0^L \left( \frac{\partial N(x)}{\partial x} A \frac{\partial u(x)}{\partial x} - N(x)B \right) dx = 0 \]

The product rule of differentiation

\[ \frac{\partial}{\partial x} (ab) = \frac{\partial a}{\partial x} b + a \frac{\partial b}{\partial x} \]

\[ a = N(x), \quad b = A \frac{\partial u(x)}{\partial x} \]
The weak formulation - 2

\[ N(x)A \frac{\partial u(x)}{\partial x}\Bigg|_0^L - \int_0^L \left( \frac{\partial N(x)}{\partial x} A \frac{\partial u(x)}{\partial x} - N(x)B \right) dx = 0 \]

The first term requires our function \( u(x) \) at the boundaries \( x=0 \) and \( x=L \) and is given by the boundary conditions. Therefore, we do not need to test \( u(x) \) at these boundary points.

The above equation is the weak form of our original equations, because it has weaker constraints on the differentiability on our solution (only first derivative compared to second derivative in original equation).
The finite element approximation - 1

We approximate our unknown solution as a sum of known functions, the so-called shape functions, multiplied with unknown coefficients. The shape functions are one at only one nodal point and zero at all other nodal points, so that the coefficient corresponding to a certain shape function is the solution at the node.
We approximate our unknown solution as a sum of known functions, the so-called shape functions, multiplied with unknown coefficients. The shape functions are one at only one nodal point and zero at all other nodal points, so that the coefficient corresponding to a certain shape function is the solution at the node.

\[ u(x) \approx u_i N(x)_i + u_{i+1} N(x)_{i+1} = \left\{N(x)_i, N(x)_{i+1}\right\} \begin{pmatrix} u_i \\ u_{i+1} \end{pmatrix} = N^T u \]
We approximate our unknown solution as a sum of known functions, the so-called shape functions, multiplied with unknown coefficients. The shape functions are one at only one nodal point and zero at all other nodal points, so that the coefficient corresponding to a certain shape function is the solution at the node.

\[
\begin{align*}
    u(x) & \approx u_i N(x)_i + u_{i+1} N(x)_{i+1} \\
    &= \left\{ N(x)_i \quad N(x)_{i+1} \right\} \begin{bmatrix} u_i \\ u_{i+1} \end{bmatrix} = N^T u
\end{align*}
\]

\[
N(x)_i = 1 - \frac{x-x_i}{x_{i+1} - x_i} = 1 - \frac{x}{dx}
\]

\[
N(x)_{i+1} = \frac{x-x_i}{x_{i+1} - x_i} = \frac{x}{dx}
\]
The finite element approximation - 2

\[ \int_0^{dx} \frac{\partial}{\partial x} \left\{ \frac{N(x)}{A} \right\} \begin{bmatrix} \frac{\partial N(x)}{\partial x} & \frac{\partial N(x)}{\partial x} \\ \frac{\partial N(x)}{\partial x} & \frac{\partial N(x)}{\partial x} \end{bmatrix} \begin{bmatrix} u_i \\ u_{i+1} \end{bmatrix} - \begin{bmatrix} N(x) \\ N(x) \end{bmatrix} B dx = 0 \]

\[ \int_0^t \left( \frac{\partial N(x)}{\partial x} A \frac{\partial u(x)}{\partial x} - N(x) B \right) dx = 0 \]

\[ u(x) \approx \{ N(x) \} \begin{bmatrix} u_i \\ u_{i+1} \end{bmatrix} \]

The FEM where the shape functions are identical to the test (weight) functions is called Galerkin method after Boris Galerkin.

\[ Ku - F = 0 \]
The finite element approximation - 3

\[ \mathbf{Ku} - \mathbf{F} = 0 \]

\[ \mathbf{K} = \int_0^d \left( \begin{array}{cc} \frac{\partial N(x)_i}{\partial x} & \frac{\partial N(x)_i}{\partial x} \\ \frac{\partial N(x)_{i+1}}{\partial x} & \frac{\partial N(x)_{i+1}}{\partial x} \end{array} \right) \begin{array}{c} d\mathbf{x}_A \\ d\mathbf{x}_{i+1} \end{array} \]

\[ A d\mathbf{x} = A \int_0^d \frac{1}{d\mathbf{x}^2} d\mathbf{x} = \]

\[ \mathbf{F} = \int_0^d \left( \begin{array}{c} N(x)_i \\ N(x)_{i+1} \end{array} \right) B d\mathbf{x} \]

\[ \mathbf{K}_{ii} = \int_0^d \frac{\partial N(x)_i}{\partial x} \frac{\partial N(x)_i}{\partial x} A d\mathbf{x} = A \int_0^d \frac{1}{d\mathbf{x}^2} d\mathbf{x} = \]

\[ A \frac{d^2}{d\mathbf{x}^2} \left|_{0}^{d} \right. = A \left( \frac{d\mathbf{x}}{d\mathbf{x}^2} - \frac{0}{d\mathbf{x}^2} \right) = A \frac{d\mathbf{x}}{d\mathbf{x}} \]

\[ \frac{\partial N(x)_i}{\partial x} = - \frac{\partial N(x)_{i+1}}{\partial x} = -\frac{1}{d\mathbf{x}} \]

\[ N(x)_i = 1 - \frac{x - x_i}{x_{i+1} - x_i} = 1 - \frac{x}{d\mathbf{x}} \]

\[ N(x)_{i+1} = \frac{x - x_i}{x_{i+1} - x_i} = \frac{x}{d\mathbf{x}} \]
The finite element approximation - 4

Local system for element 1

\[
\begin{pmatrix}
-\frac{A}{dx} & 0 & 0 \\
\frac{A}{dx} & -\frac{A}{dx} & 0 \\
0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
-\begin{pmatrix}
-\frac{Bdx}{2} \\
\frac{Bdx}{2} & 0
\end{pmatrix}
= \begin{pmatrix}
0 \\
0
\end{pmatrix}
\]

Local system for element 2

\[
\begin{pmatrix}
0 & 0 & 0 \\
0 & -\frac{A}{dx} & \frac{A}{dx} \\
0 & \frac{A}{dx} & -\frac{A}{dx}
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
-\begin{pmatrix}
0 \\
-\frac{Bdx}{2} \\
\frac{Bdx}{2}
\end{pmatrix}
= \begin{pmatrix}
0 \\
0
\end{pmatrix}
\]

Global system for sum of all elements

\[
\begin{pmatrix}
-\frac{A}{dx} & \frac{A}{dx} & 0 \\
\frac{A}{dx} & -2\frac{A}{dx} & \frac{A}{dx} \\
0 & \frac{A}{dx} & -\frac{A}{dx}
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
-\begin{pmatrix}
\frac{Bdx}{2} \\
-\frac{Bdx}{2} \\
\frac{Bdx}{2}
\end{pmatrix}
= \begin{pmatrix}
0 \\
0
\end{pmatrix}
\]
The finite element approximation - 5

Global system without boundary conditions

\[
\begin{pmatrix}
-\frac{A}{dx} & \frac{A}{dx} & 0 \\
\frac{A}{dx} & -2\frac{A}{dx} & \frac{A}{dx} \\
0 & \frac{A}{dx} & -\frac{A}{dx}
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
= \begin{pmatrix}
-\frac{Bdx}{2} \\
-Bdx \\
2-Bdx
\end{pmatrix}
\]

Implementing boundary conditions:
\[u_1 \text{ and } u_3 = 0\]

\[
\begin{pmatrix}
1 & 0 & 0 \\
\frac{A}{dx} & -2\frac{A}{dx} & \frac{A}{dx} \\
0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
= \begin{pmatrix}
0 \\
-Bdx \\
0
\end{pmatrix}
\]
The finite element approximation

Global system without boundary conditions

\[
\begin{bmatrix}
-\frac{A}{dx} & \frac{A}{dx} & 0 \\
\frac{A}{dx} & -2\frac{A}{dx} & \frac{A}{dx} \\
0 & \frac{A}{dx} & -\frac{A}{dx}
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
- \begin{bmatrix}
\frac{Bdx}{2} \\
-\frac{Bdx}{2}
\end{bmatrix}
= \begin{bmatrix}0 \\
0
\end{bmatrix}
\]

Implementing boundary conditions: \( u_1 \) and \( u_3 = 0 \)

\[
\begin{bmatrix}
1 & 0 & 0 \\
\frac{A}{dx} & -2A & \frac{A}{dx} \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
= \begin{bmatrix}0 \\
-Bdx \\
0
\end{bmatrix}
\]

Calculate solution using \( L=10 \) (\( dx=L/2=5 \)), \( A=1 \), \( B=-1 \)

\[
\begin{bmatrix}
1 & 0 & 0 \\
1 & -2 & 1 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
= \begin{bmatrix}0 \\
5 \\
0
\end{bmatrix}
\]

Analytical solution at \( x=5 \)

\[
u(x) = -\frac{B}{2A}x^2 + \frac{BL}{2A}x
\]

\[
u(x = 5) = \frac{1}{2}5^2 - \frac{10}{2}5 = -12.5
\]
Programming finite elements - 1

Nine finite elements

1 2 3 4 5 6 7 8 9 10

Global nodes
Local nodes

Which nodes belong to what element?
Introduce a node matrix!

The NODE matrix assigns to each local element the corresponding global node

1 2

\[
\begin{align*}
1 & \rightarrow \begin{cases} 
1 & \rightarrow 1 \\
2 & \rightarrow 2 
\end{cases} \\
3 & \rightarrow \begin{cases} 
3 & \rightarrow 3 \\
4 & \rightarrow 4 
\end{cases} \\
5 & \rightarrow \begin{cases} 
5 & \rightarrow 5 \\
6 & \rightarrow 6 
\end{cases} \\
6 & \rightarrow \begin{cases} 
6 & \rightarrow 6 \\
7 & \rightarrow 7 
\end{cases} \\
7 & \rightarrow \begin{cases} 
7 & \rightarrow 7 \\
8 & \rightarrow 8 
\end{cases} \\
8 & \rightarrow \begin{cases} 
8 & \rightarrow 8 \\
9 & \rightarrow 9 
\end{cases} \\
9 & \rightarrow \begin{cases} 
9 & \rightarrow 9 \\
10 & \rightarrow 10 
\end{cases}
\end{align*}
\]
Programming finite elements - 2

Nine finite elements

Global nodes
Local nodes

Local system for finite element 4

\[
\begin{bmatrix}
A & A \\
\frac{A}{dx} & \frac{A}{dx}
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2
\end{bmatrix} =
\begin{bmatrix}
-Bdx \\
\frac{Bdx}{2}
\end{bmatrix}
\]

Ten nodes
The global stiffness matrix is the sum of all local stiffness matrixes. In our code we will first setup a global stiffness matrix \( K_{\text{Global}} \) full of zeros. We will then

1) loop through the finite elements,
2) identify where the local element is positioned within the global matrix and
3) add the local stiffness matrix at the correct position to the global matrix.

\[
\begin{array}{c}
\begin{bmatrix}
\frac{A}{dx} & \frac{A}{dx} \\
\frac{A}{dx} & -\frac{A}{dx}
\end{bmatrix}
\end{bmatrix}
+ \begin{bmatrix}
\frac{A}{dx} & \frac{A}{dx} \\
\frac{A}{dx} & -\frac{A}{dx}
\end{bmatrix}
+ \begin{bmatrix}
\frac{A}{dx} & \frac{A}{dx} \\
\frac{A}{dx} & -\frac{A}{dx}
\end{bmatrix}
+ \cdots = K_{\text{Global}}
\end{array}
\]

Exactly the same is done for the right hand side vector \( \mathbf{F} \).
> restart;
> # FEM
# Derivation of small matrix for 1D 2nd order steady state
> # Number of nodes per element
nonel := 2:
> # Shape functions
N[1] := (dx-x)/dx:
N[2] := x/dx:
> # FEM approximation
u := sum( t[j]*N[j],j=1..nonel):
> # Weak formulation of 1D equation
for i from 1 to nonel do
Eq_weak[i] := int( -A*diff(u,x)*diff(N[i],x) + B*N[i], x=0..dx);
od:
> # Create element matrixes for conductive and transient terms
K:=matrix(nonel,nonel,0):
F:=matrix(nonel,1,0):
for i from 1 to nonel do
  for j from 1 to nonel do
    K[i,j] := coeff(Eq_weak[i], t[j]):
  od:
  F[i,1] := -coeff(Eq_weak[i], B)*B:
od:
> # Display K and F
matrix(K);
matrix(F);

\[
\int_0^L \left( \frac{\partial N(x)}{\partial x} A \frac{\partial u(x)}{\partial x} - N(x)B \right) dx = 0
\]
Numerical modeling of rock deformation: FEM Introduction. Stefan Schmalholz, ETH Zurich

The FEM Matlab code

In the numerical solution all parameters can be made variable easily, while it gets more difficult to find analytical solutions for variable parameters, e.g., A and B.
FEM applications

Obtain the algebraic equations for each element (this is easy!)

Put all the element equations together

\[
[K^E] \{u^E\} = \{F^E\}
\]

Taken from MIT lecture, de Weck and Kim
FEM element types

1D

2D

3D

Figure 6.3. Typical finite element geometries in one through three dimensions.

Taken from unknown web script
FEM applications

Shortening of the continental lithosphere with:
(i) viscoelastoplastic rheology, (ii) transient temperature field, (iii) viscous heating and (iv) gravity.
FEM applications

Three-dimensional viscous folding

Time=0.13212 My