Introduction to Finite Element Modelling in Geosciences

Basics of the FE method

Boris Kaus (kaus@erdw.ethz.ch)
Dave May (may@erdw.ethz.ch)

FS 2011

The model equation

We consider the equation

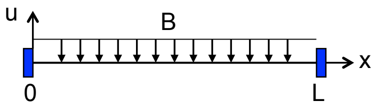
$$A\frac{\partial^2 u(x)}{\partial x^2} + B = 0$$

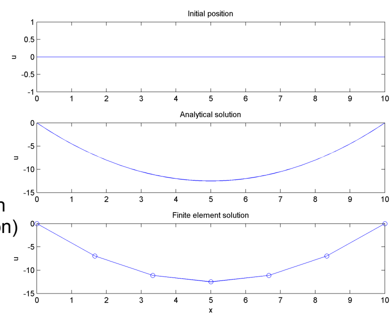
This equation is a first order, inhomogeneous ordinary differential equation.

Physically it describes for example:

- 1) The deflection of a stretched wire under a lateral load
  (u = deflection, A = tension, B = lateral load)
- 2) Steady state heat conduction with radiogenic heat production
  (u = temperature, A = thermal diffusivity, B = heat production)
- 3) Viscous fluid flow between parallel plates
  (u = velocity, A = viscosity, B = pressure gradient)

Case A)





The analytical solution

We find the general analytical solution by integration

$$\frac{\partial^2 u(x)}{\partial x^2} = -\frac{B}{A} \quad \text{Integrate } \int_x dx$$

$$\frac{\partial u(x)}{\partial x} = -\frac{B}{A}x + C_1 \quad \text{Integrate } \int_x dx$$

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

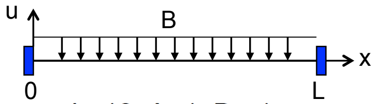
The two integration constants are found by two boundary conditions

$$u(x=0) = 0 \Rightarrow C_2 = 0$$

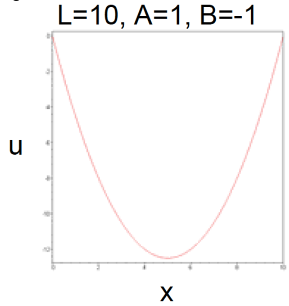
$$u(x=L) = 0 \Rightarrow C_1 = \frac{BL}{2A}$$

The special solution is

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



L=10, A=1, B=-1



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

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

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

Integrate by parts

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

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

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

The product rule of differentiation

$$\frac{\partial}{\partial x}(ab) = \frac{\partial a}{\partial x}b + a\frac{\partial b}{\partial x} \quad \Rightarrow \quad a\frac{\partial b}{\partial x} = \frac{\partial}{\partial x}(ab) - \frac{\partial a}{\partial x}b$$

$$a = N(x), \quad b = A\frac{\partial u(x)}{\partial x}$$

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. Yet, here we specified the boundary conditions u(x=0)=0 and u(x=L)=0. Therefore, we do not need to test u(x) at these boundary points.

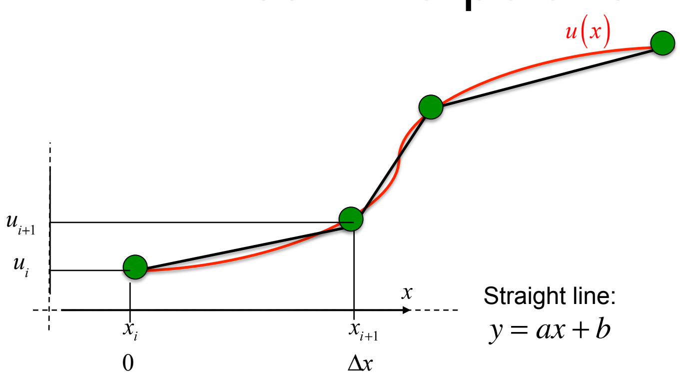
$$\int_0^L \left(\frac{\partial N(x)}{\partial x} A \frac{\partial u(x)}{\partial x} - N(x)B \right) dx = 0$$

The original equation

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

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).

Linear interpolation



Straight line:

$$y = ax + b$$

$$u(x) \approx \hat{u}(x) = \frac{u_{i+1} - u_i}{\Delta x} x + u_i$$

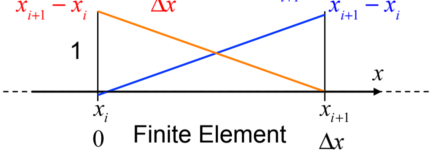
$$\hat{u}(x) = \frac{x}{\Delta x} u_{i+1} + \left(1 - \frac{x}{\Delta x}\right) u_i$$

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.

$$\boxed{u(x) \approx u_i N(x)_i + u_{i+1} N(x)_{i+1}} = \{N(x)_i \quad N(x)_{i+1}\}\begin{Bmatrix} u_i \\ u_{i+1} \end{Bmatrix} = \mathbf{N}^{\mathbf{T}}\mathbf{u}$$

$$N(x)_i = 1 - \frac{x - x_i}{x_{i+1} - x_i} = 1 - \frac{x}{\Delta x} \qquad N(x)_{i+1} = \frac{x - x_i}{x_{i+1} - x_i} = \frac{x}{\Delta x}$$



The finite element approximation - 2

$$\int_0^{\Delta x} \frac{\partial}{\partial x}\begin{Bmatrix} N(x)_i \\ N(x)_{i+1} \end{Bmatrix} A \frac{\partial}{\partial x}\left[\{N(x)_i \quad N(x)_{i+1}\}\begin{Bmatrix} u_i \\ u_{i+1} \end{Bmatrix}\right] - \begin{Bmatrix} N(x)_i \\ N(x)_{i+1} \end{Bmatrix} B dx = 0$$

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

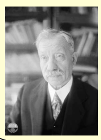
$$u(x) \approx \{N(x)_i \quad N(x)_{i+1}\}\begin{Bmatrix} u_i \\ u_{i+1} \end{Bmatrix}$$

$$\int_0^{\Delta x} \begin{Bmatrix} \frac{\partial N(x)_i}{\partial x} \\ \frac{\partial N(x)_{i+1}}{\partial x} \end{Bmatrix} A \left\{\frac{\partial N(x)_i}{\partial x} \quad \frac{\partial N(x)_{i+1}}{\partial x}\right\}\begin{Bmatrix} u_i \\ u_{i+1} \end{Bmatrix} - \begin{Bmatrix} N(x)_i \\ N(x)_{i+1} \end{Bmatrix} B\, dx = 0$$

$$\int_0 \begin{Bmatrix} \frac{\partial N(x)_i}{\partial x}\frac{\partial N(x)_i}{\partial x} & \frac{\partial N(x)_i}{\partial x}\frac{\partial N(x)_{i+1}}{\partial x} \\ \frac{\partial N(x)_{i+1}}{\partial x}\frac{\partial N(x)_i}{\partial x} & \frac{\partial N(x)_{i+1}}{\partial x}\frac{\partial N(x)_{i+1}}{\partial x} \end{Bmatrix} A \begin{Bmatrix} u_i \\ u_{i+1} \end{Bmatrix} - \begin{Bmatrix} N(x)_i \\ N(x)_{i+1} \end{Bmatrix} B\, dx = 0$$

$$\int_0 \begin{Bmatrix} \frac{\partial N(x)_i}{\partial x}\frac{\partial N(x)_i}{\partial x} & \frac{\partial N(x)_i}{\partial x}\frac{\partial N(x)_{i+1}}{\partial x} \\ \frac{\partial N(x)_{i+1}}{\partial x}\frac{\partial N(x)_i}{\partial x} & \frac{\partial N(x)_{i+1}}{\partial x}\frac{\partial N(x)_{i+1}}{\partial x} \end{Bmatrix} A\, dx \begin{Bmatrix} u_i \\ u_{i+1} \end{Bmatrix} - \int_0^{\Delta x} \begin{Bmatrix} N(x)_i \\ N(x)_{i+1} \end{Bmatrix} B\, dx = 0$$

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



$\mathbf{Ku - F} = 0$

The finite element approximation - 3

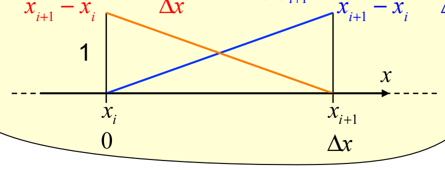
$$\mathbf{Ku} - \mathbf{F} = 0$$

$$\mathbf{K} = \int_0^{\Delta x} \begin{Bmatrix} \frac{\partial N(x)_i}{\partial x}\frac{\partial N(x)_i}{\partial x} & \frac{\partial N(x)_i}{\partial x}\frac{\partial N(x)_{i+1}}{\partial x} \\ \frac{\partial N(x)_{i+1}}{\partial x}\frac{\partial N(x)_i}{\partial x} & \frac{\partial N(x)_{i+1}}{\partial x}\frac{\partial N(x)_{i+1}}{\partial x} \end{Bmatrix} A\,dx, \quad \mathbf{F} = \int_0^{\Delta x} \begin{Bmatrix} N(x)_i \\ N(x)_{i+1} \end{Bmatrix} B\,dx$$

$$K_{ii} = \int_0^{\Delta x} \frac{\partial N(x)_i}{\partial x}\frac{\partial N(x)_i}{\partial x} A\,dx = A\int_0^{\Delta x} \frac{1}{\Delta x^2}\,dx = A\frac{x}{\Delta x^2}\Big|_0^{\Delta x} = A\left(\frac{\Delta x}{\Delta x^2} - \frac{0}{\Delta x^2}\right) = \frac{A}{\Delta x}$$

$$\frac{\partial N(x)_i}{\partial x} = -\frac{\partial N(x)_{i+1}}{\partial x} = -\frac{1}{\Delta x}$$

$$N(x)_i = 1 - \frac{x - x_i}{x_{i+1} - x_i} = 1 - \frac{x}{\Delta x} \qquad N(x)_{i+1} = \frac{x - x_i}{x_{i+1} - x_i} = \frac{x}{\Delta x}$$

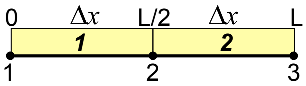


$$-\mathbf{K} = \begin{Bmatrix} -\frac{A}{\Delta x} & \frac{A}{\Delta x} \\ \frac{A}{\Delta x} & -\frac{A}{\Delta x} \end{Bmatrix}, \quad -\mathbf{F} = \begin{Bmatrix} -\frac{B\Delta x}{2} \\ -\frac{B\Delta x}{2} \end{Bmatrix}$$

The finite element approximation - 4

Local system for element ***1***

$$\begin{Bmatrix} -\frac{A}{\Delta x} & \frac{A}{\Delta x} & 0 \\ \frac{A}{\Delta x} & -\frac{A}{\Delta x} & 0 \\ 0 & 0 & 0 \end{Bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} - \begin{Bmatrix} -\frac{B\Delta x}{2} \\ -\frac{B\Delta x}{2} \\ 0 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$



Local system for element ***2***

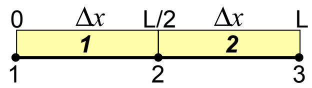
$$\begin{Bmatrix} 0 & 0 & 0 \\ 0 & -\frac{A}{\Delta x} & \frac{A}{\Delta x} \\ 0 & \frac{A}{\Delta x} & -\frac{A}{\Delta x} \end{Bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} - \begin{Bmatrix} 0 \\ -\frac{B\Delta x}{2} \\ -\frac{B\Delta x}{2} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$

Global system for sum of all elements

$$\begin{Bmatrix} -\frac{A}{\Delta x} & \frac{A}{\Delta x} & 0 \\ \frac{A}{\Delta x} & -\frac{2A}{\Delta x} & \frac{A}{\Delta x} \\ 0 & \frac{A}{\Delta x} & -\frac{A}{\Delta x} \end{Bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} - \begin{Bmatrix} -\frac{B\Delta x}{2} \\ -B\Delta x \\ -\frac{B\Delta x}{2} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$

The finite element approximation - 5

Global system without boundary conditions



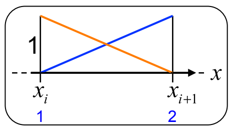
$$\begin{bmatrix} -\frac{A}{\Delta x} & \frac{A}{\Delta x} & 0 \\ \frac{A}{\Delta x} & -\frac{2A}{\Delta x} & \frac{A}{\Delta x} \\ 0 & \frac{A}{\Delta x} & -\frac{A}{\Delta x}\end{bmatrix}\begin{Bmatrix}u_1\\u_2\\u_3\end{Bmatrix} - \begin{Bmatrix}-\frac{B\Delta x}{2}\\ -B\Delta x\\ -\frac{B\Delta x}{2}\end{Bmatrix} = \begin{Bmatrix}0\\0\\0\end{Bmatrix}$$

Impose boundary conditions:
u_1 and u_3 =0

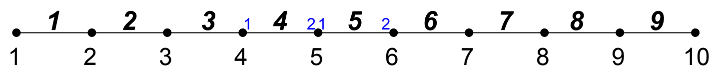
$$\begin{bmatrix} 1 & 0 & 0 \\ \frac{A}{\Delta x} & -\frac{2A}{\Delta x} & \frac{A}{\Delta x} \\ 0 & 0 & 1\end{bmatrix}\begin{Bmatrix}u_1\\u_2\\u_3\end{Bmatrix} = \begin{Bmatrix}0\\ -B\Delta x\\ 0\end{Bmatrix}$$

Put zeros on rows 1,3 and add 1 on the diagonal

Programming finite elements - 1



Nine finite elements



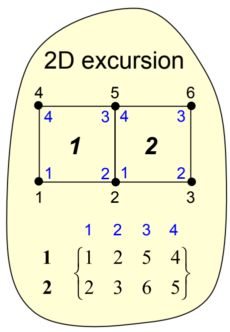
Ten nodes

Global nodes
Local nodes

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

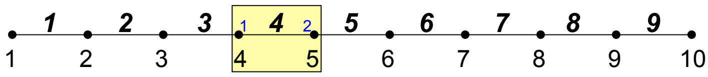
The **NODE** matrix assigns to each local element the corresponding global node

	1	2
1	1	2
2	2	3
3	3	4
4	4	5
5	5	6
6	6	7
7	7	8
8	8	9
9	9	10



Programming finite elements - 2

Nine finite elements



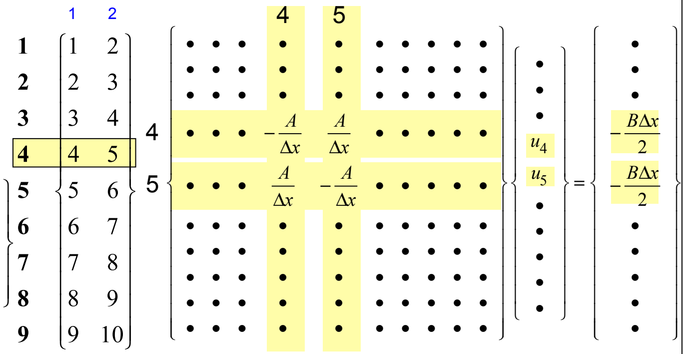
Global nodes

Local nodes

Ten nodes

Local system for finite element ***4***

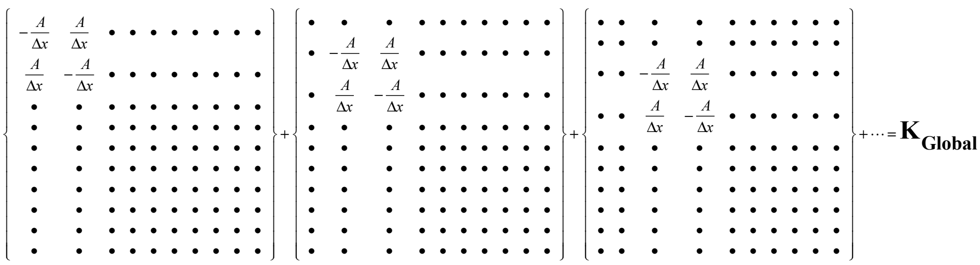
$$\begin{bmatrix} -\frac{A}{\Delta x} & \frac{A}{\Delta x} \\ \frac{A}{\Delta x} & -\frac{A}{\Delta x} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \begin{Bmatrix} -\frac{B\Delta x}{2} \\ -\frac{B\Delta x}{2} \end{Bmatrix}$$



Programming finite elements - 3

The global stiffness matrix is the sum of all local stiffness matrixes.
In our code we will first setup a global stiffness matrix $\mathbf{K}_{\mathbf{Global}}$ 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.



Exactly the same is done for the right hand side vector **F**.

The FEM Maple code

```
> 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);
```

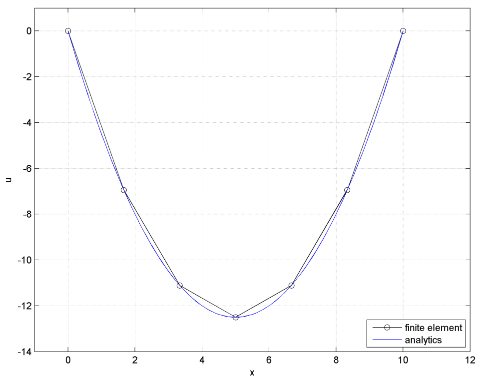
$$\int_0^L \left(\frac{\partial N(x)}{\partial x} A \frac{\partial u(x)}{\partial x} - N(x)B \right) dx = 0$$

$$\begin{bmatrix} -\frac{A}{dx} & \frac{A}{dx} \\ \frac{A}{dx} & -\frac{A}{dx} \end{bmatrix} \qquad \begin{bmatrix} -\frac{B\,dx}{2} \\ -\frac{B\,dx}{2} \end{bmatrix}$$

The FEM Matlab code

```matlab
% FEM code for 1D 2nd order steady state equation. 04.10.2006, Stefan Schmalholz, ETH Zurich
clear all, close all, clc
% Numerical parameters
no_nodes = 7;  no_el = no_nodes-1;   no_nodes_el = 2;
width    = 10; X     = [0:width/(no_el):width];
dx       = X(2)-X(1);
% Physical parameters
A                                = 1;
B                                = -1;
% Initialization of matrixes and vectors
KK  = zeros(no_nodes,no_nodes);
u   = zeros(no_nodes,1);
F   = zeros(no_nodes,1);
% Setup element matrix
K_loc   = [-A/dx  A/dx; A/dx  -A/dx];     % Matrix for conductive terms, see Maple script
F_loc   = [-B*dx/2 -B*dx/2];              % Right hand side vector
% Node numbering
for i=1:no_el
   iel = i; nodes(iel,1) = i; nodes(iel,2) = i+1;
end

% Setup system matrix
for iel=1:no_el
   for i=1:no_nodes_el
      ii                         = nodes(iel,i);
      for j=1:no_nodes_el
         jj                      = nodes(iel,j);
         KK(ii,jj)               = KK(ii,jj)      + K_loc(i,j);
      end
      F(ii)                      = F(ii)          + F_loc(i);
   end
end

% Boundary conditions
KK(1,:)   = 0; KK(1,1)     = 1; F(1)   = 0; % Left  boundary
KK(end,:) = 0; KK(end,end) = 1; F(end) = 0; % Right boundary

% Solve matrix
u                                = KK\F;

% Analytical solution
L                                = width;
x                                = [0:0.1:L];
u_ana                              = -1/2*B/A.*x.^2+1/2*B*L/A*x;%-B/A.*(1/2.*x.^2-1/2*L.*x);

% Graphics
plot(X,u,'-ok',x,u_ana,'-b');
axis([-1 width+2 -14 1]); xlabel('x'); ylabel('u'); grid on;
legend('finite element','analytics',4); drawnow;
```

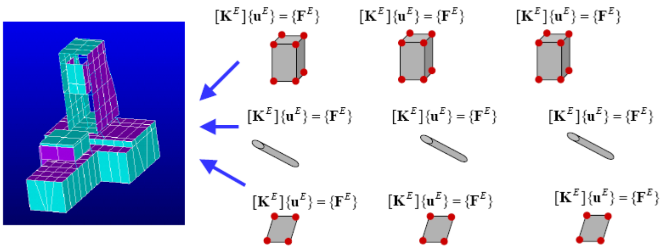
In the numerical solution all parameters can be made variable easily, while it gets more difficult to find analytical solutions for variable parameters, e.g., A and B.



FEM applications

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

Put all the element equations together



$$[\mathbf{K}]\{\mathbf{u}\} = \{\mathbf{F}\}$$

Taken from MIT lecture, de Weck and Kim

FEM element types

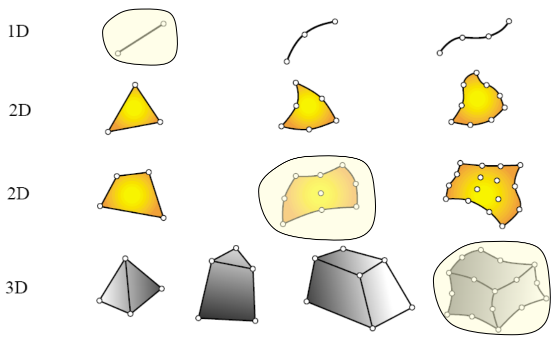
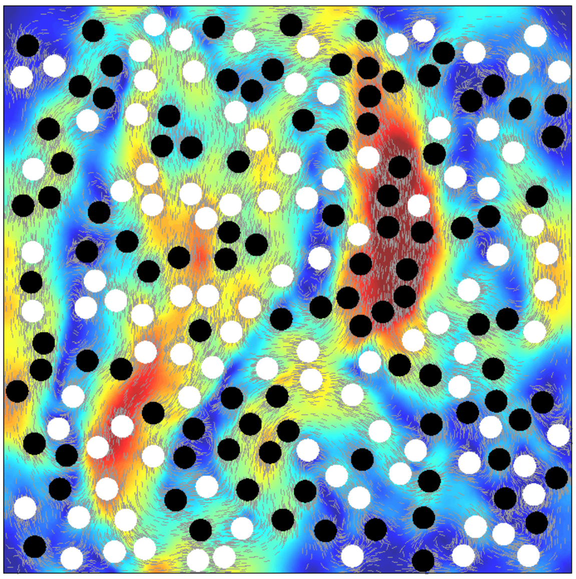


FIGURE 6.3. Typical finite element geometries in one through three dimensions.

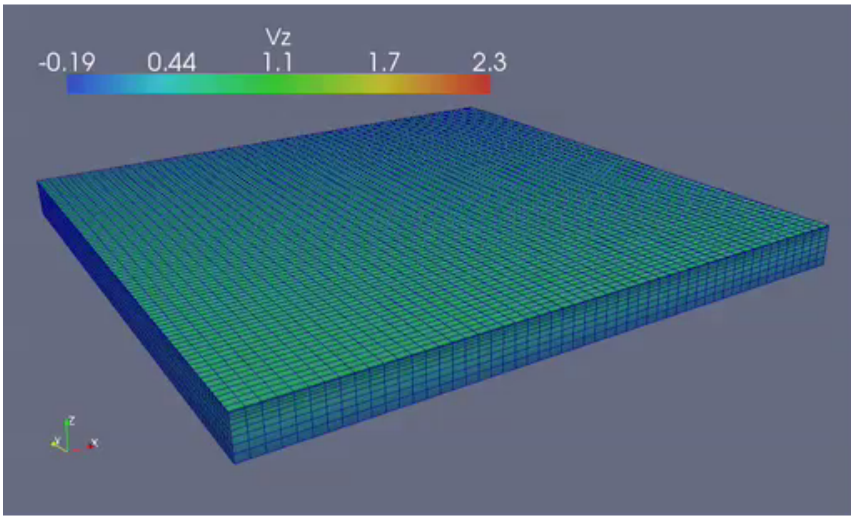
Taken from unknown web script

FEM applications



Separation of bubbles under influence of gravity

FEM applications



Viscous detachment folding (3D Stokes problem – you'll do it in 2D)