Matfem

A Matlab finite element code

Ignacio Romero

Technical University of Madrid

October 8, 2014

Description

`Matfem` is a simple finite element code written in Matlab developed for teaching purposes

- Interpreted, not compiled
- Easy to modify
- Only for 2D problems
- Not too fast, not able to solve large problems.
- Limited output and postprocessing

What can it analyze?:

- Linear and nonlinear trusses
- Linear and nonlinear frames
- Laplace problem
- Plain strain, elasticity

Writing input files

An input `.m` file must be written for every analysis, with an arbitrary name.

The user must define

- Basic information about the analysis (dofs/node, max number of nodes/elmt, ...)
- The mesh itself
- Boundary conditions (both essential and natural)
- The element templates.

Input file 1

Global variables

```
global Kff U R;
```

Basic definitions

```
%------------------------------------------------------------
%                    Problem properties
%------------------------------------------------------------
nen    = 3;    %nodes per element
ndofn  = 1;    %max dof per node
method = 1;    %solution strategy: 1-linear, 2-newton-raphson
```

Input file 2

Mesh

```
node = [ 0,    0; %1       %global coordinates of all nodes
         2,    0;
         2,    0.8;
         0,    1;
         0,    2;
         2,    2];

element = [1, 1,2,3; %1    %properties, nodes
             1, 1,3,4;
             1, 4,3,6;
             1, 5,4,6];
```

Input file 3

Mesh processing

```
%extract information from mesh
% do not write anything in these 4 lines
nn    = size(node, 1);
nel   = size(element,1);
bc    = zeros(nn,ndofn);
force = zeros(nn,ndofn);
```

Element templates

```
etype = [9,  3.0, 0.0];    % thermal triangle, conductivity, heat sc
```

Input file 3

Boundary conditions

```
% first indicate with a 1 if there is a boundary condition
% '1' is just a label, it does not indicate the value of the fixed
% if the field is free, just don't write anything
% bc means bc( nodelabel, doflabel)
 bc(1,1) = 1;  %node, dof=1 if constrained
 bc(2,1) = 1;  %node, dof=1 if constrained
 bc(5,1) = 1;
 bc(6,1) = 1;

% then set the value. If the dof is constrained by the previous co
% then the force is really the fixed value of the field.
 force(1,1) = 0.0;
 force(2,1) = 0.0;
 force(5,1) = 50.0;
 force(6,1) = 50.0;
```

Input file 4

Solution and postprocessing

```
%mesh processing and solution strategy
meshplot(node, element);
sol = solveproblem(method,nen,ndofn,node,element,etype,bc,force);


%solution postprocessing
postprocess(node, element, ndofn,  bc, force, sol);
```

Defining new boundary value problems

- There is a limited class of problems that `Matfem` can currently solve. The file `el_info.m` serves as dictionary of all the problems implemented. New ones can be added, if the corresponding tasks are programmed.
- For each additional set of equations, a file must be programmed that implements the computation of force vector, stiffness and energy.

Example: elastic problem

Problem definition

```
ndofn  = 2;     %max dof per node
method = 1;     %solution strategy: 1-linear, 2-newton-raphson
```

Element properties

```
etype = [14,  3e4, 1e4];     % 1? Lame coefficients (lambda, mu)
```

$$\lambda = \frac{\nu E}{(1-2\nu)(1+\nu)} \qquad \mu = \frac{E}{2(1+\nu)}$$

Example: elastic problem (2)

Boundary conditions

```
bc(1,1) = 1;
bc(1,2) = 1;
bc(2,2) = 1;

bc(3,1) = 1;
bc(4,1) = 1;

force(1,1) = 0.0;
force(1,2) = 0.0;
force(2,2) = 0.0;
force(3,1) = 1.0;
force(4,1) = 1.0;
```

Matfem

A Matlab finite element code

Ignacio Romero

Technical University of Madrid

October 8, 2014