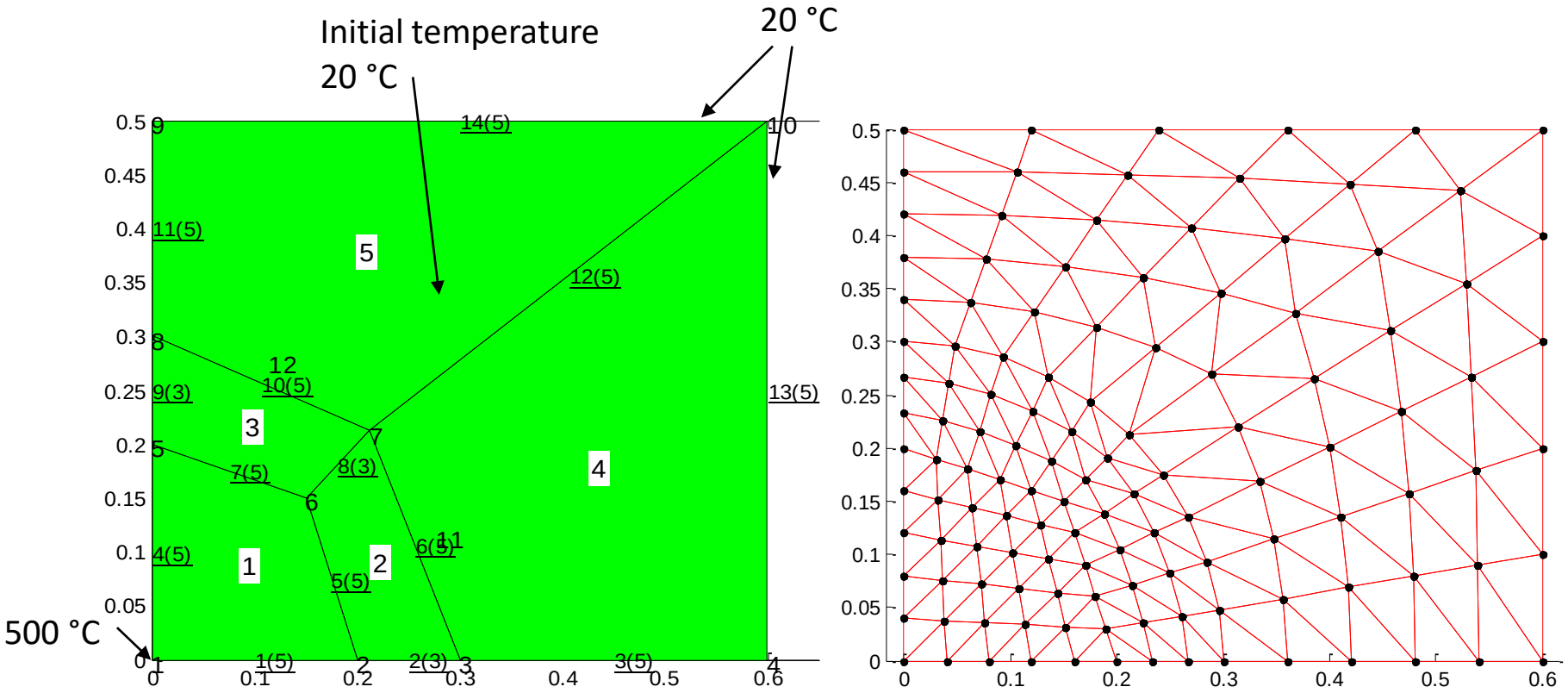
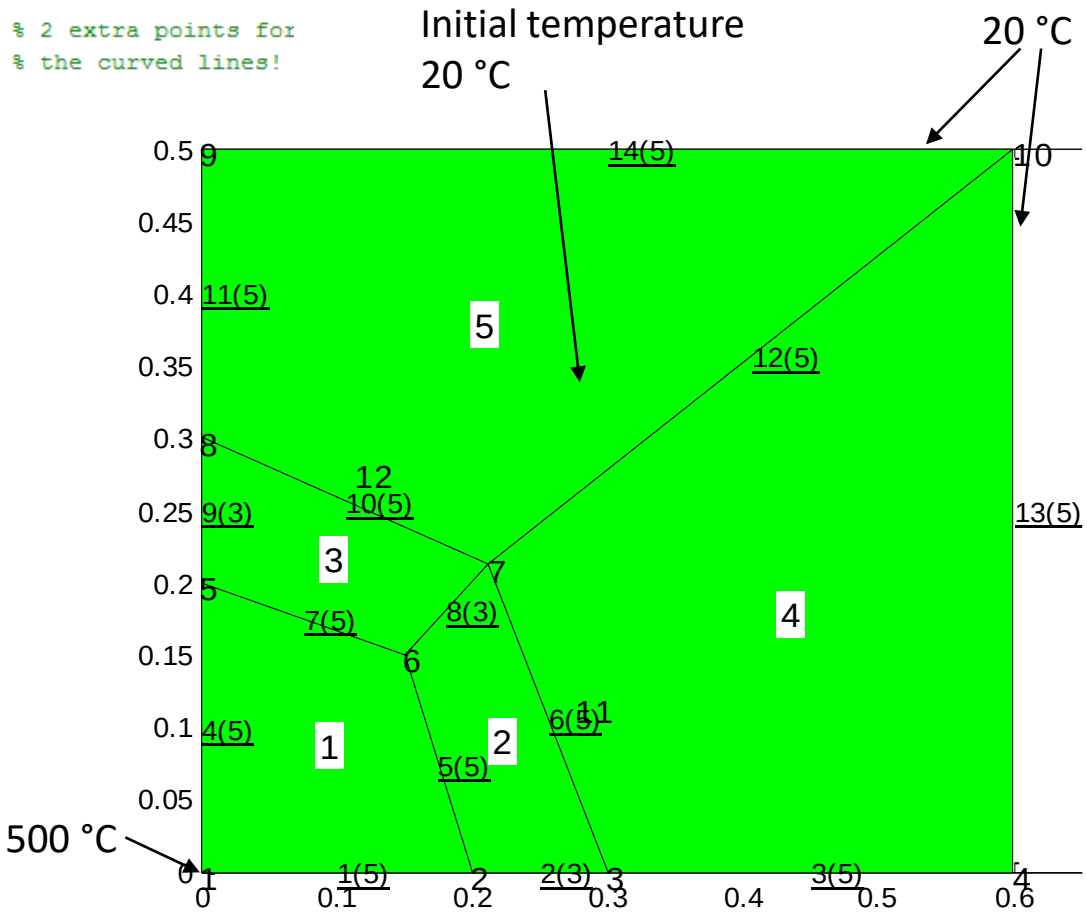
Transient heat flow in Calfem

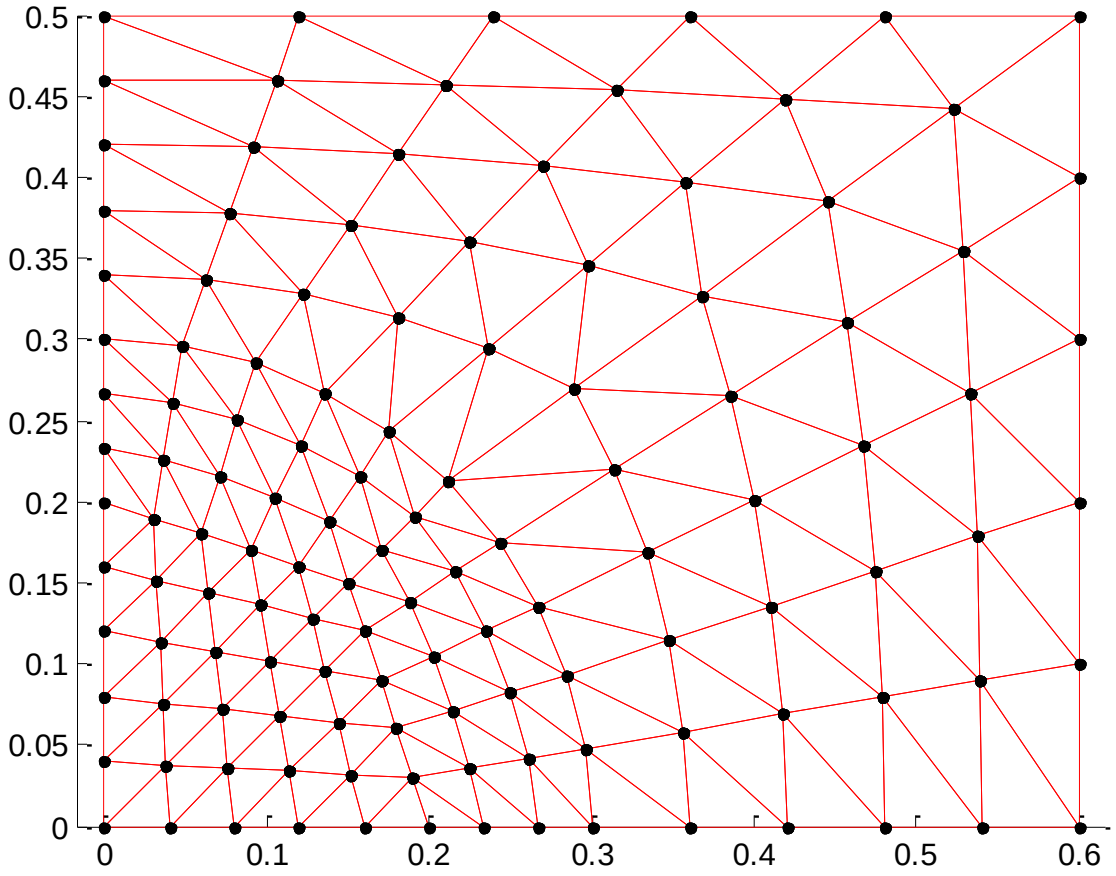


```
% Define the vertex coordinates
Vertices=[0 0
          0.2 0
          0.3 0
          0.6 0
          0 0.2
          0.15 0.15
          0.3/sqrt(2) 0.3/sqrt(2)
          0 0.3
          0 0.5
          0.6 0.5
          0.3*cos(22.5*pi/180),0.3*sin(22.5*pi/180)    % 2 extra points for
          0.3*cos(67.5*pi/180),0.3*sin(67.5*pi/180)];  % the curved lines!
% Define the line segments from vertex numbers
Segments=[1 2
          2 3
          3 4
          1 5
          2 6
          7 3
          5 6
          6 7
          5 8
          8 7
          8 9
          7 10
          4 10
          9 10];
% Define the surfaces from the segment numbers
Surfaces=[1 5 7 4
       2 6 8 5
       7 8 10 9
       3 13 12 6
       10 12 14 11];
% Define the number of elements on each segment
Seed=round([5 3 5 5 5 5 5 3 3 5 5 5 5 5]);
% Define the curved lines
iso8=zeros(1,14); iso8(6)=11; iso8(10)=12;
% Combine seed and curved line definitions
Segp(1:2:28)=Seed; Segp(2:2:28)=iso8
```

```
nen=3;   % Triangle elements
dofsPerNode=1;  % 1 dof per node
mp=[ dofsPerNode, nen];

% Draw the geometry
geomdraw2(Vertices,Segments,Surfaces,Segp,mp)
```



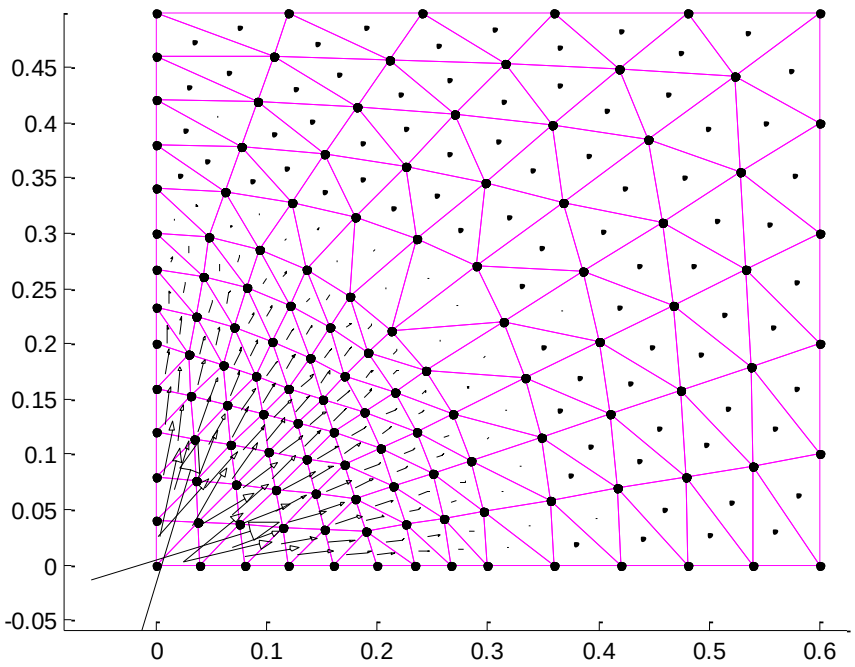
```
% Generate the element mesh
[Coord Edof Dof meshdb ]=strMeshgen(Vertices,Segments,Surfaces,Segp,mp);
% Generate the element coordinates
[Ex,Ey]=coordxtr(Edof,Coord,Dof,nen);
% Draw the element mesh with numbering of the elements
figure(2)
eldraw2(Ex,Ey,[1 4 0])%,Edof(:,1));
```

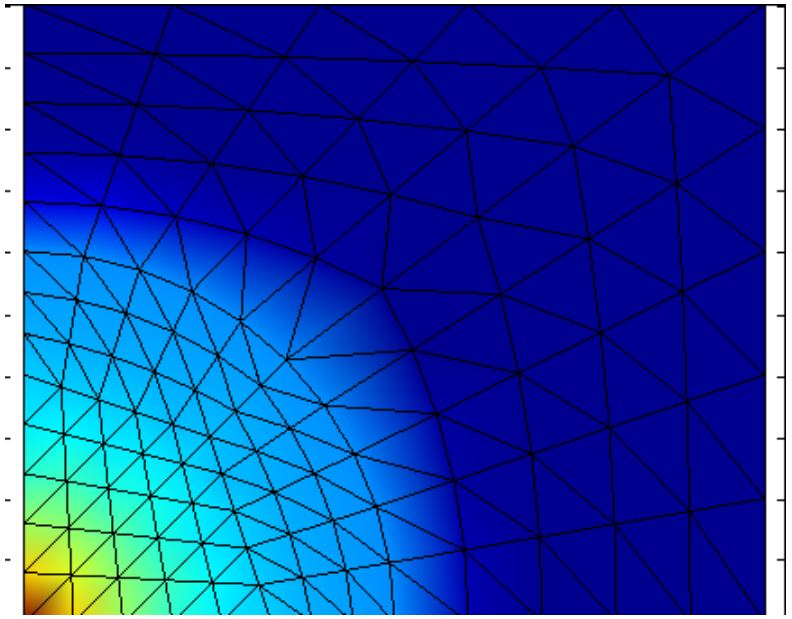


```
% Create element stiffness, assemble to global stiffness, create bc's
% and solve the equation system
nel=size(Edof,1);
ndof=length(Dof);
%
K=sparse(ndof,ndof);
C=sparse(ndof,ndof);
f=sparse(ndof,1);

% Create element stiffness and assemble for steel surfaces
for j=[1 2 3]  % surfaces 1,2 and 3 contain a steel material
  t=1;
  rho=7800;
  c=480;
  ep=[t,rho,c];
  k=43.0;
  Ds=k*[1 0;0 1];
  elems=extrSurf(j,meshdb);  % extract elements on surface j
  for k=elems'  % Loop over elements in elems
    [Ke,Ce]=flw2tt(Ex(k,:),Ey(k,:),ep,Ds);
    K=sparse_assem(Edof(k,:),K,Ke);
    C=sparse_assem(Edof(k,:),C,Ce);
  end
end

% Create element stiffness and assemble for concrete surfaces
for j=[4 5]  % surfaces 4 and 5 contain a steel material
  t=1;
  rho=2400;
  c=880;
  ep=[t,rho,c];
  k=1.0;
  Dc=k*[1 0;0 1];
  elems=extrSurf(j,meshdb);   % extract elements on surface j
  for k=elems'   % Loop over elements in elems
    [Ke,Ce]=flw2tt(Ex(k,:),Ey(k,:),ep,Dc);
    K=sparse_assem(Edof(k,:),K,Ke);
    C=sparse_assem(Edof(k,:),C,Ce);
  end
end
```

```
% Determine degree of fredom along in point for boundary condition
fn=extrPoint(1,meshdb,1)
% Determine degrees of fredom along segments for boundary conditions
bc1=extrSeg([13,14]',meshdb,1);
% Define boundary conditions
bc=[bc1, ones(length(bc1),1)*20;fn,50];
% Define initial temperature at time t=0
a0=ones(ndof,1)*20;
%
% Define and perform time stepping for 36 hours (given in seconds)
% Define input for the time-stepping command
ip=[3600 3600*36 0.5];
% Times at which the results is extracted
snap=[0:3600:3600*36];
% Perform time-stepping
[D,V]=step1(K,C,a0,ip,snap,f,bc);
%
```

```
% Extract element temperatures after 2 hours
Ed2=extract(Edof,Tsnap(:,2));
% Calculate element heat fluxes after 2 hours
for j=1:5  % surfaces 1,2 and 3 is steel and 4 and 5 is concrete
  if ((j==1) | (j==2) | (j==3))
      Dm=Ds;
  else
      Dm=Dc;
  end
  elems=extrSurf(j,meshdb);
  for i=elems'
    [Es2(i,:),Et2(i,:)]=flw2ts(Ex(i,:),Ey(i,:),Dm,Ed2(i,:));
  end
end
```

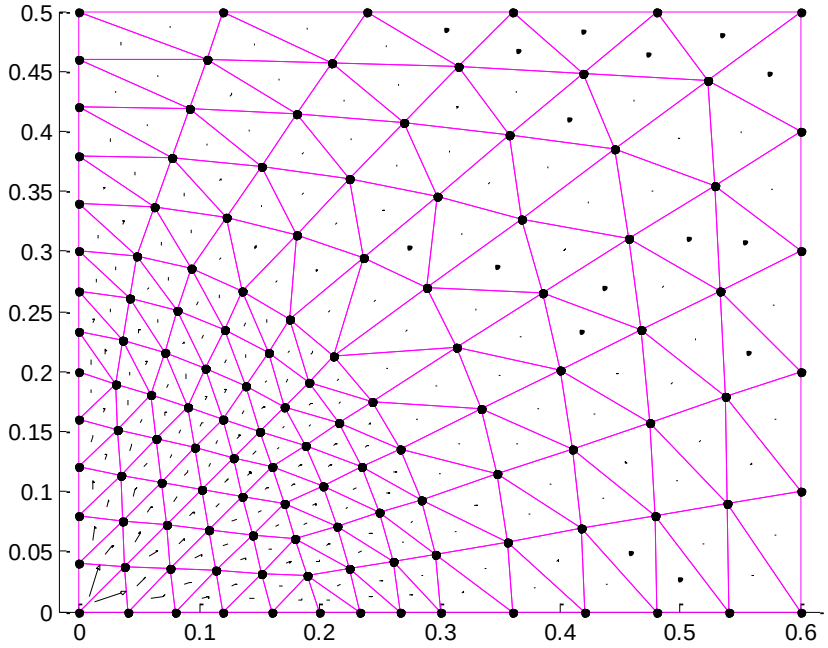


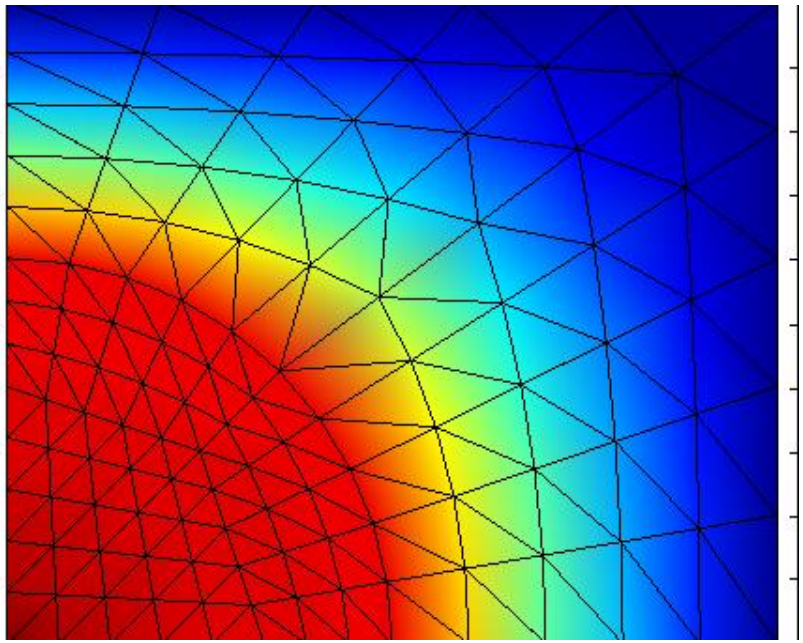


```
% Plot flux vectors and temperatures after 2 hours
figure(4)
eldraw2(Ex,Ey,[1 3 0]);
elflux2(Ex,Ey,Es2,[1,1],1e-6);
```

```
figure(5)
fill(Ex',Ey',Ed2')
axis('equal')
```

```
% Extract element temperatures after 36 hours
Ed36=extract(Edof,Tsnap(:,36));

% Calculate element heat fluxes after 36 hours
for j=1:5  % surfaces 1,2 and 3 is steel and 4 and 5 is concrete
  if (j==1) | (j==2) | (j==3)
      Dm=Ds;
  else
      Dm=Dc;
  end
  elems=extrSurf(j,meshdb);
  for i=elems'
    [Es36(i,:),Et36(i,:)]=flw2ts(Ex(i,:),Ey(i,:),Dm,Ed36(i,:));
  end
end
```





```
% Plot flux vectors and temperatures after 36 hours
figure(6)
eldraw2(Ex,Ey,[1 3 0]);
elflux2(Ex,Ey,Es36,[1,1],1e-6);
```

```
figure(7)
fill(Ex',Ey',Ed36')
axis('equal')
```