
Engineering Analysis using the Finite Element Method

University of California, Berkeley, Spring 2019

Instructor: Sanjay Govindjee

GSI: Linus Mettler

Lab 2

In this lab, you will examine the influence of mesh size on the accuracy of the finite element solution and how the FE result converges to the exact solution. To this end, we will consider the one-dimensional heat equation for a slender bar subjected to a distributed heat source.

In order to check off this lab you will provide a convergence plot. No report is required.

1 Problem statement and exact solution

Consider a slender bar of length L and thermal conductivity K_0 subjected to a distributed heat source (with units of power per volume)

$$h_{gen}(x) = r_1 + r_2 \sin(kx) \quad (1)$$

where $k = \frac{2\pi}{L}$ and r_1, r_2 are given constants. The governing differential equation reads (note the similarity with homework 1, problem 1)

$$c\rho \frac{\partial T(x,t)}{\partial t} = h_{gen}(x) + K_0 \frac{\partial^2 T(x,t)}{\partial x^2} = 0 \quad \text{on } \Omega = (0, L) \quad (2)$$

Since we are limiting ourselves to a static analysis of the problem, the specific heat c and density ρ are irrelevant for the solution. The boundary conditions are

$$T(0) = 0 \quad \text{and} \quad T(L) = 0 \quad (3)$$

The corresponding weak form statement is

$$\int_{\Omega} \frac{dT(x)}{dx} K_0 \frac{d\nu(x)}{dx} dx = \int_{\Omega} h_{gen}(x) \nu(x) dx \quad \forall \text{ admissible } \nu(x) \quad (4)$$

Use 1 - 3 to compute the exact solution in terms of L , K_0 , r_1 , r_2 . Compute the value of the maximum temperature. Confirm our expectation that a higher conductivity K_0 leads to smaller temperatures inside the domain.

2 FE model

2.1 Geometry and material properties

Start a steady thermal analysis in Ansys. In DesignModeler, define a circular cross-section of radius $r = 1 \text{ mm}$ ¹ by clicking *concept* → *cross section* → *circular*. Create a line body of $L = 100 \text{ mm}$ along the x-axis (*concept* → *lines from sketches*) and assign the previously defined cross-section to the line body. Unlike in Lab 1, do **not** set the *advanced geometry options* to 2D.

We are going to use the predefined material *copper alloy* which can be found in the material library under *engineering data sources* → *general materials*. The only relevant material parameter is the thermal conductivity of $K_0 = 401 \text{ W/mK}$. Note that the thermal conductivity is defined to be constant, i.e. independent of temperature, ensuring that our differential equation and FE problem are linear.

2.2 FE mesh and loads

Open Ansys Mechanical. Use size controls on the line body to define a fixed number of divisions along the rod. To ensure that the exact specified number of elements is generated, change the edge sizing behavior from soft to *hard*. Note that the default element order for thermal analyses in Ansys is *linear*. Higher-order elements are not supported.

You can click *view* → *annotation preferences* → *element / node numbers* to inspect the element or node numbering of your mesh.

Apply the essential boundary conditions on the two end points of the bar. Note that at least *one* essential boundary condition is necessary for a unique solution to exist².

The built-in *heat generation* loading only supports constant or tabular values, not functions of the spatial variables. So instead, we will use a *command snippet* to programmatically apply heat to each element as a function of its location. To this end, insert a command snippet to the *steady-state thermal* branch of the tree (right

¹To compute the temperature distribution it turns out that the actual cross-section area is irrelevant because the heat source is given *per volume*.

²This is similar to the rigid-body motions of improperly supported solid bodies in Lab 1.

click on the branch \rightarrow *commands*). Copy the following lines of code into the command snippet to apply h_{gen} according to (1) with constants $r_1 = 10 \text{ mW/mm}^3$ and $r_2 = 50 \text{ mW/mm}^3$. Make sure that your units are consistent (for this code snippet I used units of mm, kg, N, s).

```

ALLSEL
*GET,Ne,ELEM,,COUNT
*SET,r1,10
*SET,r2,50
*SET,L,100
*SET,pi,4*atan(1)

*DO,k,1,Ne,1
  BFE,k,HGEN,1,r1+r2*sin(2*pi*(k-1)/Ne),r1+r2*sin(2*pi*k/Ne)
*ENDDO

```

2.3 Solution

Solve your FE model and plot the temperature distribution. In order to plot and compare our FE result insert the following command snippet into the *solution* branch. This code will print the x-coordinate and temperature of each node into a text file. You will find the text file in the folder where you saved your Ansys model.

```

ALLSEL
NSEL,S,LOC,Y,0
NSEL,R,LOC,Z,0
*GET,Nn,NODE,,COUNT
*CFOPEN,.. \OutputData.txt,,
*VWRITE ('x-coordinate and temperature at every node')

*DO,k,1,Nn,1
  *GET,Xk,NODE,k,LOC,X
  *GET,Tk,NODE,k,TEMP,
  *VWRITE,Xk,Tk
  (E17.10,2X,E22.15)
*ENDDO
*CFCLOSE

```

Repeatedly refine your mesh, for example by increasing the number of elements by a factor of two. The process of refining the element size is called *h-refinement*³. For

³As opposed to p-refinement, in which the element *order* is increased with the goal of obtaining

each simulation, modify the name of the output text file as to retain the result of the previous simulations.

3 Convergence

For each of the simulations compute the relative error of the maximum temperature with respect to the analytical solution

$$e_1^N = \left| \frac{T_{\max}^{\text{exact}} - T_{\max}^{\text{FEM}}}{T_{\max}^{\text{exact}}} \right| \quad (5)$$

as well as the relative error in L^2 -norm

$$e_2^N = \frac{\|T^{\text{exact}}(x) - T^{\text{FEM}}(x)\|_{L^2}}{\|T^{\text{exact}}(x)\|_{L^2}} \quad (6)$$

where the L^2 -norm is defined as

$$\|f(x)\|_{L^2} = \sqrt{\int_{\Omega} (f(x))^2 dx} \quad (7)$$

Hint: If you are using Matlab to process the FE data you may want to use the following commands to read the data from the text files:

```
filename = 'yourfilename.txt';
data = importdata(filename, ' ',1,0);
x = data.data(:,1);
T = data.data(:,2);
[x,ind] = sort(x);
T = T(ind);
```

To compute the L^2 -norm, you could do:

```
Tfe = @(x) interp1(xx,TT,x); % convert data to piecewise linear curve
L2 = @(f) sqrt( integral(@(x) f(x).^2,0,L) ); % L2 norm of f(x)
eN = L2(@(x) Tex(x)-Tfe(x)) / L2(Tex); % error in L2 norm
```

Plot the exact and approximate solutions in one graph. Compute the errors e_1^N and e_2^N for each mesh size and plot vs N (# of elements) using logarithmic axes. Do you observe a trend? Find the number of elements necessary for the relative error to be $e^N \leq 10^{-9}$.

a more accurate solution.