Lab 7

In this lab you will amend a MATLAB FEA program for heat transfer to accommodate transient phenomena. To start, download `lab7_student.m` from bCourses. This file needs to be modified to solve the following transient heat transfer problem:

$$\rho c \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2} \qquad x \in (-1\text{ m}, 1\text{ m}) \qquad t \in (0, t_f] \tag{1}$$

$$q(-1) = 0 \qquad t \in [0, t_f] \tag{2}$$

$$q(1) = 0 \qquad t \in [0, t_f] \tag{3}$$

$$T(x, 0) = T_o \exp[-x^2/\alpha] \qquad x \in [-1\text{ m}, 1\text{ m}] \tag{4}$$

where $\rho = 7833\ \text{kg/m}^3$, $c = 465\ \text{J/kg}$, $T_o = 100\ \text{C}$, and $\alpha = 0.05\ \text{m}^2$. This problem represents a one-dimensional low-carbon steel bar that is initially centrally heated and allowed to evolve in time wherein the ends of the bar are insulated so that no energy is allowed to escape from the bar.

The FEA equations for this problem are

$$\boldsymbol{M}\dot{\boldsymbol{T}} + \boldsymbol{K}\boldsymbol{T} = \boldsymbol{0} \tag{5}$$

where $\boldsymbol{K} = \int \boldsymbol{B}^T k \boldsymbol{B}\, dx$ and $\boldsymbol{M} = \int \boldsymbol{N}^T \rho c \boldsymbol{N}\, dx$.

As given, `lab7_student.m` contains all the necessary code for the *non-transient terms*. You will need to add the transient parts to the code as follows.

1. Define variables for the mass density and the heat capacity.
2. Allocate memory for the thermal mass matrix.
3. Initialize and compute the element thermal mass matrix.
4. Assemble the element thermal mass matrix into the global thermal mass matrix.
5. After the global matrices have been assembled, compute the critical time step, $\Delta t_{\max} = 2/\lambda_{\max}$, for use with a Forward Euler time stepping scheme in two ways:
   (a) Use `eigs` on your global thermal stiffness matrix and global thermal mass matrix. [Note `eigs` is the eigenvalue solver for sparse matricies and by default only computes the 6 largest eigenvalues.]

(b) Use `eig` on the element thermal stiffness and element thermal mass matrix. It is enough to just check the last element, since with a uniform mesh all elements have the same element-level matrices. Note the mesh given in the file is uniform.

(c) Perform these two steps for `numel` equal to 200, 400, and 800 and observe the rapid (quadratic) drop in $\Delta t_{\text{max F.E.}}$. Also observe that using the full global matrices can cause numerical problems when they increase in size.

6. Reset `numel` equal to 200, initialize the temperature field for the time stepping procedure, and time step the solution with Forward Euler[1] to a time $t_f = 95\Delta t_{\text{max F.E.}}$ using $\Delta t = 0.95\Delta t_{\text{max F.E.}}$. In this very short period of time and the solution hardly changes but is completely stable. Check by plotting.

7. Now time step the solution with Forward Euler from time zero to $t_f = 95\Delta t_{\text{max F.E.}}$ using $\Delta t = 1.1875\Delta t_{\text{max F.E.}}$. Note the unstable behavior by plotting the solution.

8. Change your time integrator to Backward Euler and time step your solution from time zero to $t_f = 10$ hours using $\Delta t = 12$ min $\gg \Delta t_{\text{max F.E.}}$. Plot the solution as you compute to see the time evolution of the temperature field. Note that the area under the temperature profile, which is proportional to the thermal energy in the system, is virtually a constant in time for this problem since the domain is fully insulated (the energy can not escape, it can only redistribute itself).

[1] Do not worry about diagonalizing/lumping the mass matrix.