ME 280A Fall 2014

HOMEWORK 3: POTENTIALS & EFFICIENT SOLUTION TECHNIQUES

- Solve the following boundary value problem, with domain $\Omega = (0, L)$, analytically. You should use appropriate boundary conditions and interface conditions to derive your answer:

$$\boxed{\begin{array}{l} \frac{d}{dx}\left(A_1 \frac{du}{dx}\right) = 256 \sin\left(\frac{3}{4}\pi x\right)\cos(16\pi x) \\ A_1 = \text{ SEE BELOW} \\ L = 1 \\ u(0) = 0 \\ A_1(L)\frac{du}{dx}(L) = 1 \end{array}} \tag{1}$$

The property A_1 is given in ten segments as:

$$\boxed{\begin{array}{ll} \text{For } 0.0 \leq x < 0.1: & A_1 = 2.0 \\ \text{For } 0.1 \leq x < 0.2: & A_1 = 2.5 \\ \text{For } 0.2 \leq x < 0.3: & A_1 = 1.25 \\ \text{For } 0.3 \leq x < 0.4: & A_1 = 0.25 \\ \text{For } 0.4 \leq x < 0.5: & A_1 = 4.0 \\ \text{For } 0.5 \leq x < 0.6: & A_1 = 1.75 \\ \text{For } 0.6 \leq x < 0.7: & A_1 = 0.5 \\ \text{For } 0.7 \leq x < 0.8: & A_1 = 0.75 \\ \text{For } 0.8 \leq x < 0.9: & A_1 = 3.25 \\ \text{For } 0.9 \leq x \leq 1.0: & A_1 = 1.0 \end{array}} \tag{2}$$

The analytical solution will need to be solved as a piecewise function.

- Solve this with the finite element method using linear equal-sized elements. Use 100, 1000 and 10000 elements. You are to write a **Preconditioned Conjugate-Gradient** solver. Use the diagonal preconditioning given in the notes. The data storage is to be element by element (symmetric) and the matrix vector multiplication is to be done element by element during the iterations. *Check your Conjugate Gradient generated results against a regular Gaussian solver, for example the one available in MATLAB*
- You are to plot the solution (nodal values) for each N.
- You are to plot the error (defined below) for each N.

$$\boxed{\begin{array}{l} e^N \stackrel{\text{def}}{=} \frac{||u-u^N||_{A_1(\Omega)}}{||u||_{A_1(\Omega)}}, \\ \\ ||u||_{A_1(\Omega)} \stackrel{\text{def}}{=} \sqrt{\int_\Omega \frac{du}{dx} A_1 \frac{du}{dx}\, dx}, \end{array}} \tag{3}$$

- You are to plot

$$\boxed{POTENTIAL\ ENERGY = \mathcal{J}(u^N)} \tag{4}$$

  for each N.

- You are to plot the number of PCG-solver iterations for each N for a stopping tolerance of 0.001. *As a further check, see what happens when you vary the tolerance for the solver*

- NOTE: Use a Gauss integration rule of level 5.

Notes:

- It will help keep your code neat to make everything modular. For example, have a seperate routine for the element-by-element matrix-vector multiplaction, instead of nesting it inside of your PCG routine. This will make it easier to test individual components.

- Your code may be easier to debug if you write and debug the core PCG solver before implementing preconditioning. The solver should return the same answer regardless of how it is preconditioned.

Report:

1. Please follow the usual instructions for submission of the report. You do not need to repeat content from Homework 1 and 2 - just refer to them in your current report.

2. For the description of procedure/implementation - please write it clearly, briefly, and in a manner such that someone else can read through your implementation details and write the program that you are describing.