AMSC661, Spring 2022 Maria Cameron

Homework 7. Due March 29

1. **(5 pts)** Consider a 1D boundary-value problem

$$-\frac{d}{dx}\left(k(x)\frac{d}{dx}u\right) = 0, \quad a < x < b, \quad u(a) = u_a, \quad u(b) = u_b, \tag{1}$$

where the heat conductance coefficient $k(x)$ is the following piecewise constant function

$$k(x) = \begin{cases} k_1, & a \le x < c \\ k_2, & c < x \le b \end{cases}. \tag{2}$$

It follows from the integral form of Fourier's law that the temperature u and the heat flux $k(x)u_u$ must be continuous at $x = c$.

   (a) Find the exact solution to this problem analytically.

   (b) Set $u_a = 0$, $u_b = 10$, $k_1 = 10$, $k_2 = 1$, $a = 0$, $b = 10$, $c = 4$. Choose $h = 1.0$. Use the finite difference scheme

$$L_h U_P = -\frac{1}{h^2}\left(U_W k_w + U_E k_e - U_P(k_e + k_w)\right) \tag{3}$$

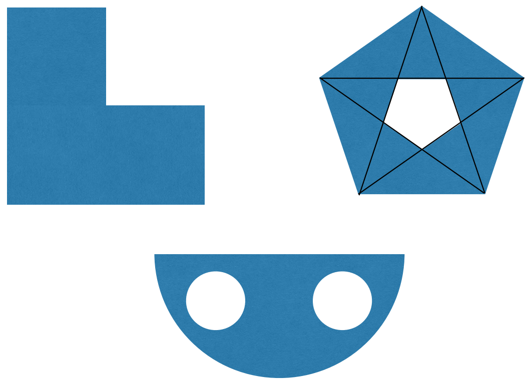
   where W is to the left of P, E is to the right of P, and e and w are the midpoints between E and P and W and P respectively. You need to evaluate $k(x)$ at these midpoints. In this case, it is not a problem that $k(x)$ is not defined at c. Compute the numerical solution U using this scheme. Plot it and also plot the exact solution in the same figure. You should see that these solutions exactly coincide.

2. **(5 pts)** *Triangular mesh generation using* `distmesh2d.m` *by P.-O. Persson.* Read [1], at least its first 12 pages (at least up to Section 6 "Mesh Generation in Higher Dimensions").

   If you prefer Matlab, download the distmesh package `distmesh.zip` available at http://persson.berkeley.edu/distmesh/.

   If you prefer Python, you can download my Python version of P.-O. Persson's code available at GitHub, user `mar1akc`, package `transition_path_theory_FEM_distmesh`, file `distmesh.py`.

   Mesh the shapes in the Figure below using distmesh2d.m following examples in [1].



You can pick arbitrary sizes as soon as topologies are preserved, and you can do uniform meshing.

3. **5 pts** Consider the following BVP in 1D:

$$-u_{xx} = f(x), \quad 0 < x < 1, \quad u(0) = u(1) = 1.$$

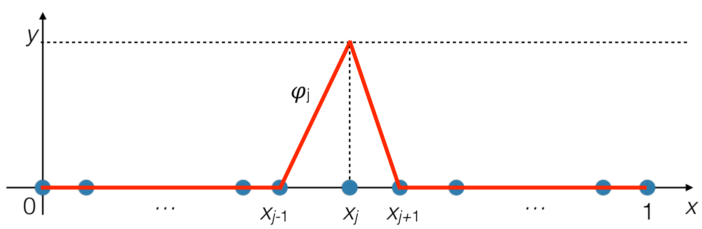
Work out all steps of the FEM on it.

(a) Let $w(x)$ be a twice continuously differentiable function on $(0,1)$ such that $w(0) = w(1) = 0$. Use integration by parts to reduce the BVP to an integral equation.

(b) Partition the interval $[0,1]$:

$$0 = x_0 < x_1 < \ldots < x_N < x_{N+1} = 1.$$

Define the basis functions $\phi_i(x)$, $1 \le i \le N$ as shown in the figure below ($\phi_i(x_i) = 0$, $\phi_i(x_j) = 0$, $j \ne i$, ϕ_i is piecewise linear).



$$\phi_i(x) = \begin{cases} 0, & x \le x_{i-1},\ x \ge x_{i+1}, \\ \frac{x - x_{i-1}}{x_i - x_{i-1}}, & x_{i-1} < x \le x_i, \\ \frac{x_{i+1} - x}{x_{i+1} - x_i}, & x_i < x < x_{i+1}. \end{cases}$$

Calculate the stiffness matrix and the load vector.

(c) In what case the FEM solution would coincide with the finite difference solution using the central difference scheme?

4. **(5 pts)**

(a) Let η_1, η_2, and η_3 be linear functions in a triangle T with vertices (x_1, y_1), (x_2, y_2), and (x_3, y_3) such that $\eta_j = 1$ at (x_j, y_j) and $\eta_j = 0$ and the other two vertices of T, $j = 1, 2, 3$. Prove that the matrix G introduced in Section 5 in [2] is equal to

$$G = \begin{bmatrix} \frac{\partial \eta_1}{\partial x} & \frac{\partial \eta_1}{\partial y} \\ \frac{\partial \eta_2}{\partial x} & \frac{\partial \eta_2}{\partial y} \\ \frac{\partial \eta_3}{\partial x} & \frac{\partial \eta_3}{\partial y} \end{bmatrix}. \tag{4}$$

(b) Let u be a finite element solution to some problem. This means that u is continuous, piecewise linear, and linear within each mesh triangle. Let T be a mesh triangle with vertices (x_1, y_1), (x_2, y_2), and (x_3, y_3), and $u(x_j, y_j) = u_j$, $j = 1, 2, 3$. Suppose that we need to compute the gradient of u. Find an exact expression for ∇u within the mesh triangle in terms of (x_j, y_j) and u_j, $j = 1, 2, 3$.

References

[1] P.-O. Persson, G. Strang, A Simple Mesh Generator in MATLAB. SIAM Review, Volume 46 (2), pp. 329-345, June 2004 (PDF)

[2] Jochen Alberty, Carsten Carstensen and Stefan A. Funken, Remarks around 50 lines of Matlab: short finite element implementation