

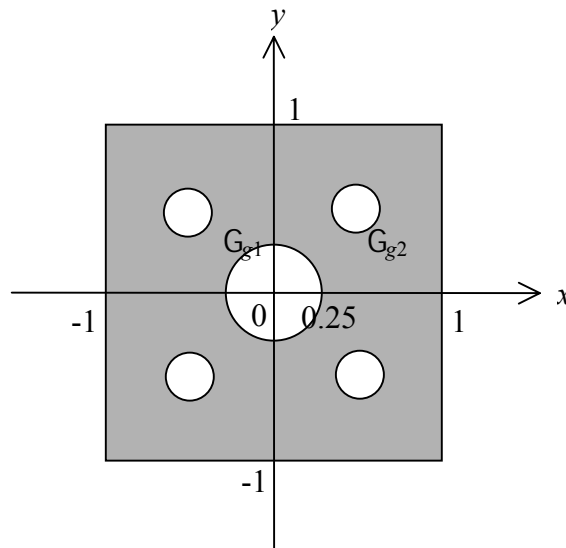
Computing Assignment #2

Handed out Nov. 19 (Mon)

Due Nov. 28 (Wed)

In this assignment, you will be solving a 2-d heat conduction problem over a square domain. The domain is over $W = (-1, 1) \times (-1, 1)$, and there are both interior and exterior boundaries. The exterior boundary is a homogeneous Neumann boundary, the holes in the interior of the element are Dirichlet boundaries with the hole in the middle with radius 0.25 having a prescribed temperature of g_1 , and the other holes located at $(0.5, 0.5)$, $(-0.5, 0.5)$, $(-0.5, -0.5)$, $(0.5, -0.5)$ with radii of 0.1 having a prescribed temperature of g_2 .

This can be thought of as a model of an engine block, where the center of the domain is in contact with a high temperature piston, and there are coolants of a lower temperature passing through the block. The distributed heat source may be due to frictional heat generated by contact with other components throughout the block.



The boundary value problem is stated as follows :

$$q_{i,i} = f \text{ on } W$$

$$u = g_1 \text{ on } G_{g1} \text{ (the center circle)}$$

$$u = g_2 \text{ on } G_{g2} \text{ (the 4 smaller circles)}$$

$$-q_i n_i = h \text{ on } G_h$$

where $f = 0.01$, $g_1 = 700$, $g_2 = 10$, and $h = 0$. The isotropic homogeneous conductivity is given as $k = 1$.

1. Write the subroutines for calculating the element stiffness, element force vector due to the distributed force and the element force vector due to the Dirichlet boundary conditions. As before, the function prototypes are given to you, all you need to do is fill in the function bodies.

(Note that you do not need to find the element force vectors due to the Neumann boundary condition as $h = 0$.)

2. Using a linear triangular mesh generated by the PDE toolbox, solve for the temperature u and plot the results.

You should be able to use the assembly routines you wrote in the first assignment. As before, you will need to manually enter the Dirichlet boundary conditions at the boundary nodes. You can find the nodal points by looking at the mesh.