Use of Matlab PDE toolbox for 2D IVPs and BVPs

If your computer system has the Matlab PDE toolbox, this provides a convenient way of solving many 2nd order PDEs with two space dimensions of the type

$$
\begin{aligned}
-\nabla \cdot (c\nabla u) + au &= f \text{ (Elliptic)}, & (1)\\
d\partial u/\partial t - \nabla \cdot (c\nabla u) + au &= f \text{ (Parabolic)}, & (2)\\
d\partial^2 u/\partial t^2 - \nabla \cdot (c\nabla u) + au &= f \text{ (Hyperbolic)}, & (3)\\
-\nabla \cdot (c\nabla u) + au &= \lambda du \text{ (Elliptic Eigenvalue)}, & (4)
\end{aligned}
$$

in a specified domain Ω using a finite element method. Here, the coefficients c, a, and the right-hand side f may be functions of x and y. The toolbox can even handle quasilinear problems in which the coefficients c and a also depend on u. The domain Ω is built up of unions of rectangles, other polygons, circles, and ellipses using a graphical user interface invoked via the matlab command `pdetool`. A mesh for the FEM is automatically generated and can be arbitrarily refined for higher accuracy. Any combination of Dirichlet BCs $u = r(x, y)$ or Neumann BCs $-c\partial u/\partial n + q(x, y)u = g(x, y)$ can be specified along each part of the boundary $\partial\Omega$.

The Matlab on-line help or the *PDE Toolbox Users Guide* from the Mathworks gives a complete tutorial on how to use the toolbox. Here we will just go through one simple example of how to use `pdetool` to solve the heat equation

$$\partial u/\partial t - \nabla^2 u = 4, t > 0 \qquad (5)$$

in the semicircle Ω: $x^2 + y^2 < 1, y > 0$ with IC $u(x, y, 0) = 0$, a Dirichlet (constant-temperature) BC on the circular edge, and a Neumann (insulating) BC on the diameter:

$$
\begin{aligned}
u(x, y, t) = 0 \text{ on } x^2 + y^2 = 1, & \qquad (6)\\
\frac{\partial u}{\partial y}(x, 0, t) = 0 \text{ on } -1 < x < 1 & \qquad (7)
\end{aligned}
$$

Note that this corresponds to $d = c = 1, a = 0, f = 4$.

After entering Matlab and typing `pdetool` we specify the problem to the pdetool GUI as follows:

1. Click on `Options`. Select `Grid` (draws a background grid), `Snap` (forces domain corners to lie exactly on top of grid) and `Axes Equal` (to get an undistorted view).

2. Click on the ellipse icon (with + in the middle). Place mouse on (0,0) and drag to (1,1) and left-click again. This defines a circular domain C1 of radius 1 centered at (0,0). Then select the rectangle (now without the +) and click on (-1,0), then drag to (1,1) to define a rectangle R1.

3. The desired semicircular domain $\Omega$is the intersection of C1 and R1. To specify this (rather than the union of C1 and R1) change the '+' to '*' in the 'Set formula' line at the top of the toolbox window.

4. Click on the $\partial\Omega$ icon. The domain boundary $\partial\Omega$ will be in red. Double-click on the bottom edge of the semicircle. This will bring up a dialogue box. Selct the 'Neumann' option ; the boxes for g and

q are already correctly set to zero (note c=1 so this will give us $\mathbf{n} \cdot \nabla \mathbf{u} = 0$. Define the Dirichlet BC $u = 0$ for the semicircular edge similarly.

5. Click on the `PDE` icon. Select 'Parabolic' under 'Type of PDE' and write 4 next to `f`; `c` and `d` should already be set to 1. Then click on OK.

6. Click on the triangle icon to get a FEM mesh, which shows up in the pdetool window.

7. On the menu bar, go to 'Solve', click on 'Parameters' and type 0:0.05:1 to get a set of times to output the solution. Note that the initial u(x,y,0) is already set to zero. If it were not you could enter a formula here in standard Matlab notation (e.g. $x^2 + y^2$)

8. Click on the = icon to get the solution as a 2D color shaded plot.

9. To view the plot as a shaded height surface, click on the mesh icon (second from right) and select 'Height' (for 3D surface plot), 'Show mesh' (to superpose the FEM grid), and 'Animate' to show the solution sequentially at all the desired output times. Note the convergence of the solution to a state of steady heat flow from the interior source to the sink at the boundary of the circle. The solution at $t = 1$ and the FEM grid chosen by `pdetool` are shown below.

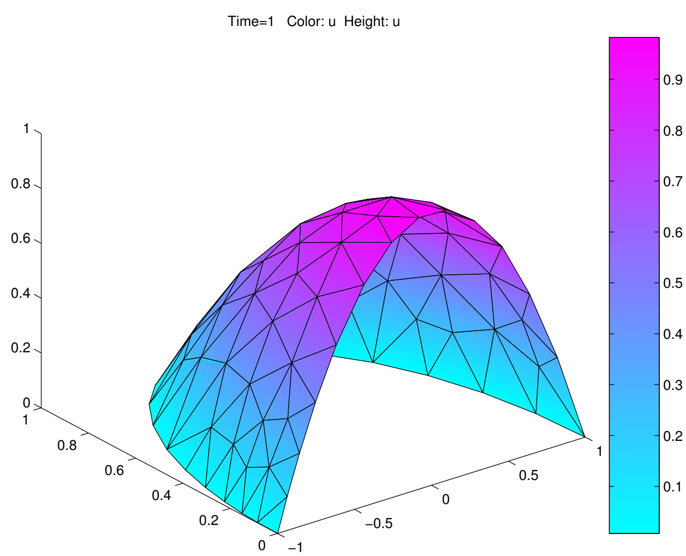

Figure 1: Solution to heat equation at $t = 1$ plotted by `pdetool`