*Partial Differential Equations*

*MA342*

*Ethan Cating*

*Ted Samore*

*Seth Troisi*

*April 28, 2013*

*Problem 1*

Problem 1 asked us to investigate how heat transfers through a 2-D solid (in specific a 10 cm square with 5 cm radius semicircles attached on the north and south side) initial 60 degrees. Both Dirichlet and Neumann boundary conditions were included. The east and west sides of the central square were heated (at 80 degrees) while the other boundaries were heated by the Neumann condition which was a 1 degree gradient.

We used finite differences to solve the model. By breaking down the geometry into small squares (dx = dy = 0.2 cm) the temperature of the square for the next time period is approximated by

u(t+1, x, y) = u(t, x, y) + k / (c \* p \* dx^2 \* dy^2) \*

( (u(t, x-1, y) - 2 \* u(t, x, y) + u(t, x+1, y)) \* dx^2

(u(t, x, y-1) - 2 \* u(t, x, y) + u(t, x, y+1)) \* dy^2)

This is averaging the heat conducted into and out of the four adjacent neighbors and calculating the temperature at the next time step based on density, thermal conductivity, specific heat, and time step. By repeatedly iterating across the grid holding time constant during each iteration the temperature for an arbitrary time in the future is calculated.

We looked at several variations to the problem including heating only one side, cooling the boundary, heating the boundary, and varying the density of the material being heated.

The density that minimized the standard deviation of the temperature of the inner points (at least 1 cm in from the edge) after 1.8 seconds was found to be 0.003214 which resulted in a mean temperature of 79.6 degrees and a standard deviation of 0.696 after the 1.8 seconds.

We utilized the power of C++ (a wonderful compiled language) to run a more massive simulation than Matlab could reasonable handle. We performed our simulations on a 111x111 grid with time steps of 0.000082 seconds which could be run in less than 5 seconds and yet had a large enough grid size to allowed for good visualization. The simulation does have some issues on the edge (especially were the semicircles meet the square).

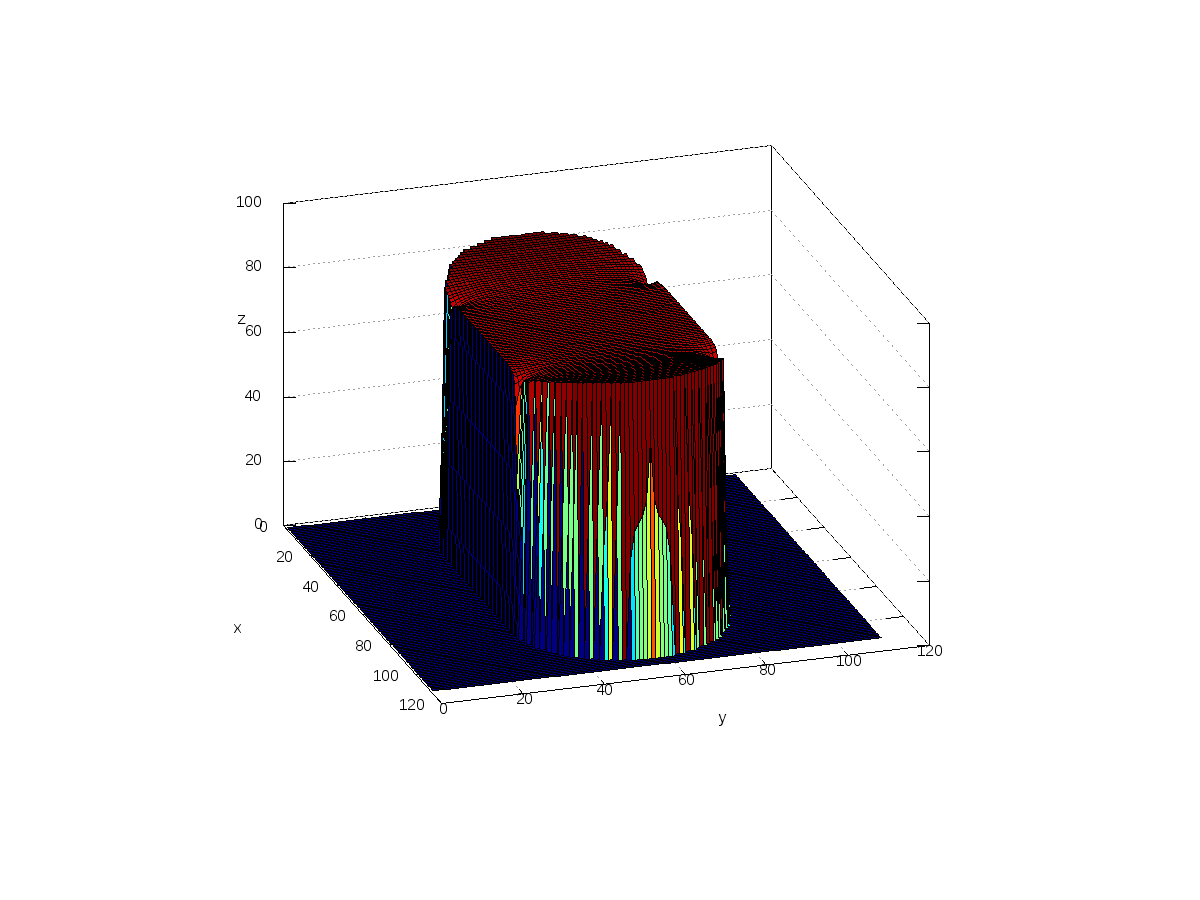


Figure 1: Temperature of the solid after 1.8 seconds with density of 0.003214, 80 degrees on both sides and heat also being added on all other boundaries.

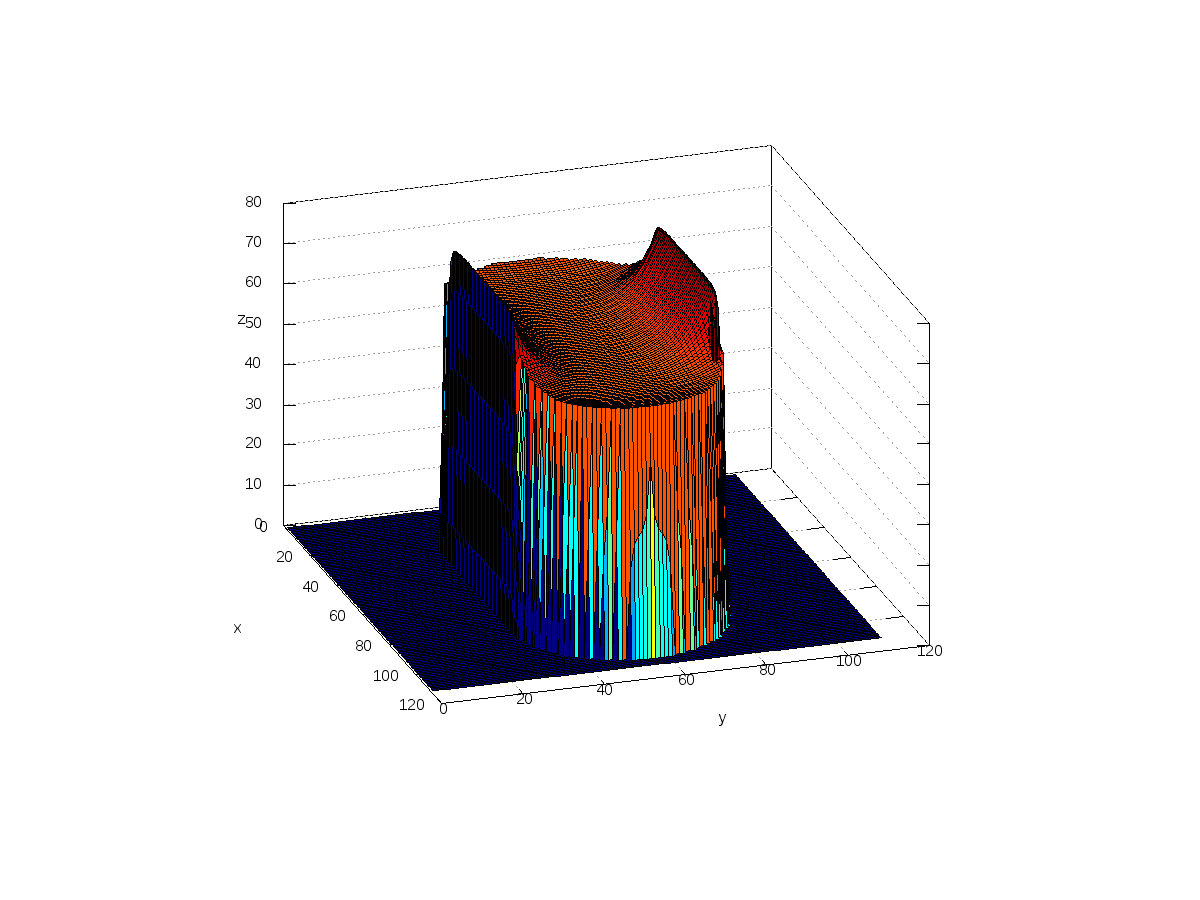


Figure 2: Density of 0.01, which significantly lowers heat transfer.

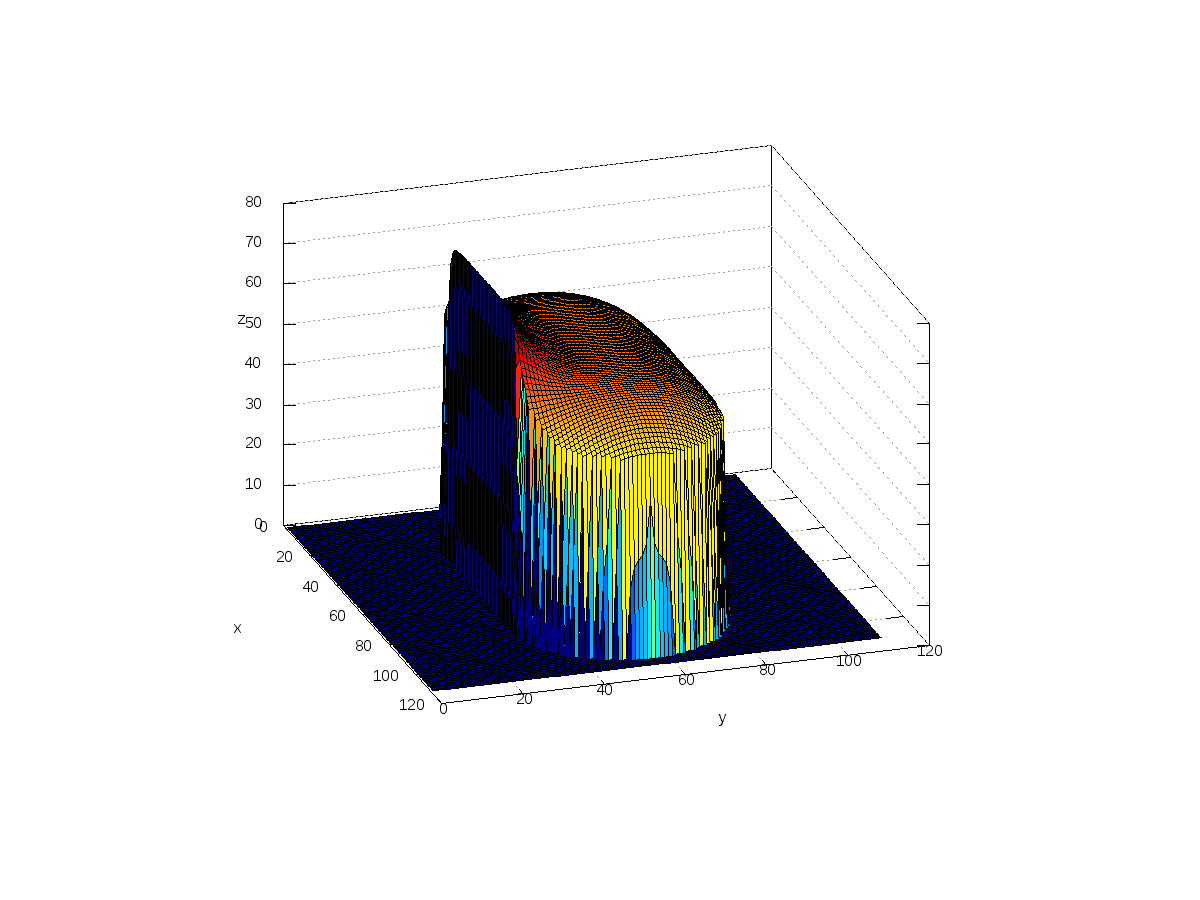


Figure 3: Boundaries being cooling with one side being heated.

Figure 1 demonstrates a minimized heat difference, Figure 2 shows a denser solid, Figure 3 demonstrates boundary cooling similar such as on a solid heatsink.

*Problem 2*

*History*

The wave equations derivation came from a long time spent in the work of partial differential equations. Many great mathematicians including Laplace, Euler, Bernoulli, and Lagrange studied this behavior intensely. The vibration of strings on musical instruments was where the problem really began for these mathematicians and where the wave phenomena were first noticed. Nowadays all kinds of complex shapes and dimensions can be calculated using computational numerical methods based on partial differential equations.

*Introduction*

The most common equation for evaluating the classical problem of the movement of a vibrating string is the wave equation. A vibrating string’s oscillatory motion gives it positive and negative displacement as it normally moves in the shape of a sine or cosine. The overall shape at full displacement depends on the initial conditions given. For this problem the string is one-dimensional with movement only along the x-coordinate, is held fixed at both ends, and includes a damping term. The damping term continually dissipates some of the strings energy as time goes on, and eventually reduces the displacement to zero. The design constraint for this problem is to find a value for the damping term κ that keeps the displacement of the string positive for all time.

*Model and Experimentation*

The wave equation adapted with the damping term is shown in the following equation:

With the initial conditions of

And the boundary conditions of

.

The final central difference method equated to

Where

When the explicit form of this was developed, it was implemented in MATLAB. The bisection method was used to converge on a solution within an error of 10^-6. With the bisection method, no error function was used, but the loop was iterated for a set number of time steps until convergence. Several different amount of iteration sizes were used to check to make sure that κ converged to the same number for different iteration sizes. The final solution for the minimum damping term is The following figures show the results graphically of the damped wave.



Figure 4: Motion plot of wave showing the damped motion keeps displacement greater than zero for all time *t.*



Figure 5: Plot of every hundredth time step for damped wave.

*Problem 3*

In problem three, we solved the wave equation in two dimensions over a unit disc. The wave constant was two, so the PDE becomes

To numerically solve this PDE, we discretized space in two dimensions rather than using polar coordinates. The wave equation requires an initial condition, and an initial velocity at all spatial points. Our initial conditions were

This means that the initial velocity is zero, and that u(x, y, 0) = u(x, y, ∆t) = . Figure 1 depicts this function

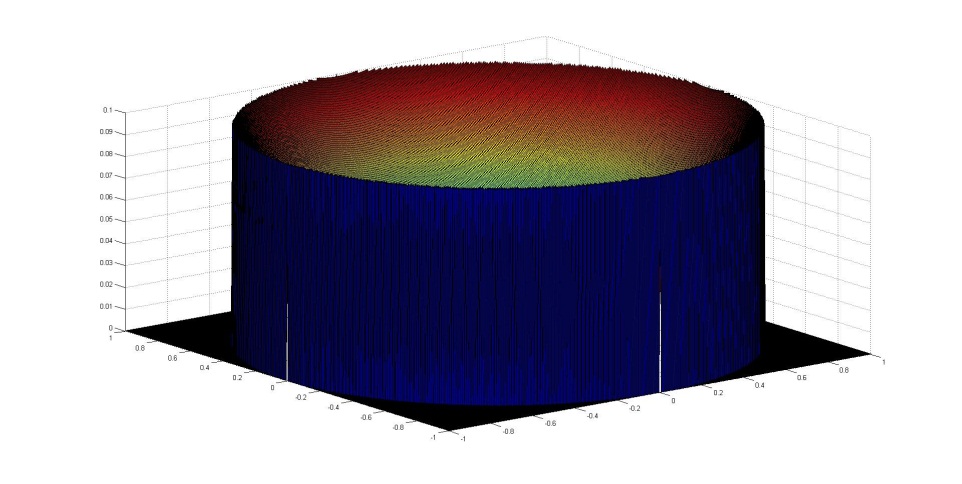


Figure 6: The function for the first two time steps.

As the solution progresses, the wave travels from the radius to the center of the circle, and then back out.

A video of the solution is attached. The wave travels from radius (as shown in figure 1) to the center, and back out.