Modeling Waves and Thermal Dissipation With Partial Differential Equations

Alan Bruner, Evan Cochrane, Theodore Guetig, Ben Wilfong

Introduction

The heat equation and wave equation are standard partial differential equations that model how the temperature of an object changes over time and how waves propagate though an object over time, respectively. In this paper, we use finite difference methods to approximate solutions to both of these models. Using a discretization of the heat equation, we simulate a two-dimensional region made of various materials to determine which material yields the most uniform temperature distribution after two seconds. A discretization of the wave equation is used to simulate the movement of a drum head using various boundary shapes defined by p-norms as well as determine the critical damping coefficient for a string.

Models

The Heat Equation

The heat equation is developed by analyzing the total heat $u(x, y, t)\Delta x\Delta y$ in a two-dimensional differential volume. Let $\boldsymbol{F}_x(x, y, t)$ be the rate at which heat flows from left to right and $\boldsymbol{F}_y(x, y, t)$ be the rate at which heat flows from bottom to top. The fate of change change in heat in the differential volume is

$$\begin{aligned}\frac{\partial(u\Delta x\Delta y)}{\partial t} \approx \Delta y\boldsymbol{F_x}\left(x-\frac{\Delta x}{2}, y, t\right) - \Delta y\boldsymbol{F}_x\left((x+\frac{\Delta x}{2}, y, t\right) \\ + \Delta x\boldsymbol{F}_y\left(x, y-\frac{\Delta y}{2}, t\right) - \Delta x\boldsymbol{F}_y\left(x, y+\frac{\Delta y}{2}, t\right). \qquad (1)\end{aligned}$$

After dividing both sides by $\Delta x\Delta y$ and taking the limit at $\Delta x \to 0$ and $\Delta y \to 0$, (1) reduces to

$$\frac{\partial u}{\partial t} + \frac{\partial \boldsymbol{F}_x}{\partial x} + \frac{\partial \boldsymbol{F}_y}{\partial y} = 0. \qquad (2)$$

Newtons law of cooling tells us that

$$\boldsymbol{F}_x = -\alpha\frac{\partial u}{\partial x} \quad \text{and} \quad \boldsymbol{F}_y = -\alpha\frac{\partial u}{\partial y} \qquad (3)$$

for some positive constant α. Substituting (3) into (2) yields

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(\alpha\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\alpha\frac{\partial u}{\partial y}\right).$$

This yields the two dimensional heat equation

$$\frac{\partial u}{\partial t} = \alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = \alpha \Delta u. \tag{4}$$

The boundary of a region can be modeled by either Dirichlet condition

$$u(x, y, t) = g(x, y, t),$$

for some known function g defined only on the boundary, or a Neumann boundary condition

$$u(x, y, t) \cdot n(x, y) = \beta,$$

where n is the outward-facing unit normal vector to the boundary and β is a scalar constant. Dirichlet boundary conditions correspond to constant temperature boundaries in the physical world while Neumann conditions correspond to applied heat fluxes.

The Wave Equation

The wave equation is developed by analyzing a finite length of a stretched string shown in Figure 1 [4].

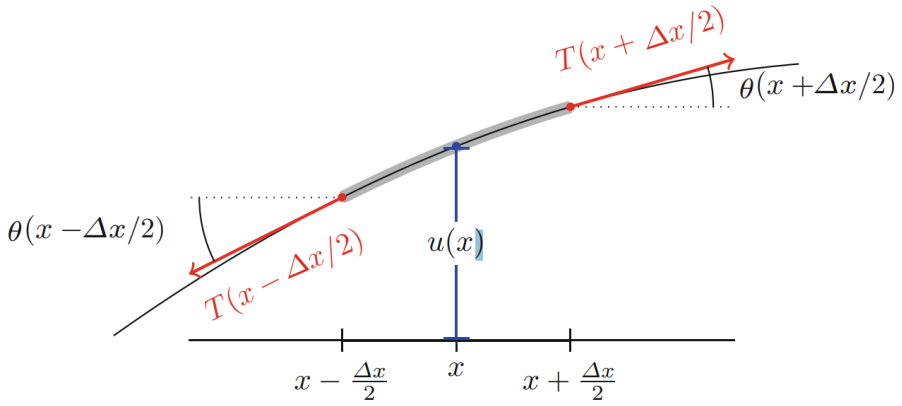

Figure 1: Forces and Displacements of a Small Length of Stretched String [4]

Newton's second law of moion tells us that

$$ma = F_{net}.$$

For the case of a string with linear density ρ, length Δx, and position $u(x, t)$, this becomes

$$\rho \Delta x \frac{\partial^2 u}{\partial t^2} = F_{net}. \tag{5}$$

For a string that only moves vertically, the horizontal forces satisfy the equality

$$T\left(x+\frac{\Delta x}{2}\right)\cos\left(\theta\left(x+\frac{\Delta x}{2}\right)\right)=T\left(x-\frac{\Delta x}{2}\right)\cos\left(\theta\left(x-\frac{\Delta x}{2}\right)\right).$$

This equality holds at all values of x, requiring that magnitude of the horizontal component of the tension H is uniform at all points of the string. This yields

$$H=\cos(\theta(x))T(x). \tag{6}$$

Summing vertical forces on the string yields the net vertical force

$$F_{net}=T\left(x+\frac{\Delta x}{2}\right)\sin\left(\theta\left(x+\frac{\Delta x}{2}\right)\right)-T\left(x-\frac{\Delta x}{2}\right)\sin\left(\theta\left(x-\frac{\Delta x}{2}\right)\right) \tag{7}$$

assuming that gravity is negligible. It follows from multiplying both sides of (6) by $\tan(\theta)$ that

$$\sin(\theta(x))T(x)=H\tan(\theta(x)).$$

Making this substitution in (7) yields

$$F_{net}=H\left[\tan\left(\theta\left(x+\frac{\Delta x}{2}\right)\right)-\tan\left(\theta\left(x-\frac{\Delta x}{2}\right)\right)\right]. \tag{8}$$

Since

$$\tan x=\frac{\text{Opposite}}{\text{Adjacent}},$$

and the tension force is acting tangent to the string's displacement,

$$\tan\theta(x)=\frac{\partial u}{\partial t}(x,t).$$

Making this substitution in (8) yields

$$F_{net}=H\left[\frac{\partial u}{\partial t}\left(x+\frac{\Delta x}{2},t\right)-\frac{\partial u}{\partial t}\left(x-\frac{\Delta x}{2},t\right)\right]. \tag{9}$$

Substituting this into (5), dividing both sides by $\rho\Delta x$, and taking the limit as $\Delta x\to 0$ yields the 1-Dimensional wave equation

$$\frac{\partial^2 u}{\partial t^2}=\frac{H}{\rho}\frac{\partial^2 u}{\partial x^2}. \tag{10}$$

Equation (10) is easily extrapolated to higher dimensions by adding second derivatives in space on the right-hand side. For two dimensions, this results in

$$\frac{\partial^2 u}{\partial t^2}=\frac{H}{\rho}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right). \tag{11}$$

The natural boundary treatment is the Dirichlet condition,

$$u_{x,y,t}=g(x,y,t)$$

for some known function $g(x,y,t)$ defined on the boundary. Physically, this condition is equivalent to holding the region at a known position at its boundary.

Solving The Models

Finite Difference Methods

Exact solutions to PDEs are generally not computable. Finite difference methods use approximations of derivatives to yield an approximate solution. The *forward difference* approximates the derivative of f at a point x using the secant slope through $(x, f(x))$ and $(x+\Delta x, f(x+\Delta x))$ for a small Δx. Similarly, the *backwards difference* uses the points with x-coordinates $x-\Delta x$ and x, and the *central difference* uses the points $x-\Delta x$ and $x+\Delta x$.

In one dimension, the heat equation is

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2},$$

the forward difference approximation for $\frac{\partial u}{\partial t}$ is

$$\frac{\partial u}{\partial t}(x,t) \approx \frac{u(x,t+\Delta x) - u(x,t)}{\Delta t},$$

and the central difference approximation for $\frac{\partial^2 u}{\partial x^2}$ is

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)}{\Delta x^2}.$$

Substituting the two approximations into the heat equation gives the approximation

$$u(x,t+\Delta t) \approx u(x,t) + \alpha \frac{\Delta t}{\Delta x^2}(u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t).$$

We approximate $u(x,t)$ using a finite number of points $x_0, ..., x_n$ all spaced a distance of Δx apart, and time steps $t_0, ..., t_m$ a distance of Δt apart. Then, we compute a matrix of temperatures $u_{i,j} = u(x_i, t_j)$ given initial values and boundary conditions using the following algorithm that uses the Dirichlet condition:

Algorithm 3.1 One-Dimensional Heat Equation Solver

Ensure: $n, m \in \mathbb{Z}^+$, $\alpha, \Delta t, \Delta x \in \mathbb{R}^+$, $f : \mathbb{R} \to \mathbb{R}$, $g : \mathbb{R}^2 \to \mathbb{R}$

for $i \in \{0, 1, ...n\}$ **do**
    $x_{i,0} \leftarrow f(0)$
end for
for $j \in \{1, ...m\}$ **do**
    $x_{0,j} \leftarrow g(0, j)$
    $x_{n,j} \leftarrow g(n, j)$
    for $i \in \{1, ...n - 1\}$ **do**
        $x_{i,j} \leftarrow \alpha \frac{\Delta t}{\Delta x^2}(u_{i+1,j-1} - 2u_{i,j-1} + u_{i-1,j-1})$
    end for
end for
return x

Solving The Heat Equation in Two Dimensions

In our simulations we solve the heat equation in 2 dimensions on the region shown in Figure 2.

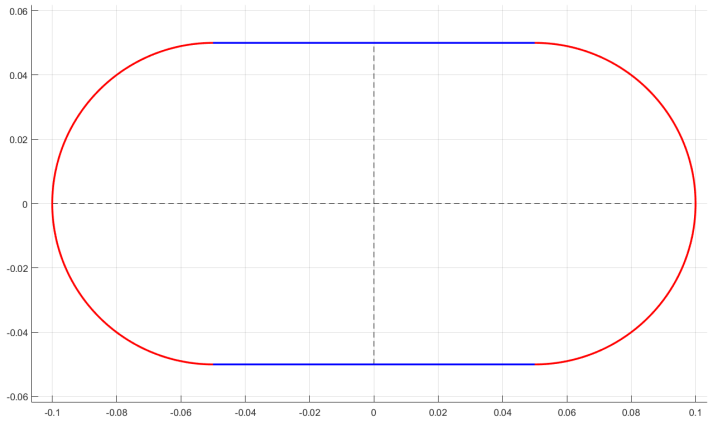

Figure 2: Heat Equation Solution Region. Both axes are in meters. The blue boundaries are solved with a Dirichlet condition and the red boundaries are solved with a Neumann condition.

A Dirichlet condition $g(t, x, y) = 60\ ^o\text{C}$ is applied to the regions of the boundary in blue and the Neumann condition $\nabla u(t, x, y) \cdot n(x, y) = -1$ is applied to the regions of the boundary in red. The initial condition is $f(x, y) = 60\ ^o\text{C}$. To reduce computational expense, we note that the region is symmetric along both the x and y axis, so only the first quadrant need be simulated. Symmetry boundary conditions are applied along the horizontal and vertical dashed lines.

We discretize the surface as in the previous section by uniformly spacing points along the x and y axis and over time using a three-dimensional temperature array or mesh $u_{i,j,k} = u(x_i, y_j, t_k)$ for all $0 \leq i \leq n$, $0 \leq j \leq m$, and $0 \leq k \leq p$ like in the previous section. Figure 3 shows the boundary of the surface in the first quadrant with the curved region indicated by a solid red line.

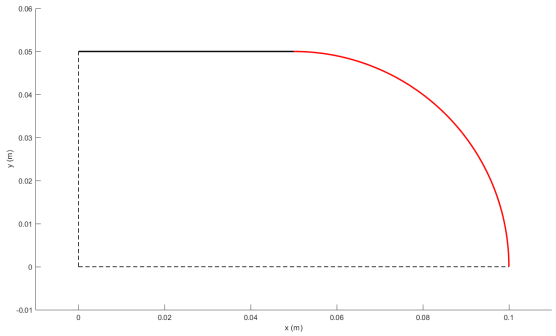

Figure 3: Heat Equation Boundary

To enforce the boundary conditions, we label each point with a number between 0 and 8. Figure 4 shows an example of each of the nine node types, seven of which are boundary types.

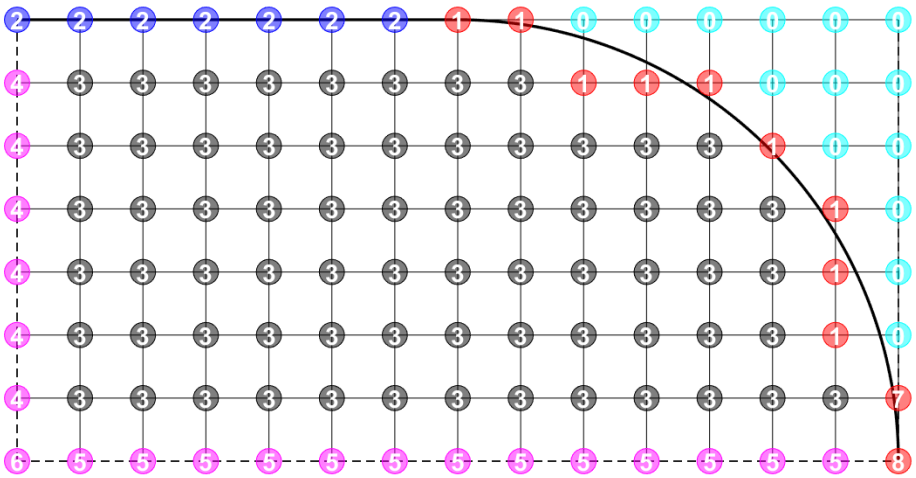

Figure 4: Labels for an example discretization. Cyan points are outside the region, red points are on the curved Neumman boundary, blue points are on the straight Dirichlet boundary, magenta points are on the symmetric boundaries, and black points are on the interior.

Assigning a label to each point, and the curved boundary points in particular, is not trivial. The curved region is defined by

$$\sigma = \{(x, y) \in \mathbb{R}^2 \mid (x - 0.05)^2 + y^2 = 0.05^2\}.$$

Point $u_{i,j}$ is added to the boundary if it minimizes

$$D(i, j) = \sqrt{d_x^2 + d_y^2},$$

where d_x and d_y are the horizontal and vertical distance between a point and σ. Figure 5 show the results of this discrete approximation for the case $n = 10$ and $m = 19$. The label assignments algorithm is described in Algorithm 3.2.

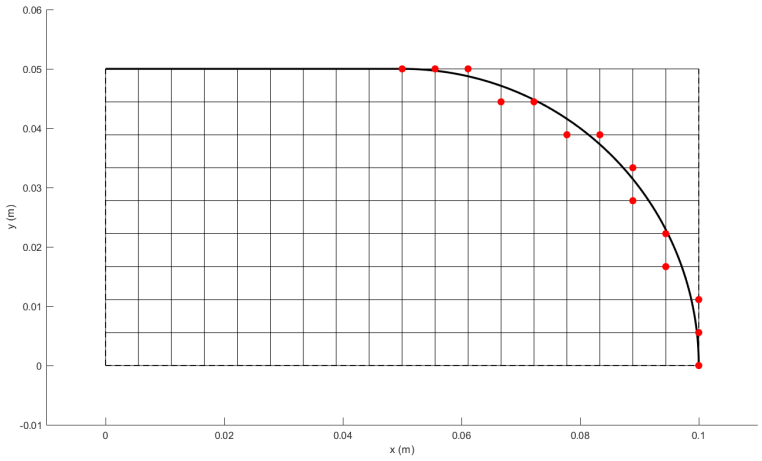

Figure 5: Discretized Boundary Points

After assigning labels, the calculation at each timestep begins. Type 0 nodes are outside the region we are computing, and as such they are set to NaN at each time step. Type three nodes are time advanced by

$$u_{x,y,t+\Delta t} = u_{x,y,t} + \left(\frac{k\Delta t}{\rho c h^2}\right)\left(u_{x-h,y,t} + u_{x+h,y,t} + u_{x,y-h,t} + u_{x,y+h,t} - 4u_{x,y,t}\right).$$

Type two nodes in blue are time advanced by

$$u_{x,y,t+\Delta t} = 60\ ^o\text{C}$$

Type four nodes in magenta are time advanced by

$$u_{x,y,t+\Delta t} = u_{x,y,t} + \left(\frac{k\Delta t}{\rho c h^2}\right)(2u_{x+h,y,t} + u_{x,y+h,t} + u_{x,y-h,t} - 4u_{x,y,t}).$$

Algorithm 3.2 Assignment of Labels to Points in Figure 4

Ensure: $n \in \mathbb{Z}^+$

$m \leftarrow 2n - 1$
$labels_{i,j} \leftarrow 0 \quad \forall 0 \leq i \leq n, 0 \leq j \leq m$
$C \leftarrow \{\}$
for $i \in \{0, ..., n\}$ **do**
    $C \leftarrow C \cup \arg\min_j D(i, j)$
end for
for $j \in \{0, ..., m\}$ **do**
    $C \leftarrow C \cup \arg\min_i D(i, j)$
end for
for $c \in C$ **do**
    $labels_c \leftarrow 1$
end for
$j \leftarrow 0$
while $labels_{n,j} \neq 1$ **do**
    $labels_{n,j} \leftarrow 2$
    $j \leftarrow j + 1$
end while
for $i \in \{0, ..., n - 1\}$ **do**
    $j \leftarrow 0$
    while $labels_{i,j} \neq 1$ **do**
        $labels_{i,j} \leftarrow 3$
        $j \leftarrow j + 1$
    end while
end for
$labels_{i,0} \leftarrow 4 \quad \forall 1 \leq i \leq n - 1$
$labels_{0,j} \leftarrow 5 \quad \forall 1 \leq j \leq m - 1$
$labels_{0,m} \leftarrow 8$
$labels_{0,m+1} \leftarrow 7$
$labels_{0,0} \leftarrow 6$
return $labels$

Type five nodes in magenta are time advanced by

$$u_{x,y,t+\Delta t} = u_{x,y,t} + \left(\frac{k\Delta t}{\rho c h^2}\right)(u_{x+h,y,t} + u_{x-h,y,t} + 2u_{x,y+h,t} - 4u_{x,y,t}).$$

Type six node in magenta is time advanced by

$$u_{x,y,t+\Delta t} = u_{x,y,t} + \left(\frac{k\Delta t}{\rho c h^2}\right)(2u_{x+h,y,t} + 2u_{x,y+h,t} - 4u_{x,y,t}).$$

The equations for types four, five, and six result from the symmetry boundary condition that requires

$$u_{x-h,y,t} = u_{x+h,y,t}$$

on the left edge of the region and

$$u_{x,y-hy,t} = u_{x,y+hy,t}$$

on the bottom edge of the region. The Neumann condition applied to the nodes marked in red is expanded as

$$\nabla u(x,y,t) \cdot n = -1 \rightarrow \frac{\partial u_{x,y,t}}{\partial x} n_x + \frac{\partial u_{x,y,t}}{\partial y} n_y = -1$$

The second order backward differences for the first derivative are

$$\frac{\partial u_{x,y,t}}{\partial x} = \frac{1}{2h}\left(3u_{x,y,t} - 4u_{x-\Delta x,y,t} + u_{x-2\Delta x,y,t}\right),$$
$$\frac{\partial u_{x,y,t}}{\partial y} = \frac{1}{2h}\left(3u_{x,y,t} - 4u_{x,y-\Delta y,t} + u_{x,y-2\Delta y,t}\right).$$

These are used to calculate the value of type one nodes in red via

$$u_{x,y,t+\Delta t} = \left[\frac{3(n_x+n_y)}{2h}\right]^{-1}\Big(-1 + \frac{n_x}{2h}\left(4u_{x-h,y,t} - u_{x-2h,y,t}\right)... + \frac{n_y}{2h}\left(4u_{x,y-h,t} - u_{x,y-2h,t}\right)\Big).$$

The corresponding equations for nodes seven and eight are

$$u_{x,y,t+\Delta t} = \left[\frac{3n_x+4n_y}{2h}\right]^{-1}\Big(-1 + \frac{n_x}{2h}\left(4u_{x-h,y,t} - u_{x-2h,y,t}\right)... + \frac{2n_y}{h}\left(u_{x,y-h,t}\right)\Big),$$

and

$$u_{x,y,t+\Delta t} = -\frac{2h}{3} + \frac{4}{3}u_{x-h,y,t} - \frac{1}{3}u_{x-2h,y,t},$$

respectively.

The solver algorithm is similar to Algorithm 3.1, except lines 4,5 and 6 are replaced with a loop over all points in the mesh, and the label is used to determine which of the above formulas are used to advance the temperature at that point.

Solving the Wave Equation in One Dimension

In one dimension, the wave equation is

$$\frac{\partial^2 u}{\partial t^2} = \frac{H}{\rho}\frac{\partial^2 u}{\partial x^2}. \tag{12}$$

The one dimensional case corresponds to a vibrating string of a fixed length. H is the horizontal tension on the string and ρ is the linear density of the string. By using a central difference to expand all second order derivatives, (12) becomes

$$\frac{1}{(\Delta t)^2}(u_{x,t+\Delta t} - 2u_{x,t} + u_{x,t-\Delta t}) = \frac{H}{(\rho\Delta x)^2}(u_{x+\Delta x,t} - 2u_{x,t} + u_{x-\Delta x,t}) \tag{13}$$

Where Δt is the selected time step and Δx is the selected spatial step. In this discretization, $u_{x,t+\Delta t}$ represents the solution at the next time step of, $u_{x+\Delta x,t}$ represents the solution in the next spatial step, and $u_{x,t}$ represent the current solution. Two initial conditions are necessary to simulate the wave equation due to its second order dependency on time. The initial conditions are as follows:

$$u(x,0) = 0 \tag{14}$$

$$\frac{\partial u}{\partial t}(x,0) = e^{-10(x-L/2)^2} - e^{-10(L/2)^2} \tag{15}$$

These initial conditions are used to predict the solution at the next time step, $u_{x,t+\Delta t}$. Solving (13) for $u_{x,t+\Delta t}$ yields,

$$u_{x,t+\Delta t} = \frac{H(\Delta t)^2}{\rho(\Delta x)^2}(u_{x+\Delta x,t} - 2u_{x,t} + u_{x-\Delta x,t}) + 2u_{x,t} - u_{x,t-\Delta t} \tag{16}$$

This approach for the solution is known as an explicit solver which introduces the problem of conditional stability in our solution. To maintain stability in the solution, the values of the selected time and spatial step must satisfy the inequality

$$\Delta t < \Delta x\sqrt{\frac{\rho}{H}}. \tag{17}$$

Failure to satisfy this inequality leads to an unstable solution that eventually violates the imposed boundary conditions. To address the spatial derivative, the string has a fixed length of L with the following Dirichlet conditions:

$$u(0,t) = u(L,t) = 0. \tag{18}$$

Now, simulation of the one dimension wave commences. The results show a wave that maintains its size and magnitude as time approaches infinity. In controls, the desired performance is for the wave to return to rest as soon as possible while minimizing its oscillation. This introduces a new dampening term in the wave equation as shown below

$$\frac{\partial^2 u}{\partial t^2} = \frac{H}{\rho}\frac{\partial^2 u}{\partial x^2} - \kappa\frac{\partial u}{\partial t} \tag{19}$$

where κ is a chosen unit-less variable that proportionally relates the dampening term to velocity. The value of κ can be adjusted to change the impact of the dampening term and need to be tested to find the optimum value. By applying a central finite difference, the wave equation becomes

$$\frac{1}{(\Delta t)^2}(u_{x,t+\Delta t} - 2u_{x,t} + u_{x,t-\Delta t}) = \frac{H}{\rho(\Delta x)^2}(u_{x+\Delta x,t} - 2u_{x,t} + u_{x-\Delta x,t}) - ... \\ \frac{k}{\Delta t}(u_{x,t+\Delta t} - u_{t,x}) \tag{20}$$

The dampening term expanded by central finite difference introduces another $u_{x,t+\Delta t}$ in the equation. By solving for the next time step, our new expression is

$$u_{x,t+\Delta t} = \frac{H(\Delta t)^2}{(\kappa\Delta t + 1)\rho(\Delta x)^2}(u_{x+\Delta x,t} - 2u_{x,t} + u_{x-\Delta x,t}) + ... \\ \frac{1}{\kappa\Delta t + 1}(2u_{x,t} - u_{x,t-\Delta t}) + \frac{\kappa\Delta t}{\kappa\Delta t + 1}u_{x,y} \tag{21}$$

Using the same initial conditions and boundary conditions as the non-dampened case, the model is fully defined and ready to solve. All that is left is to find the best value of κ to provide the most stable return to rest.

Solving the Wave Equation in Two Dimensions

Transitioning from the one dimensional case to the two dimensional case introduces a new spatial term and dependency on the equations. To account for this addition, equation 12 is updated to the following form.

$$\frac{\partial^2 u}{\partial t^2} = \alpha\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \tag{22}$$

Where α is held constant at 2. The same central finite difference from earlier is applied the model.

$$\frac{1}{(\Delta t)^2}(u_{x,y,t+\Delta t} - 2u_{x,y,t} + u_{x,y,t-\Delta t}) = \frac{2}{(\Delta x)^2}(u_{x+\Delta x,y,t} - 2u_{x,y,t} + u_{x-\Delta x,y,t})... \\ - \frac{2}{(\Delta y)^2}(u_{x,y+\Delta y,t} - 2u_{x,y,t} + u_{x,y-\Delta y,t}) \tag{23}$$

Where Δt is the selected time step, Δx is the selected spatial step in the x direction, and Δy is the selected spatial step in the y direction. As expected, the modelling equation become more complex but still remains manageable. To help simplify the model, n uniformed mesh is used such that $\Delta x = \Delta y$. Keep in mind that $u_{x+\Delta x,y,t}$ is the next spatial step in the x direction whereas $u_{x,y+\Delta y,t}$ is the next spatial step in the y direction. Similar logic applies to other alternations of the solution. Following the same procedures, the equation is rearranged to solve for $u_{x,y,t+\Delta t}$.

$$u_{x,y,t+\Delta t} = \frac{2(\Delta t)^2}{(\Delta x)^2}(u_{x+\Delta x,y,t} - 2u_{x,y,t} + u_{x-\Delta x,y,t}) + ... \\ \frac{2(\Delta t)^2}{(\Delta y)^2}(u_{x,y+\Delta y,t} - 2u_{x,y,t} + u_{x,y-\Delta y,t}) + 2u_{x,y,t} - u_{x,y,t-\Delta t} \quad (24)$$

Using an explicit solver to model the problem introduces the concern of conditional stability. Refer to Appendix B for more information about the stability of the two dimensional wave equation. To simulate the model, boundary conditions and initial conditions are applied. The initial conditions are:

$$u(x, y, 0) = \frac{||(x, y)||_p}{10}, \quad (25)$$

$$\frac{\partial u}{\partial t}(x, y, 0) = 0. \quad (26)$$

The boundary condition is,

$$u(x, y, t) = \frac{||(x, y)||_p}{10}. \quad (27)$$

Figure 6 shows the boundary conditions in the model region.

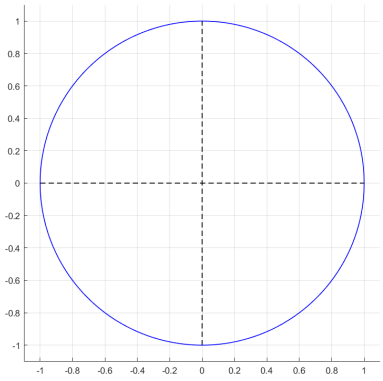

Figure 6: Wave Equation Solution Region. Both axes are in meters. The blue boundaries are solved with a Dirichlet condition.

As with the heat equation, taking advantage of the problem's symmetry allows for the mesh to be one fourth of the model region. New boundary conditions are applied to the interior of the mesh to protect its symmetry by labeling the boundaries with values 0 to 5 as shown in Figure 7.

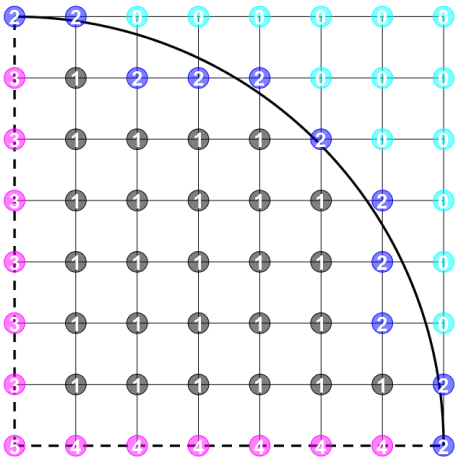

Figure 7: Labels for wave equation discretization. Cyan points are outside the region, blue points are on the Dirichlet boundary, magenta points are on the symmetric boundaries, and black points are on the interior

Using the node labels, the appropriate external and internal boundary conditions are applied. Type zero nodes are set to NaN since they are outside the region of interest. Type 1 nodes are subject to equation 24 since they are can

use information from the nodes surrounding them. Type two nodes are enforced by the Dirichlet condition $u(x, y, t) = 0.1$. Type three nodes cannot use equation 24 since they are missing nodes on their left. Instead, symmetry on the left boundary is applied and results in the following equation

$$u_{x,y,t+\Delta t} = \frac{2(\Delta t)^2}{(\Delta x)^2}(u_{x+\Delta x,y,t} - 2u_{x,y,t} + u_{x-\Delta x,y,t}) + ... \\ \frac{2(\Delta t)^2}{(\Delta y)^2}(2u_{x,y+\Delta y,t} - 2u_{x,y,t}) + 2u_{x,y,t} - u_{x,y,t-\Delta t}.$$

Likewise, symmetry is applied to type four nodes but on the bottom bounds yielding.

$$u_{x,y,t+\Delta t} = \frac{2(\Delta t)^2}{(\Delta x)^2}(2u_{x+\Delta x,y,t} - 2u_{x,y,t}) + ... \\ \frac{2(\Delta t)^2}{(\Delta y)^2}(u_{x,y+\Delta y,t} - 2u_{x,y,t} + u_{x,y-\Delta y,t}) + 2u_{x,y,t} - u_{x,y,t-\Delta t}$$

Type five nodes are missing information below them and to their left. Two applications of symmetry yield

$$u_{x,y,t+\Delta t} = \frac{2(\Delta t)^2}{(\Delta x)^2}(2u_{x+\Delta x,y,t} - 2u_{x,y,t}) + \frac{2(\Delta t)^2}{(\Delta y)^2}(2u_{x,y+\Delta y,t} - 2u_{x,y,t})... \\ + 2u_{x,y,t} - u_{x,y,t-\Delta t}.$$

With the proper conditions applied to the nodes, initialization of the model can begin with a positive p-norm. Different p-norms generate difference mesh shapes to simulate the two dimensional wave equation. Exploration of the p-norm continues in the results sections.

Results

Heat Equation

The temperature distribution after two seconds was calculated for seventeen different materials. This exploration included metals, plastics, glasses, and gasses. Table 1 summarizes the material properties of eat material as well as each materials temperature range after 2 seconds. The material with the most uniform temperature distribution after 2 seconds was acrylic with a temperature range of only 0.000677 ^{o}C after two seconds. The material with the least uniform temperature distribution after 2 seconds was type IIA diamond, with a temperature range of 0.0343^{o}C. The temperature distributions of acrylic and diamond

Table 1: Materials Explored

Material	k (W/m·K)	ρ (kg/m^3)	c (J/kg·^{o}C)	Range(T) ^{o}C
Pure Aluminum [2]	220.0	2.707e3	896.0	0.0174
Pure Copper [2]	386.0	8.954e3	380.0	0.0197
Pure Gold [2]	318.0	18.9e3	130.0	0.0213
Pure Lead [2]	35.0	11.373e3	130.0	8.39e-3
Tungsten Steel (W5%) [2]	54.0	8.073e3	435.0	6.74e-3
Acrylic [3]	0.06	1.19e3	1.5e3	3.93e-4
Type IIA Diamond [3]	2.0e3	3.5e3	2.0e3	0.0343
Pyrex Glass [3]	1.02	2.23e3	837.0	1.29e-3
Nylon [3]	0.242	1.1e3	1.7e3	6.77e-4
LD Polyethylene [3]	0.35	920.0	2.3e3	7.52e-4
HD Polyethylene [3]	0.5	950.0	2.3e3	8.66e-4
RT Hydrogen [1]	0.182	0.08185	14.314e3	0.0238
RT Oxygen [1]	0.02676	1.3007	920.3	8.15e-3
RT Nitrogen [1]	0.0262	1.1421	1.0408e3	8.09e-3
RT Carbon Dioxide [1]	0.016572	1.7973	871.0	5.60e-3
Water Vapor (577^{o}C) [1]	0.0637	0.2579	2.186e3	0.0197
Water Vapor (107^{o}C) [1]	0.0246	0.5863	2.06e3	7.77e-3

RT - Room Temperature | HD - High Density | LD - Low Density

after two seconds are shown in Figures 8 and 9 respectively. Animations for each material can be found on YouTube [1].

MATLAB is optimized for matrix operations and not for "for" loops, as it is an interpreted language. To improve computational speed, a mex file was created, allowing the algorithm that processes a single timestep to be run in a faster, complied C environment. Table 2 shows a comparison of run-times for several T sizes and time-steps. In every case, there was an improvement in overall run-time. However, this run-time decrease was limited in most cases. This is likely due to the mex file only being run on a single timestep, which caused the overhead cost of switching environments to negate the increase gained by the increased loop speed. By creating a mex file that could run every timestep, computation times could further be reduced.

Table 2: Mex vs. MATLAB runtimes

Array Size	Time-Steps	Steps per frame	Run-time (sec)	
			Mex	MATLAB
100x200	10,000	100	94.28	94.33
1000x2000	1000	100	205.42	223.68
1500x3000	2000	250	467.73	493.68

[1] https://youtu.be/utg9CNeSCK0

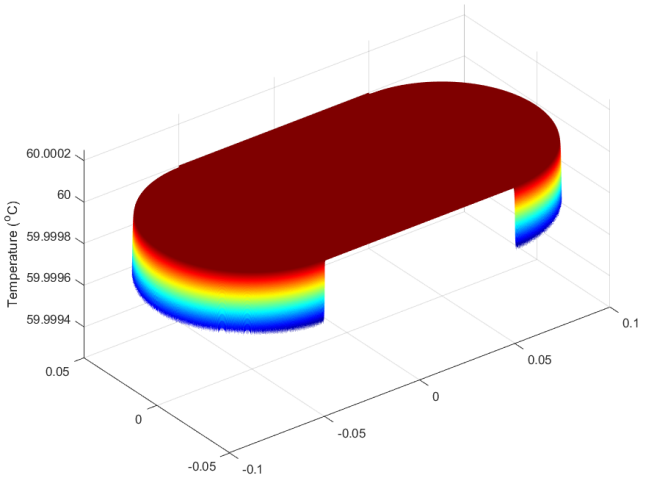

Figure 8: Temperature Distribution of Acrylic After 2 Seconds

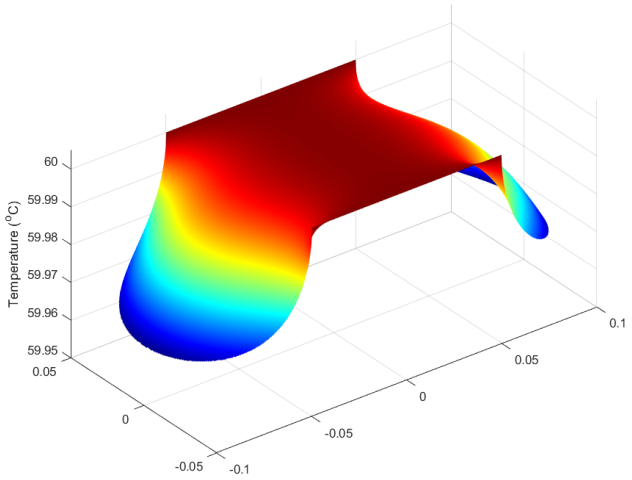

Figure 9: Temperature Distribution of Type IIA Diamond After 2 Seconds

Wave Equation

Modeling the non-dampened case results in a continuous and steady oscillation as time approaches infinity. However, the dampened case requires an extensive search to find the best value for κ. Optimizing κ minimizes oscillation and

maintains a positive solution for most of the simulation. For a simulation of 30 seconds, the κ search is outlined in Figure 10.

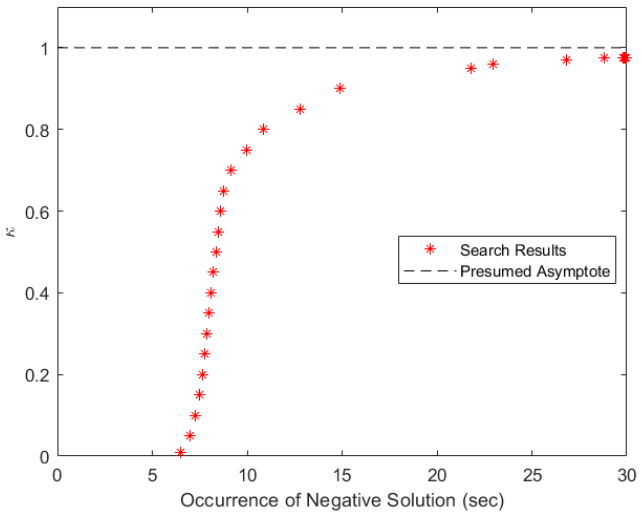

Figure 10: Optimizing Kappa

As κ approaches a value of 1, the time at which a negative solution occurs approaches the end simulation time. Based on the trend of the collected data, there is a presumed asymptote at a value of $\kappa = 1$. The optimum value for κ is 0.9758 which results in the negative solution at $t = 29.988$ and minimizes oscillation. Annimations of two-dimensional solutions are available in the footnotes.

Conclusion

We have successfully modeled thermal dissipation and waves using partial differential equations in both one and two dimensions. We conclude that acrylic yields the most uniform temperature distribution and diamond yields the least uniform temperature distribution after two seconds for the given region and boundary conditions. We also found that a minimum damping coefficient $\kappa = 0.9758$ yields non-negative solutions for the one-dimensional system analyzed. We also calculated transient solutions to the two dimensional wave equation representing drum heads defined by varying norms. Lastly, we created galleries of transient solutions for the two dimensional heat equation and the two-dimensional wave equation available on YouTube [2] [3].

[2] https://youtu.be/utg9CNeSCK0
[3] https://youtu.be/BQBj8Z1xO1E

References

[1] Engineers Edge. *Thermal Conductivity of Gasses Chart*. 2000. URL: https://www.engineersedge.com/heat_transfer/thermal-conductivity-gases.htm.

[2] Engineers Edge. *Thermal Properties of Metals, Conductivity, Thermal Expansion, Specific Heat*. 2000. URL: https://www.engineersedge.com/properties_of_metals.htm.

[3] Engineers Edge. *Thermal Properties of Non-Metals*. 2000. URL: https://www.engineersedge.com/heat_transfer/thermal_properties_of_nonmetals_13967.htm.

[4] Allen Holder and Joseph Eichholz. *An Introduction To Computational Science*. International Series in Operations Research & Management Science. Cham, Switzerland: Springer, 2019.

[5] Randall J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations*. Other Titles in Applied Mathematics. SIAM, 2007. Chap. 12.6 von Neumann Analysis.

Appendix A - Summary

In our paper, we used linear partial differential equation to model waves and heat transfer in one and two dimensions. For the heat equation, temperature as a function of position and time needed to satisfy the heat equation

$$\frac{\partial u}{\partial t} = \alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \tag{A.1}$$

given a scalar constant α (a material property), an initial temperature distribution $u(x, y, 0)$ and Dirichlet or Neumann boundary conditions. For the wave equation, displacement as a function of position and time needed to satisfy the wave equation

$$\frac{\partial^2 u}{\partial t^2} = \frac{H}{\rho} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \tag{A.2}$$

given scalars H and ρ corresponding to tension and density, an initial position $u(x, y, 0)$, an initial velocity $\frac{\partial u}{\partial t}(x, y, 0)$, and Dirichlet boundary conditions. (A.1) and (A.2) were both developed by applying first principles to a differential volume and taking the resulting limits as the volume approach zero.

Once the models were established, we applied finite difference methods to compute transient solutions. Finite difference methods are easily implemented on simple boundaries and have well known convergence characteristics, making them ideal for the problems we studied. For more complex geometries, galerkin, spectral, and finite element methods can be used at the expense of these well-known convergence characteristics. These models, along with finite difference methods, helped us to identify acrylic as a material that yields nearly uniform temperature distributions for a simple boundary containing both constant temperature and outward heat flux boundary conditions. Conversely, type IIA diamond yielded the least uniform temperature distribution. The transient solution to these and 15 other materials are available on YouTube [4]. We also determined the smallest damping value κ for a one dimensional wave with initial conditions

$$u(x, 0) = 0 \text{ and } \frac{\mathrm{d}u}{\mathrm{d}t}(x, 0) = e^{-10(x-L/2)^2} - e^{-10(L/2)^2}$$

for which displacement remains non-negative to be 0.9758. Lastly, we created a suite of solutions to the two dimensional wave equation that represent the striking of a drum head with boundaries defined by different p-norms available on YouTube [5].

[4] https://youtu.be/utg9CNeSCK0
[5] https://youtu.be/BQBj8Z1xO1E

Appendix B - 2 Dimensional Wave Equation Stability

The stability constraint for the 2 dimensional wave equation discretization

$$u_{i,j}^{k+1} = 2u_{i,j}^{k} - u_{i,j}^{k-1} + r\left(u_{i+1,j}^{k} + u_{i-1,j}^{k} + u_{i,j+1}^{k} + u_{i,j-1}^{k} - 4u_{i,j}^{k}\right) \tag{B.1}$$

where

$$r = \left(\frac{H\Delta t^2}{\rho h^2}\right) \tag{B.2}$$

and $h = \Delta x = \Delta y$ is determined via Von-Neumann analysis [5] by substituting $u(x, y, t_n) = g(\xi)^n e^{I\zeta h^2 ij}$ (where $I = \sqrt{-1}$) into equation B.1, yielding

$$\begin{aligned} g(\xi)^{k+1} e^{I\xi h^2 ij} &= 2g(\xi)^k e^{I\xi h^2 ij} - g(\xi)^{k-1} e^{I\xi h^2 ij} + ... \\ &\quad rg(\xi)^k \left(e^{I\xi h^2 (i+1)j} + e^{I\xi h^2 (i-1)j} + e^{I\xi h^2 i(j+1)} + e^{I\xi h^2 i(j-1)} - 4e^{I\xi h^2 ij}\right). \end{aligned}$$

Dividing both sides by $g(\xi)^{k-1} e^{Ih^2 ij}$ and simplifying yields

$$g(\xi)^2 = 2g(\xi) - 1 + rg(\xi)\left(2e^{I\xi h^2} + 2e^{-I\xi h^2} - 4\right).$$

Applying Euler's identity yields the further simplification

$$0 = -1 + rg(\xi)\left(4\cos(\xi h^2) - 4 + \frac{2}{r}\right) - g(\xi)^2$$

after some rearrangement. Solving for $g(\xi)$ when $\xi h^2 = \pi$, the worst case, yields

$$g(\xi) = 1 - 4r \pm 2\sqrt{4r^2 - 2r}.$$

Solving for $|g(\xi)| = 0$ yields $r = 1/2$, the maximum value of r for which the method is stable. Substituting this into equation B.2 and rearranging yields

$$\Delta t \leq \sqrt{\frac{1}{2}\frac{\rho h^2}{H}}. \tag{B.3}$$

For the case $H/\rho = 2$, B.3 reduces to

$$\Delta t \leq \frac{1}{2}h$$