Investigation of Heat Conduction Phenomena

Lana Janson

April 10, 2025

1 Abstract

My project focuses on the numerical computation of temperature distributions in one-dimensional (1D) and two-dimensional (2D) structures using computational methods. The goal is to model heat conduction within a rod and a plate, respectively, by solving the heat equation numerically. The investigation involves discretizing the heat equation and implementing numerical methods such as NumPy's linalg.solve to solve the resulting system of equations. The temperature distributions are then visualized through graphical representations. My project aims to provide insights into the behavior of heat transfer phenomena in simple geometric configurations and demonstrate the application of computational techniques in solving thermal conduction problems.

2 Introduction

Heat conduction is the transfer of internal thermal energy by the collisions of microscopic particles and movement of electrons within a body. The microscopic particles in the heat conduction can be molecules, atoms, and electrons. Understanding heat conduction phenomena is crucial for designing efficient thermal systems, such as building heaters or even to effectively send ships into space.

Heat can only be transferred through three means: conduction, convection and radiation. Conduction is one of the primary mechanisms by which heat is transferred within materials. It occurs through direct physical contact between adjacent particles or molecules within a substance. When a temperature gradient exists, with one region of the material being warmer than another, heat energy is transferred from higher to lower temperature regions through molecular collisions. This process continues until thermal equilibrium is reached, resulting in a uniform distribution of temperature throughout the material. Conduction is highly dependent on the material properties such as thermal conductivity, which determines the rate at which heat is conducted through the substance.

In this project I am investigating heat conduction in one-dimensional (1D) and two-dimensional (2D) scenarios. I will focus on determining the temperature distribution along a rod and a plate, respectively. By analyzing the behavior of heat conduction in these geometries, I can gain insights into the underlying mechanisms and factors influencing thermal transport.

My motivation for this project stems from a combination of my interest in numerical solving of matrices and practical uses in future classes. Understanding heat conduction is crucial for various engineering and scientific applications, which could help me in my classes to come (such as thermal physics that I will be taking next semester). Graphing accurate numerical modeling can inform me of the design of thermal systems, and help me understand the relevance of the system of equations solved. The project also aligns with my interest in computational methods and numerical analysis. I enjoyed completing Homework 4, where we were using matrices to compute a system of equations using Kirchhoff's laws.

3 Methods

3.1 1D Heat Conduction

In the 1D scenario, I modeled heat conduction along a rod using a finite difference method. Through using this method, I can discretized the rod into a series of nodes, with spatial steps determined by the desired resolution. The heat conduction equation, which describes how temperature changes over time, was then discretized using central differencing to approximate the second derivative with respect to space:

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2} \tag{1}$$

where T_i represents the temperature at node i, and Δx is the spatial step size.

The discretized equation was then solved numerically using an iterative approach. At each time step, the temperature at each node was updated based on the temperature of neighboring nodes, as dictated by the finite difference scheme. This process was repeated until the system reached a steady state or a specified time limit.

The iterative update process in the 1D scenario can be expressed as follows:

$$T_i^{(n+1)} = T_i^{(n)} + \alpha \frac{T_{i+1}^{(n)} - 2T_i^{(n)} + T_{i-1}^{(n)}}{\Delta x^2} \tag{2}$$

where $T_i^{(n)}$ represents the temperature at node i at the n-th time step, and α is the thermal diffusivity coefficient. This update equation was applied iteratively until the temperature distribution converged to a steady state or the specified time limit was reached. This set of solved equations were found by: *Finite Difference Solution of the Heat Equation Adam Powell 22.091 March 13–15, 2002.*

3.2 2D Heat Conduction

In the 2D scenario, heat conduction across a plate was modeled using a similar finite difference method. The plate was discretized into a grid of nodes, with spatial steps determined by the desired resolution in both the x and y directions. The heat conduction equation in 2D, which describes how temperature changes over time, was discretized using central differencing to approximate the Laplacian:

$$\nabla^2 T = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} \tag{3}$$

where $T_{i,j}$ represents the temperature at node (i, j), and Δx and Δy are the spatial step sizes in the x and y directions, respectively.

The discretized equation was then solved numerically using an iterative approach similar to the 1D scenario. At each time step, the temperature at each node was updated based on the temperature of neighboring nodes, as dictated by the finite difference scheme. This process continued until the system reached a steady state or a specified time limit.

The iterative update process can be expressed as follows:

$$T_{i,j}^{(n+1)} = T_{i,j}^{(n)} + \alpha \left(\frac{T_{i+1,j}^{(n)} - 2T_{i,j}^{(n)} + T_{i-1,j}^{(n)}}{\Delta x^2} + \frac{T_{i,j+1}^{(n)} - 2T_{i,j}^{(n)} + T_{i,j-1}^{(n)}}{\Delta y^2} \right) \tag{4}$$

where $T_{i,j}^{(n)}$ represents the temperature at node (i, j) at the n-th time step, and α is the thermal diffusivity coefficient. This update equation was applied iteratively until the temperature distribution converged to a steady state or the specified time limit was reached. Similarly to my 1D scenario, I used *Adam Powel's Finite Difference Solution of the Heat Equation*, to find the solved equations to use in my code.

3.3 Description of the code

To investigate the heat conduction in both scenarios, I used numerical solutions supplied by Python packages such as NumPy. I implemented the discretized equations in code, and then ran a simple chosen set of parameters, such as thermal conductivity, boundary conditions, and time step size to observe the distribution of temperature over the two objects. After choosing my set of parameters, I constructed the original matrices, then solved the matrices using a time-stepping loop and the function np.linalg.solve which is found in NumPy. Lastly, I visualized the results using plots to gain insights into the temperature distribution across the rod and plate over time.

3.3.1 Python Code for 1D scenario

```python
import numpy as np
import matplotlib.pyplot as plt

#parameters
L = 1.0  #length of the rod
T_initial = 0.0  #initial temperature
T_boundary = 100.0  #boundary temperature
k = 1.0  #thermal conductivity
alpha = 0.01  #thermal diffusivity
t_max = 1.0  #maximum time
dt = 0.001  #time step
dx = 0.1  #spatial step

#grid parameters
Nx = int(L / dx) + 1
Nt = int(t_max / dt) + 1

#initialize temperature array
T = np.ones(Nx) * T_initial

#boundary conditions
T[0] = T_boundary  #left boundary
T[-1] = T_boundary  #right boundary

#construct original matrix
A = np.zeros((Nx, Nx))
for i in range(Nx):
    for j in range(Nx):
        if i == j:
            A[i, j] = 1 + 2 * alpha * dt / (dx ** 2)
        elif abs(i - j) == 1:
            A[i, j] = -alpha * dt / (dx ** 2)

#print original matrix
print("Original Matrix:")
print(A)

#time-stepping loop
for n in range(1, Nt):
    #solve for next time step using the original matrix A
    T = np.linalg.solve(A, T)

#print final temperature distribution
```

```python
print("\nFinal Temperature Distribution along the Rod:")
print(T)

# Plotting
x = np.linspace(0, L, Nx)

plt.figure(figsize=(8, 6))
plt.plot(x, T, 'b-', label='Temperature Distribution')
plt.title('Temperature Distribution along the Rod')
plt.xlabel('Position')
plt.ylabel('Temperature')
plt.legend()
plt.grid(True)
plt.show()
```

3.3.2 Python Code for 2D Scenario

```python
import numpy as np
import matplotlib.pyplot as plt

# Parameters
Lx = 1.0  # Length of the plate in the x-direction
Ly = 1.0  # Length of the plate in the y-direction
T_initial = 0.0  # Initial temperature
T_boundary = 100.0  # Boundary temperature
k = 1.0  # Thermal conductivity
alpha = 0.01  # Thermal diffusivity
t_max = 1.0  # Maximum time
dt = 0.001  # Time step
dx = 0.1  # Spatial step in the x-direction
dy = 0.1  # Spatial step in the y-direction

# Grid parameters
Nx = int(Lx / dx) + 1
Ny = int(Ly / dy) + 1
Nt = int(t_max / dt) + 1

# Initialize temperature array
T = np.ones((Nx, Ny)) * T_initial

# Boundary conditions
T[:, 0] = T_boundary  # Bottom boundary
T[:, -1] = T_boundary  # Top boundary
T[0, :] = T_boundary  # Left boundary
T[-1, :] = T_boundary  # Right boundary

# Construct original matrix
A = np.zeros((Nx * Ny, Nx * Ny))
for i in range(Nx):
    for j in range(Ny):
        idx = i * Ny + j
        A[idx, idx] = 1 + 4 * alpha * dt / (dx ** 2) + 4 * alpha * dt / (dy ** 2)
        if i > 0:
            A[idx, idx - Ny] = -alpha * dt / (dx ** 2)
```

```python
        if i < Nx - 1:
            A[idx, idx + Ny] = -alpha * dt / (dx ** 2)
        if j > 0:
            A[idx, idx - 1] = -alpha * dt / (dy ** 2)
        if j < Ny - 1:
            A[idx, idx + 1] = -alpha * dt / (dy ** 2)

# Print original matrix
print("Original Matrix:")
print(A)

# Time-stepping loop
for n in range(1, Nt):
    # Solve for next time step using the original matrix A
    T = np.linalg.solve(A, T.flatten()).reshape((Nx, Ny))

# Print final temperature distribution
print("\nFinal Temperature Distribution on the Plate:")
print(T)

# Plotting
x = np.linspace(0, Lx, Nx)
y = np.linspace(0, Ly, Ny)
X, Y = np.meshgrid(x, y)

plt.figure(figsize=(8, 6))
plt.contourf(X, Y, T, cmap='hot')
plt.colorbar(label='Temperature')
plt.title('Temperature Distribution on the Plate')
plt.xlabel('x')
plt.ylabel('y')
plt.grid(True)
plt.show()
```

4 Results

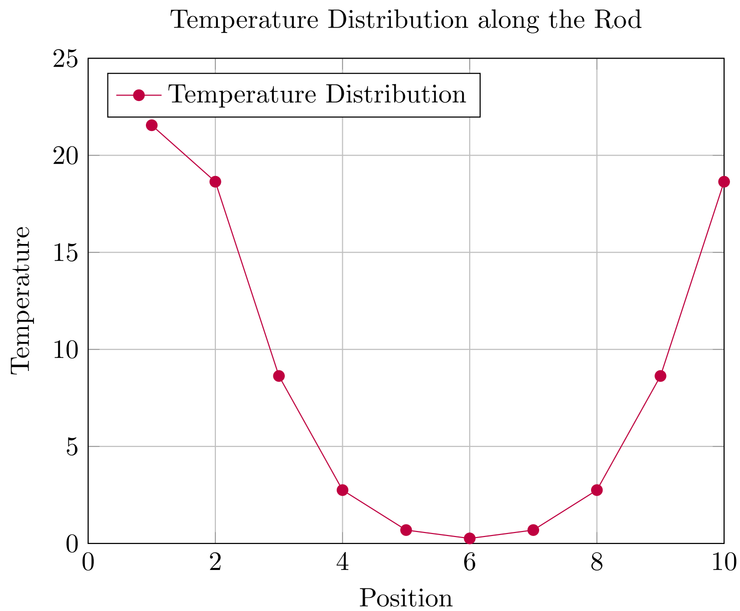
4.1 1D Heat Conduction

The original matrix obtained for the one-dimensional scenario is given by:

$$\begin{bmatrix} 1.002 & -0.001 & 0 & 0 & \cdots & 0 \\ -0.001 & 1.002 & -0.001 & 0 & \cdots & 0 \\ 0 & -0.001 & 1.002 & -0.001 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & -0.001 & 1.002 & -0.001 \\ 0 & \cdots & 0 & 0 & -0.001 & 1.002 \\ 0 & \cdots & 0 & 0 & 0 & -0.001 \end{bmatrix}$$

and the final temperature distribution along the rod is:

$$[21.55, 18.64, 8.63, 2.75, 0.69, 0.26, 0.69, 2.75, 8.63, 18.64, 21.55]$$



4.2 2D Heat Conduction

Original Matrix:

$$\begin{bmatrix} 1.008 & -0.001 & 0 & 0 & \cdots & 0 \\ -0.001 & 1.008 & -0.001 & 0 & \cdots & 0 \\ 0 & -0.001 & 1.008 & -0.001 & \cdots & 0 \\ 0 & 0 & -0.001 & 1.008 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & -0.001 & 1.008 & -0.001 \\ 0 & \cdots & 0 & 0 & -0.001 & 1.008 \end{bmatrix}$$

Final Temperature Distribution on the Plate:

$$\begin{bmatrix} 0.33299684 & 0.43928138 & 0.43057754 & 0.41122183 & \cdots & 0.33299684 \\ 0.43928138 & 0.51006181 & 0.43192121 & 0.37393014 & \cdots & 0.43928138 \\ 0.43057754 & 0.43192121 & 0.28997317 & 0.20129062 & \cdots & 0.43057754 \\ 0.41122183 & 0.37393014 & 0.20129062 & 0.09852516 & \cdots & 0.41122183 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0.33299684 & 0.43928138 & 0.43057754 & 0.41122183 & \cdots & 0.33299684 \end{bmatrix}$$

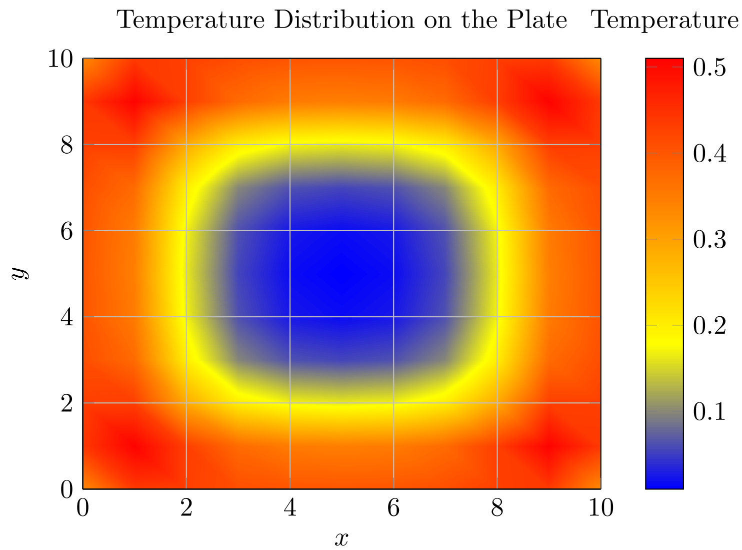


Figure 1: Temperature Distribution on the Plate

5 Analysis/Discussion

5.1 1D Heat Conduction

The 1D model represents the temperature distribution along a rod with specified chosen boundary conditions and thermal properties. The temperature is calculated at discrete points along the rod, with the initial and boundary temperatures defined. The final temperature distribution along the rod shows how heat is transferred from the boundaries towards the center. The temperature decreases gradually from the boundaries towards the center, following the expected behavior of heat diffusion in a one-dimensional system. This simple 1D model provides insights into how temperature changes over time within the rod. It can be used to study various scenarios by adjusting parameters such as thermal conductivity, diffusivity, and boundary conditions.

5.2 2D Heat Conduction

The 2D model represents the temperature distribution on a plate with specified boundary conditions and thermal properties. The temperature is calculated at discrete points on the plate's surface. The final temperature distribution on the plate exhibits more complex patterns compared to the 1D case. Heat diffuses not only along the length of the plate but also across its surface, leading to variations in temperature across different regions. The 2D model allows for the study of spatial variations in temperature, which is crucial for applications involving heat transfer in two-dimensional systems. It can be used to analyze temperature gradients, identify hotspots, and optimize thermal management strategies.

6 Conclusion

My project investigated the temperature distribution in one-dimensional (1D) and two-dimensional (2D) systems using computational modeling. Through the implementation of numerical methods, I analyzed the heat transfer behavior in a rod and a plate under specified boundary conditions and thermal properties.

The results obtained from the simulations provide valuable insights into the dynamics of heat diffusion and temperature distribution in both 1D and 2D scenarios. In the 1D case, I observed a gradual decrease in temperature along the length of the rod, highlighting the fundamental principles of one-dimensional heat conduction. Meanwhile, in the 2D case, the temperature distribution on the plate exhibited more complex patterns, reflecting the influence of spatial variations on heat transfer.

Overall, this project helped me understand heat transfer phenomena and provided a foundation for further research in thermal analysis.

7 References

1. Newman, Mark. *Computational Physics*. CreateSpace, 2013.

2. "Scipy.Linalg.Solve#." *Scipy.Linalg.Solve - SciPy v1.13.0 Manual*, `docs.scipy.org/doc/scipy/reference/generated/scipy.linalg.solve.html`. Accessed 30 Apr. 2024.

3. Pietrak, Karol, & Tomasz S. Wiśniewski. "A review of models for effective thermal conductivity of composite materials." *Journal of Power Technologies* [Online], 95.1 (2015): 14–24. Web. 30 Apr. 2024.

4. Finite Difference Solution of the Heat Equation. Adam Powell. 22.091 March 13–15, 2002. Available at: `https://dspace.mit.edu/bitstream/handle/1721.1/35256/22-00JSpring-2002/NR/rdonlyres/Nuclear-Engine` `22-00JIntroduction-to-Modeling-and-SimulationSpring2002/55114EA2-9B81-4FD8-90D5-5F64F20/lecture_16.pdf`.