COE352 Final Project

Alex Witt (alw4364)

December 2020

Contents

1 Introduction

This project was created for COE 352 at UT Austin taught by Dr. Corey Trahan. The goal is to use a 1D Galerkin approximation to model the evolution of a transient heat flow problem. Code can be found here.

2 Background

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = f(x,t) \tag{1}$$

where

$$f(x,t) = (\pi^2 - 1)e^{-t} sin(\pi x)$$

2.1 1D Spatial Discretization

Apply the test function to develop the weak form:

$$\int_0^L v(x)\left(\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2}\right)\, dx = \int_0^L v(x) f(x,t)\, dx \tag{2}$$

Use integration by parts to simplify

$$-\int_0^L v(x)\left(\frac{\partial^2 u}{\partial x^2}\right)\, dx = \left[-v(x)\frac{\partial u}{\partial x}\right]_0^L + \int_0^L \frac{\partial v}{\partial x}\frac{\partial u}{\partial x}\, dx$$

Rearrange

$$\int_0^L v(x)\left(\frac{\partial u}{\partial t}\right)+\frac{\partial v}{\partial x}\frac{\partial u}{\partial x}\,dx=\int_0^L v(x)f(x,t)\,dx+\left[v(x)\frac{\partial u}{\partial x}\right]_0^L$$

Our test function will be defined as a weighted linear combination:

$$v=\sum_{i=1}^N \beta_i\phi_i(x)$$

$$\frac{\partial v}{\partial x}=\sum_{i=1}^N \beta_i\phi_i'(x)$$

Using the Galerkin method, we also approximate u with these same functions.

$$u=\sum_{j=1}^N \alpha_j\phi_j(x)$$

$$\frac{\partial u}{\partial x}=\sum_{j=1}^N \alpha_j\phi_j'(x)$$

Taking the left hand side of equation 2:

$$\int_0^L v(x)\left(\frac{\partial u}{\partial t}\right)+\frac{\partial v}{\partial x}\frac{\partial u}{\partial x}\,dx=\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)\left(\frac{\partial u}{\partial t}\right)+\sum_{i=1}^N \beta_i\phi_i'(x)\sum_{j=1}^N \alpha_j\phi_j'(x)\,dx \tag{3}$$

$$=\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)\left(\frac{\partial u}{\partial t}\right)\,dx+\int_0^L\sum_{i=1}^N \beta_i\phi_i'(x)\sum_{j=1}^N \alpha_j\phi_j'(x)\,dx$$

$$=\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)\left(\frac{\partial u}{\partial t}\right)\,dx+\sum_{i=1}^N\beta_i\int_0^L\sum_{j=1}^N \alpha_j\phi_i'(x)\phi_j'(x)\,dx$$

2.2 Time Discretization

2.2.1 Forward Euler

Apply a Forward Euler approximation to discretize time.

$$\int_0^L v(x)\left(\frac{\partial u}{\partial t}\right)\,dx=\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)\frac{u(t+\Delta t)-u(t)}{\Delta t}\,dx$$

$$=\frac{1}{\Delta t}\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)u(t+\Delta t)\,dx-\frac{1}{\Delta t}\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)u(t)\,dx$$

$$=\frac{1}{\Delta t}\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)\sum_{j=1}^N\alpha_j^{n+1}\phi_j(x)\,dx-\frac{1}{\Delta t}\int_0^L\sum_{i=1}^N \beta_i\phi_i(x)\sum_{j=1}^N\alpha_j\phi_j(x)\,dx$$

$$=\frac{1}{\Delta t}\sum_{i=1}^N\sum_{j=1}^N\int_0^L\phi_i(x)\phi_j(x)\,dx\,\alpha_j^{n+1}-\frac{1}{\Delta t}\sum_{i=1}^N\sum_{j=1}^N\int_0^L\phi_i(x)\phi_j(x)\,dx\,\alpha_j$$

Plugging this into equation 3

$$\frac{1}{\Delta t}\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^{n+1} - \frac{1}{\Delta t}\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j + \sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i'(x)\phi_j'(x)\ dx\ \alpha_j = \sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^{n+1} - \sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j + \Delta t\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i'(x)\phi_j'(x)\ dx\ \alpha_j = \Delta t\sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^{n+1} = \sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j - \Delta t\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i'(x)\phi_j'(x)\ dx\ \alpha_j + \Delta t\sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

$$M\alpha_j^{n+1} = M\alpha_j - \Delta t K\alpha_j + \Delta t\sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

2.2.2 Backward Euler

Apply a Forward Euler approximation to discretize time.

$$\begin{aligned}
\int_0^L v(x)\left(\frac{\partial u}{\partial t}\right)\ dx &= \int_0^L \sum_{i=1}^{N}\beta_i\phi_i(x)\frac{u(t)-u(t-\Delta t)}{\Delta t}\ dx \\
&= \frac{1}{\Delta t}\int_0^L \sum_{i=1}^{N}\beta_i\phi_i(x)u(t)\ dx - \frac{1}{\Delta t}\int_0^L \sum_{i=1}^{N}\beta_i\phi_i(x)u(t-\Delta t)\ dx \\
&= \frac{1}{\Delta t}\int_0^L \sum_{i=1}^{N}\beta_i\phi_i(x)\sum_{j=1}^{N}\alpha_j^n\phi_j(x)\ dx - \frac{1}{\Delta t}\int_0^L \sum_{i=1}^{N}\beta_i\phi_i(x)\sum_{j=1}^{N}\alpha_j^{n-1}\phi_j(x)\ dx \\
&= \frac{1}{\Delta t}\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^n - \frac{1}{\Delta t}\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^{n-1}
\end{aligned}$$

Plugging this into equation 3

$$\frac{1}{\Delta t}\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^n - \frac{1}{\Delta t}\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^{n-1} + \sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i'(x)\phi_j'(x)\ dx\ \alpha_j^n = \sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^n - \sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^{n-1} + \Delta t\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i'(x)\phi_j'(x)\ dx\ \alpha_j^n = \Delta t\sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^n = \sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i(x)\phi_j(x)\ dx\ \alpha_j^{n-1} - \Delta t\sum_{i=1}^{N}\sum_{j=1}^{N}\int_0^L \phi_i'(x)\phi_j'(x)\ dx\ \alpha_j^n + \Delta t\sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

$$M\alpha_j^n + \Delta t K\alpha_j^n = M\alpha_j^{n-1} + \Delta t\sum_{i=1}^{N}\int_0^L \phi_i(x)f(x,t)\ dx$$

3 Results

In this section, the results required for the project are described.

3.1 Forward Euler

When solving the problem with a Forward Euler time discretization with $\Delta t = \frac{1}{551}$, the results are not great; the solution becomes unstable near $t = 1.0$ as seen in Figure 1. The final temperature values can be plotted to show the numerical explosion in Figure 2.

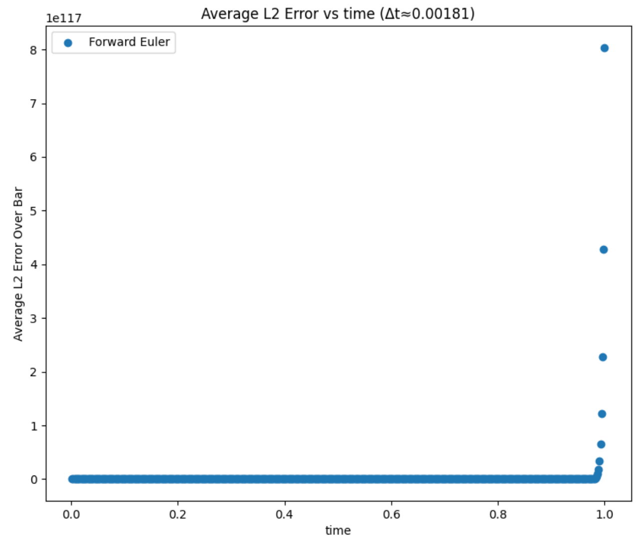

Figure 1: Averaged L2 error across nodes at each time.

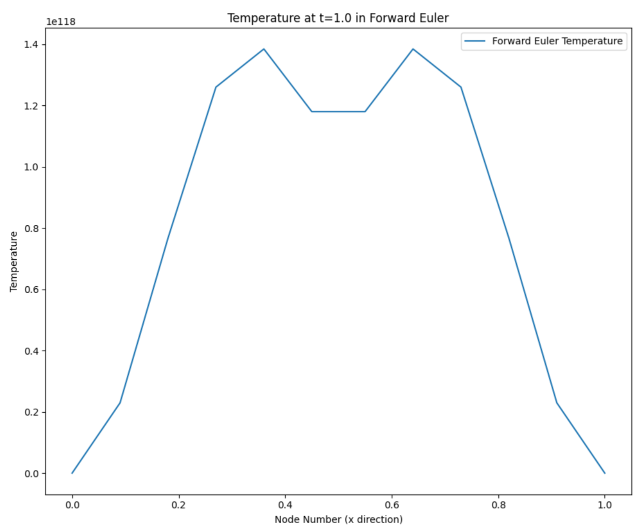

Figure 2: Temperature at t=1 in Forward Euler.

3.2 Forward Euler Time Step Adjustment

To combat the numerical instability, the time step can be made smaller. Through some trial and error, $\Delta t = \frac{1}{640}$ was found to suffice for stable modeling, though the solution is still poor.

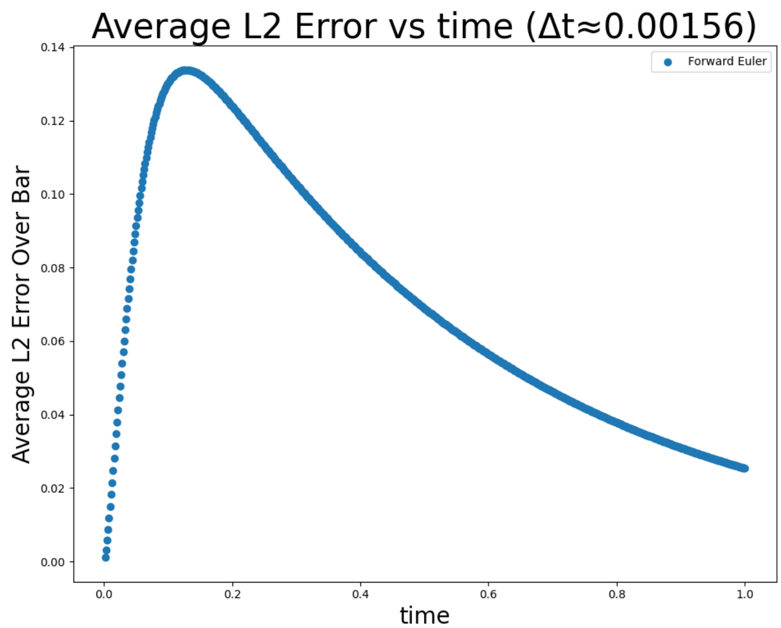

Figure 3: Forward Euler with $\Delta t = \frac{1}{640}$.

3.3 Forward Euler Number of Node Adjustment

When visualizing the evolution of the system as a surface when $\Delta t = \frac{1}{640}$ and number of nodes $N = 11$, we find the system does not evolve in unison with the analytical solution. The simulation sharply drops initially, unlike the analytical progression which is smoother.

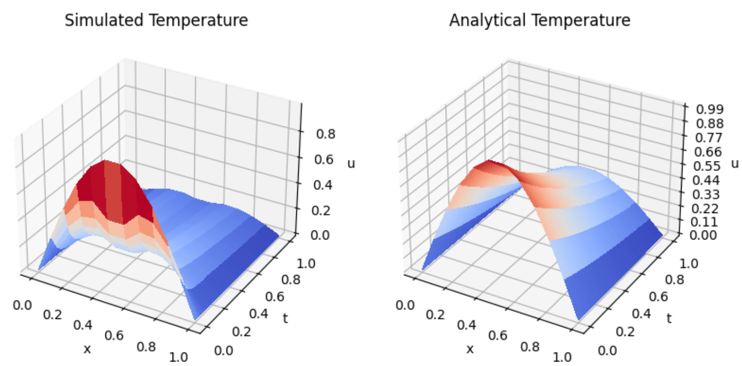

Figure 4: Forward Euler Surface vs Analytical Solution with $\Delta t = \frac{1}{640}$ and $N = 11$.

The spatial and temporal resolution of simulation can be decreased to $N = 10$ while maintaining $\Delta t = \frac{1}{640}$, and the results are much better aligned with the analytical solution. *Decreasing* the spatial resolution decreases the importance of temporal error.

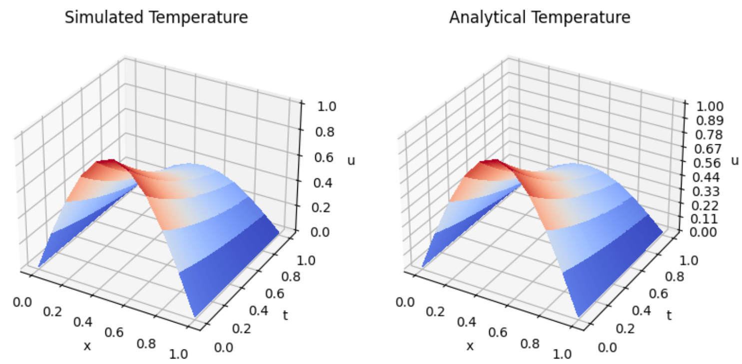

Figure 5: Forward Euler Surface vs Analytical Solution with $\Delta t = \frac{1}{640}$ and $N = 10$.

The averaged L2 error is also much smaller.

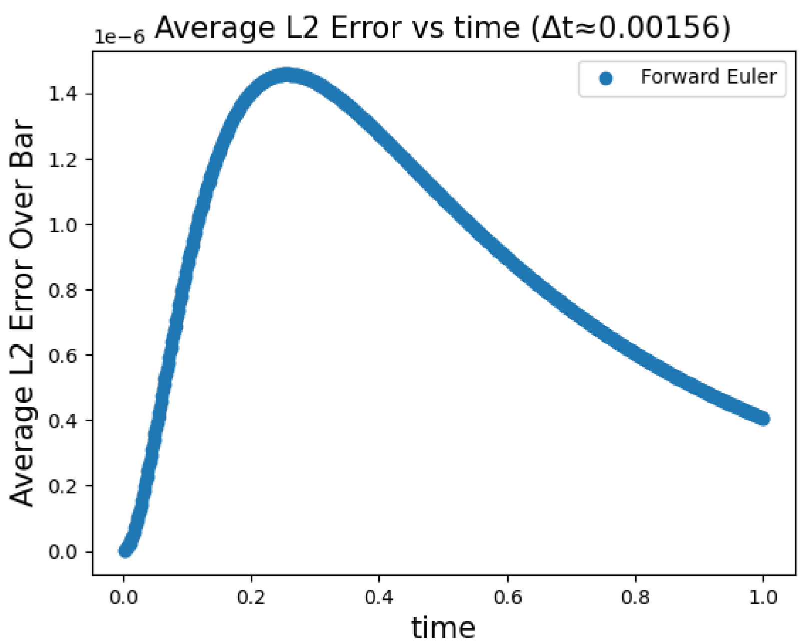

Figure 6: Forward Euler Surface vs Analytical Solution with $\Delta t = \frac{1}{640}$ and $N = 10$.

From this plot, we see the error is much smaller than before, by multiple orders of magnitude.

3.4 Backward Euler

Using the original $N = 11$ and $\Delta t = \frac{1}{551}$, we see the solution is stable, but tappers off rapidly compared to the analytical solution.

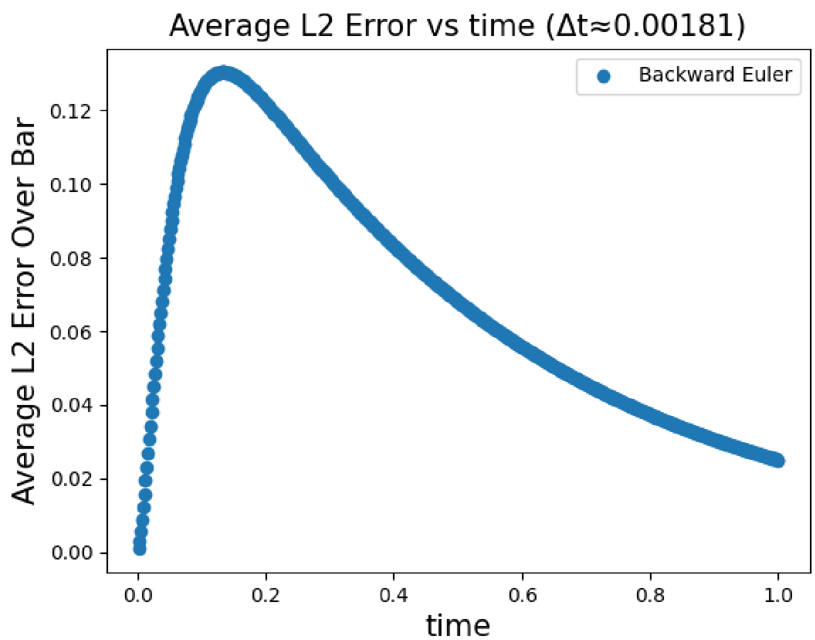

Figure 7: Backward Euler Averaged L2 error through time with $\Delta t = \frac{1}{551}$ and $N = 11$.

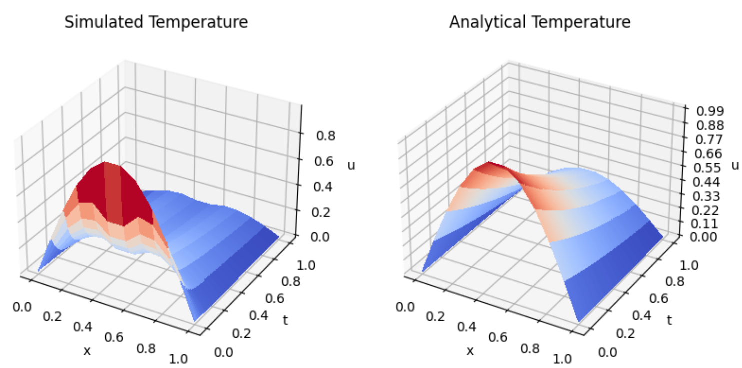

Figure 8: Backward Euler Surface vs Analytical Solution with $\Delta t = \frac{1}{551}$ and $N = 11$.

3.5 Backward Euler Spatial Resolution Adjustments

The Backward Euler solution can be drastically improved by setting the number of nodes to $N = 10$.

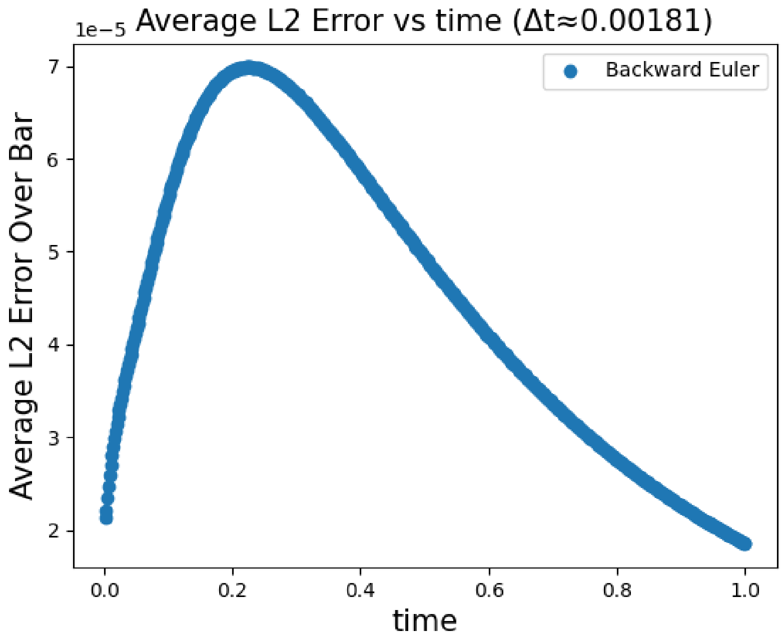

Figure 9: Backward Euler Averaged L2 error through time with $\Delta t = \frac{1}{551}$ and $N = 10$.

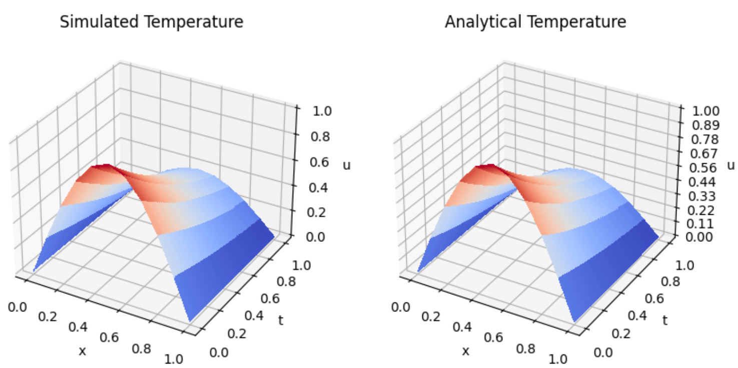

Figure 10: Backward Euler Surface vs Analytical Solution with $\Delta t = \frac{1}{551}$ and $N = 10$.

3.6 Backward Euler Stability

Fortunately, the backward Euler method is implicit and unconditionally stable, meaning any temporal step size works. However, if the timestep is too large, the accuracy of the solution degrades because we are not capturing enough information in the system evolution.

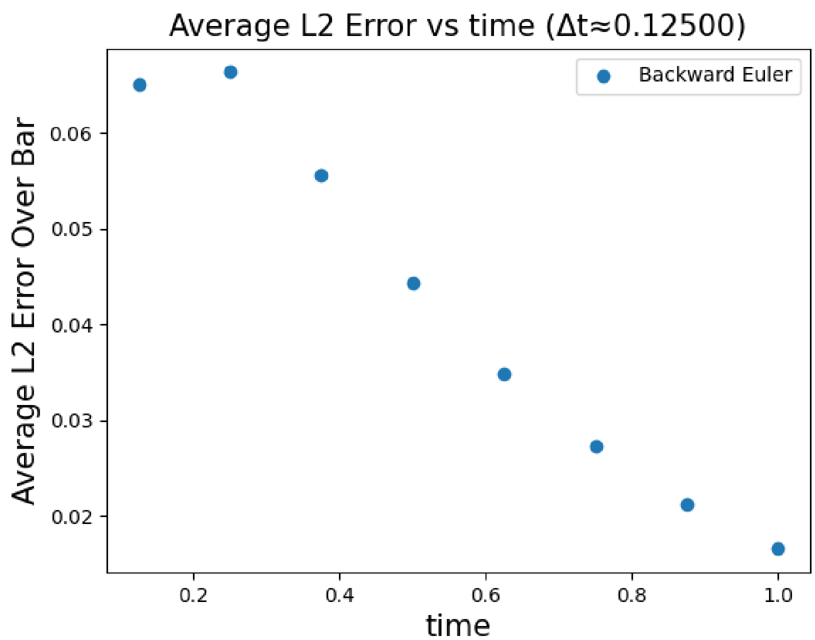

Figure 11: Backward Euler Averaged L2 error through time with $\Delta t = \frac{1}{8}$ and $N = 11$.

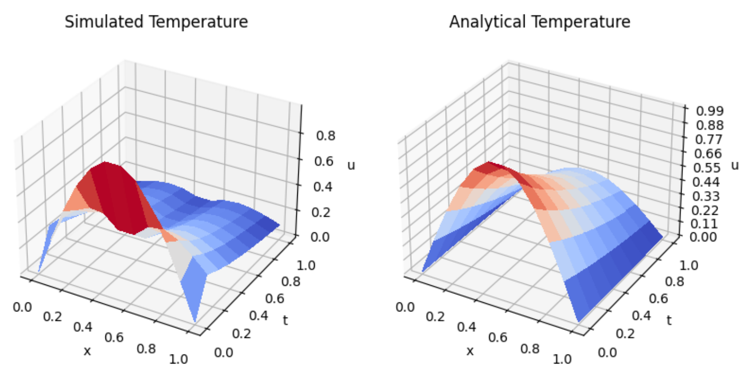

Figure 12: Backward Euler Surface vs Analytical Solution with $\Delta t = \frac{1}{8}$ and $N = 11$.

If we increase the timestep such that $\Delta t > dx$, we find solutions that are still stable but lagging behind the analytical solution (figures 13 14).

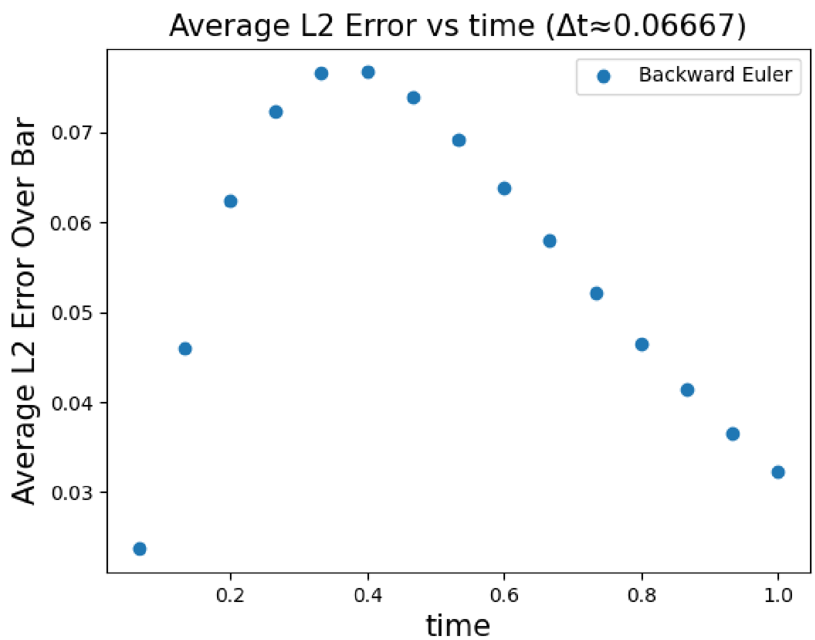

Figure 13: Backward Euler Averaged L2 error through time with $\Delta t = \frac{1}{15}$ and $N = 20$.

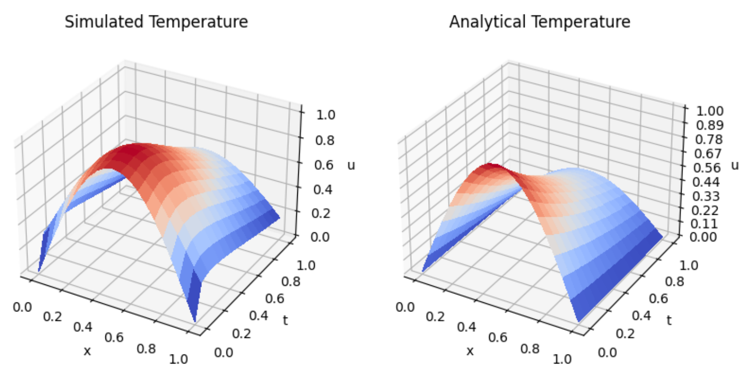

Figure 14: Backward Euler Surface vs Analytical Solution with $\Delta t = \frac{1}{15}$ and $N = 20$.