COE 352 Project 2

Anna Victoria Lavelle

GitHub Link: https://github.com/avlavelle/COE352_Proj2

Weak Form Derivation

The weak form is derived on the first notebook page. The following pages demonstrate how using forward Euler and backward Euler, the weak form equation was rewritten in matrix form which was essential for solving the problem and plotting it.

1. Weak form of $u_t - u_{xx} = f(x,t)$

initial + Dirichlet: $u(x,0) = \sin(\pi x)$

$u(0,t) = u(1,t) = 0$

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

solution: $u(x,t) = e^{-t}\sin(\pi x)$

$$u_t - u_{xx} - f(x,t) = 0$$

Multiply by a test function $\phi_i(x)$:

$$u_t \phi_i(x) - u_{xx}\phi_i(x) - f(x,t)\phi_i(x) = 0$$

Integrate over the domain:

$$\int_0^1 u_t \phi_i(x)\,dx - \int_0^1 u_{xx}\phi_i(x)\,dx - \int_0^1 f(x,t)\phi_i(x)\,dx = 0$$

Integrate by parts:

$f(x) = \phi_i(x)$

$g'(x) = u_{xx}$ $\quad g(x) = u_x$

$$\int_0^1 u_t \phi_i(x)\,dx - \left(\phi_i(x)u_x\Big]_0^1 - \int u_x \phi_i'(x)\,dx\right) - \int_0^1 f(x,t)\phi_i(x)\,dx = 0$$

weak form:

$$\boxed{R_i = \int_0^1 u_t \phi_i(x)\,dx + \int_0^1 \frac{\partial u}{\partial x}\frac{\partial \phi_i}{\partial x}\,dx - \int_0^1 f(x,t)\phi_i(x)\,dx}$$

Approximating u_t w/ Forward Euler:

$$R_i = \int_0^1 \left(\frac{u(t+\Delta t) - u(t)}{\Delta t} \right) \phi_i(x)\,dx + \int_0^1 \frac{\partial u}{\partial x} \frac{\partial \phi_i}{\partial x}\,dx - \int_0^1 f(x,t)\phi_i(x)\,dx$$

Galerkin expansion:

n = time-step index

j = spatial index

$$R_i = \frac{1}{\Delta t} \int_0^1 \sum_{j=1}^{n} u_j^{n+1} \phi_j \phi_i\,dx - \frac{1}{\Delta t} \int_0^1 \sum_{j=1}^{n} u_j^{n} \phi_j \phi_i\,dx +$$

$$\int_0^1 \sum_{j=1}^{n} u_j^n \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x}\,dx - \int_0^1 f(x,t)\phi_i(x)\,dx$$

$$R_i = \frac{1}{\Delta t} \overbrace{\sum_{j=1}^{n} u_j^{n+1}}^{\vec{u}^{n+1}} \overbrace{\int_0^1 \phi_j \phi_i\,dx}^{M} - \frac{1}{\Delta t} \overbrace{\sum_{j=1}^{n} u_j^{n}}^{\vec{u}_n} \overbrace{\int_0^1 \phi_j \phi_i\,dx}^{M} +$$

$$\underbrace{\sum_{j=1}^{n} u_j^n}_{\vec{u}^n} \underbrace{\int_0^1 \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x}\,dx}_{K} - \underbrace{\int_0^1 f(x,t)\phi_i(x)\,dx}_{\vec{F}^n}$$

Matrix Form:

$$\frac{1}{\Delta t} M\vec{u}^{n+1} - \frac{1}{\Delta t} M\vec{u}^{n} + K\vec{u}^{n} - \vec{F}^n = 0$$

$$\vec{u}^{n+1} = \vec{u}^n - \Delta t M^{-1} K \vec{u}^n + \Delta t M^{-1} \vec{F}^n$$

$$= \left[I - \Delta t M^{-1} K \right] \vec{u}^n + \Delta t M^{-1} \vec{F}^n$$

With Backward Euler:

$$R_i = \int_0^1 \left(\frac{u(t+\Delta t) - u(t)}{\Delta t} \right) \phi_i(x)\,dx + \underbrace{\int_0^1 \frac{\partial u(t+\Delta t)}{\partial x} \frac{\partial \phi_i}{\partial x}\,dx}_{\text{implicit}}$$

$$\underbrace{- \int_0^1 f(x, t+\Delta t)\,\phi_i(x)\,dx}_{\text{implicit}}$$

Galerkin expansion:

n= time-step index

j= spatial index

$$R_i = \frac{1}{\Delta t} \int_0^1 \sum_{j=1}^{n} u_j^{n+1} \phi_j \phi_i\,dx - \frac{1}{\Delta t} \int_0^1 \sum_{j=1}^{n} u_j^n \phi_j \phi_i\,dx +$$

$$\int_0^1 \sum_{j=1}^{n} u_j^{n+1} \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x}\,dx - \int_0^1 f(x, t+\Delta t)\,\phi_i(x)\,dx$$

$$R_i = \frac{1}{\Delta t} \underbrace{\sum_{j=1}^{n} u_j^{n+1}}_{\vec{u}^{n+1}} \underbrace{\int_0^1 \phi_j \phi_i\,dx}_{M} - \frac{1}{\Delta t} \underbrace{\sum_{j=1}^{n} u_j^n}_{\vec{u}^n} \underbrace{\int_0^1 \phi_j \phi_i\,dx}_{M}$$

$$+ \underbrace{\sum_{j=1}^{n} u_j^{n+1}}_{\vec{u}^{n+1}} \int_0^1 \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x}\,dx - \underbrace{\int_0^1 f(x, t+\Delta t)\,\phi_i(x)\,dx}_{\vec{F}^{n+1}}$$

Matrix Form:

$$\frac{1}{\Delta t} M \vec{u}^{n+1} - \frac{1}{\Delta t} M \vec{u}^n + K \vec{u}^{n+1} - \vec{F}^{n+1}$$

$$= \left[\frac{1}{\Delta t} M + K \right] \vec{u}^{n+1} = \frac{1}{\Delta t} M \vec{u}^n + \vec{F}^{n+1}$$

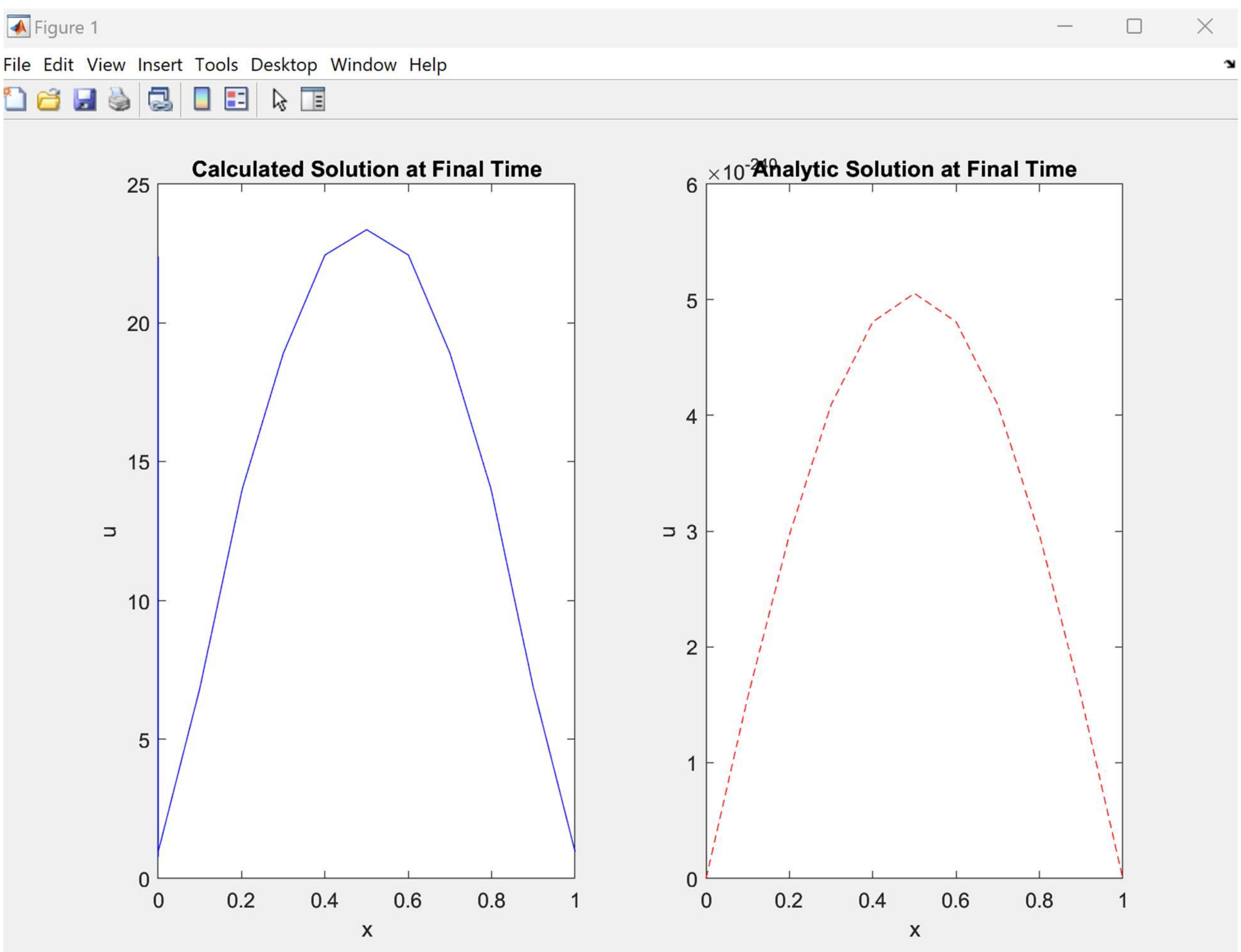
$$\vec{u}^{n+1} = \frac{1}{\Delta t} B^{-1} M \vec{u}^n + B^{-1} F^{n+1}$$

Forward Euler Time Derivative Discretization

The first step was generating the stiffness (K) and mass (M) matrices using steps from the 1D Poisson code since they are both independent of time. This required mapping from the x to the xi space, as well as the use of basis functions and solving with 2nd order Gaussian quadrature.

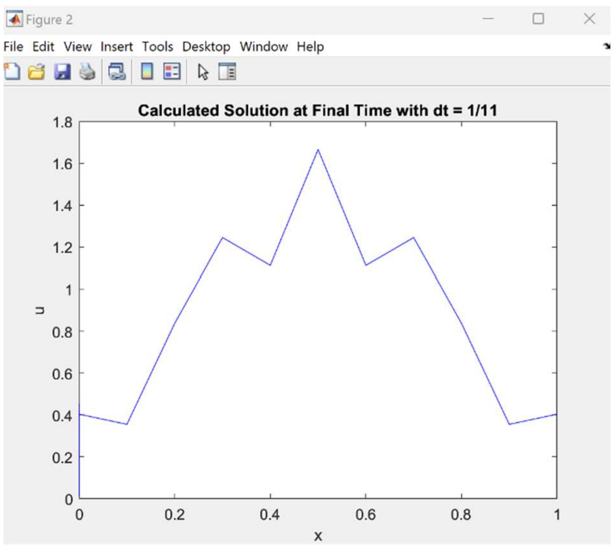
After assembling the K and M matrices, the time step, boundary conditions, and function were input. A loop using the time step helped build the time-dependent right hand side vectors. Once the vectors and matrices were all assembled, they were input to the matrix form of the weak form of the heat transfer equation using forward Euler.

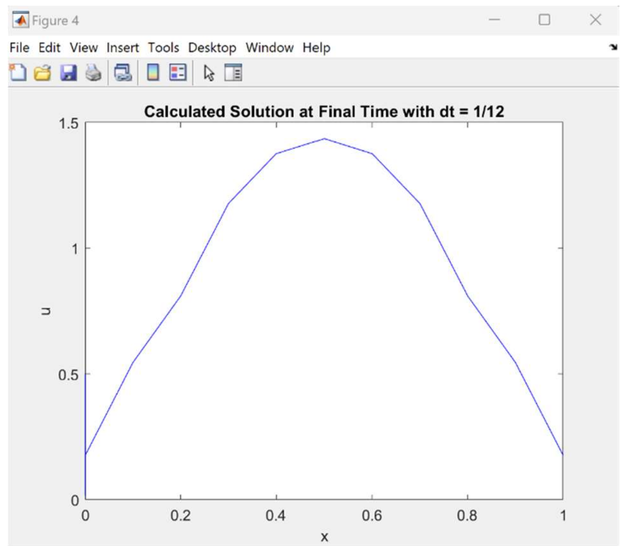
The figure below depicts both the calculated solution and the analytic solution at the final time. There is a large discrepancy between the u axis values. I think the difference in values comes from an incorrect implementation of the time step somewhere in the code.



Instability

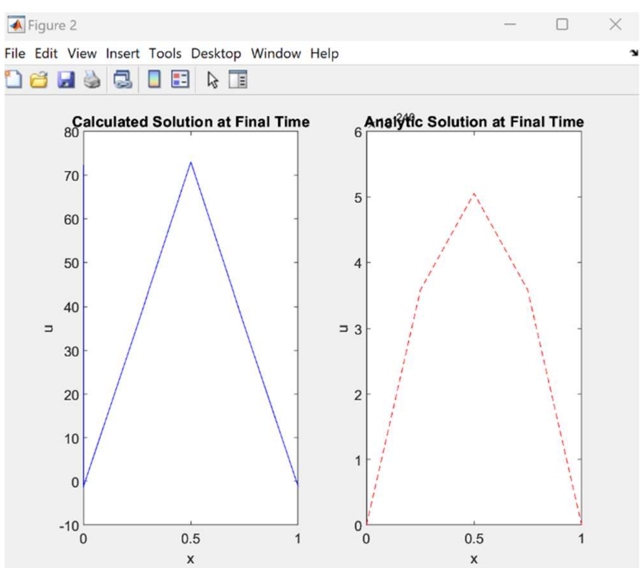
The figures below depict when instability begins by increasing the time step. At a time step of 1/12, the plot still looks stable, as shown on the right. However, at a time step of 1/11, instability begins, as shown on the left.





Decreasing N

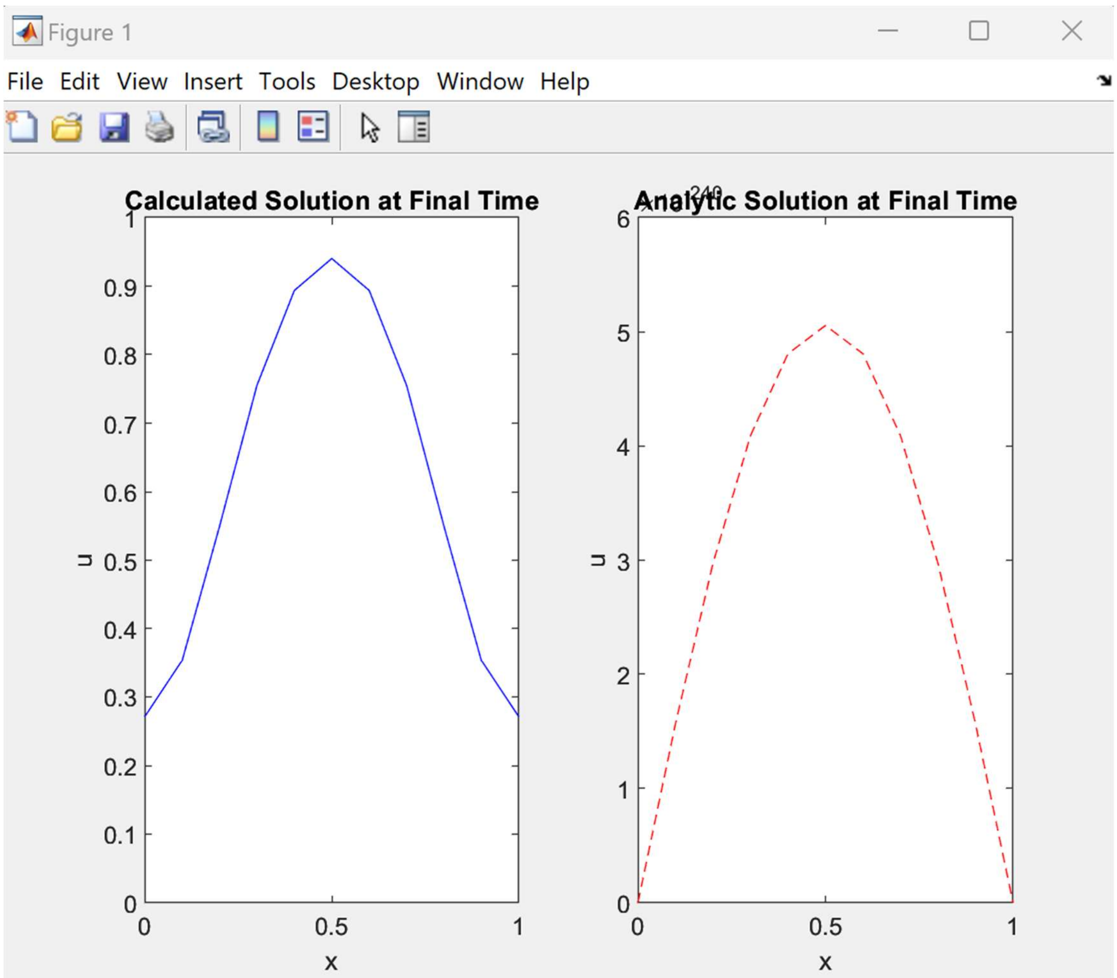
In decreasing N, we decrease the number of nodes making our solution less accurate because our grid is less refined, and it fails to capture finer details in the solution. Below is a depiction of the plots with 5 nodes. Both plots are less accurate.



Implicit Backward Euler

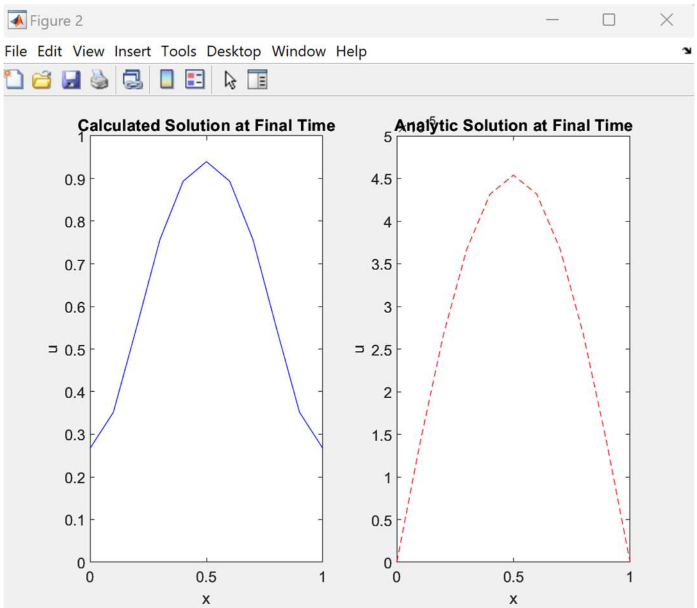
Developing the solution using backward Euler required most of the same steps as solving the problem with forward Euler, but the resulting elements were used to solve a different matrix form that provided a similar, but slightly different approximation. Below is the backward Euler solution plotted at the final time next to the analytic solution.

There is, once again, a large discrepancy between the u axis values. I think the difference in values comes from an incorrect implementation of the time step somewhere in the code.

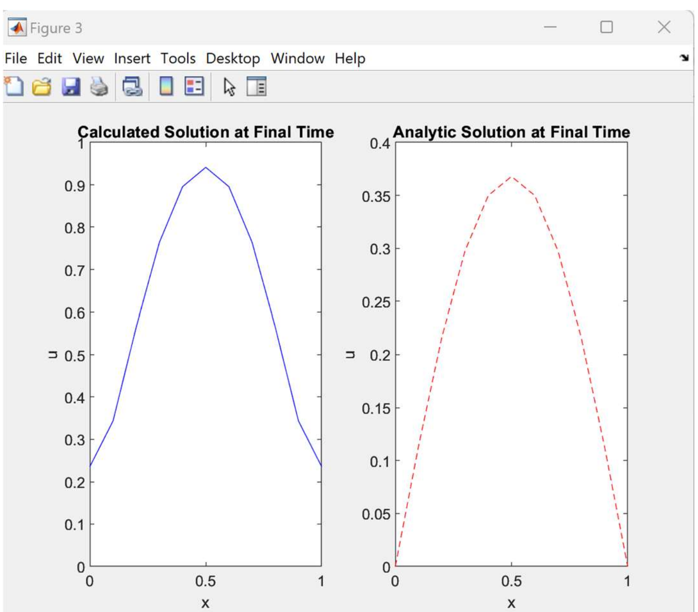


Time and Spatial Step Sizes

The graph below depicts when the time and spatial step sizes are both 1/10, so the spatial and time step sizes are equal.



The graph below depicts when the time step is 1 and the spatial step is 1/10, so the time step size is greater than the spatial step size.



Neither change in the step sizes appears to cause much of a difference in the plots. However, a time step that is equal to or greater than the spatial step would probably lead to instability. Without stability, our solution is not accurate or reliable. Instability would likely manifest itself in the plot through oscillations such as the ones depicted from changing the time step in the forward Euler solution.