Chad Henry

MECE 5397

Dr. Prosperetti & Dr. Amritkar

Diffusion Equation

**Abstract**

With the help of computers, complex simulations and discretizations have been able to help solve in minutes the very same problems that would have taken months to solve by hand. As an exercise in scientific computing, this project compares two different discretizations, the Explicit and Crank Nicolson and some of their respective features. The project conducted a brief grid convergence study on what the effect of the number of nodes and the degree of accuracy had on the time to compute. Furthermore, this project will compare the experimental graphs with the expected theoretical behavior. Included with the mathematical analysis is a history of the project via a git repository and some associated folders.

**Problem Statement**

Solve the 2D diffusion equation given the following domain and boundary conditions using the Crank Nicolson and Implicit Discretization methods.

*Rectangular Domain*

*Boundary Conditions*

*Given*

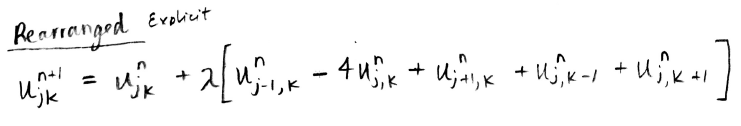
**Discretization**

Explicit

Crank Nicolson

**Description of the Numerical Method**

After rearranging and combining the discretized terms the deisred form is achieved. Notice how the point Uj,k is calculated by the surrounding terms on the left, right, top, and bottom. This serves as a template for all interior points.



The following figure shows how to handle a Neumann boundary condition by implementation of a ghost node at the boundary. By creating a fictitious point at Uj,N+1 then eliminating it through substitution, the desired form is upheld and the boundary is calculated with a higher degree of accuracy.

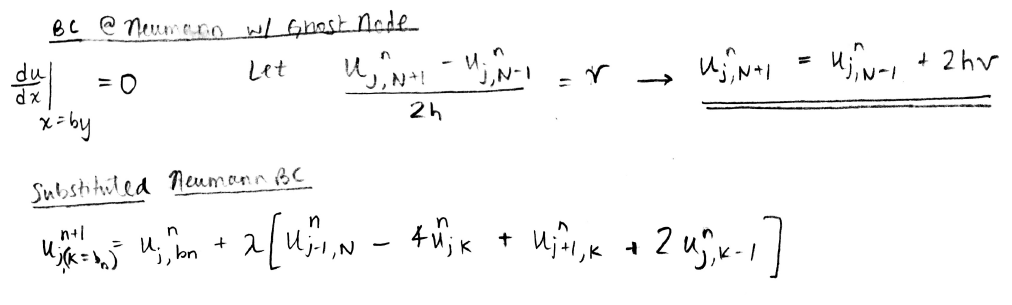


Figure 1: Ghost Node Boundary Calculation

**Technical specifications of the computer used**

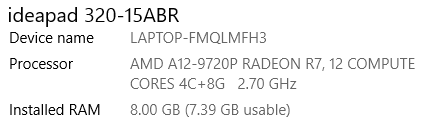


Figure 2: Computer Specifications

• **Results**

The following figures show the development of the solution as the number of nodes increases.

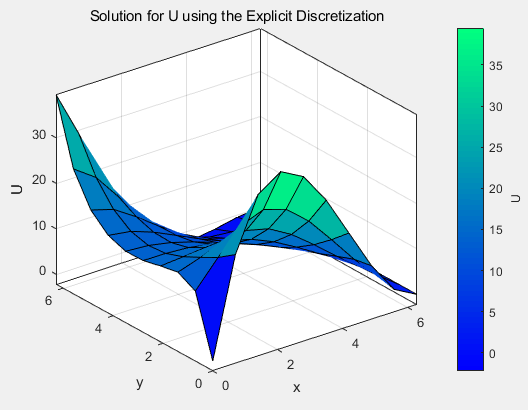


Figure 3: Explicit Discretization N=10

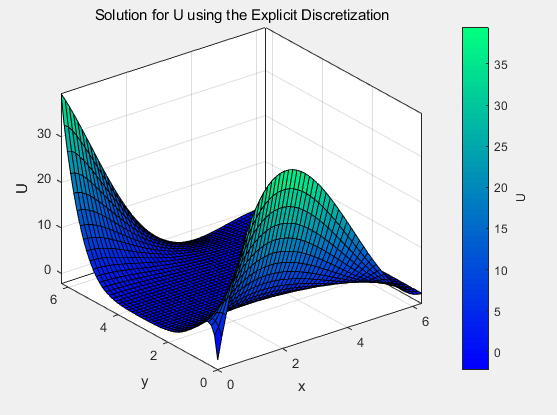


Figure 2: Explicit Discretization N=50

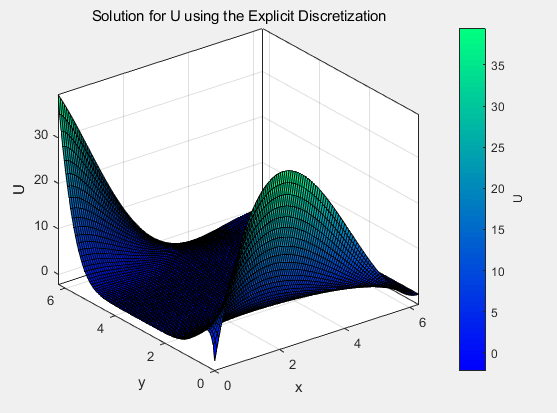


Figure 3: Explicit Discretization N=100

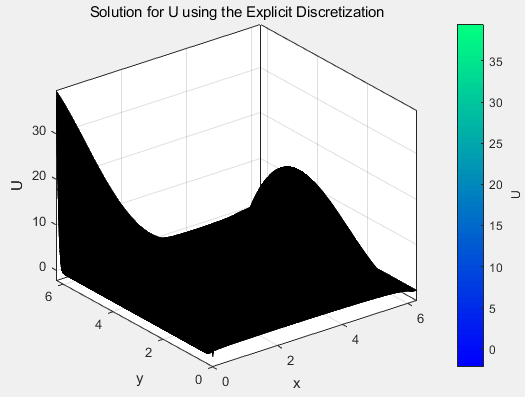


Figure 4: Explicit Discretization N=700

**Grid Convergence Study**

The time to compute increased non-linearly because the size of the data being calculated would not fit compactly within the memory hierarchy. As the size of the data exceeded the max storage in a level of memory, it had to be moved into the next category of memory where the access time is much higher. Furthermore, as the data progresses to the right, notice how in figure 3 the oscillations in the values dampen over time as the data converges to a steady state value.

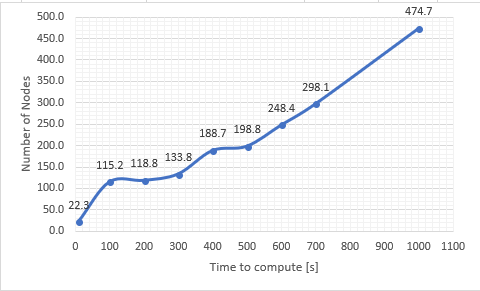
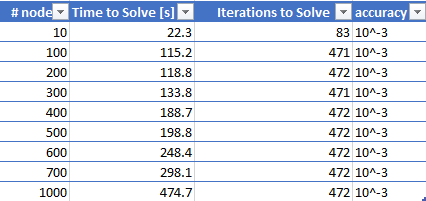


Figure 5: Relationship of time to compute vs. # nodes

Table 1: Solution Dependence on # of nodes:



The accuracy correlated very precisely with the required time to compute. An order of magnitude increase in computational accuracy yielded an order of magnitude decrease in both computing time and number of iterations as displayed in figure 4 and table 2.

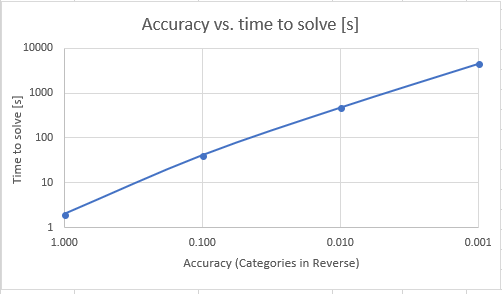


Figure 6: Desired Accuracy vs. Time to compute

Table 2: Accuracy vs. Time to Compute

