MARMARA
UNIVERSITY
FACULTY OF ENGINEERING
MECHANICAL ENGINEERING

ME4095
COMPUTATIONAL FLUID DYNAMICS

MIDTERM POJECT REPORT
2D – Heat Conduction Using ADI Method

NAME Oğuzalp ÇETİN

NUMBER 150418031

SUBMITTED TO Prof. Dr. Mehmet Zafer GÜL

INTRODUCTION

The Alternating Direction Implicit (ADI) method is a numerical technique widely used for solving partial differential equations (PDEs) in various engineering applications. This report is about the implementation of the ADI method in MATLAB App Designer for solving 2D steady heat conduction problem.

PROCEDURE

The procedure involves setting up the initial parameters, including geometry dimensions, material properties, and boundary conditions. The app employs a graphical user interface (GUI) to simplify user inputs. The ADI method discretizes the problem, and the resulting system of equations is solved iteratively until convergence is achieved or a maximum number of iterations is reached.

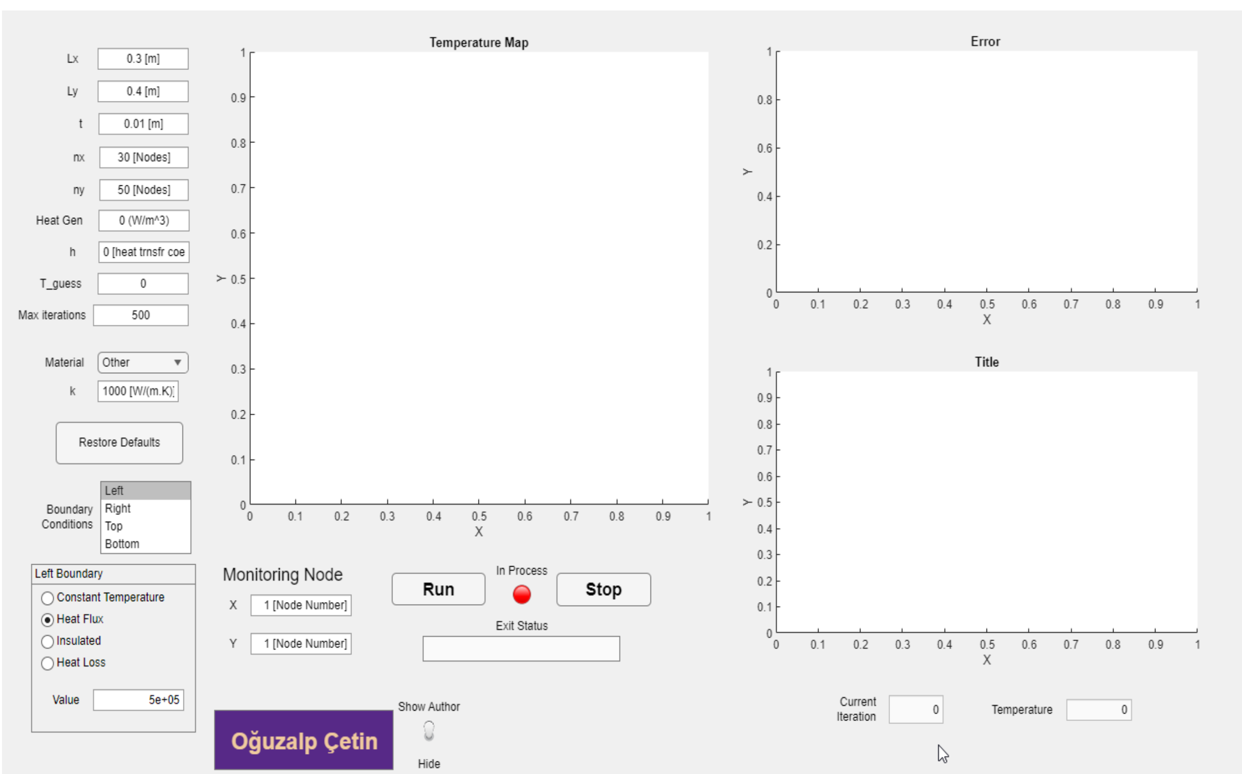

Figure 1: The GUI

To implement the ADI algorithm, TDMA is used. This algorithm is highly economical for tri-diagonal systems. It consists of a forward elimination and a back-substitution stage

$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right)=0$$

This can be written in discretised form as

$$a_P T_P = a_W T_W + a_E T_E + a_S T_S + a_N T_N$$

where

$$a_W = \frac{k}{\Delta x}A_w \quad a_E = \frac{k}{\Delta x}A_e \quad a_S = \frac{k}{\Delta y}A_s \quad a_N = \frac{k}{\Delta y}A_n$$

$$a_P = a_W + a_E + a_S + a_N$$

- *Forward elimination*
  - arrange system of equations in the form of (7.2):

    $$-\beta_j \phi_{j-1} + D_j \phi_j - \alpha_j \phi_{j+1} = C_j$$
  - calculate coefficients α_j, β_j, D_j and C_j
  - starting at $j = 2$ calculate A_j and C'_j using (7.6b–c):

    $$A_j = \alpha_j/(D_j - \beta_j A_{j-1})^{-1} \quad \text{and} \quad C'_j = (\beta_j C'_{j-1} + C_j)/(D_j - \beta_j A_{j-1})^{-1}$$
  - repeat for $j = 3$ to $j = n$
- *Back-substitution*
  - starting at $j = n$ obtain ϕ_n by evaluating (7.6a):

    $$\phi_j = A_j \phi_{j+1} + C'_j$$
  - repeat for $j = n - 1$ to $j = 2$ giving ϕ_{n-1} to ϕ_2 in reverse order

For two-dimensional problems, the TDMA must be applied iteratively in a line-by-line fashion.

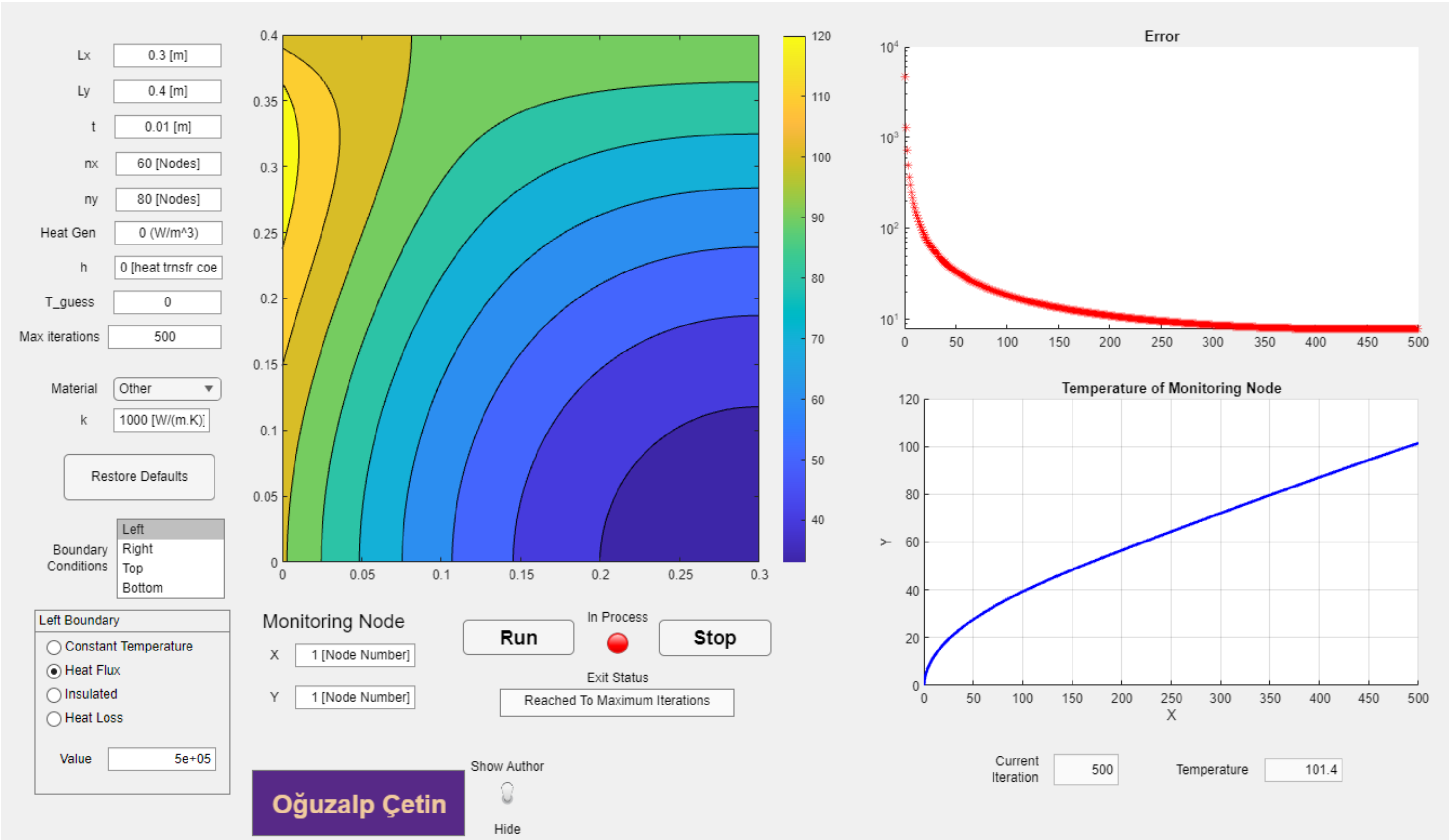

Figure 2: Solution of Example 7.2 using 60 x 80 nodes set-up

RESULTS

The ADI method is applied to a heat conduction problem, taken from Example 7.2 in the book "An Introduction to Computational Fluid Dynamics" by H. Versteeg. This example serves as a reference case for evaluating the ADI method's correct. The problem involves heat conduction within a defined spatial domain, subject to specific material properties, initial conditions, and boundary constraints.

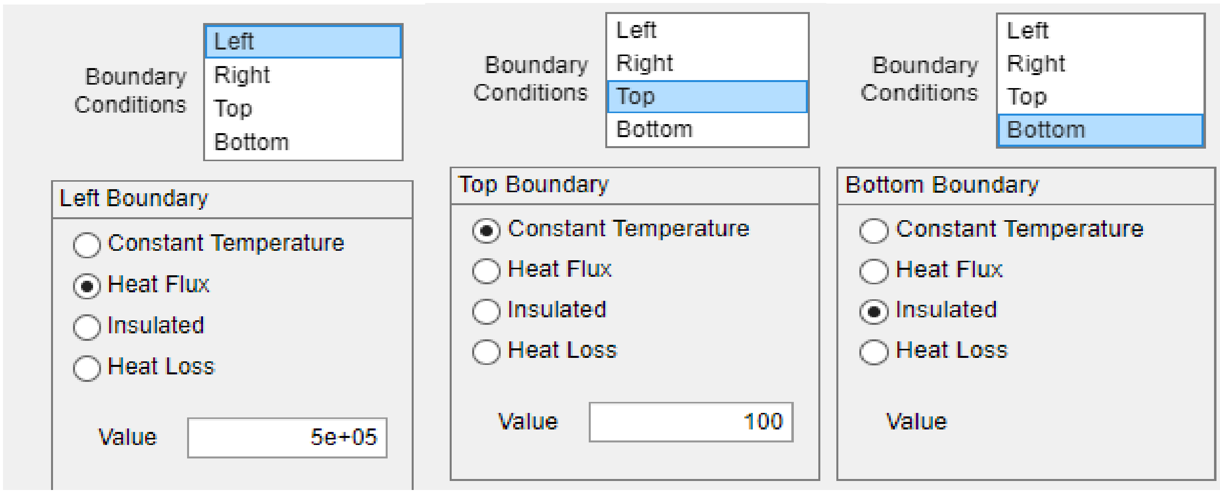

Figure 3: Boundary Conditions (Right Boundary is insulated)

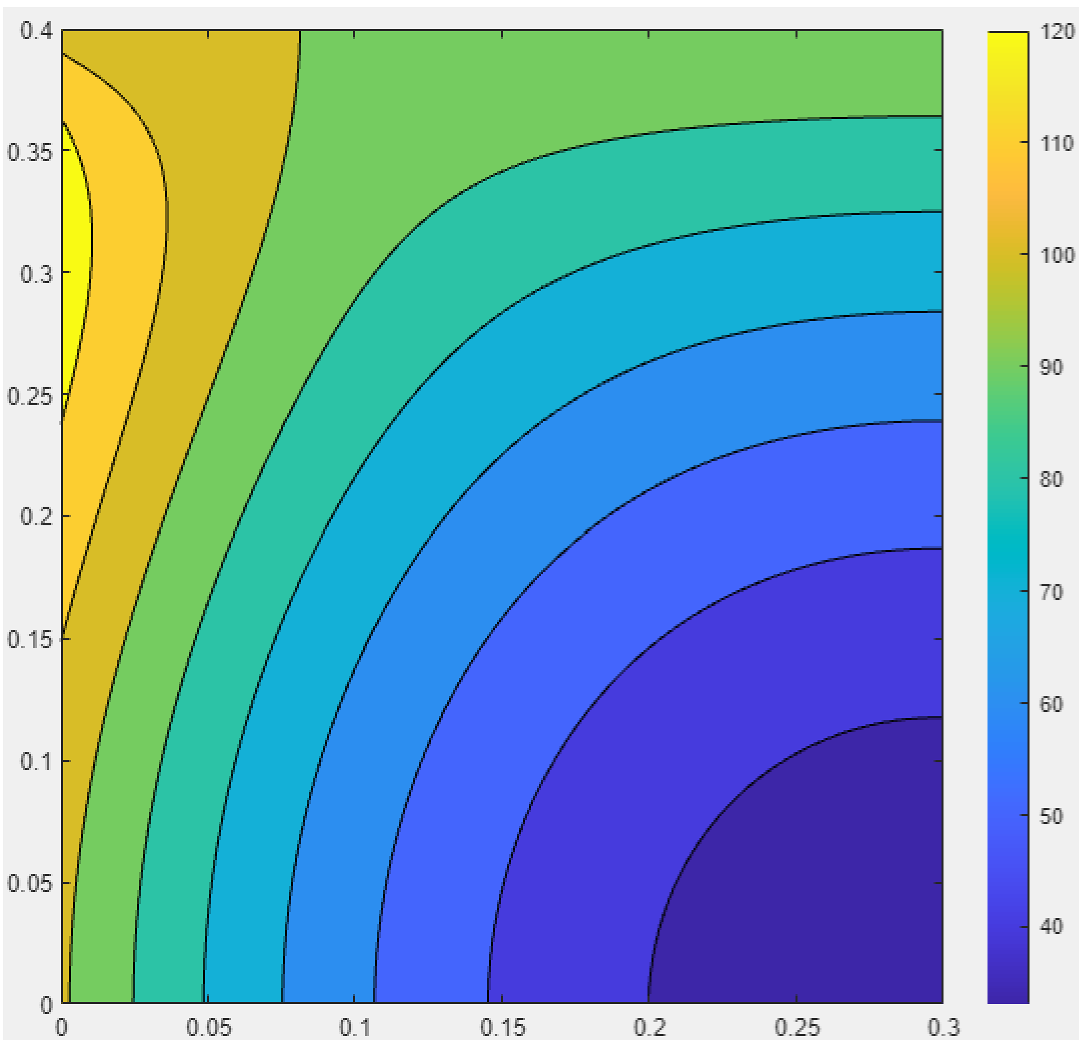

Figure 4: Temperature map after 500 iterations

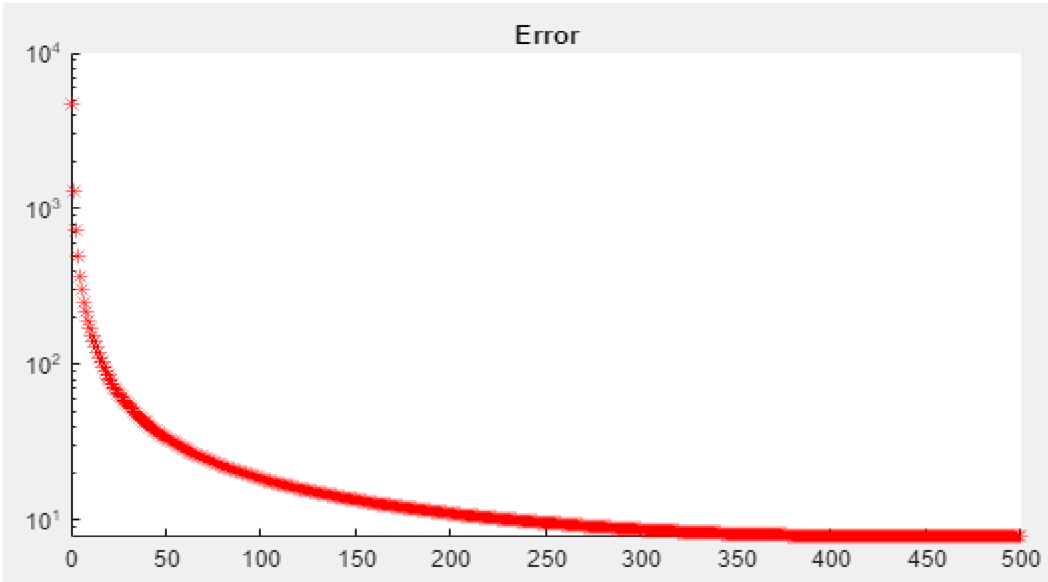

Figure 5: Plot of absolute error vs iteration no

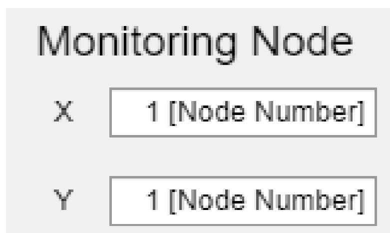

Figure 6: Temperature-Monitored Node

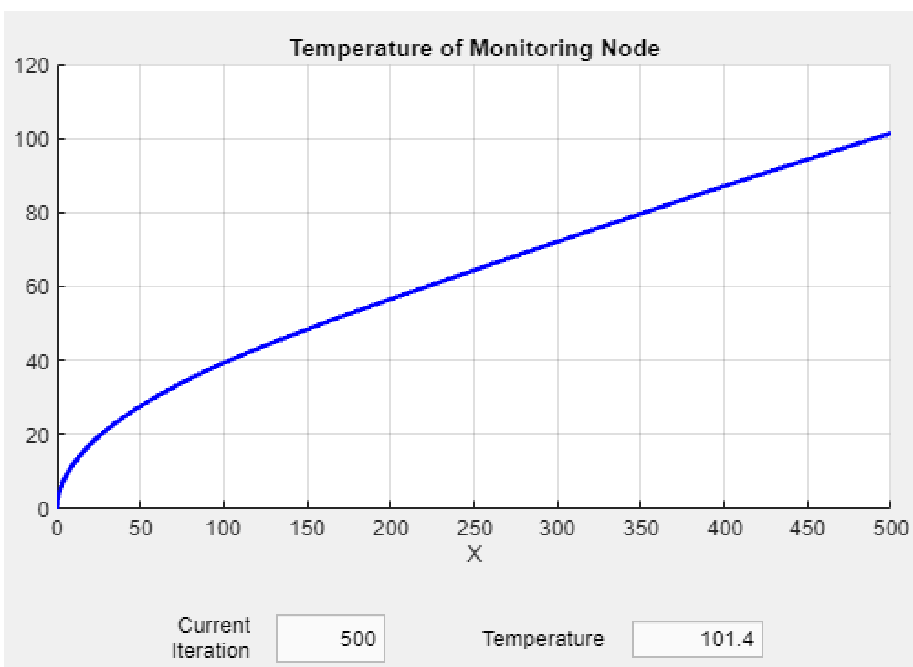

Figure 7: Temperature - Iteration Plot of monitored node (1, 1)

DISCUSSION

The application yielded successful outcomes, especially when referencing a specific example from a book. This suggests that the ADI method can generally be applicable to various heat transfer problems.

The specified boundary conditions during the application has a significant role in the results. These conditions directly affect the temperature distribution.

CONCLUSION

In conclusion, in this project, I implemented the Alternating Direction Implicit (ADI) method using MATLAB App Designer for simulating heat conduction problems. The objective was to validate the effectiveness of the ADI method, and the results were compared with a reference problem from H. Versteeg's computational fluid dynamics book.