International Journal of Scientific Research and Engineering Development-– Volume 3 Issue 6, Nov-Dec 2020

Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

2-D Heat Conduction Analysis of a Square Slab for Steady and Transient Behaviour Using Various Iterative Solvers

Rajesh Kumar*, Vidyanand Kumar**, Animesh Sinha***

*(Dept. of Mechanical Engineering, CVR college of Engineering, Telangana, India)

** (Dept. of Mechanical Engineering, CVR college of Engineering, Telangana, India)

*** (Dept. of Mechanical Engineering, CVR college of Engineering, Telangana, India)

--************************----------------------------------

Abstract:

In this paper, a 2D heat conduction equation was solved numerically using the point iterative techniques such as Jacobi, Gauss Seidel & Successive over-relaxation for both implicit and explicit schemes. To analyse the heat transfer from the surrounding, a 2D rectangular slab of any material was taken and was divided into a grid. The 2D heat equation was solved for both steady and unsteady state and after comparing the results was found that Successive over-relaxation method is the most effective iteration method when compared to Jacobi and Gauss-Seidel.

Keywords **—Heat conduction, 2D slab, MATLAB, Jacobi, Gauss-Seidel, SOR**

--************************----------------------------------

I. INTRODUCTION

Heat transfer illustrates the flow of heat due to differences in the temperature. The study of this transport phenomenon concerns the exchange of momentum, energy, and mass in the form of convection, conduction and radiation. All these processes can be described with the help of mathematical formulae. Heat transfer processes are important for all energy systems including electronic products, aerospace, automobiles and heat exchangers. In automobile industry heat transfer plays a vital role because the automobile is exposed to various temperature conditions. Regardless of more information in composite materials for the chassis of the automobile, temperature for the comfort of the passenger is considered the top priority.

In this paper a MATLAB program is written to study the 2D heat conduction problem of a slab using various iterative methods. The 2D heat equation was solved to visualize the estimate the temperature variations and transfer of heat at different points of the grid on the slab in order to find the stabilized temperature at the different points on the grid.

II. PROBLEM STATEMENT

1. Computational area is to be taken as square shape
2. It is assumed nx=ny; [no of points on x-direction is equivalent to no of points on y-direction]
3. Boundary conditions are

Left = 400 K

Right=600 k

Top=800 K

Bottom=900 k

4. The initial assumed temperature is 300 K

There are various schemes for solving implicit equations

i) Jacobi

ii) Gauss-Seidel

iii) Successive over relaxation

III. METHOD

The heat equation is a partial differential equation that describes the distribution of heat evolves over time in a solid medium.

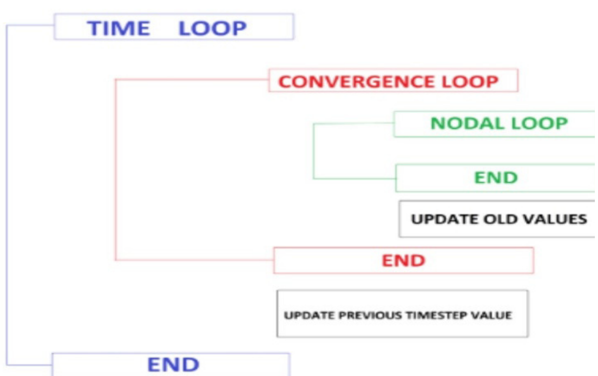
For a function u(x,y,z,t) of three spatial variables (x,y,z) and the time variable t, the heat equation is

$\frac{\partial u}{\partial t} = \alpha\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)$ where, α is called the thermal diffusivity of the medium.

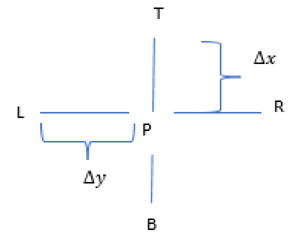
This type of heat transfer may occur under two conditions

- Steady state condition
- Unsteady/Transient state condition

Time Logic Loop



Δx- the distance between point T and P
Δy- the distance between point L and P



Δx- the distance between point T and P
Δy- the distance between point L and P

IV. MATHEMATICAL FORMULATION

In order to solve the 2D Heat conduction problem, the governing equations should be discretised to write the code in MATLAB.

Steady State Equation Discretization

In steady state conduction, the amount of heat entering any region of an object is equal to the amount of heat coming out of it.

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$$

By discretizing the equation, it becomes

$$T_p = \frac{1}{k}\left[\frac{T_L + T_R}{\Delta x^2} + \frac{T_T + T_B}{\Delta y^2}\right]$$

Here the value of k is,

$$K = 2\left(\frac{\Delta x^2 + \Delta y^2}{\Delta x^2 \Delta y^2}\right)$$

On further simplifying the equation for MATLAB code it becomes

$$T_{i,j} = 0.25 \cdot \left(T_{i-1,j} + T_{i+1,j} + T_{i,j-1} + T_{i,j+1}\right)$$

Implicit scheme unsteady/ Transient -state condition

Unsteady -state Equation discretization:

Non-steady-state situations appear after an imposed change in temperature at a boundary as well as inside of an object, as a result of new source or sink of heat suddenly introduced within an object, causing temperatures near the source or sink to change in time.

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}$$

On discretizing the equation, it gives,

$$\frac{T_P^{n+1} - T_P^n}{\Delta t} = \alpha \frac{T_L - 2T_P + T_R}{\partial x^2} + \frac{T_T - 2T_P + T_B}{\partial y^2}$$

On further simplifying it,

$$(T_P^{n+1}) = \frac{1}{1 + k1 + k2}(T_P^n + k1(T_L + T_R)^{n+1} + k2(T_T + T_B)^{n+1})$$

Where, $k1 = \alpha \frac{\Delta t}{\Delta x^2} k2 = \alpha \frac{\Delta t}{\Delta y^2}$ and α is the thermal diffusivity.

$$T_{i,j} = \frac{1}{1 + k1 + k2}\left(T_{i,j} + k1\left(T_{i-1,j} + T_{i+1,j}\right) + k2\left(T_{i,j-1} + T_{i,j+1}\right)\right)$$

Explicit Scheme-Unsteady / Transient state Condition

Unsteady-state Equation discretization (EXPLICIT):

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}$$

If we discretize the equation it becomes,

$$\frac{T_P^{n+1} - T_P^n}{\Delta t} = \alpha \frac{T_L - 2T_P + T_R}{\partial x^2} + \frac{T_T - 2T_P + T_B}{\partial y^2}$$

On further simplifying it further,

$$(T_P^{n+1}) = T_P^n + k1(T_L - 2T_P + T_R)^n + k2(T_T - 2T_P + T_B)^n$$

Where, $k1 = \alpha \frac{\Delta t}{\Delta x^2} k2 = \alpha \frac{\Delta t}{\Delta y^2}$ and α is the thermal diffusivity.

For better utilization in the MATLAB code

$$T_{i,j} = T_{i,j} + k1\left(T_{i-11,j} - 2T_{i,j} + T_{i+1,j}\right) + k2\left(T_{i,j-1} - 2T_{i,j} + T_{i,j+1}\right)$$

V. NUMERICAL CALCULATION

To study the temperature distribution on the slab, the domain was divided into grids. The boundary conditions are taken with the discretized equations and including the boundary temperature given in the problem. To solve the equation MATLAB programming language was used. These equations are solved by using point iterative methods like Gauss Seidel, Jacobi and Successive Over Relaxation to identify which methods give the higher convergence rate.

Steady State

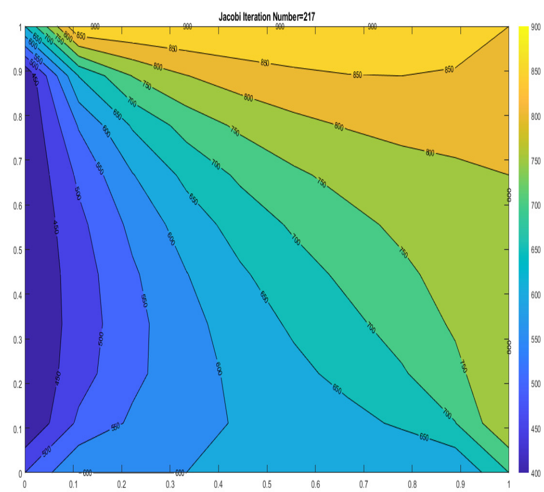


Fig-1: Jacobi Method

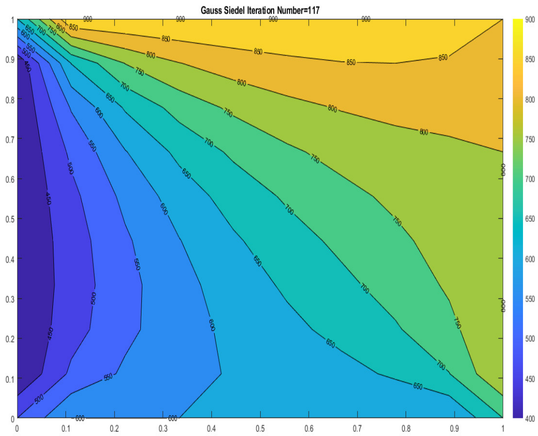


Fig-2: Gauss- Seidel Method

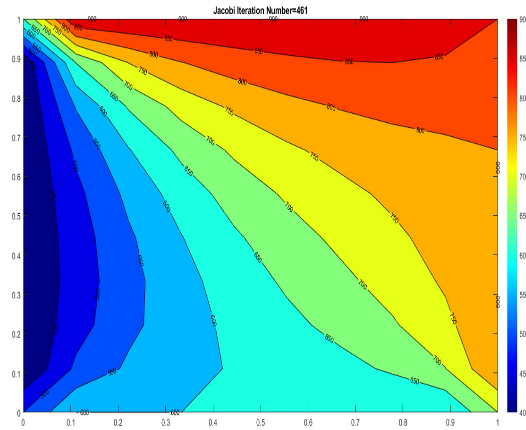


Fig-4: Jacobi Method

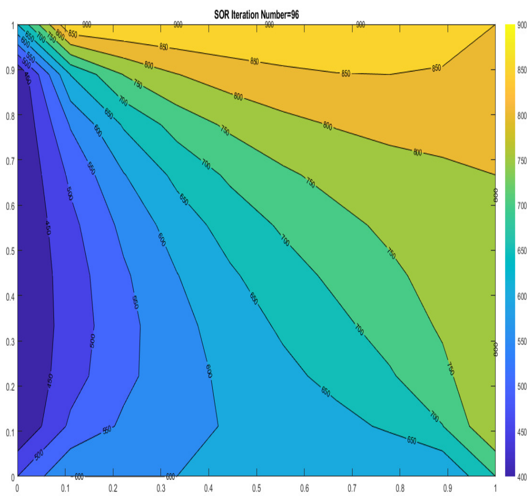


Fig-3: SOR(ω=1.1)Method

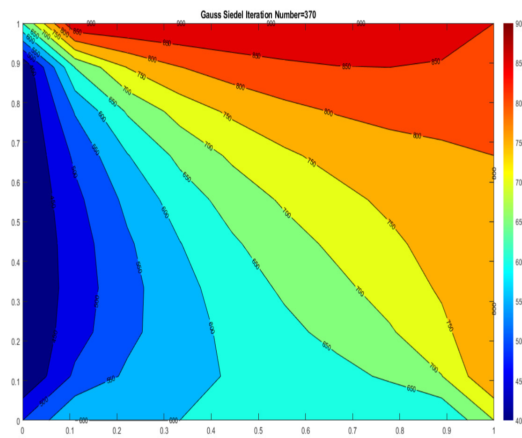


Fig-5: Gauss- Seidel Method

Unsteady State/ Transient State

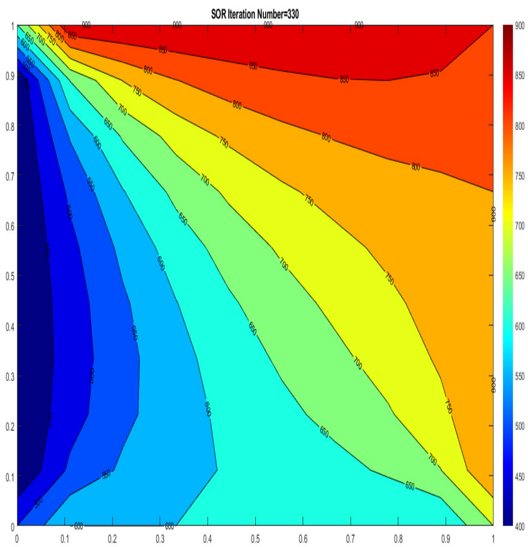
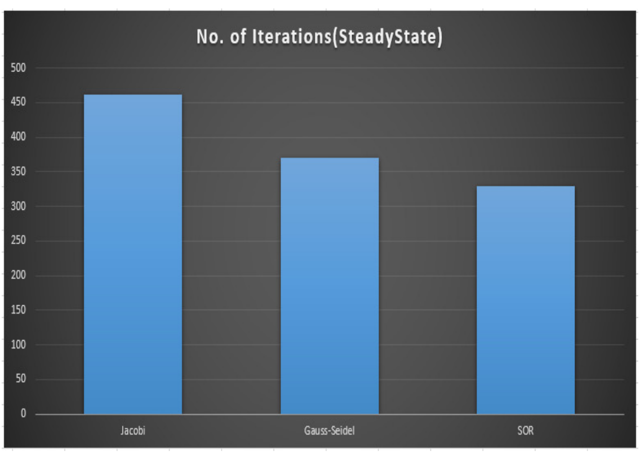


Fig-6: SOR(ω=1.1)Method

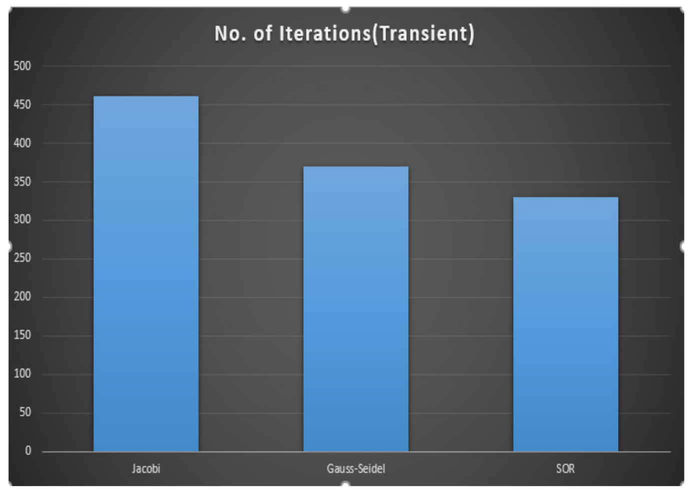
VI. CONCLUSIONS

Table -1: Steady State



On comparing the three iterative techniques in the figure 1,2 and 3, it was found that the computational time required to converge the solution was highest in the Jacobi Method as it doesn't update its values and continues with the guess value. Whereas the Gauss-Seidel Method, it uses the updated value of the previous iteration the no of iterations required to converge the solution is less in the SOR Method due to the use of a relaxation factor. If, the alpha >1, then it is called over relaxation and the solution get converge faster. If the correction factor is higher than required, then the solution can blow instead of converging and the solution diverges.

Table -2: Unsteady state/ Transient state



The transient analysis gives better performance compared to the steady-state as it depicts the real-world scenario. The transient analysis takes into consideration the varying rate of heat transfer with respect to time.
Similar to steady-state analysis, the no of iterations required to converge the solution was falling from the Jacobi method to the SOR Method as depicted by the bar graph.

ACKNOWLEDGMENT

The authors are thankful to CVR college of engineering, Hyderabad for their kind cooperation to accomplish this study. We are even thankful to Dr. B. K. Patle for his guidance and support on this project.

REFERENCES

[1] Parag V. Patil and J. S. V. R. Krishna Prasad, "Numerical Solution for Two Dimensional Laplace Equation with Dirichlet Boundary Conditions", International Organization of Scientific Research- Journal of Mathematics, Volume- 6, Issue- 4, Page no.(66-75), May 2013. J.

Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[2] Mohammed Hasnat, NourddineKaid, Mohammed Bensafi, AbdellahBelkacem , "A numerical Technique Finite Volume Method for Solving Diffusion 2D Problem", The International Journal Of Engineering And Science, Volume- 4,Issue- 10,Page no.(35-41),October 2015.

[3] J. D. Anderson,"Computational Fluid Dynamics: The Basics with Applications", McGraw-Hill Publications, First(Ist) edition, ISBN-0070016852, April 1995.

[4] Parag V. Patil, J. S. V. R. Krishna Prasad, "The unsteady state finite volume numerical grid technique for multidimensional problems", Int J. Adv. Appl. Math. and Mech, Volume-. 2, Issue-2, Page no. (78-87), December 2014.

[5] J. S. V. R. Krishna Prasad and Parag V. Patil , "Finite Volume Numerical Grid Technique for Multidimensional Problems", International Journal of Science and Research , Volume-3, Issue-7, Page no. (518-523), July 2014

[6] H.K. Versteeg, W. Malalsekra, "An Introduction to Computational Fluid Dynamics: The Finite Volume Method", Prentice Hall, Second(2nd) Edition, ISBN0131274988, February 2007.

[7] A.W.M. van Schijndel, H.L. Schellen. (2009). The Use of COMSOL in Teaching Heat and Moisture Transport, Modeling in Building Constructions, Excerpt from the Proceedings of the COMSOL Conference 2009 Milan.

[8] MATLAB, MathWorks , Natick, Massachusetts, U.S.A. [Online] available: http://www.mathworks.com/products/matlab/description2.html

