Jurnal Inovasi Pendidikan Matematika

Volume 3, Nomor 1, Juli 2022

Iterative Method of Thomas Algorithm on The Case Study of Energy Equation

Mohamad Tafrikan[1], Mohammad Ghani[2]

Department of Mathematics, Universitas Islam Negeri Walisongo[1]; tafrikan@walisongo.ac.id[1]
Faculty of Advanced Technology and Multidiscipline, Universitas Airlangga[2];
mohammad.ghani@ftmm.unair.ac.id[2]

Abstract

Implicit method is one of the finite difference method and is widely used for discretization some of ordinary or partial differential equations, such like: advection equation, heat transfer equation, burger equation, and many others. Implicit method is unconditionally stable and has been proved with the approximation of Von-Neumann stability criterion. Actually, implicit method is always identical to block matrices (tri-diagonal matrices or penta-diagonal matrices). These matrices can be solved numerically by Thomas algorithm including Gauss elimination using pivot or not, backward or forward substitution. Furthermore, it can be also solved using LU decomposition method with the elimination of lower triangle matrices first and then the elimination of upper triangle matrices. In this research, Thomas algorithm is used to solve numerically for the problem of convective flow on boundary layer, especially for energy equation with the variation of Prandtl number (P_r).

Keywords: *Implicit method, Thomas algorithm, Gauss elimination, Backward or Forward substitution*

Abstrak

Metode implisit merupakan salah satu metode beda hingga dan banyak digunakan untuk diskritisasi beberapa persamaan diferensial biasa atau parsial, seperti: persamaan adveksi, persamaan perpindahan panas, persamaan burger, dan lain-lain. Metode implisit stabil tanpa syarat dan telah dibuktikan dengan pendekatan kriteria stabilitas Von-Neumann. Sebenarnya, metode implisit selalu identik dengan matriks blok (matriks tri-diagonal atau matriks penta-diagonal). Matriks-matriks ini dapat diselesaikan secara numerik dengan algoritma Thomas termasuk eliminasi Gauss menggunakan pivot atau tidak, substitusi mundur atau maju. Selanjutnya dapat juga diselesaikan menggunakan metode dekomposisi LU dengan eliminasi matriks segitiga bawah terlebih dahulu kemudian eliminasi matriks segitiga atas. Dalam penelitian ini, algoritma Thomas digunakan untuk menyelesaikan secara numerik permasalahan aliran konvektif pada lapisan batas, khususnya untuk persamaan energi dengan variasi bilangan Prandtl (P_r).

Kata kunci: Metode implisit, Algoritma Thomas, Eliminasi Gauss, Substitusi Mundur atau Maju

INFO ARTIKEL

ISSN : 2733-0597 *e-ISSN : 2733-0600* *DOI : http://dx.doi.org/10.30587/postulat.v3i1.4346*	***Jejak Artikel*** Submit Artikel: 3 Juni 2022 Submit Revisi: 8 Juli 2022 Upload Artikel: 26 Juli 2022

INTRODUCTION

Differential equations are equations that involve an unknown function and derivatives. There will be times when solving the exact solution for the equation may be unavailable or the means to solve it will be unavailable. At these times and most of the time explicit and implicit methods will be used in place of exact solution (Bui, 2010), (Němec et al, 2017). Hasan et al (2013) explained that implicit method is one of the finite difference method and is widely used for discretization some of ordinary or partial differential equations, such like: advection equation, heat transfer equation, burger equation, and many others.

Actually, implicit method is always identical to block matrices, its means tri-diagonal matrices or penta-diagonal matrices (Zhu, 1994). These matrices can be solved numerically by Thomas algorithm including Gauss elimination using pivot or not, backward or forward substitution. The Thomas algorithm requires two recursive loops, forward and backward (Hollig, et al 2001), (Chapra, 2010). Furthermore, the Gaussian Elimination is a simple, systematic algorithm to solve systems of linear equations. It is the workhorse of linear algebra, and, as such, of absolutely fundamental importance in applied mathematics (Peter, 2008).

In this research, Thomas algorithm is used to solve numerically for the problem of convective flow on boundary layer, especially for energy equation with the variation of Prandtl number Pr. Finally, the implicit method is unconditionally stable under the Von-Neumann stability criteria which gives also the same results of equation (8). This statement can be seen in Figure 3 which always give converge results for any input of Prandtl number

(Pr). Thomas algorithm is very appropriate to make an easy of discretization iteratively (in the tri-diagonal matrices) based on the implicit method.

PRELIMINARIES

Implicit method is one of the finite difference method and is unconditionally stable. This can be shown by the following simple example of one-dimensional heat transfer equation

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

where α is thermal conductivity, initial condition: $T(x,0) = f(x)$ and boundary condition: $T(0,t) = T(L,t) = 0$.

By using Von-Neuman stability criterion with the following truncation error

$$T_{i,j} = \varepsilon^j e^{mn(i\Delta x)}$$

Furthermore, substitute this truncation error function into the following discretization

$$(1+2r)T_{i,j+1} - rT_{i-1,j+1} - rT_{i+1,j+1} = T_{i,j}$$

then obtained

$$(1+2r)\varepsilon^{j+1}e^{mn(i\Delta x)} - r\varepsilon^{j+1}e^{mn(i-1)\Delta x} - r\varepsilon^{j+1}e^{mn(i+1)\Delta x} = \varepsilon^j e^{mn(i\Delta x)}$$

where the parameter $r = \frac{\alpha \Delta t}{\Delta x^2}$. Furthermore, by dividing the two terms with the $\varepsilon^j e^{mn(i\Delta x)}$, then obtained

$$(1+2r)\varepsilon - r\varepsilon(e^{-mn\Delta x} + e^{mn\Delta x}) = 1$$

Based on the properties of exponential in complex variable, then obtained

$$\varepsilon = \frac{1}{1+2r(1-\cos(k\Delta x))}$$

According to Von-Neumann stability criterion that the stable criterion is achieved when $|\varepsilon| \leq 1$. To check this condition, then the following two conditions are analyzed

a). Condition I: for the maximum value of $\cos(k\Delta x) = 1$, then it is clearly obtained $\varepsilon = 1$.

b). Condition II: for the minimum value of $\cos(k\Delta x) = -1$, then obtained $\varepsilon = \frac{1}{1+2r(2)}$. Every element of ε in positive real number, then it is still in range of $|\varepsilon| \leq 1$.

Furthermore, it can be concluded that Implicit method is unconditionally stable. After studying Von-Neumann stability for 1D unsteady Heat Transfer, then the next steps are to create tri-diagonal matrix by doing discretization first for the following 1D unsteady Heat Transfer

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \tag{1}$$

where α is thermal conductivity. Furthermore, defined an initial condition: $T(x, 0) = f(x)$ and boundary conditions: $T(0, t) = T(L, t) = 0$. By doing approximation for time with the forward difference and for space with the central difference, then obtained

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = \alpha \left[\frac{T_{i+1}^{n+1} - 2T_i^{n+1} + T_{i-1}^{n+1}}{\Delta x^2} \right] \tag{2}$$

By rearranging the above discretization, then obtained

$$T_i^{n+1} - T_i^n = \frac{\alpha \Delta \mathrm{t}}{\Delta \mathrm{x}^2} (T_{i+1}^{n+1} - 2T_i^{n+1} + T_{i-1}^{n+1}) \tag{3}$$

Once again, by rearranging and grouping the same terms $n + 1$ and n, then obtained

$$(1 + 2r)T_{i,j+1} - rT_{i-1,j+1} - rT_{i+1,j+1} = T_{i,j} \tag{4}$$

where parameter r is defined as

$$r = \frac{\alpha \Delta t}{\Delta x^2} \tag{5}$$

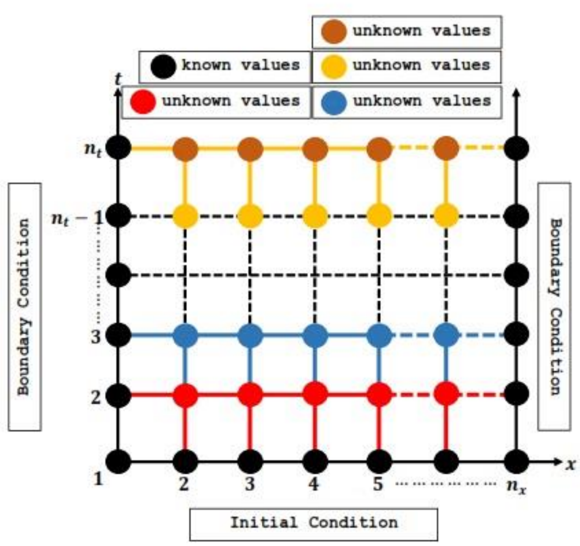

Figure 1. Implicit Scheme

Because of the boundary conditions: $T(0,t) = T(n_x,t) = 0$ and index on **MATLAB** is started from index 1, then Equation (4) is iterated by $i = 2, 3, 4, \dots, n_x - 1$. In this case, the boundary conditions can be rewritten as: $T(1,t) = T(n_x,t) = 0$. To give some descriptions, the following implicit scheme for this case is given in Figure 1.

Figure 1 gives illustration for implicit scheme, it shows that the value at step 2 of time needs the value at step 1 of time for every iterations of space started with the iteration of $1, 2, 3, \dots, n_x$. Finding the unknown values at step $3, 4, 5, \dots, n_t$ is analogue with the previous step. Based on this implicit scheme, then the discretization of 1D unsteady Heat Transfer is iterated and obtained as follows

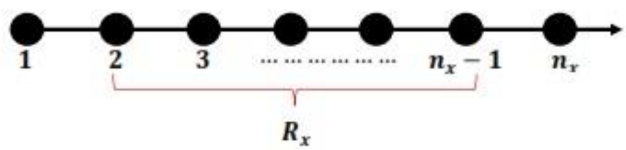

Figure 2. Domain Implicit Scheme

According to Figure 1, then Figure 2 gives description that the domain of implicit scheme is $R_x = [2, 3, \dots, n_x - 1]$. This case is caused that the boundary conditions are at point 1 and point n_x. Furthermore, based on the two Figures, then obtained the following discretization of $(1+2r)T_{i,j+1} - rT_{i-1,j+1} - rT_{i+1,j+1} = T_{i,j}$

Table 1. Iteration for Implicit Method in 1D unsteady Heat Transfer

iteration	result
$i = 2$	$(1+2r)T_{2,j+1} - rT_{1,j+1} - rT_{3,j+1} = T_{2,j}$
$i = 3$	$(1+2r)T_{3,j+1} - rT_{2,j+1} - rT_{4,j+1} = T_{3,j}$
$i = 4$	$(1+2r)T_{4,j+1} - rT_{3,j+1} - rT_{5,j+1} = T_{4,j}$
$\vdots$	$\vdots$
$i = n_x - 2$	$(1+2r)T_{n_x-2,j+1} - rT_{n_x-3,j+1} - rT_{n_x-1,j+1} = T_{n_x-2,j}$
$i = n_x - 1$	$(1+2r)T_{n_x-1,j+1} - rT_{n_x-2,j+1} - rT_{n_x,j+1} = T_{n_x-1,j}$

By rearranging the above iteration, then obtained

$$\begin{bmatrix} (1+2r)T_{2,j+1} - rT_{1,j+1} - rT_{3,j+1} \\ (1+2r)T_{3,j+1} - rT_{2,j+1} - rT_{4,j+1} \\ (1+2r)T_{4,j+1} - rT_{3,j+1} - rT_{5,j+1} \\ \vdots \\ (1+2r)T_{n_x-2,j+1} - rT_{n_x-3,j+1} - rT_{n_x-1,j+1} \\ (1+2r)T_{n_x-1,j+1} - rT_{n_x-2,j+1} - rT_{n_x,j+1} \end{bmatrix} = \begin{bmatrix} T_{2,j} \\ T_{3,j} \\ T_{4,j} \\ \vdots \\ T_{n_x-2,j} \\ T_{n_x-1,j} \end{bmatrix} \tag{6}$$

Based on the boundary conditions: $T(1,t) = T(n_x,t) = 0$, then the above matrix can be rewritten as

$$\begin{bmatrix} (1+2r)T_{2,j+1} - rT_{3,j+1} \\ (1+2r)T_{3,j+1} - rT_{2,j+1} - rT_{4,j+1} \\ (1+2r)T_{4,j+1} - rT_{3,j+1} - rT_{5,j+1} \\ \vdots \\ (1+2r)T_{n_x-2,j+1} - rT_{n_x-3,j+1} - rT_{n_x-1,j+1} \\ (1+2r)T_{n_x-1,j+1} - rT_{n_x-2,j+1} \end{bmatrix} = \begin{bmatrix} T_{2,j} \\ T_{3,j} \\ T_{4,j} \\ \vdots \\ T_{n_x-2,j} \\ T_{n_x-1,j} \end{bmatrix}$$

Or rewritten as follows

$$\begin{bmatrix} (1+2r) & -r & 0 & 0 & 0 \\ -r & (1+2r) & -r & 0 & 0 \\ 0 & -r & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & (1+2r) & -r \\ 0 & 0 & 0 & -r & (1+2r) \end{bmatrix} \begin{bmatrix} T_{2,j+1} \\ T_{3,j+1} \\ T_{4,j+1} \\ \vdots \\ T_{n_x-1,j+1} \end{bmatrix} = \begin{bmatrix} T_{2,j} \\ T_{3,j} \\ T_{4,j} \\ \vdots \\ T_{n_x-1,j} \end{bmatrix} \tag{7}$$

MAIN RESULTS

Based on the previous section, we can apply the similar ways to the following equation which is our attention in this paper

$$\frac{1}{P_r}\,\theta'' + 2f\theta' = 0$$

with the following boundary conditions.

$$\theta'(0) = -1$$

$$\theta \to 0 \; at \; y \to \infty$$

By doing discretization with central difference into the energy equation and the related boundary conditions, then obtained

$$\frac{1}{P_r}\left(\frac{\theta_{i+1}-2\theta_i+\theta_{i-1}}{\Delta y^2}\right)+2f_i\left(\frac{\theta_{i+1}-\theta_{i-1}}{2\Delta y}\right)=0$$

By simplifying the discretization, then obtained

$$(r_1+r_2f_i)\theta_{i+1}+(r_1-r_2f_i)\theta_{i-1}-2r_1\theta_i=0$$

where $r_1=\frac{1}{P_r\Delta y^2}$ and $r_2=\frac{1}{\Delta y}$. The boundary conditions can be written as follows

$$\frac{\theta_{i+1}-\theta_{i-1}}{2\Delta y}=-1 \Leftrightarrow \theta_{i-1}=2\Delta y+\theta_{i+1}\; at\; y=0$$

and

$$\theta \rightarrow 0\; at\; y \rightarrow \infty$$

Furthermore, it is iterated by $i=1,2,3,\dots,N$, then obtained

iteration	**result**
$i=1$	$(r_1+r_2f_1)\theta_2+(r_1-r_2f_i)\theta_0-2r_1\theta_1=0$
$i=2$	$(r_1+r_2f_1)\theta_3+(r_1-r_2f_i)\theta_1-2r_1\theta_2=0$
$i=3$	$(r_1+r_2f_1)\theta_4+(r_1-r_2f_i)\theta_2-2r_1\theta_3=0$
$\vdots$	$\vdots$
$i=N$	$(r_1+r_2f_1)\theta_{N+1}+(r_1-r_2f_i)\theta_{N-1}-2r_1\theta_N=0$

where $f_i=1+\frac{i}{N}$. By rearranging the above discretization, then obtained

$$\begin{bmatrix}(r_1+r_2f_i)\theta_2-2r_1\theta_1\\(r_1+r_2f_i)\theta_3+(r_1-r_2f_i)\theta_1-2r_1\theta_2\\(r_1+r_2f_i)\theta_4+(r_1-r_2f_i)\theta_2-2r_1\theta_3\\\vdots\\(r_1-r_2f_i)\theta_{N-1}-2r_1\theta_N\end{bmatrix}=\begin{bmatrix}-(r_1-r_2f_i)\theta_0\\0\\0\\\vdots\\-(r_1+r_2f_i)\theta_{N+1}\end{bmatrix}$$

Based on the boundary conditions, then obtained

$$\begin{bmatrix}(r_1+r_2f_i)\theta_2-2r_1\theta_1\\(r_1+r_2f_i)\theta_3+(r_1-r_2f_i)\theta_1-2r_1\theta_2\\(r_1+r_2f_i)\theta_4+(r_1-r_2f_i)\theta_2-2r_1\theta_3\\\vdots\\(r_1-r_2f_i)\theta_{N-1}-2r_1\theta_N\end{bmatrix}=\begin{bmatrix}-(r_1-r_2f_i)(2\Delta y+\theta_2)\\0\\0\\\vdots\\0\end{bmatrix}$$

Or rewritten as follows

$$\begin{bmatrix} -2r_1 & (r_1+r_2f_i) & 0 & 0 & 0 \\ (r_1-r_2f_i) & -2r_1 & (r_1+r_2f_i) & 0 & 0 \\ 0 & (r_1-r_2f_i) & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & -2r_1 & (r_1+r_2f_i) \\ 0 & 0 & 0 & (r_1-r_2f_i) & -2r_1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_N \end{bmatrix} = \begin{bmatrix} -(r_1-r_2f_i)(2\Delta y+\theta_2) \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{8}$$

Before we establish the numerical simulation of our main problem, we firstly introduce the Thomas Algorithm which is employed to the discretization of our main problem. Thomas algorithm is one of technique to solve numerically the problem of partial or ordinary differential equations that has been changed in tri-diagonal or penta-diagonal matrices. This algorithm is actually identical to Gauss elimination and backward substitution. Based on the tri-diagonal matrix that has been obtained from the discretization of energy equation on convective flow of boundary layer, then the tri-diagonal matrix must be written as shown this below

$$\begin{bmatrix} \mathrm{b}_1 & c_1 & 0 & 0 & \dots & 0 \\ a_2 & \mathrm{b}_2 & c_2 & 0 & \dots & 0 \\ 0 & a_3 & \mathrm{b}_3 & c_3 & \dots & \vdots \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & \vdots & a_{N-1} & \mathrm{b}_{\mathrm{N}-1} & c_{N-1} \\ 0 & 0 & \dots & 0 & a_N & \mathrm{b}_{\mathrm{N}} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{N-1} \\ v_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{N-1} \\ d_N \end{bmatrix} \tag{9}$$

Furthermore, this matrix is eliminated using Gauss elimination, then obtained

$$\begin{bmatrix} \mathrm{b'}_1 & c_1 & 0 & 0 & \dots & 0 \\ 0 & \mathrm{b'}_2 & c_2 & 0 & \dots & 0 \\ 0 & 0 & \mathrm{b'}_3 & c_3 & \dots & \vdots \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & \vdots & 0 & \mathrm{b'}_{\mathrm{N}-1} & c_{N-1} \\ 0 & 0 & \dots & 0 & 0 & \mathrm{b'}_{\mathrm{N}} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{N-1} \\ v_N \end{bmatrix} = \begin{bmatrix} d'_1 \\ d'_2 \\ d'_3 \\ \vdots \\ d'_{N-1} \\ d'_N \end{bmatrix} \tag{10}$$

where

$$\mathrm{b}'_1 = \mathrm{b}_1 \tag{11}$$

$$d'_1 = d_1 \tag{12}$$

$$\mathrm{b}'_2 = \mathrm{b}_2 - c_1 \frac{a_2}{\mathrm{b}'_1} \tag{13}$$

$$d_2' = d_2 - d'_1 \frac{a_2}{\mathrm{b'}_1} \tag{14}$$

Equations (3) and (4) can be written recursively as shown this below

$$\mathrm{b}_\mathrm{i}' = \mathrm{b}_\mathrm{i} - c_{i-1} \frac{a_i}{\mathrm{b'}_{\mathrm{i}-1}} \tag{15}$$

$$d_i' = d_i - d'_{i-1} \frac{a_i}{\mathrm{b'}_{\mathrm{i}-1}} \tag{16}$$

where $i = 2, 3, \dots, N$.

For the solution, then backward substitution is used recursively as shown this below

$$v(N) = \frac{d'(N)}{\mathrm{b}'(N)} \tag{17}$$

$$v(i) = \frac{d'(i) - c(i)v(i+1)}{\mathrm{b}'(\mathrm{i})} \tag{18}$$

where $i = N-1, N-2, \dots, 2, 1$.

When the tri-diagonal matrix of Thomas algorithm in Equation (9) is applied into the tri-diagonal matrix in Equation (8), then obtained the following similarity

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{N-1} \\ v_N \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_N \end{bmatrix} \tag{19}$$

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{N-1} \\ d_N \end{bmatrix} = \begin{bmatrix} -(r_1 - r_2 f_i)(2\Delta y + \theta_2) \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{20}$$

$$\begin{bmatrix} b_1 & c_1 & 0 & 0 & \dots & 0 \\ a_2 & b_2 & c_2 & 0 & \dots & 0 \\ 0 & a_3 & b_3 & c_3 & \dots & \vdots \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & \vdots & a_{N-1} & b_{N-1} & c_{N-1} \\ 0 & 0 & \dots & 0 & a_N & b_N \end{bmatrix}$$

$$= \begin{bmatrix} -2r_1 & (r_1 + r_2 f_i) & 0 & 0 & 0 \\ (r_1 - r_2 f_i) & -2r_1 & (r_1 + r_2 f_i) & 0 & 0 \\ 0 & (r_1 - r_2 f_i) & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & -2r_1 & (r_1 + r_2 f_i) \\ 0 & 0 & 0 & (r_1 - r_2 f_i) & -2r_1 \end{bmatrix} \qquad (21)$$

Furthermore, Thomas algorithm is applied on tri-diagonal matrix of energy equation that is a derivation of convective flow on boundary layer, then obtained the following figure

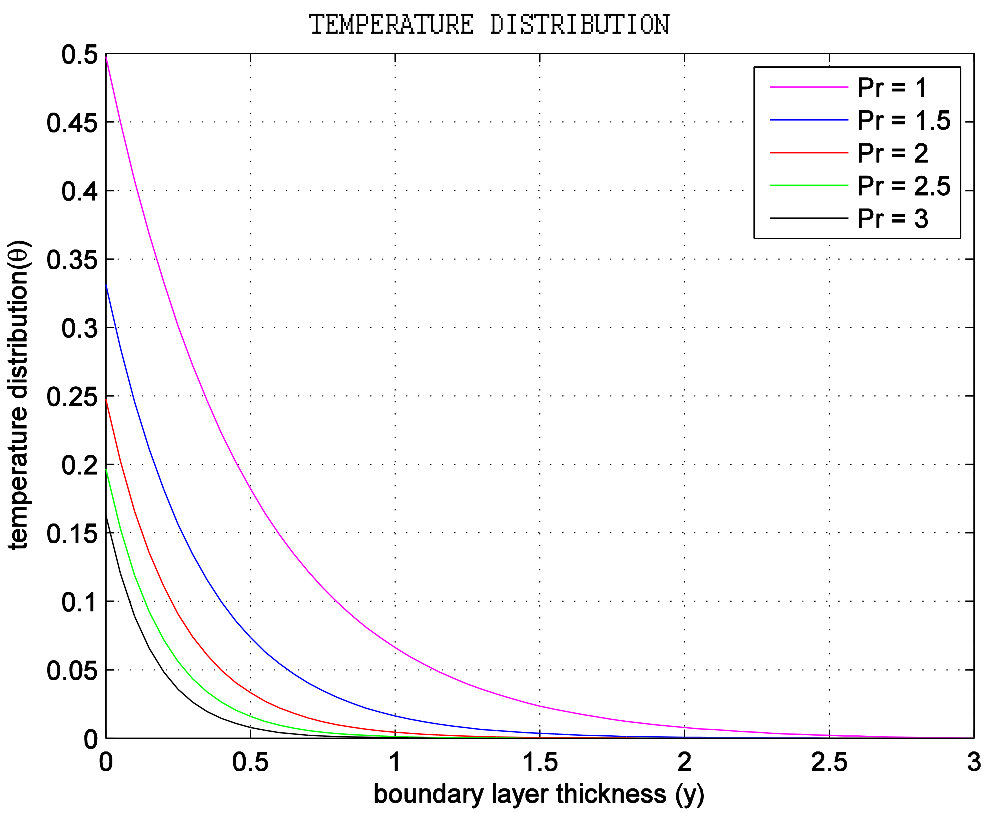

Figure 3. Temperature distribution with the thickness of boundary layer y

Figure 3 shows that when the Prandtl number (P_r) is increased then temperature distribution is decreased. In this case, it is related to heat transfer that is more increased, so that it causes temperature near bluff body is more decreased. This numerical result is iterated until 1000 iterations with the step size of y is equal to 0.005.

CONCLUSIONS

Based on the preliminaries and main results of this paper, we can give some conclusions as follows:

1. The implicit method is unconditionally stable under the Von-Neumann stability criteria which gives also the same results of equation (8). This statement can be seen in Figure 3 which always give convergent results for any input of Prandtl number (P_r).
2. Thomas algorithm is very appropriate to make an easy of discretization iteratively (in the tri-diagonal matrices) based on the implicit method.

AKCKNOWLEDGEMENTS

The authors would like to say thanks to reviewers for the useful comments to upgrade the quality of this paper.

REFERENCES

Bui, T. (2010). *Explicit and Implicit Methods in Solving Differential Equations.* University of Connecticut. US.

Chapra. (2010). *Numerical Methods, 6th Edition.* USA.

Hasan et al (2013). *An Implicit Method for Numerical Solution of Singular and Stiff Initial Value Problems.* Journal of Computational Engineering. Bangladesh.

Hollig et al. (2001). *Weighted extended B-spline approximation of Dirichlet problems.* SIAM Journal Numeric Analysis. USA.

Němec et al. (2017). *Explicit and Implicit Method in Nonlinear Seismic Analysis.* MATEC Web of Conferences, Czech Republic.

Peter, J. (2008). *Numerical Analysis Lecture Notes.* University of Minnesota Publishing, UK.

Zhu, J. (1994). *Solving Partial Differential Equations on Parallel Computer.* World Scientific Publishing. Singapore.