Journal of Mountain Area Research

Vol. 5, 2020

http://journal.kiu.edu.pk/index.php/JMAR

Full length article

A COMPARISON OF SERIAL AND PARALLEL SOLUTIONS OF TWO DIMENSIONAL HEAT CONDUCTION EQUATION

A. A. Khan[2], S. Abbas[1,,] and B. Shakia[3]*

1 Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing China.
2 Diamond jubilee Institute, hyderabad. Hunza
3 Elysian Academy, Gahkuch, ghizer

ABSTRACT

We study a comparison of serial and parallel solution of 2D-parabolic heat conduction equation using a Crank-Nicolson method with an Alternating Direction Implicit (ADI) scheme. The two-dimensional Heat equation is applied on a thin rectangular aluminum sheet. The forward difference formula is used for time and an averaged second order central difference formula for the derivatives in space to develop the Crank-Nicolson method. FORTRAN serial codes and parallel algorithms using OpenMP are used. Thomas tridigonal algorithm and parallel cyclic reduction methods are employed to solve the tridigonal matrix generated while solving heat equation. This paper emphasize on the run time of both algorithms and their difference. The results are compared and evaluated by creating GNU-plots (Command-line driven graphing utility).

KEYWORDS: Heat Conduction Equation, ADI Scheme, Crank-Nicolson Method, OpenMP.

*Corresponding author: (Email: suhailkiu156@gmail.com)

1. INTRODUCTION

In this work, the problem we have considered is to find the temperature distribution on a two dimensional thin aluminum sheet using a parabolic Heat Conduction equation, which is homogeneous second order partial differential equation (PDE). We have applied Crank-Nicolson method along ADI scheme. The discritization method we employed is finite difference method and grid size is considered uniform along both axis. All the solution procedure is done in FORTRAN using serial and parallel programming(OpenMP).

Results are represented in the form of graphs using GNU-plots. Main focus of this work is to analyze the differences in serial and parallel algorithms with respect to run time cost.

1.1 Literature Background

Since the concept of numerical computation is ancient, as it is evident from discovery of the tablet from Babylonian times which describes a numerical method to find the value for square of 2. Since then numerical computation is evolved and developed throughout centuries. Several mathematicians invented various methods during their contemporary times, From

Archimedes "Method of exhaustion" to Issac Newton's various methods and from Swiss mathematician Leonhard Euler to Carl Friedrich Gauss and so on.

Mathematics of diffusion inspired many related mathematical models. Jean Joseph Fourier developed a partial differential equation describing transient process of heat diffusion through solids in his publication 'Analytic theory of heat" 1822 [1]. Finding analytical solutions to ODE's and PDE's is limited by linearity, order and type of boundary conditions, therefore we require numerical approximations. In 1947 John Crank and Phyllis Nicolson published a paper on numerical solution to nonlinear differential equations, in which they proposed a method which is second order accurate and highly convergent [2]. When this scheme is applied on two dimensional parabolic equation it lacks quality of producing tridigonal system. Therefore, to overcome this limitation D.W peaceman and H.H. Rachford proposed alternating-direct implicit method which retains the advantages of Crank Nicolson scheme and generates tridiagonal system [3]. Almost every numerical method based on finite difference method to produce system of linear equations to compute and some efficient methods generate tridigonal system of equations, which take relatively less time to compute. Recent developments in parallel computing and enhancements in computer processors, numerical analysis improved so much. Due to the usefulness of tridigonal schemes several such schemes have been developed and parallelism is introduced to take advantage of parallel computing [4]. A Gaussian elimination based tridiagonal solver was proposed by Llewellyn H. Thomas in 1949[5], is an efficient method that only take $O(m)$ steps and with $O(m^3)$ accuracy. In 1960 R.W. Hockney developed a tridigonal solver which can be manipulated to introduce parallelism in it [6]. This method was modified and co-published with C.R. Jesshope in 1981 [7]. Since then cyclic reduction had gone through different modifications and various versions of it are available.

1.2 Heat Conduction Equation

Parabolic PDE's can represent heat conduction across a material. We can drive a balance heat for differential element of a rectangular aluminum sheet by taking in to account the concept of conservation of heat. The parabolic equation also examines the total heat stored in the element over a unit time period Δt. Following equation represents two dimensional time dependent heat conduction equation [5].

$$\frac{\partial V}{\partial t} = \alpha(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2}) \qquad (1)$$

Where V represents temperature, $\alpha = k/C\rho$ and k is referred to as coefficient of thermal conductivity of the material considered. ρ is density of the material and C is heat capacity of the substance.

2. CRANK-NICOLSON METHOD ALONG WITH ADI SCHEME

The Crank-Nicolson method is an implicit numerical scheme, which has second order accuracy both is spatial dimension and time. It is usually used to approximate first order differential equations. The method generates tridigonal matrices in one dimension, which are fairly easy to compute and takes less storage and time. Moreover, when Crank-Nicolson method is applied on two or more dimensions it gives system of equations, to solve such matrices we require to allocate huge amount of space to store and it takes more time for computation. Therefore, to overcome this difficulty we require ADI scheme to employ on Crank-Nicolson method [8].

Crank-Nicolson method along with ADI scheme is unconditionally convergent and second order accurate both in space and time. It generates tridigonal matrices, which can be solve in less time and using less storage.

Following equation are numerical approximations to heat equation using Crank-Nicolson method along with ADI scheme.

$$-\beta V_{i,j-1}^{k+1} + 2(1+\beta)V_{i,j}^{k+1} - \beta V_{i,j+1}^{k+1}$$
$$= -\beta V_{i-1,j}^{k} + 2(1-\beta)V_{i,j}^{k} - \beta V_{i+1,j}^{k} \qquad (2)$$

$$-\beta V_{i-1,j}^{k+1} + 2(1+\beta)V_{i,j}^{k+1} - \beta V_{i+1,j}^{k+1}$$
$$= -\beta V_{i,j-1}^{k+\frac{1}{2}} + 2(1-\beta)V_{i,j}^{k+\frac{1}{2}} - \beta V_{i,j+1}^{k+\frac{1}{2}} \qquad (3)$$

Where $\beta = \alpha\frac{\Delta t}{(\Delta x)^2}$ and k represents iterations with respect to time.

3. TRIDIAGONAL SOLVERS

Tridiagonal solvers have been extensively studied in last few decades. The motive for this curiosity is of two parts. First is academic because it is fairly straight forward and easy to understand yet challenging due to almost no intrinsic parallelism. Other is practical application, tridigonal system are required to solve in various fields such as ADI solver for partial differential equations, approximation of cubic splines and so on. There are several methods to solve such system, for example Gaussian elimination, LU decomposition, Thomas tridigonal algorithm, cyclic reduction and et cetera.

3.1 Thomas Algorithm

Thomas algorithm is particular case of Gaussian Elimination method. This is solely created to solve tridigonal matrices and solution can be acquired in $O(m)$ operations while $O(m^3/3)$ operations are needed to obtain a solution using Gaussian Elimination method. The solution consists of two parts, first is forward elimination and the other is backward substitution.

3.2 Cyclic Reduction method

The Cyclic Reduction was introduced by Hockney in 1965 together with Golub [6]. The method was preferred over Gaussian Elimination scheme because it dealt with periodic boundary conditions efficiently and additional matrices were not require to compute as in Gaussian elimination. This method has inherent parallelism, which can be exploited using vector and parallel computers.

Cyclic reduction recursively computes new coefficients and right hand side values of the $AX = d$. The different levels of recursion (represented by 'j') continue to produces smaller and smaller tridigonal systems until a single equation is lefts. OpenMP parallelism can be implemented on the cyclic reduction algorithm, which is thoroughly studied in [9].

4. RESULTS

We provide solutions to two dimensional heat conduction equation at different time steps, comparison between Thomas algorithm and Cyclic Reduction run time, and efficiency of parallel Cyclic reduction.

4.1 Heat Conduction through 2-D Aluminum plate

Two dimensional grid of a $(100 \times 100\ cm2)$ aluminum sheet with Dirichlet boundary conditions. The initial condition we consider at time $t = 0$ is

$V\ (x, y, t) = 0$ and the boundary conditions are (temperature is in Celsius):

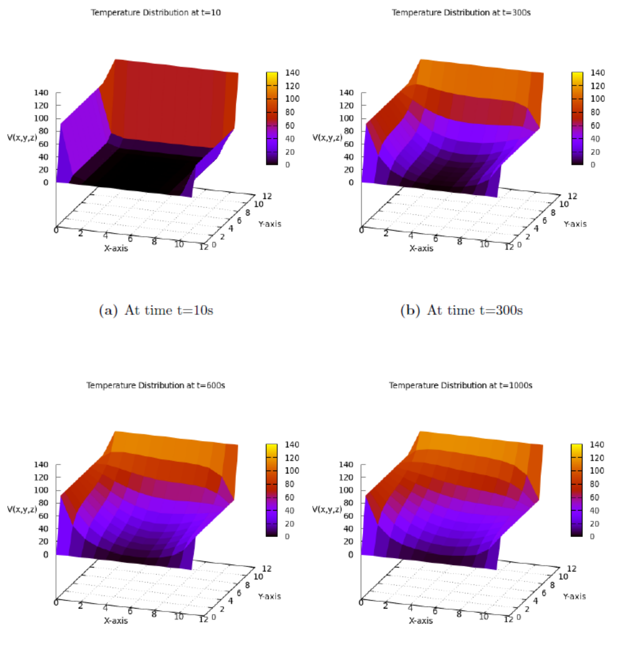
$$V\ (x, 0, t) = 0$$

$$V\ (x, y_n, t) = 125$$

$$V\ (0, y, t) = 85$$

$$V\ (x_n, y, t) = 45$$

Where as $\Delta x = \Delta y = 10cm$, $\Delta t = 10s$. We have 3D GNU-plots for the temperature distribution at each node at $t = 10$, $t = 300$, $t = 600$ and $t = 1000$. These temperature distributions at different times have been shown in Figure 1.



(a) At time t=10s

(b) At time t=300s

(a) At time t=600s

(b) At time t=1000s

Figure 1: Temperature Distribution in 2D Aluminum plate at various values of t.

4.2 Runtime Comparison Between serial Thomas algorithm and serial cyclic reduction

Here we discuss the time taken by serial versions of the two methods for computation with respect to different number of equations. It is to be noted that the run-time is taken for both schemes is of whole Crank Nicolson computation algorithm, moreover the both tridiagonal solvers are called twice in each time iteration. Once for y-sweep and second time for x-sweep, each sweep has the same number of equations or same matrix size. Following time measurements are taken on a system having an Intel core i3 processor with allocated 4GB random-access memory (RAM) as shown in table 1. Where n is number of equation or size of the square matrix and the time is taken in seconds. Solution is obtained by applying 100 iterations.

Table 1: Execution time for Crank Nicolson method using two different tridigonal Schemes.

N	Thomas Algorithm	Cyclic Reduction
0	0	0
3	0.023	0.026
20	0.486	0.371
40	1.518	1.541
80	4.960	4.857
160	19.086	20.478
320	78.150	80.386
640	300.326	314.761

Solution is obtained by applying 100 iterations. The data suggests that run-time for the computation is almost identical for the small number equations; however for large matrices Thomas algorithm takes less time as compare to serial cyclic reduction. Figure 2 illustrates the differences in run-time.

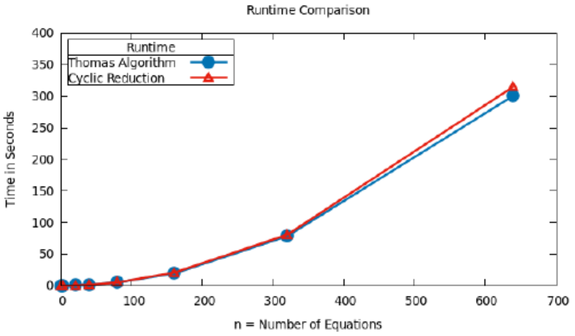


Figure 2: Computational Time comparison for Crank Nicolson method along with ADI scheme

4.3 Parallel and sequential execution time

During computation of implicit finite difference models more than ninety percent of the execution time is taken by recursive tridigonal matrix generated during solution. In 1991 Marilyn Santiago and David R. Kincaid in their publication [10] discussed the improvements in execution time for wave equation by using parallel cyclic reduction method. They used Alliant computer for implementation of parallel cyclic reduction. Following measurements are taken from [10], which shows difference in run-time of scalar method and parallel implementations of cyclic reduction. Hockney and Jesshope [9] proposed application of parallel version of cyclic reduction, moreover its implementation is widely known. The scalar method used by Marilyn Santiago and David R. Kincaid is based on Gaussian elimination method, which is basically Thomas Algorithm. The FORTRAN code for scalar method is given in [5] and it is thoroughly studied. Execution time in table 2 is taken for single iteration, time is in seconds and for parallel execution number of processors is 4. Where n is number of equations or order of tridiagonal system.

Table 2: Runtime comparison between scalar and parallel tridiagonal Schemes.

N	Thomas Algorithm	Cyclic Reduction
0	0	0
100	0.00127053	0.00166068
150	0.00189453	0.00203853
166	0.00208513	0.00208557
175	0.00219822	0.00210630
200	0.00250197	0.00217863
250	0.00312197	0.00232407
300	0.00374203	0.00277244
400	0.00496727	0.00308521
500	0.00621368	0.00338320
700	0.00867713	0.00433912
1000	0.01234523	0.00530377
2000	0.02469464	0.00908263
3000	0.03874320	0.01331816

Thomas algorithm runs faster when $n \leq 166$ and cyclic reduction performs better for $n > 166$. Figure 3 and figure 4 perfectly shows how significant difference in execution time among both the methods. For 100 number of equations speedup is 0.77 which is not significant however for 3000 order of matrix system the speedup is 2.91, which is considerable.

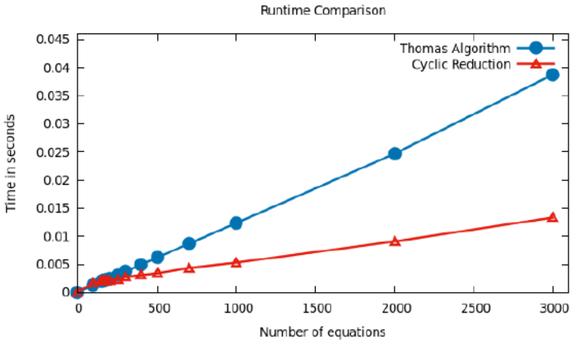


Figure 3: Computational Time comparison between scalar and parallel method

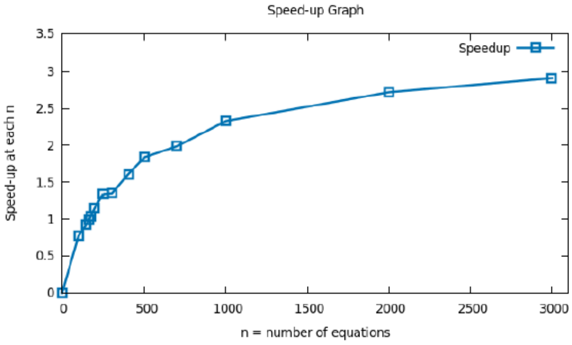


Figure 4: Speed-up for the computation

4. CONCLUSION

We have solved 2D-parabolic heat conduction equation using a Crank-Nicolson method with an Alternating Direction Implicit (ADI) scheme. Thomas Algorithm and cyclic reduction solver have been used to solve the tridiagonal systems. The serial and parallel solutions have been applied on a 2D aluminum plate. We sum up this work with following observations.

1) We have generated graphs for numerical approximations using GNU-plot for time iteration at $t = 10s$, $t = 300s, t = 600s$ and $t = 1000s$. Which is a visual representation of heat conduction through each grid point of $(100 \times 100cm^2)$ 2D aluminum sheet at single time iteration with grid size of $\Delta x = \Delta y = 10cm$.

2) We have developed two FORTRAN codes for Crank Nicolson's method with different tridigonal solvers. We have compared run-time of the two codes for 100 time iterations and various number of equations. The comparison of sequential algorithms of both tridigonal solvers Thomas Algorithm and cyclic reduction. We observed that for small number of equations both performs almost same, however for $n < 160$ numerical approximation code for Crank Nicolson's method with Thomas algorithm runs faster as compare to code with sequential cyclic reduction.
3) For an order of matrix $n \leq 166$ serial method performs better, moreover for $n > 166$ parallel implementation of cyclic reduction accomplishes the task in less time which is evident from the Figure 3. Speedup is the ratio between run-time of scalar and parallel methods. Which provides a clear picture of execution time difference (see: Figure 4). Speedup for $n = 100$ is 0.77 and for $n = 3000$ speedup is 2.91 which is quite remarkable.

REFERENCES

[1] T.N. Narasimhan, Fourier heat conduction: History, influence and connections. In: Rev. Geophys, pp. 151-172, (1999).

[2] J. Crank P. Nicolson, A practical method for numerical evaluation of solutions of heat conduction type. In: Cambridge Philosophical Society, pp. 50-64, (1947).

[3] D. Peaceman, M. Rachford, The numerical solution of parabolic and elliptic differential equations, In: J. SIAM 3, pp. 28-41, (1955).

[4] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical Recipes in FORTRAN 90: The Art of Parallel Scientific Computing. The Pitt Building, Trumpington Street, Cambridge: Cambridge University Press, (1996).

[5] H. A. Luther, B. Carnahan and J. O. Wilkes, Applied Numerical Methods, in: John Wiley and Sons, Inc, pp. 440-446, (1969).

[6] R.W. Hockney. A fast direct solution of Poisson's equation using Fourier analysis, In: Journal of Association or Computing Machinery 12, pp. 95-113, (1965).

[7] R.W. Hockney C.R. Jesshope, Parallel Computers, In: Adam Hilger Ltd (1981).

[8] S. C. Chapra and R. P. Canale, Numerical Methods for Engineers. 2 Penn Plaza, New York, NY 10121: McGraw-Hill Education, pp. 853-885, (2015).

[9] C.R Jesshope, R .W Hockney, Parallel Computers 2. 6000 Broken Sound Parkway
NW, Suite 300 Boca Raton, FL 33487-2742: CRC, Press Taylor Francis Group, pp. 1-6, (2019).

[10] M. Santiago, D. R. Kincaid, Using Cyclic Reduction on a parallel computer to improve the performance of an underwater sound implicit finite difference model, In: Computers Math. Applic. Vol. 21, No. 5, pp. 83-94, (1991).

Received: 13 September 2020. Revised/Accepted: 6 November 2020.

This work is licensed under a Creative Commons Attribution 4.0 International License.