IASJ
المجلات الاكاديمية العراقية

Tikrit Journal of Pure Science
ISSN: 1813 – 1662 (Print) --- E-ISSN: 2415 – 1726 (Online)

Journal Homepage: http://tjps.tu.edu.iq/index.php/j

Hybrid Crank-Nicolson numerical method to solve Heat Diffusion problems

Omar Abdullah Ajeel, Awni M.Gaftan
Department of Mathematics, College of Computer Science and Mathematics, Tikrit University, Tikrit, Iraq

https://doi.org/10.25130/tjps.v28i2.1347

ARTICLE INFO.

Article history:
-Received: **22 / 8 / 2022**
-Accepted: **13 / 9 / 2022**
-Available online: **26 / 4 / 2023**

Keywords: Crank-Nicolson hybrid method, implicit method, Heat−Diffusion, Numerical Method

Corresponding Author:
Name: **Omar Abdullah Ajeel**
E-mail:
omar.alajeel1@gmail.com
awny.muhammed@tu.edu.iq
Tel:

ABSTRACT

In this paper, we derive the hybrid Crank-Nicolson method based on the implicit method and the usual Crank-Nicolson method, where we get more accurate results and faster access to results. We also used Maple to implement the hybrid method and the results were identical

Introduction

The partial differential equation can be defined as an equation whose derivatives include an unknown function for two or more independent variables, and partial differential equations can describe many phenomena, including wave propagation, fluid flow (air or liquid), vibration, solid mechanics, flow and distribution of heat, fields, electric potentials, and the diffusion of chemicals. In air or water, electromagnetism and quantum mechanics [1– 3].

One of the most interesting features of solving partial differential equations using finite difference methods is that many methods based on linear equations with constant coefficients. it is partial derivatives. Thus, finite differences methods can be used, which are classified into three categories: the explicit, implicit, and semi-implicit methods (Crank-Nicolson), and most of the methods have defects in stability and accuracy [4–6]

The first who invented the Crank-Nicolson method were the two scientists John Crank and Phyllis Nicolson in the 1940s, who originally applied it to the heat equation and approached the solution of the heat equation to some finite networks by approximating the derivatives in space x and time t with finite differences, and much earlier, The Richardson devised the Finite Difference Chart which was easy to calculate but was numerically unstable and was not useful[7]. It was not until John Crank, Nicolson Phyllis, and others performed extended numerical computations that the instability of the difference graph was recognized. Therefore, the Crank Nicolson method is numerically stable and requires only solving a very simple system of linear equations at each time level. Thus, the Crank Nicolson method has become one of the most common methods[4], [5], [8], [9] . One of the downsides to the Crank Nicholson method is that it produces a complex system of linear equations despite the availability of approximate solutions. Efficiency.

The implicit method is characterized by being unconditionally stable, the time variable is precisely controlled, and it can extract the value of the time variable with fewer steps. In any case, it is considered one of the difficult methods when calculating two-dimensional and three-dimensional. It has the same number of steps as the explicit method in the first dimension[5], [10]

The aim of this study was to use a hybrid numerical equation, between implicit and Crank-Nicolson method, in order to solve heat diffusion problems and investigate its stability.

Hybrid Method Algorithm

1- Making a Mesh for the solution area.

2- Finding the values of the node points (the intersection of the mesh points) using the (explicit or implicit method).

Explicit formula

$u_j^{k+1} = ru_{j+1}^k + (1-2r)u_j^k + ru_{i-1}^j$ …..1

Implicit formula

$u_j^k = -ru_{j-1}^{k+1} + (1+2r)u_i^{k+1} - ru_{j+1}^{k+1}$…..2

3- Finding the values of the midpoints between every two points on the horizontal axis and the vertical axis using the traditional Crank-Nicolson general formula.

$-ru_{j-1}^{k+1} + 2(1+r)u_j^{k+1} - ru_{j+1}^{k+1} = ru_{j-1}^k + 2(1-r)u_j^k + ru_{j+1}^k$…..3

Now in this section we will implement a hybrid method by on the traditional Crank-Nicolson general formula based on the results of values from nodal points using other differences method (previous results) and then finding the values of the median points using only Crank-Nicolson general formula.

Now we applied the hybrid Crank-Nicolson to solve the heat diffusion equation

$\frac{\partial u}{\partial t} = a\frac{\partial^2 u}{\partial x^2}$ with the following date

- The difference between the units of times is ($\Delta t = 2$) sec.
- The difference on x-axis is ($\Delta x = 2.5cm$).

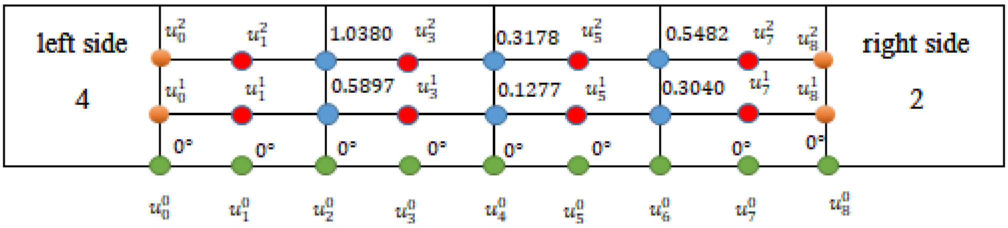
In addition, the constant of diffusions is ($a = 0.625$.) and the initial and Boundary conditions use as bellow

Initial condition $u(x,t) = 0°$ $0 < x < 10$ $t = 0.$

Boundary conditions $u(0,t) = 4°$, $u(10,t) = 2°$

Solution:

First: we use the following mesh for the values of node point from apply the implicit method as bellow [11].



Second: find the midpoint by The Crank-Nicolson general formula equation (3)

$-ru_{j-1}^{k+1} + 2(1+r)u_j^{k+1} - ru_{j+1}^{k+1} = ru_{j-1}^k + 2(1-r)u_j^k + ru_{j+1}^k$

To find r

$r = a\frac{\Delta t}{\Delta x^2} \rightarrow r = 0.625\frac{2}{2.5^2} = 0.2$

Where $\Delta t = k$

We substitute the value of j = 1,3,5,7 and k = 0 and the value of r = 0.2 into equation (3) to find value $u_1^1, u_3^1, u_5^1, u_7^1$

$2.4u_1^1 = 0.2(u_0^0 + u_2^0 + u_0^1 + u_2^1) + 1.6u_1^0$

$2.4u_3^1 = 0.2(u_2^0 + u_4^0 + u_2^1 + u_4^1) + 1.6u_3^0$

$2.4u_5^1 = 0.2(u_4^0 + u_6^0 + u_4^1 + u_6^1) + 1.6u_5^0$

$2.4u_7^1 = 0.2(u_6^0 + u_8^0 + u_6^1 + u_8^1) + 1.6u_7^0$

Substituting the initial boundary and known points

$2.4u_1^1 = 0.2(0 + 0 + 4 + 0.589665654) + 1.6(0)$

$2.4u_1^1 = 0.917933131$

$u_1^1 = 0.382472138$

$2.4u_3^1 = 0.2(0 + 0 + 0.589665654 + 0.127659575) + 1.6(0)$

$2.4u_3^1 = 0.143465046$

$u_3^1 = 0.059777102$

$2.4u_5^1 = 0.2(0 + 0 + 0.127659575 + 0.303951368) + 1.6(0)$

$2.4u_5^1 = 0.086322189$

$u_5^1 = 0.035967579$

$2.4u_7^1 = 0.2(0 + 0 + 0.303951368 + 2) + 1.6(0)$

$2.4u_7^1 = 0.460790274$

$u_7^1 = 0.191995947$

also to Substitute from j = 1,3,5,7 and k = 1 and r = 0.2 value in equation (3) to find $u_1^2, u_3^2, u_5^2, u_7^2$

$2.4u_1^2 = 0.2(u_0^1 + u_2^1 + u_0^2 + u_2^2) + 1.6u_1^1$

$2.4u_3^2 = 0.2(u_2^1 + u_4^1 + u_2^2 + u_4^2) + 1.6u_3^1$

$2.4u_5^2 = 0.2(u_4^1 + u_6^1 + u_4^2 + u_6^2) + 1.6u_5^1$

$2.4u_7^2 = 0.2(u_6^1 + u_8^1 + u_6^2 + u_8^2) + 1.6u_7^1$

Substituting the initial and boundary and known point's values and $u_1^1, u_3^1, u_5^1, u_7^1$ value

$$2.4u_1^2 = 0.2(4 + 0.589665654 + 4 + 1.038017018) + 1.6(0.382472138)$$

$2.4u_1^2 = 1.737491955$

$u_1^2 = 1.057288315$

$2.4u_3^2 = 0.2(0.589665654 + 0.127659575 + 1.038017018 + 0.317790856) + 1.6(0.059777102)$

$2.4u_3^2 = 0.510269984$

$u_3^2 = 0.212612493$

$2.4u_5^2 = 0.2(0.127659575 + 0.303951368 + 0.317790856 + 0.548221099) + 1.6(0.035967579)$

$2.4u_5^2 = 0.317072706$

$u_5^2 = 0.132113628$
$2.4u_7^2 = 0.2(0.303951368 + 2 + 0.548221099 + 2) + 1.6(0.191995947)$
$2.4u_7^2 = 1.277628009$
$u_7^2 = 0.532345004$

As well as applying the hybrid method algorithm using the Maple, program

Example:

Let us have rectangular plate by the dimensions (2*4m) and it is considered on x and y-axis at the original point:

Where $\Delta t = 2\ sec$, $\Delta x = 2.5cm$, $a = 0.625$.

Initial condition $u(x\ ,t) = 0°$ $0 < x < 10$ $t = 0$.

Boundary conditions $u(\,0\,,t) = 4°$, $u(\,10\,,t\,) = 2°$

Solution

step : 1. *where* $k = 0$

1. *where* $k = 0$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=0\,, j=1\,, r=0.2\}\right)$$

$$\{2.4\,u_{1,\,1} = 0.2\,u_{0,\,0} + 1.6\,u_{1,\,0} + 0.2\,u_{2,\,0} + 0.2\,u_{0,\,1} + 0.2\,u_{2,\,1}\}$$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=0\,, j=3\,, r=0.2\}\right)$$

$$\{2.4\,u_{3,\,1} = 0.2\,u_{2,\,0} + 1.6\,u_{3,\,0} + 0.2\,u_{4,\,0} + 0.2\,u_{2,\,1} + 0.2\,u_{4,\,1}\}$$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=0\,, j=5\,, r=0.2\}\right)$$

$$\{2.4\,u_{5,\,1} = 0.2\,u_{4,\,0} + 1.6\,u_{5,\,0} + 0.2\,u_{6,\,0} + 0.2\,u_{4,\,1} + 0.2\,u_{6,\,1}\}$$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=0\,, j=7\,, r=0.2\}\right)$$

$$\{2.4\,u_{7,\,1} = 0.2\,u_{6,\,0} + 1.6\,u_{7,\,0} + 0.2\,u_{8,\,0} + 0.2\,u_{6,\,1} + 0.2\,u_{8,\,1}\}$$

$$eval(\{2.4\,u_{1,\,1} = 0.2\,u_{0,\,0} + 1.6\,u_{1,\,0} + 0.2\,u_{2,\,0} + 0.2\,u_{0,\,1} + 0.2\,u_{2,\,1}, 2.4\,u_{3,\,1} = 0.2\,u_{2,\,0} + 1.6\,u_{3,\,0} + 0.2\,u_{4,\,0} + 0.2\,u_{2,\,1} + 0.2\,u_{4,\,1}, 2.4\,u_{5,\,1} = 0.2\,u_{4,\,0} + 1.6\,u_{5,\,0} + 0.2\,u_{6,\,0} + 0.2\,u_{4,\,1} + 0.2\,u_{6,\,1}, 2.4\,u_{7,\,1} = 0.2\,u_{6,\,0} + 1.6\,u_{7,\,0} + 0.2\,u_{8,\,0} + 0.2\,u_{6,\,1} + 0.2\,u_{8,\,1}\}, \{u_{0,\,0} = 0\,, u_{1,\,0} = 0, u_{2,\,0} = 0, u_{3,\,0} = 0, u_{4,\,0} = 0, u_{5,\,0} = 0, u_{6,\,0} = 0, u_{7,\,0} = 0, u_{8,\,0} = 0, u_{0,\,1} = 4, u_{0,\,2} = 4, u_{0,\,3} = 4, u_{8,\,1} = 2, u_{8,\,2} = 2, u_{8,\,3} = 2, u_{2,\,1} = 0.5896656535, u_{4,\,1} = 0.1276595745, u_{6,\,1} = 0.3039513678, u_{2,\,2} = 1.0380170176, u_{4,\,2} = 0.3177908556, u_{6,\,2} = 0.5482210992, u_{2,\,3} = 1.3890208804, u_{4,\,3} = 0.5330610751, u_{6,\,3} = 0.7534523673\})$$

$$\{2.4\,u_{1,\,1} = 0.9179331307, 2.4\,u_{3,\,1} = 0.1434650456, 2.4\,u_{5,\,1} = 0.08632218846, 2.4\,u_{7,\,1} = 0.4607902736\}$$

$$solve(\{2.4\,u_{1,\,1} = 0.9179331307, 2.4\,u_{3,\,1} = 0.1434650456, 2.4\,u_{5,\,1} = 0.08632218846, 2.4\,u_{7,\,1} = 0.4607902736\}, \{u_{1,\,1}, u_{3,\,1}, u_{5,\,1}, u_{7,\,1}\})$$

$$\{u_{1,\,1} = 0.3824721378, u_{3,\,1} = 0.05977710233, u_{5,\,1} = 0.03596757852, u_{7,\,1} = 0.1919959473\}$$

step : 1. *where* $k = 1$

1. *where* $k = 1$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=1\,, j=1\,, r=0.2\}\right)$$

$$\{2.4\,u_{1,\,2} = 0.2\,u_{0,\,1} + 1.6\,u_{1,\,1} + 0.2\,u_{2,\,1} + 0.2\,u_{0,\,2} + 0.2\,u_{2,\,2}\}$$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=1\,, j=3\,, r=0.2\}\right)$$

$$\{2.4\,u_{3,\,2} = 0.2\,u_{2,\,1} + 1.6\,u_{3,\,1} + 0.2\,u_{4,\,1} + 0.2\,u_{2,\,2} + 0.2\,u_{4,\,2}\}$$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=1\,, j=5\,, r=0.2\}\right)$$

$$\{2.4\,u_{5,\,2} = 0.2\,u_{4,\,1} + 1.6\,u_{5,\,1} + 0.2\,u_{6,\,1} + 0.2\,u_{4,\,2} + 0.2\,u_{6,\,2}\}$$

$$eval\left(\left\{+2\,(1+r)\mathrm{u}_{j,\,k+1} = \mathrm{r\,u}_{j-1,\,k} + 2(1-r)\mathrm{u}_{j,\,k} + \mathrm{r\,u}_{j+1,\,k} + \mathrm{r\,u}_{j-1,\,k+1} + \mathrm{r\,u}_{j+1,\,k+1}\right\}, \{k=1\,, j=7\,, r=0.2\}\right)$$

$$\{2.4\,u_{7,\,2} = 0.2\,u_{6,\,1} + 1.6\,u_{7,\,1} + 0.2\,u_{8,\,1} + 0.2\,u_{6,\,2} + 0.2\,u_{8,\,2}\}$$

$$eval(\{2.4\,u_{1,\,2} = 0.2\,u_{0,\,1} + 1.6\,u_{1,\,1} + 0.2\,u_{2,\,1} + 0.2\,u_{0,\,2} + 0.2\,u_{2,\,2}, 2.4\,u_{3,\,2} = 0.2\,u_{2,\,1} + 1.6\,u_{3,\,1} + 0.2\,u_{4,\,1} + 0.2\,u_{2,\,2} + 0.2\,u_{4,\,2}, 2.4\,u_{5,\,2} = 0.2\,u_{4,\,1} + 1.6\,u_{5,\,1} + 0.2\,u_{6,\,1} + 0.2\,u_{4,\,2} + 0.2\,u_{6,\,2}, 2.4\,u_{7,\,2} = 0.2\,u_{6,\,1} + 1.6\,u_{7,\,1} + 0.2\,u_{8,\,1} + 0.2\,u_{6,\,2} + 0.2\,u_{8,\,2}\}, \{u_{0,\,0} = 0\,, u_{1,\,0} = 0, u_{2,\,0} = 0, u_{3,\,0} = 0, u_{4,\,0} = 0, u_{5,\,0} = 0, u_{6,\,0} = 0, u_{7,\,0} = 0, u_{8,\,0} = 0, u_{0,\,1} = 4, u_{0,\,2} = 4, u_{0,\,3} = 4, u_{8,\,1} = 2, u_{8,\,2} = 2, u_{8,\,3} = 2, u_{2,\,1} = 0.5896656535, u_{4,\,1} = 0.1276595745, u_{6,\,1} = 0.3039513678, u_{2,\,2} = 1.0380170176, u_{4,\,2} = 0.3177908556, u_{6,\,2} = 0.5482210992, u_{2,\,3} = 1.3890208804, u_{4,\,3} = 0.5330610751, u_{6,\,3} = 0.7534523673, u_{1,\,1} = 0.3824721378, u_{3,\,1} = 0.05977710233, u_{5,\,1} = 0.03596757852, u_{7,\,1} = 0.1919959473\})$$

$$\{2.4\,u_{1,\,2} = 2.537491955, 2.4\,u_{3,\,2} = 0.5102699840, 2.4\,u_{5,\,2} = 0.3170727050, 2.4\,u_{7,\,2} = 1.277628009\}$$

$$solve(\{2.4\,u_{1,\,2} = 2.537491955, 2.4\,u_{3,\,2} = 0.5102699840, 2.4\,u_{5,\,2} = 0.3170727050, 2.4\,u_{7,\,2} = 1.277628009\}, \{u_{1,\,2}, u_{3,\,2}, u_{5,\,2}, u_{7,\,2}\})$$

$$\{u_{1,\,2} = 1.057288315, u_{3,\,2} = 0.2126124933, u_{5,\,2} = 0.1321136271, u_{7,\,2} = 0.5323450038\}$$

Table 1: comparison of the results of the Crank-Nicolson hybrid method with the Crank-Nicolson hybrid method using Maple

Crank-Nicolson Hybrid	Crank-Nicolson Hybrid with maple
0.382472138	0.3824721378
0.059777102	0.05977710233
0.035967579	0.03596757852
0.191995947	0.1919959473
1.057288315	1.057288315
0.212612493	0.2126124933
0.132113628	0.1321136271
0.532345004	0.5323450038

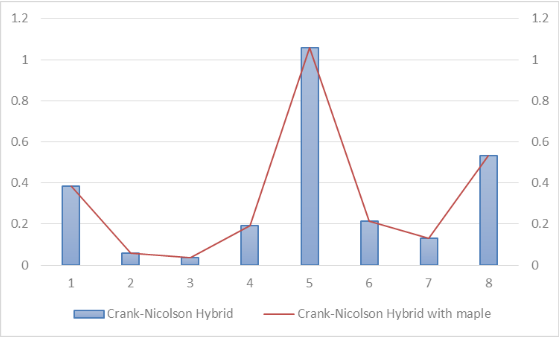


Diagram of matching results of the Crank-Nicolson hybrid method with the Crank-Nicolson hybrid method using Maple

Conclusion

The results of the solution showed that the hybrid method algorithm saved effort and time because one of the negative indicators of Crank-Nicolson method is the complexity of the solution.

Resources

[1] A. Kaw and L. Snyder, “Introduction to scientific computing,” 2000.

[2] S. A. Manna and B. J. Salem, “Numerical Analysis of the Burger Equation using Finite Differences,” *AL-Rafidain Journal of Computer Sciences and Mathematics*, vol. 4, no. 1, pp. 119–132, 2007.

[3] A. R. Mitchell and D. F. Griffiths, “The finite difference method in partial differential equations,” *A Wiley-Interscience Publication*, 1980.

[4] S. Sathyapriya, G. Hamsavarthini, M. Meenakshi, and E. Tanushree, “A Study on Crank Nicolson Method for Solving Parabolic Partial Differential Equations,” 2021.

[5] Q. Feng and B. Zheng, “Alternating Group Explicit-Implicit Method And Crank-Nicolson Method For Convection-Diffusion Equation,” 2009.

[6] K. Kakuda and N. Tosaka, “The generalized boundary element approach to Burgers’ equation,” *Int J Numer Methods Eng*, vol. 29, no. 2, pp. 245–261, 1990.

[7] T. Cebeci, *Convective heat transfer*. Springer, 2002.

[8] C. E. Abhulimen and B. J. Omowo, “Modified Crank-Nicolson Method for solving one dimensional Parabolic equations,” *IOSR Journal of Mathematics, Volume15*, no. 6, pp. 60–66, 2019.

[9] D. J. Duffy, “A Critique of the Crank Nicolson Scheme Strengths and Weaknesses for Financial Instrument Pricing,” 2004.

[10] D. Britz and J. Strutwolf, “The Explicit Method,” 2016, pp. 89–100. doi: 10.1007/978-3-319-30292-8_5.

[11] Bushra shakir mahmood, “Using the Theta Differences Method for solving the Bi-harmonic Equation,” thesis, University of Tikrit, iraq, 2019.

الطريقة الهجينة كرانك– نيكلسون لحل مسائل انتشار الحرارة

عمر عبدالله عجيل ، عوني محمد كفطان

قسم الرياضيات ، كلية علوم الحاسوب والرياضيات ، جامعة تكريت ، تكريت ، العراق

الملخص

في هذا البحث، نشتق طريقة Crank-Nicolson الهجينة بناءً على الطريقة الضمنية وطريقة Crank-Nicolson المعتادة، حيث نحصل على نتائج أكثر دقة ووصول أسرع إلى النتائج. استخدمنا أيضًا Maple لتنفيذ الطريقة الهجينة وكانت النتائج متطابقة بنسبة 98%.