American Journal of Computational Mathematics, **2011, 1, 285-293**
doi:10.4236/ajcm.2011.14035 Published Online December 2011 (http://www.SciRP.org/journal/ajcm)

An Efficient Direct Method to Solve the Three Dimensional Poisson's Equation

Alemayehu Shiferaw, Ramesh Chand Mittal
Department of Mathematics, *Indian Institute of Technology*, *Roorkee*, *India*
E-mail: *abelhaim@gmail.com*, *mittalrc@iitr.ernet.in*
Received August 24, 2011; *revised October* 11, 2011; *accepted October* 20, 2011

Abstract

In this work, the three dimensional Poisson's equation in Cartesian coordinates with the Dirichlet's boundary conditions in a cube is solved directly, by extending the method of Hockney. The Poisson equation is approximated by 19-points and 27-points fourth order finite difference approximation schemes and the resulting large algebraic system of linear equations is treated systematically in order to get a block tri-diagonal system. The efficiency of this method is tested for some Poisson's equations with known analytical solutions and the numerical results obtained show that the method produces accurate results. It is shown that 19-point formula produces comparable results with 27-point formula, though computational efforts are more in 27-point formula.

Keywords: Poisson's Equation, Finite Difference Method, Tri-Diagonal Matrix, Hockney's Method, Thomas Algorithm

1. Introduction

Poisson's equation in three dimensional Cartesian coordinates system plays an important role due to its wide range of application in areas like ideal fluid flow, heat conduction, elasticity, electrostatics, gravitation and other science fields especially in physics and engineering. For Dirichlet's and mixed boundary conditions, the solution of Poisson's equation exists and it is unique. Using some existing methods like variable separable or Green's function we can find the solutions of Poisson's equation analytically even though at times it is difficult and tedious from the point view of practical applications for some boundary conditions [1-4]. For further applications, it seems very plausible to treat numerically in order to obtain good and accurate solution of Poisson's equation. The advantages of numerical treatment is to reduce complexities of the problem, secure more accurate results and use modern computers for further analysis [1,2,5].

If possible, direct methods are certainly preferable to iterative methods when several sets of equations with the same coefficients matrix but different right-hand sides have to be solved. It is well known that direct methods solve the system of equations in a known number of arithmetic operations, and errors in the solution arise entirely from rounding-off errors introduced during the computation [1,5-7].

Researchers in this area have tried to solve Poisson's equation numerically by transforming the partial differential equation to its equivalent finite difference (or finite element or others) approximation to get in terms of an algebraic equation. When we approximate the Poisson's equation by its finite difference approximation, in fact, we obtain a large number of system of linear equations [2,5-7]. In order to solve the two dimensional Poisson's equation numerically several attempts have been made, Hockney [8] has devised an efficient direct method which uses the reduction process, Buneman developed an efficient direct method for solving the reduced system of equations.[5,6,9(unpublished)]; Buzbee *et al*. [10] developed an efficient and accurate direct methods to solve certain elliptic partial difference equations over a rectangle with Dirichlet's, Neumann or periodic boundary conditions; Averbuch *et al*. [11] on a rectangular domain and McKenney *et al*. [12] on complex geometries have developed a fast Poisson Solver. The fast Fourier transform can also be used to compute the solution to the discrete system very efficiently provided that the number of mesh points in each dimension is a power of small prime (This technique is the basis for several "fast Poisson solver" software packages) [7]. Skolermo [13] has developed a method based on the relation between the Fourier coefficients for the solution and those for the right-hand side. In this method the Fast Fourier Transform is used for the computation and its influence on the accuracy of the solution. Greengard and Lee [14] have developed a direct, adaptive solver for the Poisson equation which can achieve any prescribed order of accuracy. Their method is based on a domain decomposition approach using local spectral approximation, as well as potential theory and the fast multipole method.

To solve the three dimensional Poisson's equations in Cartesian coordinate systems using finite difference approximations; for instance, Spotz and Carey [15] have developed an approximation using central difference scheme to obtain a 19-point stencil and a 27-point stencil with some modification on the right hand side terms; Braverman *et al*. [16] established an arbitrary order accuracy fast 3D Poisson Solver on a rectangular box and their method is based on the application of the discrete Fourier transform accompanied by a subtraction technique which allows reducing the errors associated with the Gibbs phenomenon; Sutmann and Steffen [17] have developed compact approximation schemes for the Laplace operator of fourth- and sixth-order based on Padé approximation of the Taylor expansion for the discretized Laplace operator; Jun Zhang [18] has developed a multigrid solution for Poisson's equation and their finite difference approximation is based on uniform mesh size and they have solved the resulting system of linear equations by a residual or multigrid method.

The aim of this paper is to develop a fourth order finite difference approximation schemes and the resulting large algebraic system of linear equations is treated systematiccally in order to get a block tri-diagonal system [19] and extend the Hockney's method to solve the three dimensional Poisson's equation on Cartesian coordinate systems. It is shown that the discussed method produces very good results. It is found that, in general, 27-points scheme produces better results than 19-points scheme but 19-point scheme also shows comparable results.

2. Finite Difference Approximation

Consider the Poisson equation

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = f(x,y,z) \quad \text{on} \quad D$$

and

$$u = g(x,y,z) \quad \text{on} \quad C \tag{1}$$

where $D = \{(x,y,z): 0 < x < a, 0 < y < b, 0 < z < c\}$ and C is the boundary of D.

Let the mesh size along the X-direction and Y-direction be h_1, and along the Z-direction be h_2 (h_1 and h_2 need not be equal).

Let δ_x be the central difference operator, and we know that

$$\begin{aligned}
\frac{\partial^2 u}{\partial x^2} &= \frac{\delta_x^2}{h_1^2\left(1+\frac{1}{12}\delta_x^2\right)} + O\left(h_1^4\right) \\
\frac{\partial^2 u}{\partial y^2} &= \frac{\delta_y^2}{h_1^2\left(1+\frac{1}{12}\delta_y^2\right)} + O\left(h_1^4\right) \\
\frac{\partial^2 u}{\partial z^2} &= \frac{\delta_z^2}{h_1^2\left(1+\frac{1}{12}\delta_z^2\right)} + O\left(h_2^4\right)
\end{aligned} \tag{2}$$

Using (2) in (1), we have

$$\left(\frac{\delta_x^2}{h_1^2\left(1+\frac{1}{12}\delta_x^2\right)} + \frac{\delta_y^2}{h_1^2\left(1+\frac{1}{12}\delta_y^2\right)} + \frac{\delta_z^2}{h_2^2\left(1+\frac{1}{12}\delta_z^2\right)} + O\left(h_1^4\right) + O\left(h_2^4\right)\right) u_{i,j,k} = f_{i,j,k} \tag{3}$$

where $i = 1,2,3,\cdots,m$, $j = 1,2,3,\cdots,n$ and $k = 1,2,3,\cdots,p$

Letting $r = \frac{h_1^2}{h_2^2}$, simplifying and neglecting the truncation error in (3), we get

$$\begin{aligned}
h_1^2 f_{i,j,k} = \Bigg(& \frac{\delta_x^2\left(1+\frac{1}{12}\delta_y^2\right)\left(1+\frac{1}{12}\delta_z^2\right)}{\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_y^2\right)\left(1+\frac{1}{12}\delta_z^2\right)} \\
&+ \frac{\delta_y^2\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_z^2\right)}{\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_y^2\right)\left(1+\frac{1}{12}\delta_z^2\right)} \\
&+ \frac{r\delta_z^2\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_y^2\right)}{\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_y^2\right)\left(1+\frac{1}{12}\delta_z^2\right)} \Bigg) u_{i,j,k}
\end{aligned} \tag{4}$$

$$\begin{aligned}
& h_1^2\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_y^2\right)\left(1+\frac{1}{12}\delta_z^2\right) f_{i,j,k} \\
&= \delta_x^2\left(1+\frac{1}{12}\delta_y^2\right)\left(1+\frac{1}{12}\delta_z^2\right) + \delta_y^2\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_z^2\right) \\
&\quad + r\delta_z^2\left(1+\frac{1}{12}\delta_x^2\right)\left(1+\frac{1}{12}\delta_y^2\right)
\end{aligned}$$

$$h_1^2\left(1+\frac{1}{12}\left(\delta_x^2+\delta_y^2+\delta_z^2\right)+\frac{1}{144}\left(\delta_x^2\delta_y^2+\delta_x^2\delta_z^2+\delta_y^2\delta_z^2\right)+\frac{1}{1728}\delta_x^2\delta_y^2\delta_z^2\right)f_{i,j,k}$$
$$=\left(\left(\delta_x^2+\delta_y^2+r\delta_z^2\right)+\frac{1}{6}\delta_x^2\delta_y^2+\frac{1+r}{12}\left(\delta_x^2\delta_z^2+\delta_y^2\delta_z^2\right)+\frac{2+r}{144}\delta_x^2\delta_y^2\delta_z^2\right)u_{i,j,k} \quad (5)$$

On simplifying (5),

***Scheme* 1**

Considering all the terms of (5), we obtain

$$h_1^2\left(144+12\left(\delta_x^2+\delta_y^2+\delta_z^2\right)+\left(\delta_x^2\delta_y^2+\delta_y^2\delta_z^2+\delta_x^2\delta_z^2\right)+\frac{1}{12}\delta_x^2\delta_y^2\delta_z^2\right)f_{i,j,k}$$
$$=-(400+200r)u_{i,j,k}+(100r-40)\left(u_{i,j,k-1}+u_{i,j,k+1}\right)$$
$$+(80-20r)\left(u_{i-1,j,k}+u_{i+1,j,k}+u_{i,j-1,k}+u_{i,j+1,k}\right)$$
$$+(20-2r)\left(u_{i-1,j-1,k}+u_{i-1,j+1,k}+u_{i+1,j-1,k}+u_{i+1,j+1,k}\right) \quad (6a)$$
$$+(8+10r)\left(u_{i-1,j,k-1}+u_{i+1,j,k-1}+u_{i-1,j,k+1}+u_{i+1,j,k+1}+u_{i,j-1,k-1}+u_{i,j+1,k-1}+u_{i,j-1,k+1}+u_{i,j+1,k+1}\right)$$
$$(2+r)\left(u_{i+1,j-1,k-1}+u_{i-1,j-1,k-1}+u_{i+1,j+1,k-1}+u_{i+1,j+1,k+1}+u_{i-1,j-1,k+1}+u_{i+1,j-1,k+1}+u_{i-1,j+1,k-1}+u_{i-1,j+1,k+1}\right)$$

This is a 27-point stencil scheme.

***Scheme* 2**

By omitting the term $\left((2+r)/(144)\right)\delta_x^2\delta_y^2\delta_z^2$ in the left side and taking only the first and second terms in the right side of (5) and simplifying it, we get

$$12h_1^2\left(1+\frac{1}{12}\left(\delta_x^2+\delta_y^2+\delta_z^2\right)\right)f_{i,j,k}$$
$$=-(32+16r)u_{i,j,k}+(8r-4)\left(u_{i,j,k-1}+u_{i,j,k+1}\right)$$
$$+(6-2r)\left(u_{i-1,j,k}+u_{i+1,j,k}+u_{i,j-1,k}+u_{i,j+1,k}\right) \quad (6b)$$
$$+2\left(u_{i-1,j-1,k}+u_{i-1,j+1,k}+u_{i+1,j-1,k}+u_{i+1,j+1,k}\right)$$
$$(1+r)\left(u_{i-1,j,k-1}+u_{i+1,j,k-1}+u_{i-1,j,k+1}+u_{i+1,j,k+1}+u_{i,j-1,k-1}+u_{i,j+1,k-1}+u_{i,j-1,k+1}+u_{i,j+1,k+1}\right)$$

and this is a 19-point stencil scheme.

The Poisson's Equation (1) now is approximated by its equivalent systems of linear equations either (6a) or (6b) and these equations now will be treated in order to form a block tri-diagonal matrix. We can find the eigenvalues and eigenvectors of these block tri-diagonal matrices easily.

Now we solve these two different systems of linear equations systematically.

Taking first in the X-direction, next Y-direction and lastly Z-direction in (6a) and (6b), we get a large system of linear equations (the number of equations actually depends on the values of m, n and p); and this system of equations can be written in matrix form (for both schemes) as

$$AU=\mathrm{B} \quad (7)$$

where

$$A=\begin{pmatrix} R & S & & & & \\ S & R & S & & & \\ & S & R & S & & \\ & & & \ddots & & \\ & & & S & R & S \\ & & & & S & R \end{pmatrix} \quad (8)$$

it has p blocks and each block is of order $mn\times mn$

$$R=\begin{pmatrix} R_1 & R_2 & & & & \\ R_2 & R_1 & R_2 & & & \\ & R_2 & R_1 & R_2 & & \\ & & & \ddots & & \\ & & & R_2 & R_1 & R_2 \\ & & & & R_2 & R_1 \end{pmatrix},$$

$$S=\begin{pmatrix} S_1 & S_2 & & & & \\ S_2 & S_1 & S_2 & & & \\ & S_2 & S_1 & S_2 & & \\ & & & \ddots & & \\ & & & S_2 & S_1 & S_2 \\ & & & & S_2 & S_1 \end{pmatrix}$$

R and S have n blocks and each block is of order $m\times m$ where for the Scheme 1

$$R_1=\begin{pmatrix} -400-200r & 80-20r & & & & \\ 80-20r & -400-200r & 80-20r & & & \\ & 80-20r & -400-200r & 80-20r & & \\ & & & \ddots & & \\ & & & 80-20r & -400-200r & 80-20r \\ & & & & 80-20r & -400-200r \end{pmatrix}$$

$$R_2 = \begin{pmatrix} 80-20r & 20-2r & & & & & \\ 20-2r & 80-20r & 20-2r & & & & \\ & 20-2r & 80-20r & 20-2r & & & \\ & & & \ddots & & & \\ & & & & 20-2r & 80-20r & 20-2r \\ & & & & & 20-2r & 80-20r \end{pmatrix}$$

$$S_1 = \begin{pmatrix} 100r-40 & 8-10r & & & & & \\ 8-10r & 100r-40 & 8-10r & & & & \\ & 8-10r & 100r-40 & 8-10r & & & \\ & & & \ddots & & & \\ & & & & 8-10r & 100r-40 & 8-10r \\ & & & & & 8-10r & 100r-40 \end{pmatrix}$$

$$S_2 = \begin{pmatrix} 8-10r & 2+r & & & & & \\ 2+r & 8-10r & 2+r & & & & \\ & 2+r & 8-10r & 2+r & & & \\ & & & \ddots & & & \\ & & & & 2+r & 8-10r & 2+r \\ & & & & & 2+r & 8-10r \end{pmatrix}$$

and for Scheme 2

$$R_1 = \begin{pmatrix} -32-16r & 6-2r & & & & & \\ 6-2r & -32-16r & 6-2r & & & & \\ & 6-2r & -32-16r & 6-2r & & & \\ & & & \ddots & & & \\ & & & & 6-2r & -32-16r & 6-2r \\ & & & & & 6-2r & -32-16r \end{pmatrix}$$

$$R_2 = \begin{pmatrix} 6-2r & 2 & & & & & \\ 2 & 6-2r & 2 & & & & \\ & 2 & 6-2r & 2 & & & \\ & & & \ddots & & & \\ & & & & 2 & 6-2r & 2 \\ & & & & & 2 & 6-2r \end{pmatrix}$$

$$S_1 = \begin{pmatrix} 8r-4 & 1+r & & & & & \\ 1+r & 8r-4 & 1+r & & & & \\ & 1+r & 8r-4 & 1+r & & & \\ & & & \ddots & & & \\ & & & & 1+r & 8r-4 & 1+r \\ & & & & & 1+r & 8r-4 \end{pmatrix}$$

and

$$S_2 = \begin{pmatrix} 1+r & & & & \\ & 1+r & & & \\ & & 1+r & & \\ & & & \ddots & \\ & & & & 1+r \end{pmatrix}$$

$$u = \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ \cdots \\ U_p \end{pmatrix} \text{ and } b = \begin{pmatrix} B_1 \\ B_2 \\ B_3 \\ \cdots \\ B_p \end{pmatrix}$$

where

$$U_k = (u_{11k} \quad u_{21k} \quad \cdots \quad u_{m1k} \quad u_{12k} \quad u_{22k} \quad \cdots \quad u_{m2k} \quad \cdots \quad u_{1nk} \quad u_{2nk} \quad \cdots \quad u_{mnk})^T$$

and b the known column vector B_k where

$$B_k = (b_{11k} \quad b_{21k} \quad \ldots \quad b_{m1k} \quad b_{12k} \quad b_{22k} \quad \ldots \quad b_{m2k} \quad \cdots \quad b_{1nk} \quad b_{2nk} \quad \cdots \quad b_{mnk})^T$$

such that each b_{ijk} represents known boundary values of u and values of f.

3. Extended Hockney's Method

Let q_i be an eigenvector of R_1, R_2, S_1 and S_2 corresponding to the eigenvalues η_i, τ_i, α_i and β_i respectively, and Q be the modal matrix $[q_1, q_2, q_3, \cdots, q_m]$ of the matrix R_1, R_2, S_1 and S_2 of order m such that, $Q^T Q = I$

$$Q^T R_1 Q = \operatorname{diag}(\eta_1, \eta_2, \eta_3, \cdots, \eta_n)$$

$$Q^T R_2 Q = \operatorname{diag}(\tau_1, \tau_2, \tau_3, \cdots, \tau_n)$$

$$Q^T S_1 Q = \operatorname{diag}(\alpha_1, \alpha_2, \alpha_3, \cdots, \alpha_n)$$

and

$$Q^T S_2 Q = \operatorname{diag}(\beta_1, \beta_2, \beta_3, \cdots, \beta_n)$$

Note that the eigenvalues of R_1, R_2, S_1 and S_2, for Scheme 1 are

$$\eta_i = -400 - 200r + 2(80 - 20r)\cos\left(\frac{i\pi}{m+1}\right)$$

$$\tau_i = 80 - 20r + 2(20 - 2r)\cos\left(\frac{i\pi}{m+1}\right)$$

$$\alpha_i = 100r - 40 + 2(8 - 10r)\cos\left(\frac{i\pi}{m+1}\right)$$

$$\beta_i = 8 - 10r + 2(2 + r)\cos\left(\frac{i\pi}{m+1}\right)$$

and for scheme 2

$$\eta_i = -32 - 16r + 2(6 - 2r)\cos\left(\frac{i\pi}{m+1}\right)$$

$$\tau_i = 6 - 2r + 4\cos\left(\frac{i\pi}{m+1}\right)$$

$$\alpha_i = 8r - 4 + 2(1 + r)\cos\left(\frac{i\pi}{m+1}\right)$$

$$\beta_i = 1 + r$$

Let $\mathbb{Q} = \operatorname{diag}(Q, Q, \cdots, Q)$ is a matrix of order $mn \times mn$

Thus $\mathbb{Q}$ satisfy $\mathbb{Q}^T \mathbb{Q} = I$

$$\mathbb{Q}^T R \mathbb{Q} = \operatorname{diag}(\psi_1, \psi_2, \psi_3, \cdots, \psi_n) = \Lambda \text{ (say)}$$

where

$$\psi_i = \operatorname{diag}(\lambda_1, \lambda_2, \lambda_3, \cdots, \lambda_m)$$

and

$$\mathbb{Q}^T \mathbb{S} \mathbb{Q} = \operatorname{diag}(\mathrm{T}_1, \mathrm{T}_2, \mathrm{T}_3, \cdots, \mathrm{T}_n) = \Upsilon \text{ (say)}$$

where

$$\mathrm{T}_i = \operatorname{diag}(\mu_1, \mu_2, \mu_3, \cdots, \mu_m)$$

Here

$$\lambda_i = \eta_i + 2\tau_i \cos\left(\frac{i\pi}{m+1}\right)$$

and

$$\mu_i = \alpha_i + 2\beta_i \cos\left(\frac{i\pi}{m+1}\right)$$

Let

$$\begin{aligned} \mathbb{Q}^T U_k &= V_k \Rightarrow U_k = \mathbb{Q} V_k \\ \mathbb{Q}^T B_k &= \bar{B}_k \Rightarrow B_k = \mathbb{Q} \bar{B}_k \end{aligned} \tag{9}$$

where

$$V_k = (v_{11k} \quad v_{21k} \quad \cdots \quad v_{m1k} \quad v_{12k} \quad v_{22k} \quad \cdots \quad v_{m2k} \quad \cdots \quad v_{1nk} \quad v_{2nk} \quad \cdots \quad v_{mnk})^T$$

in order to find the solutions of Equation (1), we solve Equations (6a) and (6b).

Consider Equation (7) and using (8) we can write it in terms of the matrices R and S as

$$\begin{aligned} RU_1 + SU_2 &= B_1 \\ SU_1 + RU_2 + SU_3 &= B_2 \\ SU_2 + RU_3 + SU_4 &= B_3 \\ SU_{p-1} + RU_p &= B_p \end{aligned} \tag{10}$$

Pre multiplying (10) by $\mathbb{Q}^T$ and using (9), we get

$$\begin{aligned} \Lambda V_1 + \Upsilon V_2 &= \bar{B}_1 \\ \Upsilon V_1 + \Lambda V_2 + \Upsilon V_3 &= \bar{B}_2 \\ \Upsilon V_2 + \Lambda V_3 + \Upsilon V_4 &= \bar{B}_3 \\ \Upsilon V_{p-1} + \Lambda V_p &= \bar{B}_p \end{aligned} \tag{11}$$

Each equation of (11) again can be written as

$$\lambda_i v_{ij1} + \mu_i v_{ij2} = \overline{b}_{ij1}$$
$$\mu_i v_{ij1} + \lambda_i v_{ij2} + \mu_i v_{ij3} = \overline{b}_{ij2}$$
$$\mu_i v_{ij2} + \lambda_i v_{ij3} + \mu_i v_{ij4} = \overline{b}_{ij3} \quad (12)$$
$$\mu_i v_{ij(p-1)} + \lambda_i v_{ijp} = \overline{b}_{ijp}$$

For $i = 1, 2, 3, \cdots, m$, $j = 1, 2, 3, \cdots, n$ and $k = 1, 2, 3, \cdots, p$.

For $k = 1, 2, 3, \cdots, p$ collect now from each equation of (12), the first equations *i.e.* for $i = 1, j = 1$, and get

$$\mu_1 v_{11(k-1)} + \lambda_1 v_{11k} + \mu_1 v_{11(k+1)} = \overline{b}_{11k} \quad (13a)$$

Again collect the equations from (12) for $i = 2, j = 1$ and get

$$\mu_2 v_{21(k-1)} + \lambda_2 v_{21k} + \mu_2 v_{21(k+1)} = \overline{b}_{21k} \quad (13b)$$

Continuing in the same fashion and collecting the equations at some point of i, j, we get

$$\mu_i v_{ij(k-1)} + \lambda_i v_{ijk} + \mu_i v_{ij(k+1)} = \overline{b}_{ijk} \quad (13c)$$

And collecting the last equations (*i.e.* for $i = m$, $j = n$), we get

$$\mu_m v_{mn(k-1)} + \lambda_m v_{mnk} + \mu_m v_{mn(k+1)} = \overline{b}_{mnk} \quad (13d)$$

All these set of Equations (13a)-(13d) are tri-diagonal ones and hence we solve for v_{ijk} by using Thomas algorithm, and with the help (9) again we get all u_{ijk} and this solves (6a) and (6b) as desired.

By doing this we generally reduce the number of computations and computational time.

4. Algorithm

1) Approximate the Poisson equation's by its fourth order finite difference approximations scheme

2) Calculate the eigenvalues and eigenvectors of the block tri-diagonal matrices;

3) Find the modal matrix Q and $\mathbb{Q}$;

4) Pre multiply R, S and B_k by $\mathbb{Q}^T$ and get systems of linear equations;

5) Solve the system by using Thomas algorithm;

6) Calculate back for $u_{i,j,k}$.

Since we used finite difference approximation to approximate the Poisson equation's and this method is direct one, it is sure that the error in the solution arises only from rounding off errors. By doing this we generally reduce the number of computations and computational time.

5. Numerical Results

A computational experiment is done on six selected examples for both schemes in which the analytical solutions of u are known to us in order to test the efficiency and adaptability of the proposed method. The computed solution is found for the entire interior grid points but results are reported with regard to the maximum absolute errors for corresponding choice of m, n, p and the computed solutions are given in **Tables 1-6**.

Example 1. Suppose $\nabla^2 u = 0$ with the boundary conditions

$$u(0, y, z) = u(x, 0, z) = u(x, y, 0)$$
$$= u(x, y, 1) = u(1, y, z) = u(x, 1, z) = 1$$

The analytical solution is $u(x, y, z) = 1$ and its results are shown in **Table 1** below.

Example 2. Consider $\nabla^2 u = 0$ with

$$u(0, y, z) = u(x, 0, z) = u(x, y, 0) = 0$$
$$u(1, y, z) = yz, u(x, 1, z) = xz, u(x, y, 1) = xy$$

The analytical solution is $u(x, y, z) = xyz$ and its results are shown in **Table 2** below.

Example 3. Suppose $\nabla^2 u = 2(xy + xz + yz)$ with the boundary conditions

$$u(0, y, z) = u(x, 0, z) = u(x, y, 0) = 0,$$

and

$$u(1, y, z) = yz(1 + y + z),\ u(z, 1, z) = (1 + z + z)$$
$$u(x, y, 1) = xy(1 + x + y)$$

The analytical solution is $u(x, y, z) = xyz(x + y + z)$ and its results are shown in **Table 3** below.

Example 4. Suppose $\nabla^2 u = 6$ and given the boundary conditions

$$u(0, y, z) = y^2 + z^2, u(x, 0, z) = x^2 + z^2,$$
$$u(x, y, 0) = x^2 + y^2, u(1, y, z) = 1 + y^2 + z^2,$$
$$u(x, 1, z) = 1 + x^2 + z^2, u(x, y, 1) = 1 + x^2 + y^2$$

The analytical solution is $u(x, y, z) = x^2 + y^2 + z^2$ and its results are shown in **Table 4** below.

Example 5. Suppose $\nabla^2 u = -\pi^2 xy \sin(\pi z)$ with the boundary conditions

$$u(0, y, z) = u(x, 0, z) = u(x, y, 0) = u(x, y, 1) = 0$$
$$u(1, y, z) = y \sin(\pi z), u(x, 1, z) = x \sin(\pi z)$$

The analytical solution is $u = xysin(\pi z)$ and its results are shown in **Table 5** below.

Example 6. Suppose

$$\nabla^2 u = -3\pi^2 \sin(\pi x) \sin(\pi y) \sin(\pi z)$$

with boundary conditions

Table 1. The maximum absolute error for Example 1.

(m,n,p)	Scheme 1	Scheme 2
(9,9,9)	1.99840e–015	1.55431e–015
(9,9,19)	1.55431e–015	2.22045e–015
(9,9,29)	1.90958e–014	7.88258e–015
(9,9,39)	5.10703e–015	5.66214e–015
(19,19,9)	7.54952e–015	9.54792e–015
(19,19,19)	1.55431e–015	1.19904e–014
(19,19,29)	6.43929e–015	2.15383e–014
(19,19,39)	7.21645e–015	234257e–014
(29,29,9)	1.69864e–014	1.11022e–014
(29,29,19)	9.43690e–015	4.66294e–015
(29,29,29)	1.63203e–014	6.66134e–015
(29,29,39)	3.96350e–014	8.43769e–015
(39,39,9)	6.43929e–015	5.77316e–015
(39,39,19)	2.33147e–014	2.88658e–015
(39,39,29)	4.46310e–014	1.62093e–014
(39,39,39)	3.29736e–014	1.39888e–014

Table 2. The maximum absolute error for Example 2.

(m,n,p)	Scheme 1	Scheme 2
(9,9,9)	5.55112e–016	5.55112e–016
(9,9,19)	7.77156e–016	5.55112e–016
(9,9,29)	2.83107e–015	1.30451e–015
(9,9,39)	1.72085e–015	1. 16573e–015
(19,19,9)	1.27676e–015	1.55431e–015
(19,19,19)	2.33147e–015	1.99840e–015
(19,19,29)	1.38778e–015	3.38618e–015
(19,19,39)	1.33227e–015	3.99680e–015
(29,29,9)	2.44249e–015	1.52656e–015
(29,29,19)	1.77636e–015	2.02616e–015
(29,29,29)	2.80331e–015	1.67921e–015
(29,29,39)	6.16174e–015	2.08167e–015
(39,39,9)	1.24900e–015	1.66533e–015
(39,39,19)	3.69149e–015	1.88738e–015
(39,39,29)	7.54952e–015	3.05311e–015
(39,39,39)	6.59195e–015	4.49640e–015

Table 3. The maximum absolute error for Example 3.

(m,n,p)	Scheme 1	Scheme 2
(9,9,9)	1.11022e–015	1.33227e–015
(9,9,19)	1.55431e–015	1.11022.e–015
(9,9,29)	5.13478e–015	2.88658e–015
(9,9,39)	3.83027e–015	2.88658e–015
(19,19,9)	2.77556e–015	3.05311e–015
(19,19,19)	4.77396e–015	4.10783e–015
(19,19,29)	3.55271e–015	6.71685e–015
(19,19,39)	3.10862e–015	7.99361e–015
(29,29,9)	4.71845e–015	3.10862e–015
(29,29,19)	3.94129e–015	4.44089e–015
(29,29,29)	5.27356e–014	4.44089e–015
(29,29,39)	1.24623e–014	4.44089e–015
(39,39,9)	3.10862e–015	4.44089e–015
(39,39,19)	7.82707e–015	4.99600e–015
(39,39,29)	1.49880e–014	6.32827e–015
(39,39,39)	1.37113e–014	1.03251e–014

Table 4. The maximum absolute error for Example 4.

(m,n,p)	Scheme 1	Scheme 2
(9,9,9)	2.55351e–015	1.77636e–015
(9,9,19)	3.10862e–015	2.22045e–015
(9,9,29)	1.74305e–014	7.43849e–015
(9,9,39)	6.88338e–015	5.88418e–015
(19,19,9)	7.54952e–015	9.32587e–015
(19,19,19)	1.44329e–014	1.28786e–015
(19,19,29)	6.99441e–015	2.14273e–014
(19,19,39)	7.77156e–015	2.28706e–014
(29,29,9)	1.48770e–014	9.99201e–015
(29,29,19)	9.32587e–015	8.43769e–015
(29,29,29)	1.58762e–014	8.88178e–015
(29,29,39)	3.67484e–014	1.02141e–014
(39,39,9)	7.32747e–015	7.43849e–015
(39,39,19)	2.17604e–014	7.10543e–015
(39,39,29)	4.39648e–014	1.78746e–014
(39,39,39)	3.39728e–014	1.79856e–014

Table 5. The maximum absolute error for Example 5.

(m,n,p)	**Scheme 1**	**Scheme 2**
(9,9,9)	5.89952e–006	5.90013e–006
(9,9,19)	3.67647e–007	3.67666e–007
(9,9,29)	7.25822e–008	7.25853e–008
(9,9,39)	2.29611e–008	2.2962e–008
(19,19,9)	5.92320e–006	5.9233e–006
(19,19,19)	3.69122e–007	3.69124e–007
(19,19,29)	7.28734e–008	7.28737e–008
(19,19,39)	2.30532e–008	2.30533e–008
(29,29,9)	5.94508e–006	5.94513e–006
(29,29,19)	3.70486e–007	3.70487e–007
(29,29,29)	7.31427e–008	7.31428e–008
(29,29,39)	2.31384e–008	2.31384e–008
(39,39,9)	5.94513e–006	5.94515e–006
(39,39,19)	3.70489e–007	3.70489e–007
(39,39,29)	7.31432e–008	7.31433e–008
(39,39,39)	2.31386e–008	2.31386e–008

Table 6. The maximum absolute error for Example 6.

(m,n,p)	Scheme 1	Scheme 2
(9,9,9)	4.07466e–005	9.56601e–005
(9,9,19)	2.80105e–005	3.9912e–005
(9,9,29)	2.73312e–005	2.79092e–005
(9,9,39)	2.72169e–005	2.35847e–005
(19,19,9)	1.52747e–005	1.01614e–005
(19,19,19)	2.53918e–006	5.93387e–006
(19,19,29)	1.85988e–006	3.47172e–006
(19,19,39)	1.74565e–006	2.48652e–006
(29,29,9)	1.3916e–005	3.32432e–006
(29,29,19)	1.18059e–006	1.88497e–006
(29,29,29)	5.01294e–007	1.17049e–006
(29,29,39)	3.87057e–007	7.96897e–007
(39,39,9)	1.36876e–005	7.87013e–006
(39,39,19)	9.52114e–007	6.34406e–007
(39,39,29)	2.72819e–007	5.30167e–007
(39,39,39)	1.58582e–007	3.70168e–007

$$u(0,y,z) = u(1,y,z) = u(x,0,z) = u(x,1,z) = u(x,y,0) = u(x,y,1) = 0$$

The analytical solution is

$$u(x,y,z) = \sin(\pi x)\sin(\pi y)\sin(\pi z)$$

and its results are shown in **Table 6** above.

This example VI was considered as a test problem in [7] and [18], and the results show that our method is more accurate than their methods and in their scheme the step size is the same for all dimensions but in our case Z-direction can have a different step length.

6. Conclusions

In this work, the three dimensional Poisson's equation in Cartesian coordinate systems is approximated by a fourth order finite difference approximation scheme. Here we used to approximate the Poisson's equation by a 27-points scheme and a 19-points scheme, and in doing this by the very nature of finite difference method for elliptic partial differential equations, it resulted in transforming the Poisson's equation (1) in to a large number of algebraic systems of linear Equations (6a) or (6b) which forms a block tri-diagonal matrix in both schemes. These block tri-diagonal matrices are quite comfortable to find the eigenvalues and eigenvectors in order to extend Hockney's method to three dimensions, and we have successfully reduced matrix A to a tri-diagonal one and by the help of Thomas Algorithm we solved the Poisson's equation. The main advantage of this method is that we have used a direct method to solve the Poisson's equation for which the error in the solution arises only from rounding off errors; because it's a direct method the solution of (1) is sure to converge as we are always solving (1) by transforming it in to a diagonally dominant tri-diagonal system of linear equations; and it reduces the number of computations and computational time. It is found that this method produces very good results for fourth order approximations and tested on six examples. Actually it is shown that the discussed method, in general, for 27-points scheme produces better results than 19-points scheme but 19-point scheme has also shown comparable results.

Therefore, this method is suitable to find the solution of any three dimensional Poisson's equation in Cartesian coordinates system.

7. References

[1] L. Collatz, "The Numerical Treatment of Differential Equations," Springer Verlag, Berlin, 1960.

[2] M. K. Jain, "Numerical Solution of Differential Equa-

tions," New Age International Ltd., New Delhi, 1984.

[3] R. Haberman, "Elementary Applied Partial Differential Equations with Fourier Series and Boundary Value Problems," Prentice-Hall Inc., Saddle River, 1987.

[4] T. Myint-U and L. Debnath, "Linear Partial Differential Equations for Scientists and Engineers," Birkhauser, Boston, 2007

[5] G. D. Smith, "Numerical Solutions of Partial Differential Equations: Finite Difference Methods," Oxford University Press, New York, 1985.

[6] G. H. Golub and C. F. van Loan, "Matrix Computations," Johns Hopkins University Press, Baltimore, 1989.

[7] J. Stoer and R. Bulirsch, "Introduction to Numerical Analysis," Springer-Verlag, New York, 2002.

[8] R. W. Hockney, "A Fast Direct Solution of Poisson Equation Using Fourier Analysis," *Journal of ACM*, Vol. 12, No. 1, 1965, pp. 95-113. doi:10.1145/321250.321259

[9] W. W. Lin, "Lecture Notes of Matrix Computations," National Tsing Hua University, Hsinchu, 2008.

[10] B. L. Buzbee, G. H. Golub and C. W. Nielson, "On Direct Methods for Solving Poisson's Equations," *SIAM Journal on Numerical Analysis*, Vol. 7, No. 4, 1970, pp. 627-656. doi:10.1137/0707049

[11] A. Averbuch, M. Israeli and L. Vozovoi, "A Fast Poisson's Solver of Arbitrary Order Accuracy in Rectangular regions," *SIAM Journal on Scientific Computing*, Vol. 19, No. 3, 1998, pp. 933-952. doi:10.1137/S1064827595288589

[12] A. McKenney, L. Greengard and A. Mayo, "A Fast Poisson Solver for Complex Geometries," *Journal of Computation Physics*, Vol. 118, No. 2, 1996, pp. 348-355. doi:10.1006/jcph.1995.1104

[13] G. Skolermo, "A Fourier Method for Numerical Solution of Poisson's Equation," *Mathematics of Computation*, Vol. 29, No. 131, 1975, pp. 697-711.

[14] L. Greengard and J. Y. Lee, "A Direct Adaptive Poisson Solver of Arbitrary Order Accuracy," *Journal of Computation Physics*, Vol. 125, No. 2, 1996, pp. 415-424. doi:10.1006/jcph.1996.0103

[15] W. F. Spotz and G. F. Carey, "A High-Order Compact Formulation for the 3D Poisson Equation," *Numerical Methods for Partial Differential Equations*, Vol. 12, No. 2, 1996, pp. 235-243. doi:10.1002/(SICI)1098-2426(199603)12:2<235::AID-NUM6>3.0.CO;2-R

[16] E. Braverman, M. Israeli, A. Averbuch and L. Vozovoi, "A Fast 3D Poisson Solver of Arbitrary Order Accuracy," *Journal of Computation Physics*, Vol. 144, No. 1, 1998, pp. 109-136. doi:10.1006/jcph.1998.6001

[17] G. Sutmann and B. Steffen, "High Order Compact Solvers for the Three-Dimensional Poisson Equation," *Journal of Computation and Applied Mathematics*, Vol. 187, No. 2, 2006, pp. 142-170. doi:10.1016/j.cam.2005.03.041

[18] J. Zhang, "Fast and High Accuracy Multigrid Solution of the Three Dimensional Poisson Equation," *Journal of Computation Physics*, Vol. 143, No. 2, 1998, pp. 449-461. doi:10.1006/jcph.1998.5982

[19] M. A. Malcolm and J. Palmer, "A Fast Method for Solving a Class of Tri-Diagonal Linear Systems," *Communications of the ACM*, Vol. 17, No. 1, 1974, pp. 14-17. doi:10.1145/360767.360777

