Applied Mathematical Sciences, Vol. 6, 2012, no. 34, 1655 – 1666

A New Method for Solving 3D Elliptic Problem with Dirichlet or Neumann Boundary Conditions Using Finite Difference Method

J. Izadian* **and S. S. Jalalian**

Department of Mathematics, Faculty of Sciences, Mashad Branch,
Islamic Azad University, Mashad, Iran.
Jalal_Izadian@yahoo.com

M. Jalili

Department of Mathematics, Neyshabur Branch,
Islamic Azad University, Neyshabur, Iran.
Jalili.maryam@yahoo.com

Abstract

In this paper, a new algorithm for solving general three dimensional linear Elliptic types P.D.E.'s applying finite difference method is introduced. In this method, the boundary conditions are considered as auxiliary equations coupling with the main equations to constitute a system of linear equations, using suitable finite difference partial derivatives. The mesh points are also generated simply by proposed algorithm. This algorithm can perform numbering of mesh points, generating matrix coefficient, and right hand side vector by a special automatic procedure. Numerical experiments are presented to show performance, reliability and efficiency of proposed algorithm.

Keywords: Elliptic equation, Neumann Boundary Conditions, Dirichlet conditions, Finite difference method

1 Introduction

There is a wide variety of elliptic solvers in numerical literatures that use finite element method, finite difference method, and spectral methods (see [1-8]).

Finite element methods are time consuming, but suitable for irregular geometries. Finite difference solvers, in contrasts, are fast but they are not appropriate for irregular geometries (in this case the main difficulty using finite difference method is calculating the numerical derivatives). The coefficient matrix in linear elliptic equation is calculated manually. In nonlinear cases, problem is more complicated. However for simple equations, there is a great class of codes in computing packages [3, 6]. But these packages are not suitable for every linear elliptic equation, or general boundary conditions.

In this paper, a simple approach for constituting the algebraic discretized equations from a given P.D.E., in mesh points, is applied, and then these equations and the algebraic equations, obtained from boundary conditions which used as a whole system of equations that their unknowns are values of approximate solution in mesh points. Similar procedures can be used for nonlinear equations that gives an algebraic nonlinear system which can be solved by using Newton method.

In fact, one constructs step by step the column of coefficient matrix, and right hand side vector of desired system. This approach will be described in section 2.

This paper is organized as fallows, in section 2, the general 3D linear elliptic equations with variable coefficients subject to Dirichlet boundary conditions, and Neumann boundary conditions, for a three dimensional cell are introduced. The method of numerating the mesh points, coding internal, boundary mesh points and the manner of introducing the whole discritized set that approximate domain are explained. In section 3, the finite difference partial derivatives are used to constitute the algebraic equations in mesh points, and to obtain a linear or nonlinear algebraic system. In section 4, numerical experiments for two and three dimensional equations for two types of mentioned boundary conditions will be presented. In section 5, conclusion and discussion will terminate the paper.

2 Description of method

The general second order linear P.D.E.'s of three variables x, y, and z is given as follows:

$$C_{211}u_{xx} + C_{222}u_{yy} + C_{233}u_{zz} + C_{212}u_{xy} + C_{213}u_{xz} + C_{223}u_{yz}$$
$$+C_{11}u_x + C_{12}u_y + C_{13}u_z + C_0 u = f\ , \qquad (x,y,z) \in D\ , \qquad (1)$$

where C_{2ij} for $i,j = 1,2,3, C_{1,i}, for\ i = 1,2,3\ , C_0$ and f are given real valued functions of variables x, y, and z, that are continuous functions on the domain $D \subset \mathbb{R}^3$.This equation can be more general, but (1) is considered as an elliptic type, then D is closed.

In this paper D is closed 3D cell in $\mathbb{R}^3$, that is given as follows:

$$D = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times [\alpha_3, \beta_3],$$

and three types of boundary conditions are considered:

i. The Dirichlet type boundary conditions,

ii. The Neumann type boundary conditions,

iii. Mixed boundary conditions.

More general boundary conditions are introduced. That describes both Dirichlet and Neumann boundary conditions in the unique equation as follows:

$$\bar{A}u + \bar{B}u_x + \bar{C}u_y + \bar{D}u_z = g(x, y, z)\,, \qquad (x, y, z) \in \partial D, \tag{2}$$

where $\bar{A}, \bar{B}, \bar{C}, \bar{D}, and\ g$ are given real functions of three real variables x, y, and z, that are continuous on ∂D .

If $\bar{A} \equiv 0$, condition (2) gives a Neumann condition for an appropriate selection of coefficients $\bar{B}\,, \bar{C}$, and $\bar{C}$ and if $\bar{B} = \bar{C} = \bar{D} \equiv 0$, (2) is a Dirichlet condition. Actually domain D is descritized in a set of small cells to find suitable mesh points for using finite difference methods. For this purpose, the following partitions of intervals$[\alpha_1, \beta_1]$, $[\alpha_2, \beta_2]$, and $[\alpha_3, \beta_3]$ are considered:

$$\alpha_1 = x_0 < x_1 < \cdots < x_{n_1} = \beta_1\ ,$$
$$\alpha_2 = y_0 < y_1 < \cdots < y_{n_2} = \beta_2\ ,$$
$$\alpha_3 = z_0 < z_1 < \cdots < z_{n_3} = \beta_3\ ,$$

with

$$h_1 = x_{i+1} - x_i\ , \quad i = 0,1,2,\cdots, n_1 - 1,$$
$$h_2 = y_{j+1} - y_j\ , \quad j = 0,1,2,\cdots, n_2 - 1,$$
$$h_3 = z_{k+1} - z_k\ , \quad k = 0,1,2,\cdots, n_3 - 1.$$

Resulting a mesh that consists a set of $(n_1 + 1)(n_2 + 1)(n_3 + 1)$ points:

$$\{(x_i, y_j, z_k) \mid 0 \le i \le n_1, 0 \le j \le n_2, 0 \le k \le n_3\}.$$

For discritizing the main equation and its relative boundary conditions, numerating the mesh points by certain order is essential. Thus these points are numerated, by starting from the plane $z = 0$ and terminating in $z = z_{n_3}$, in each plane the points will be numerated from $y = 0$ to $y = y_{n_2}$, and on each segment, $y = y_j$ $(0 \le j \le n_2)$, points are numerated from $x = 0$ to $x = x_{n_1}$. This procedure is started from point (x_0, y_0, z_0) and continued in a zigzag form in each plane on mesh points, from plane $z = 0$ until $z = z_{n_3}$.

Having an ordering, these mesh points provide a vector of unknown values of solution on mesh points, and permit to constitute a linear system for numerical solution of the given problem. However this method of numerating is not unique. There are various approaches for performing this task (see [1, 7]).

After ordering the mesh points, the labeling of interior and boundary points of mesh (descritized set) is needed. This can be done by considering an additional parameter $ip = 0$, for boundary points and $ip = 1$ for interior points. Thus for each mesh point the information can be considered by denoting the vector of 8 components as follows:

$$(m, \tilde{x}_m, \tilde{y}_m, \tilde{z}_m, i, j, k, ip) \tag{3},$$

where $m = 1,2,\cdots,(n_1+1)(n_2+1)(n_3+1)$, $(\tilde{x}_m, \tilde{y}_m, \tilde{z}_m)$ is a mesh point, i, j, and k are respectively the indices of Cartesian location of the mesh points, say (x_i, y_j, z_k), and $ip = 0$ if mesh point is a boundary point, and $ip = 1$ if mesh point is an interior point. Now by writing the P.D.E. on interior mesh points and the boundary condition equations on boundary point a system of $M = (n_1+1)(n_2+1)(n_3+1)$ equations, with the same number of unknowns, starting from point number 1 to last point number M, is obtained. Then the system of equations of the following form is found:

$$F_i(v_1, v_2, \cdots, v_M) = 0\,, \qquad i = 1,2,\cdots,M,$$

where

$$\begin{gathered} v_l = u_{ijk} = u(x_i, y_j, z_k), \\ i = 1,2,\cdots,n_1+1 \\ j = 1,2,\cdots,n_2+1 \\ k = 1,2,\cdots,n_3+1 \\ l = i + (j-1)n_1 + (k-1)n_1 n_2\,. \end{gathered}$$

With following correspondence

$$(\tilde{x}_l, \tilde{y}_l, \tilde{z}_l) = (x_i, y_j, z_k),$$

the mesh points data can be given by a $M \times 8$ matrix, which lth row is a matrix given by (3), and the first column contains the number of mesh points. The second, third, and fourth columns are x, y, and z, the Cartesian coordinates of mesh points, with respect to ox, oy, and oz, respectively. Columns 5-7 are the indices of mesh points relative to three axes, i.e. 1 for x-axis, 2 for y-axis, and 3 for z-axis. Finally 8th column denotes the information of boundary or interior points.

Then these data are used for constituting desired algebraic equations, for these purpose finite differences partial derivatives are required. These partial derivatives for interior points can be written by central differences. But for boundary conditions particular approximate partial derivatives using backward or forward finite differences are necessary. First and second order central difference derivatives are recalled as follows:

$$
\begin{aligned}
u_x(x,y,z) &= \frac{u_{i+1,j,k} - u_{i-1,j,k}}{2h_1} + O(h_1^2),\\
u_y(x,y,z) &= \frac{u_{i,j+1,k} - u_{i,j-1,k}}{2h_2} + O(h_2^2),\\
u_z(x,y,z) &= \frac{u_{i,j,k+1} - u_{i,j,k-1}}{2h_3} + O(h_3^2),\\
u_{xx}(x,y,z) &= \frac{u_{i+1,j,k} + u_{i-1,j,k} - 2u_{i,j,k}}{h_1^2} + O(h_1^2),\\
u_{yy}(x,y,z) &= \frac{u_{i,j+1,k} + u_{i,j-1,k} - 2u_{i,j,k}}{h_2^2} + O(h_2^2),\\
u_{zz}(x,y,z) &= \frac{u_{i,j,k+1} + u_{i,j,k-1} - 2u_{i,j,k}}{h_3^2} + O(h_3^2),\\
u_{xy}(x,y,z) &= \frac{u_{i+1,j+1,k} + u_{i-1,j-1,k} - u_{i-1,j+1,k} - u_{i+1,j-1,k}}{4h_1h_2} + O(h^2),\\
u_{xz}(x,y,z) &= \frac{u_{i+1,j,k+1} + u_{i-1,j,k-1} - u_{i-1,j,k+1} - u_{i+1,j,k-1}}{4h_1h_3} + O(h^2),\\
u_{yz}(x,y,z) &= \frac{u_{i,j+1,k+1} + u_{i,j-1,k-1} - u_{i,j+1,k-1} - u_{i,j-1,k+1}}{4h_2h_3} + O(h^2).\\
h &= \max(h_1,h_2,h_3)
\end{aligned}
\tag{4}
$$

For applying boundary conditions, numerical derivatives in first point or final point of intervals are needed. These derivatives are first order. For the first point numerical derivatives, are obtained as follows:

$$
\begin{aligned}
u_x(x,y,z) &= \frac{-1.5u_{i,j,k} + 2u_{i+1,j,k} - .5u_{i+2,j,k}}{h_1} + O(h_1^2)\\
u_y(x,y,z) &= \frac{-1.5u_{i,j,k} + 2u_{i,j+1,k} - .5u_{i,j+2,k}}{h_2} + O(h_2^2)\\
u_z(x,y,z) &= \frac{-1.5u_{i,j,k} + 2u_{i,j,k+1} - .5u_{i,j,k+2}}{h_3} + O(h_3^2)
\end{aligned}
\tag{5}
$$

And for final point numerical derivatives are given as follows:

$$
\begin{aligned}
u_x(x,y,z) &= \frac{1.5u_{i,j,k} - 2u_{i-1,j,k} + .5u_{i-2,j,k}}{h_1} + O(h_1^2),\\
u_y(x,y,z) &= \frac{1.5u_{i,j,k} - 2u_{i,j-1,k} + .5u_{i,j-2,k}}{h_2} + O(h_2^2),\\
u_z(x,y,z) &= \frac{1.5u_{i,j,k} - 2u_{i,j,k-1} + .5u_{i,j,k-2}}{h_3} + O(h_3^2).
\end{aligned}
\tag{6}
$$

After that, the system of linear equations of unknown values v_l in mesh points $(\tilde{x}_l, \tilde{y}_l, \tilde{z}_l)$, $l = 1,2,\dots,M$ is constituted. For this purpose, the forming of equations is started from the first point of the mesh and continued until the final point of mesh, in increasing order, from 1 to M, following the numerating order discussed in beginning of this section.

3 Application of finite differences

Considering lth mesh point, if l is an interior point, the lth equation is described as follows:

$$
\begin{aligned}
F_l(v_1, v_2, \cdots, v_M)
&= C_{211}(x_i, y_j, z_k)u_{xx}(x_i, y_j, z_k) + C_{222}(x_i, y_j, z_k)u_{yy}(x_i, y_j, z_k) \\
&+ C_{233}(x_i, y_j, z_k)u_{zz}(x_i, y_j, z_k) + C_{212}(x_i, y_j, z_k)u_{xy}(x_i, y_j, z_k) \\
&+ C_{213}(x_i, y_j, z_k)u_{xz}(x_i, y_j, z_k) + C_{223}(x_i, y_j, z_k)u_{yz}(x_i, y_j, z_k) \\
&+ C_{11}(x_i, y_j, z_k)u_x(x_i, y_j, z_k) + C_{12}(x_i, y_j, z_k)u_y(x_i, y_j, z_k) \\
&+ C_{13}(x_i, y_j, z_k)u_z(x_i, y_j, z_k) + C_0(x_i, y_j, z_k)u(x_i, y_j, z_k) \\
&\quad -f(x_i, y_j, z_k) = 0\,, \qquad (7) \\
&\quad l = i + (j-1)n_1 + (k-1)n_1 n_2,
\end{aligned}
$$

Substituting suitable numerical derivatives given by (4) in (7) these equations can be determined. If lth point is a boundary point, for instance assuming that is (x_i, y_j, z_k), and if x_i, y_j, and z_k are the starting points of intervals $[\alpha_1, \beta_1], [\alpha_2 \beta_2]$, and $[\alpha_3, \beta_3]$, respectively, then the lth equation is constructed using the first point numerical derivatives as follows:

$$
\begin{aligned}
F_l(v_1, v_2, \cdots v_M) = {}& \\
&\bar{A}(x_i, y_j, z_k)u_{i,j,k} + \bar{B}(x_i, y_j, z_k)u_x(x_i, y_j, z_k) \\
&+ \bar{C}(x_i, y_j, z_k)u_y(x_i, y_j, z_k) + \bar{D}(x_i, y_j, z_k)u_z(x_i, y_j, z_k) \\
&- g(x_i, y_j, z_k) \\
&= \bar{A}(x_i, y_j, z_k)u_{ijk} + \bar{B}(x_i, y_j, z_k)\left(\frac{-1.5u_{i,j,k} + 2u_{i-1,j,k} - .5u_{i-2,j,k}}{h_1}\right) \\
&+ \bar{C}(x_i, y_j, z_k)\left(\frac{-1.5u_{i,j,k} + 2u_{i,j-1,k} - .5u_{i,j-2,k}}{h_2}\right) \\
&+ \bar{D}(x_i, y_j, z_k)\left(\frac{-1.5u_{i,j,k} + 2u_{i,j,k+1} - .5u_{i,j,k+2}}{h_3}\right) \\
&- g(x_i, y_j, z_k) = 0\,.
\end{aligned}
$$

If point $x_i,\ y_j, and\ z_k$ is the final point of intervals $[\alpha_1, \beta_1]$, $[\alpha_2, \beta_2]$, and $[\alpha_3, \beta_3]$, respectively,

$$F_l(v_1, v_2, \ldots, v_m) = \bar{A}(x_i, y_j, z_k)u_{ijk}$$

$$+\bar{B}(x_i, y_j, z_k)\left(\frac{1.5u_{i,j,k} - 2u_{i-1,j,k} + 0.5u_{i-2,j,k}}{h_1}\right)$$
$$+\bar{C}(x_i, y_j, z_k)\left(\frac{1.5u_{i,j,k} - 2u_{i,j-1,k} + 0.5u_{i,j-2,k}}{h_2}\right)$$
$$+\bar{D}(x_i, y_j, z_k)\left(\frac{-1.5u_{i,j,k} + 2u_{i,j,k+1} - 0.5u_{i,j,k+2}}{h_3}\right)$$
$$-g(x_i, y_j, z_k) = 0 \tag{8}$$

If (x_i, y_j, z_k) is such that only x_i is a first point of last point of interval $[\alpha_1, \beta_1]$ then in equation (8) forward derivatives or backward derivatives of u_x (respectively), and for other derivatives central derivatives are used, if y_j or z_k are first point or last point of relative interval, the same procedure is applied. Repeating the same way of writing the equations, the desired linear system of M equations with M unknowns is obtained. In this system of equations $u_{i,j,k}$ is replaced by of v_l for each $l = i + (j-1)n_1 + (k-1)n_1 n_2$ where $i = 1,2,\cdots,n_1+1$, $j = 1,2,\cdots,n_2+1$, $and\ k = 1,2,\cdots,n_3$. Next, this linear system can be written in standard form:

$$AV = B,$$

where A is $M \times M$ matrix and $V = (v_1, v_2, \cdots, v_M)$ and B is $M \times 1$ matrix that will be determined.

After that the following system of equations is considered:

$$\begin{cases} F_1(v_1, v_2, \cdots v_M) = 0, \\ F_2(v_1, v_2, \cdots, v_M) = 0, \\ \qquad \vdots \\ F_M(v_1, v_2, \cdots v_M) = 0. \end{cases}$$

For determining matrices $A\ and\ B$ automatically we can apply two approaches. The first that can be applied for M not very large, is using jacobian matrix of $F = (F_1, F_2, \cdots, F_M)$ that is a constant matrix because F is linear function of $v_1, v_2, \cdots, v_M$,

$$A = DF(V)\,, \quad V = (v_1, v_2, \cdots, v_M)\,,$$

for determining B using Newton iteration

$$DF\big(V^{(k)}\big)\big(V^{(k+1)} - V^{(k)}\big) = -F\big(V^{(k)}\big)$$
$$V^{(k)} = (v_1^{(k)}, \dots, v_m^{(k)}), \qquad k = 0,1,2, \dots .$$

Accepting $V^{(0)} = 0$ we have

$$DF(0)\big(V^{(1)}\big) = -F(0)\,,$$

by denoting $B = -F(0)$, it follows:

$$AV^{(1)} = B.$$

Solving this equation can be performed by only one iteration. But this method cannot be applied for large systems, because in such cases, it is essential to perform symbolic calculation, for example applying MATLAB, MAPLE, or MATHEMATICA. However, there is also the following approach.

For large systems a particular approach is proposed that it seems to be new. For this purpose the canonical basis of $\mathbb{R}^M$ with elements

$$e_i = (0,\cdots,0,1,0,\cdots,0), \qquad i = 1,2,\cdots,M\ ,$$

is considered, it can be simply proved that

$$\hat{a}_i = F(e_i) + B,$$

where $\hat{a}_i$ is ith column of matrix A. In fact, function F is defined as follows:

$$F(V) = AV - B,$$

then $F(V) = 0$ deduces

$$AV = B,$$

where

$$F(0) = -B.$$

4 Numerical experiments

In this section three examples of second order P.D.E's, with Neumann boundary conditions or Dirichlet boundary conditions are considered. The numerical results are given in tables and for 2D examples the solution surface is presented. CPU-time and errors relative to exact results are also given. In all test examples, n_1, n_2, and n_3 are respectively the number of equidistant division of $[\alpha_1, \beta_1]$, $[\alpha_2, \beta_2]$, and $[\alpha_3, \beta_3]$.

4.1 Example

We consider the following partial differential equations with a given mixed boundary conditions:

$$x^2u_{xx} + xyu_{xy} + y^2u_{yy} + xu_x + yu_y + 4x^2u$$
$$= 4e^{(-x^2-y^2)}(x^4 + y^4 - y^2 + x^2y^2),$$
$$(x,y) \in D = [0,1] \times [0,1],$$

with boundary condition:

$$u(x,y) = e^{-x^2-y^2}\ , \quad (x,y) \in \partial D\ .$$

The numerical results of this example are shown in table 4.1, and the surfaces of error with respect to exact solution $u(x,y) = e^{-x^2-y^2}$, and nonzero elements of coefficient matrix A is shown in figure 4.1.

Table 4.1 the numerical results of example 4.1.

	n_1	n_2	np	$cpu - time$	$max - error$
Example 1	5	5	36	22.276	10^{-3}
Example 2	10	10	121	35.877	10^{-3}
Example 3	15	15	256	173.488	10^{-4}
Example 4	20	20	441	394.950	10^{-4}

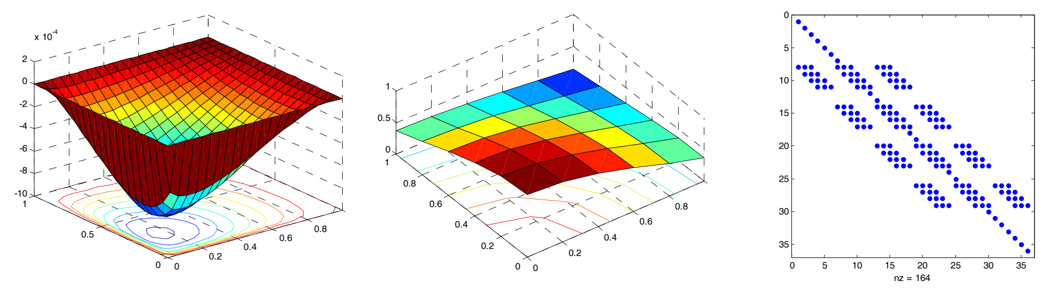

Figure 4.1 surfaces of error and solution $u = e^{-x^2-y^2}$, and nonzero elements of coefficient matrix A.

4.2 Example

In this test example the following there dimensional P.D.E. are considered

$$x^2u_{xx} + y^2u_{yy} + z^2u_{zz} + xyu_{xy} + xzu_{xz} + yzu_{yz}$$
$$+ xu_x + yu_y + zu_z + u = 25x^2y^2z^2,$$
$$(x,y,z) \in \Omega = [0,1] \times [0,1] \times [0,1] .$$

Subject to following boundary condition:

$$u_x + u_y + u_z + u - xyz(xyz + 2yz + 2xy + 2xz) = 0 ,$$
$$(x,y,z) \in \partial\Omega.$$

The numerical results are given in table 4.2. and $u(x,y,z) = x^2y^2z^2$ is the exact solution of problem.The surfaces of solution for 6 fixed values of k is shown in figure 4.2, and the error for vector solution and nonzero elements of coefficient matrix A is shown in figure 4.3.

Table 4.2 the numerical results of example 4.2.

	n_1	n_2	n_3	np	$cpu-time$	Max error
Example 1	5	5	5	216	106	10^{-12}
Example 2	8	8	6	567	4809.265	10^{-10}

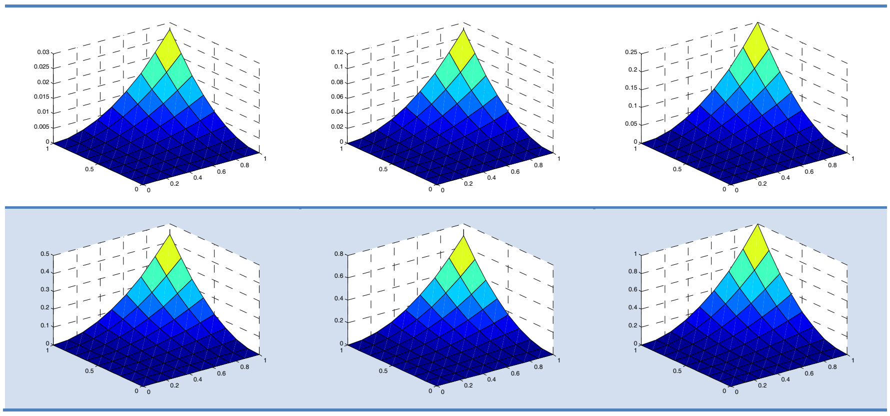

Figure 4.2 surfaces of solution for 6 fixed values of k.

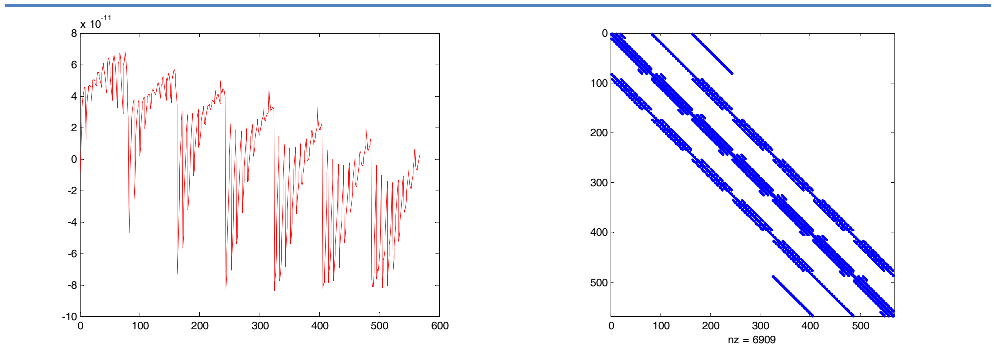

Figure 4.3 error for vector solution and nonzero elements of coefficient matrix A.

4.3 Example

In this test example the three dimensional elliptic equation

$$x^2u_{xx} + y^2u_{yy} + z^2u_{zz} + xyu_{xy} + xzu_{xz} + yzu_{yz}$$

$$+xu_x + yu_y + zu_z + u = (6xyz + 6x^2y^2z^2 + 1)e^{xyz},$$

$$(x, y, z) \in \Omega = [0,1] \times [0,1] \times [0,1].$$

Subject to following boundary condition is considered:

$$u_x + u_y + u_z + u - (xy + yz + zx + 1)e^{xyz} = 0,$$

$$(x, y, z) \in \partial\Omega.$$

The numerical result is given in table 4.3. $u(x, y, z) = e^{xyz}$, is the exact solution. The surfaces of solution for 6 fixed values of k is shown in figure 4.4, and the error for vector solution and nonzero arrays of coefficient matrix A is shown in figure 4.5.

Table 4.3 the result of example 4.3.

	n_1	n_2	n_3	np	$cpu - time$	$\max error$
Example 1	5	5	5	216	171.396140	10^{-3}

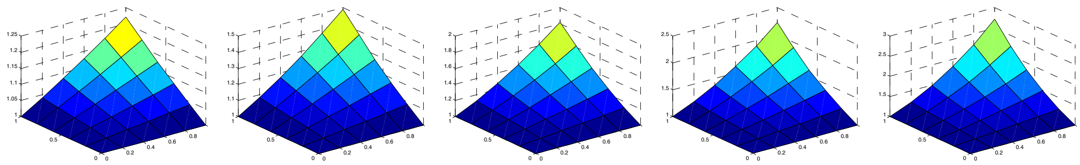

Figure 4.4 surfaces of solution for 6 fixed values of k.

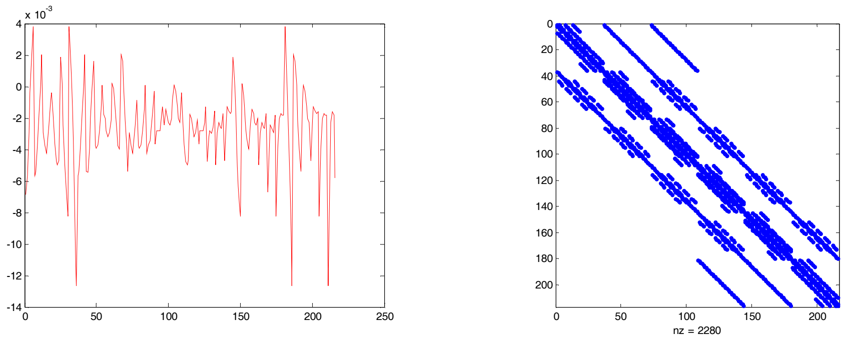

Figure 4.5 error for vector solution and nonzero elementsof coefficient matrix A.

5 Conclusion

In this paper a method for solving two and three dimensional linear second order P.D.E.'s with three type of boundary condition are presented. A particularity of method is to profit the boundary conditions as auxiliary equations joining principal linear equations that constitute a linear system that gives the desired solution. Another particularity of this method is its automatic character of mesh generation and forming of linear system. The numerical results show a good behavior of algorithm and its performance for very complicated linear second order P.D.E.'s. The same researches can be performed to generalize this method to nonlinear P.D.E.'s.

References

[1] W. F. Ames, Numerical Methods for partial Differential Equation, Academic Press. 1NC. 1967.

[2] J. Adams, Muligrid Software for Elliptic Partial Differential Equations, MUOPACK, NCAR Technical Note-357+STR, Feb, 51 pages, 1991.

[3] G. Bencheva, S. Margenov, Parallel Incomplete Factorization Preconditioning of Rotated Linear FEM Systems, J. comput. Appl. Mech. 4(2), 105 117, 2003.

[4] T. Belytschk, Y. Kringonz, D. Organ, M, Fleming, p. Krysl, Meshless Method, An Overview and Recent Developments. Computing Methods in Applied Mechanics and Engineering Vol. (139): 3.47, 1996.

[5] M. M. Gupta, J. Kouatcou, J. Zhang, Comparison of Second and Fourth, Order Discretization for Multigrid Poisson Solvers. Journal of Computational Physics Vol. (132(2)), 226-232. 1990.

[6] I. Lirkov. Y. Vutov, Parallel Performance of a 3D Elliptic Solver, Proceedings of the International Multiconference on Computer Science and Information Technology, Volume1. ISSSN. 1896 7094, 579-590, 2006.

[7] J. M. McDonough. Lectures on Computational Numerical Analysis of Partial Differential Equations, available from the website http:/www.engk.uky.edu/ 2008.

[8] M. Othman. A. R. Abdullah, An Efficient Four Points Modified Explicit Group Poisson Solver. International Journal Computer Mathematics Vol.(76): 203-217, 2000.

Received: October, 2011