ANALELE ŞTIINŢIFICE ALE UNIVERSITĂŢII "AL.I.CUZA" IAŞI
Tomul XLV, s.I a, Matematică, 1999, f.2.

THE RELAXATION SCHEMES FOR THE 2D AND 3D ANISOTROPIC PARTIAL DIFFERENTIAL EQUATIONS USING MULTIGRID METHOD

BY

A. HASSAN NASR* **and M. FATEMA HASSAN****

Abstract. For the efficient multigrid solution of 2D and 3D partial differential equations different types of relaxation schemes have been implemented and applied successfully for some problems with arbitrry anistropies. The previous work considered only the anisotropic elliptic cases (without including mixed derivatives), see for example [1], [2], [7], [13], [18]. Our work discusses various types of anisotropic elliptic cases, more general differential operators (including mixed derivatives) and also discusses the parabolic and hyperbolic equations. Only very few systemtical invetigations on how to treat such anisotropic 3D problems have been carried out so far. Ute Gartel explains in [6] how plane relaxation has been implemented by making use of parallel 2D multigrid code. We explain in this paper how the plane relaxation can be performed making use of 3D multigrid methods. FORTRAN program has been achieved by use of the personal computer. This paper includes the conditions for convergence of the used line and plane relaxation schemes in 2D and 3D cases for the elliptic, parabolic and hyperbolic types. Numerical experiments and results to indicate the validity of these conditions are given.

1. Introduction. Multigrid methods are known as very fast solvers of a large class of discretized partial differential equations. However, the multigrid method cannot be understood as a fixed algorithm. Usually, the components of the multigrid iteration have to be adapted to the given problem and sometimes the problems are modified in order to make them acceptable for multi–grid methods. In particular, the smoothing iteration is the most delicate part of the multi–grid processes, see [9], [15].

The efficiency of the multigrid methods depends essentially on the ability of the 'smoothing operator' to reduce high frequency error components. Whereas for isotropic problems red black relaxation scheme has reasonable smoothing properties, in anisotropic cases more sophisticated methods such

as line and alternating relaxation schemes (in the 2D case) or plane relaxation schemes (in the 3D case) are considerably of better results compared with red black relaxation scheme, see [7], [11] and [18].

The case of a linear boundary value problem is considered. The discretization of such problem on a uniform grid of mesh size h yields a linear system of equations denoted by

$$Lu = f \qquad (\Omega)$$

with Dirichlet boundary conditions

$$u = g \qquad (\partial\Omega)$$

where

$$Lu = \sum_{\alpha=1}^{m} \sum_{\beta=1}^{m} a_{\alpha\beta} u_{x_\alpha x_\beta} + \sum_{\alpha=1}^{m} b_\alpha u_{x_\alpha} + au$$

$$a_{\alpha\beta} = a_{\beta\alpha},\ u_{x_\alpha x_\beta} = u_{x_\beta x_\alpha} \text{ for } \alpha \neq \beta$$

and

$$x_1 = x,\ x_2 = y,\ x_3 = z.$$

Ω denotes a 2D or 3D rectangular domain and L is a 2nd order differential operator with variable coefficients. The variable coefficients of the differential equation are allowed to have arbitrary anisotropies, e.g., to differe considerably from each other in various parts of Ω, see [1], [2], [5], [9], [13], [14], [17] and [18].

The multigrid algorithm is characterized by its components: correction scheme, multigrid iterations or Full Multigrid, standard coarsening, difference operators L^H on coarse grid Ω^H, are built in the same way as L^h on Ω^h, linear interpolation (for the coarse–to–fine transfer of corrections), full weighting (for the fine–to–coarse restriction of residuals) and relaxation scheme for error smoothing, [7]. The type of relaxation used depends on the kind of anisotropy of the given coefficients.

In Section 2 we recall the problems for a standard multigrid method applied to the anisotropic equation. Sections 3, 4 and 5 of the paper contain description of the techniques for multigrid processes of relaxation, residual transfer and interpolation, respectively. Section 6 describes the anisotropic problems for 2D and 3D case, numerical examples are given and numerical results are presented.

2. The problem. The general linear partial differential equation of order two in m variables, $m = 2$ (in the 2D case) and $m = 3$ (in the 3D case), has the form

$$(1) \qquad \sum_{\alpha=1}^{m}\sum_{\beta=1}^{m} a_{\alpha\beta}u_{x_\alpha x_\beta} + \sum_{\alpha=1}^{m} b_\alpha u_{x_\alpha} + au = f(x_1, ..., x_m).$$

If $u_{x_\alpha x_\beta} = u_{x_\beta x_\alpha}$, then the principal part of (1) can always be arranged so that $a_{\alpha\beta} = a_{\beta\alpha}$, thus the $(m\times m)$ matrix $E = [a_{\alpha\beta}]$ can be assumed symmetric. In linear algebra it is shown that every real, symmetric mth-degree polynomial in λ, $\det(E - \lambda I)$ where I is $(m\times m)$ identity matrix, let P denote the number of positive eigenvalues, and Z the number of zero eigenvalues of the matrix E then equation (1) is

hyperbolic :	if $Z = 0$ and $P = 1$ or $Z = 0$ and $P = m - 1$
parabolic :	if $Z > 0$ (equivalently if $\det E = 0$)
elliptic :	if $Z = 0$ and $P = m$ or $Z = 0$ and $P = 0$
ultra hyperbolic :	if $Z = 0$ and $1 < P < m - 1$.

If any of $a_{\alpha\beta}$ is non constant, the type of (1) can vary with position.

The principal phenomena in the 3D–case are similar to those in the 2D–case. However the number of different kinds of anisotropic behavior to be distinguished is larger. The given problems is discretized on a grid Ω^h of uniform mesh size h by use of finite differences. This leads to grid equations

$$(2) \qquad Lu = f \qquad (\Omega)$$

with Dirichlet boundary conditions

$$(3) \qquad u = g \qquad (\partial\Omega)$$

$\Omega = (x, y);\ x \in (0, 1)$ and $y \in (0, 1)$, for the 2D problems
$\Omega = (x, y, z);\ x \in (0, 1),\ y \in (0, 1)$ and $z \in (0, 1)$, for the 3D problems.

Multigrid methods are known to be optimal for the solution of a wide class of partial differential equations, i.e., the number of arithmetic operations is proportional to the number N of the discrete unknowns [6], [14].

3. Relaxation. Pointwise Gauss–Seidel steps are performed in a red–black ordering of points, line and plane–relaxation steps are performed in an even ordering of line, or plane. In line relaxation, an entire row or column of the grid is updated simultaneously, which generally requires the solution of a tridiagonal system for each line. One step of even–odd line

relaxation consists of two half steps. In the first half step all even lines are relaxed. Using the new values at the even lines, the second half step relaxes all odd lines in an analogous way. As the same ways the even–odd plane relaxation is performed see [6], [13] and [14]. Gauss–Seidel relaxation, point and line relaxation, for the 2D case, have been implemented in [11]. In this section we explain how line relaxation scheme and plane–relaxation have been implemented.

The line relaxation scheme for the 2D case: In the 2D case we use *line relaxation* and *alternating line relaxation*. The system of equations (1), with $m = 2$, may be defined as a tridiagonal system of equations,

$$\begin{bmatrix} B1 & C1 & 0 & \cdots & 0 \\ A1 & B1 & C1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & A1 & B1 & C1 \\ 0 & \cdots & 0 & A1 & B1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \vdots \\ \beta_{N-1} \end{bmatrix} \tag{4}$$

where

$$A1 = -2(2a_{22} - b_2h),\ B1 = 4(2a_{11} + 2a_{22} - h^2a),\ C1 = -2(2a_{22} + b_2h),$$

$$\beta_j = 4h^2 f_{ij} + 2(2a_{11} + b_1h)u_{i+1,j} + 2(2a_{11} - b_1h)u_{i-1,j} +$$

$$+2a_{12}(u_{i+1,j+1} - u_{i+1,j-1} - u_{i-1,j+1} + u_{i-1,j-1})$$

where $1 \leq i, j \leq N - 1$, i is constant for each x–line, as the same way for y–line relaxation scheme.

We shall call a matrix ill–conditiononed if its corresponding inverse matrix is unstable. We shall call the inverse matrix stable if "small" changes in the initial matrix elements corresponds to small changes in the inverse matrix elements. It is obvious that for stability of any inverse matrix it is necessary in every case that the determinant of the matrix not be too small (or too large, which will make the determinant of the inverse small) [4].

The plane relaxation scheme for the 3D case: Replacing the derivatives of (1) by a finite difference leads to the system of linear equations

$$
\begin{aligned}
&-\Bigg[\frac{(a_{11}(u_{i+1,j,k}-2u_{i,j,k}+u_{i-1,j,k})}{h^2}+\\
&+\frac{(a_{22}(u_{i,j+1,k}-2u_{i,j,k}+u_{i,j-1,k})}{h^2}+\\
&+\frac{(a_{33}(u_{i,j,k+1}-2u_{i,j,k}+u_{i,j,k-1})}{h^2}+\\
&+\frac{2a_{12}(u_{i+1,j+1,k}-u_{i+1,j-1,k}-u_{i-1,j+1,k}+u_{i-1,j-1,k})}{4h^2}+\\
&+\frac{2a_{13}(u_{i+1,j,k+1}-u_{i+1,j,k-1}-u_{i-1,j,k+1}+u_{i-1,j,k-1})}{4h^2}+\\
&+\frac{2a_{23}(u_{i,j+1,k+1}-u_{i,j+1,k-1}-u_{i,j-1,k+1}+u_{i,j-1,k-1})}{4h^2}+\\
&+\frac{b_1(u_{i+1,j,k}-u_{i-1,j,k})}{2h}+\frac{b_2(u_{i,j+1,k}-u_{i,j-1,k})}{2h}+\\
&+\frac{b_3(u_{i,j,k+1}-u_{i,j,k-1})}{2h}+au_{i,j,k}\Bigg]=f_{i,j,k},
\end{aligned}
\tag{5}
$$

where $u_{i,j,k}$ is an approximation to the exact solution $u(x_i, y_j, z_k)$, h is the interval between successive grid points in each of the three directions (x, y and z), $h = 1/(N-1)$ and $f_{i,j,k} = f(x_i, y_j, z_k)$ is the right hand side function, i, j, k varies from 2 to $N-1$, where N is the number of grid points along the x, y, z axis. In the case of point relaxation the system of equations (5) may then be given as

$$
\begin{aligned}
u_{i,j,k} &= \frac{1}{T_1}4h^2 f_{i,j,k}+T_2u_{i+1,j,k}+T_3u_{i-1,j,k}+T_4u_{i,j+1,k}+\\
&+T_5u_{i,j-1,k}+T_6u_{i,j,k+1}+T_7u_{i,j,k-1}+\\
&+a_{12}(u_{i+1,j+1,k}-u_{i+1,j-1,k}-u_{i-1,j+1,k}+u_{i-1,j-1,k})+\\
&+a_{13}(u_{i+1,j,k+1}-u_{i+1,j,k-1}-u_{i-1,j,k+1}+u_{i-1,j,k-1})+\\
&+a_{23}(u_{i,j+1,k+1}-u_{i,j+1,k-1}-u_{i,j-1,k+1}+u_{i,j-1,k-1})
\end{aligned}
\tag{6}
$$

where

$$
\begin{aligned}
T_1 &= -4(2a_{11}+2a_{22}+2a_3-4h^2a)\\
T_2 &= 4a_{11}+2hb_1, \quad T_3 = 4a_{11}-2hb_1\\
T_4 &= 4a_{22}+2hb_2, \quad T_5 = 4a_{22}-2hb_2\\
T_6 &= 4a_{33}+2hb_3, \quad T_7 = 4a_{33}-2hb_3.
\end{aligned}
$$

The point relaxation is used on the coarsest level because it contains only one interior point, making relaxation a direct solver. In the case of plane

relaxation, an entire plane of the grid is updated simultaneously, we assumed that all the points on the plane are unknown, $(y-z)$–planes for example (the unknowns lying on planes that cut the x–axis at right angles), so we obtain a system of linear equations given by equation (7).

$$(7)\qquad \begin{bmatrix} B & C & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ A & B & C & 0 & & & & \vdots \\ 0 & A & B & C & 0 & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & & & 0 & A & B & C \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & A & B \end{bmatrix} \begin{bmatrix} u_{i,j,2} \\ u_{i,j,3} \\ u_{i,j,4} \\ \vdots \\ u_{i,j,k} \\ \vdots \\ \vdots \\ u_{i,j,N-1} \end{bmatrix} = \begin{bmatrix} \beta_{i,j,2} \\ \beta_{i,j,3} \\ \beta_{i,j,4} \\ \vdots \\ \beta_{i,j,k} \\ \vdots \\ \vdots \\ \beta_{i,j,N-1} \end{bmatrix}$$

where $2 \le i \le N-1,\ 2 \le j \le N-1,\ 2 \le k \le N-1$.

The values of $\beta_{i,j,k}$ at $3 \le j \le N-2,\ 3 \le k \le N-2$ are given by

$$\begin{aligned} \beta_{i,j,k} &= 4h^2 f_{i,j} + T_2 u_{i+1,j,k} + T_3 u_{i-1,j,k} + \\ &+ a_{12}(u_{i+1,j+1,k} - u_{i+1,j-1,k} - u_{i-1,j+1,k} + u_{i-1,j-1,k}) + \\ &+ a_{13}(u_{i+1,j,k+1} - u_{i+1,j,k-1} - u_{i-1,j,k+1} + u_{i-1,j,k-1}) \end{aligned}$$

and at the points near to the boundary the value of $\beta_{i,j,k}$ is given by

$$\begin{aligned} \beta_{i,j,k} &= 4h^2 f_{i,j} + T_2 u_{i+1,j,k} + T_3 u_{i-1,j,k} + \\ &+ a_{12}(u_{i+1,j+1,k} - u_{i+1,j-1,k} - u_{i-1,j+1,k} + u_{i-1,j-1,k}) + \\ &+ a_{13}(u_{i+1,j,k+1} - u_{i+1,j,k-1} - u_{i-1,j,k+1} + u_{i-1,j,k-1}) + \\ &+ 4h^2 \text{ (The known value of the lift hand side of equation (5))} \end{aligned}$$

A, B and $C(N-2)\times(N-2)$ are tridiagonal matrices and are given by

$$A = \begin{bmatrix} -T_7 & a_{23} & 0 & \cdots & 0 \\ -a_{23} & -T_7 & a_{23} & & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & -a_{23} & -T_7 & a_{23} \\ 0 & & 0 & -a_{23} & -T_7 \end{bmatrix},$$

$$B = \begin{bmatrix} T_1 & -T_4 & 0 & \cdots & 0 \\ -T_5 & T_1 & -T_4 & & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & -T_5 & T_1 & -T_4 \\ 0 & & 0 & -T_5 & T_1 \end{bmatrix},$$

$$C = \begin{bmatrix} -T_6 & -a_{23} & 0 & \cdots & 0 \\ a_{23} & -T_6 & -a_{23} & & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & a_{23} & -T_6 & -a_{23} \\ 0 & & 0 & a_{23} & -T_6 \end{bmatrix}.$$

Equation (7) represents $(y-z)$–plane relaxation, where i is constant for each $y-z$ plane, k is constant for each z–line and j varies from 2 to $N-1$ for each line. As the same way $(x-z)$–plane and $(x-y)$–plane is performed.

4. Fine–to–coarse residual transfer. Residuals must be computed on the fine grid and transferred to the coarser grid (where the interval between consecutive grid points is doubled). To each interior point on the coarse grid there corresponds a "block" of 27 fine–grid points centered on a fine–grid red point coinciding with the coarse–grid interior point.

Figure 1. 'Full weighting' scheme for residual transfer–weights, w_i, assigned to fine–grid points involved in computation for coarse–grid points at center, denotes fine–grid red points; ∘ denotes fine–grid black points. Fine–grid red points on even–numbered plans have no corresponding coarse–grid points.

There is a number of ways of defining the weights assigned to the points. Figure 1 depicts the so–called 'full–weighting' scheme for residual transfer. In this scheme the residual, r_i at each point is weighted by w_i shown in the figure. The residual, R, injected into the coarse–grid point at the center of the block is then given by [18]

$$R = \sum_{1}^{27} w_i r_i / \sum_{1}^{27} w_i, \quad \sum_{1}^{27} w_i = 64$$

5. Interpolation. Two types of interpolation are utilized in the 3D code: type I, when correction is to be interpolated from the coarse grid and added to the finer grid, it is called the linear interpolation. Type II, when current values on the coarser grid are to be interpolated as first approximation to the solutions on the finer grid [18]. The interpolation operator has the form

$$I_{2h}^{h} u^h = u^{2h}$$

then the components of u are given as follows

1. At the common points between fine– and coarse–grid (common points)
   - ∘ $u^h_{2i,2j,2k} = u^{2h}_{i,j,k}$

2. The points that lie on the center of the coarser line (intermediate points):
   - ⊗ $u^h_{2i+1,2j,2k} = \frac{1}{2}\left[u^{2h}_{i,j,k} + u^{2h}_{i+1,j,k}\right]$
   - ⊗ $u^h_{2i,2j+1,2k} = \frac{1}{2}\left[u^{2h}_{i,j,k} + u^{2h}_{i,j+1,k}\right]$
   - ⊗ $u^h_{2i,2j,2k+1} = \frac{1}{2}\left[u^{2h}_{i,j,k} + u^{2h}_{i,j,k+1}\right]$

These equations produce the six unknown function values.

3. The points that lie on the center of the coarser plane. This produces 12 unknown function values
   - ■ $u^h_{2i+1,2j+1,2k} = \frac{1}{4}\left[u^{2h}_{i,j,k} + u^{2h}_{i+1,j,k} + u^{2h}_{i,j+1,k} + u^{2h}_{i+1,j+1,k}\right]$
   - ■ $u^h_{2i+1,2j,2k+1} = \frac{1}{4}\left[u^{2h}_{i,j,k} + u^{2h}_{i+1,j,k} + u^{2h}_{i,j,k+1} + u^{2h}_{i+1,j,k+1}\right]$
   - ■ $u^h_{2i,2j+1,2k+1} = \frac{1}{4}\left[u^{2h}_{i,j,k} + u^{2h}_{i,j+1,k} + u^{2h}_{i,j,k+1} + u^{2h}_{i,j+1,k+1}\right]$

4. The remaining points are those at the center of the cube in fine plane but are not lying on the coarser plane nor on coarser lines. We will obtain them as average of the 8–edges about each point

$$\times \quad u^h_{2i+1,2j+1,2k} = \frac{1}{8}[u^{2h}_{i,j,k} + u^{2h}_{i+1,j,k} + u^{2h}_{i,j+1,k} + u^{2h}_{i+1,j+1,k} + \\ + u^{2h}_{i,j,k+1} + u^{2h}_{i+1,j,k+1} + u^{2h}_{i,j+1,k+1} + u^{2h}_{i+1,j+1,k+1}]$$

Figure 2. Example of linear interpolation, at $n = 5$ in each direction, $\circ$ denotes the common points, $\otimes$ denotes the intermediate point (that lie on the center of the coarser line), $\blacksquare$ that lie on the center of the coarser plane, $\times$ denotes the points that lie at the center of the cube.

6. Full multigrid method. Once a multigrid cycle with all its necessary components is specified, as with any other iterative method, their is the significant problem of finding a good first approximation. A convenient approach to finding a reasonable first guess is the combination of an iterative solver with the idea of nested iteration. This essentially means that, before starting to iterate on the given grid, one computes approximations on coarser grids that are interpolated to fine grid until one finally reaches the actual computational grid. The combination of this idea with multigrid iterations is called full multigrid method (FMG). Because of the h-independent

convergence of proper multigrid iterations a fixed number of iteration s is sufficient [13].

7. The anisotropic problems. In this section the anisotropic problems for 2D case and 3D case are investigated with different variable coefficients.

7.1. The 2D anisotropic problems. The anisotropic of the possible forms for the 2D problems.

$$
\begin{aligned}
&1) \quad Lu = \varepsilon u_{xx} + 2u_{xy} + u_{yy} + u_x + u_y + u\\
&1') \quad Lu = u_{xx} + \varepsilon(2u_{xy} + u_{yy} + u_x + u_y + u)\\
&2) \quad Lu = \varepsilon(u_{xx} + 2u_{xy}) + u_{yy} + u_x + u_y + u\\
&2') \quad Lu = u_{xx} + 2u_{xy} + \varepsilon(u_{yy} + u_x + u_y + u)\\
&3) \quad Lu = \varepsilon(u_{xx} + u_x) + 2u_{xy} + u_{yy} + u_y + u\\
&3') \quad Lu = u_{xx} + u_x + \varepsilon(2u_{xy} + u_{yy} + u_y + u)\\
&4) \quad Lu = \varepsilon(u_{xx} + 2u_{xy} + u_x) + u_{yy} + u_y + u\\
&4') \quad Lu = u_{xx} + 2u_{xy} + u_x + \varepsilon(u_{yy} + u_y + u)\\
&5) \quad Lu = \varepsilon(u_{xx} + u_{yy}) + 2u_{xy} + u_x + u_y + u\\
&5') \quad Lu = u_{xx} + u_{yy} + \varepsilon(2u_{xy} + u_x + u_y + u)\\
&6) \quad Lu = \varepsilon(u_{xx} + u_{yy} + 2u_{xy}) + u_x + u_y + u\\
&6') \quad Lu = u_{xx} + u_{yy} + 2u_{xy} + \varepsilon(u_x + u_y + u)\\
&7) \quad Lu = \varepsilon(u_{xx} + u_{yy} + u_x + u_y) + 2u_{xy} + u\\
&7') \quad Lu = u_{xx} + u_{yy} + u_x + u_y + \varepsilon(2u_{xy} + u)
\end{aligned}
$$

Note.

1) The interchange between x and y reduces to an equivalent problem.

2) Putting $1/\varepsilon$ instead of ε gives an equivalent problem.

Table (1) summarizes the results of using the multigrid algorithm with line relaxation and alternating line relaxation scheme.

In the numerical computations we use two types of relaxation schemes with $N = 65$. It is noted that the x–line relaxation scheme fails if

$$\frac{|a_{11}|}{|a_{12}|} \ll 1 \quad \text{or} \quad \frac{|a_{11}|}{|b_1|} \ll 1$$

i.e., the x-line relaxation and alternating line relaxation schemes succeed in the problems that satisfy the conditions

$$\begin{array}{llll} |a_{11}| \sim |a_{12}| & \text{and} & |a_{11}| \sim |b_1| & \text{or} \\ |a_{11}| \geq |a_{12}| & \text{and} & |a_{11}| \geq |b_1| & \end{array}$$

and the use of y-line relaxation scheme fails if

$$\frac{|a_{22}|}{|a_{12}|} \ll 1 \quad \text{or} \quad \frac{|a_{22}|}{|b_2|} \ll 1.$$

In this section we investigate the three types of equations, elliptic, parabolic and hyperbolic partial differential equations, that satisfy the following conditions

$$\frac{|a_{11}|}{|a_{12}|} \ll 1 \quad \text{or} \quad \frac{|a_{11}|}{|b_1|} \ll 1$$

i.e., if a_{11} is not very small compared to a_{12} and b_1.

To examine the algorithm we will consider the following problems with known analytical solution.

1) Examples for the elliptic partial differential equations:

P1) $Lu = \varepsilon(u_{xx} + 6u_{xy} + u_x) + 2u_{yy} + u_y + u,$
P2) $Lu = \varepsilon(u_{xx} + u_x) + 2u_{yy} + u_y + u,$

where $\varepsilon > 0$.

Note. The 2D elliptic problems have been investigated in our previous research, [11].

2) Examples for the hyperbolic partial differential equations:

P1) $Lu = \varepsilon(u_{xx} + 2u_{xy} + u_x) + u_{yy} + u_y + u,$
P2) $Lu = \varepsilon(u_{xx} + u_x) + u_{yy} + u_y + u,$
P3) $Lu = \varepsilon u_{xx} + u_{yy},$

where $\varepsilon < 0$.

3) Examples for the parabolic partial differential equations:

P1) $Lu = \varepsilon u_{xx} + 2\sqrt{\varepsilon}\, u_{xy} + \varepsilon u_x + u_{yy} + u_y + u,$
P2) $Lu = \varepsilon^2 u_{xx} + 2\varepsilon u_{xy} + u_x + u_{yy} + u_y + u,$
P3) $Lu = \varepsilon u_{xx} + 2\sqrt{\varepsilon}\, u_{xy} + u_{yy},$

where $\varepsilon > 0$.

In the numerical computations the L_2 norm, max. norm of the error, L_2 norm and max. norm of the defect are calculated at the end of each v-cycle. The tables give the results of using the multigrid algorithm, described by the components: $V(2,1)$–cycles (i.e., V-cycle, two relaxation steps before and one after the coarse grid correction), full weight residual restriction, linear interpolation and line relaxation. The following analytical solutions are used with different values of ε for each problem, such that $|\varepsilon| < 1$.

1) $u_{\text{exact}} = \cos(x^2 + 3y)$
2) $u_{\text{exact}} = e^{xy} + 5\cos(xy)$
3) $u_{\text{exact}} = x^2y^2(1 - x^2)(1 - y^2)$
4) $u_{\text{exact}} = \sin(xy)$
5) $u_{\text{exact}} = e^{xy}$
6) $u_{\text{exact}} = \sin(x^2 + y)$

The tables (2), (3) and (4) summarize the results for the first problem from each type, where the other problems have approximately the same order of the error. The norm of the error is given by $\|\text{error}\| = \|u^h_{\text{exact}} - u^h_{\text{approx}}\|$ and the current residual is given by: $r^h = F^h - A^h u^h$

Note.

1) For the hyperbolic problems the use of x-line relaxation scheme fails if $|a_{11}| \not\ll |a_{22}|$ i.e., if a_{11} is very small compared to a_{22}.

2) If any one of x-line relaxation or y-line relaxation scheme fails then alternating line relaxation scheme fails also.

7.2. The 3D anisotropic problems. The anisotropic of the possible forms for the 3D case:

$$
\begin{aligned}
&1) \quad Lu=\varepsilon u_{xx}+u_{yy}+u_{zz}+2u_{xy}+2_{xz}+2u_{yz}+u_x+u_y+u_z+u\\
&1') \quad Lu=u_{xx}+\varepsilon(u_{yy}+u_{zz}+2u_{xy}+2u_{xz}+2u_{yz}+u_x+u_y+u_z+u)\\
&2) \quad Lu=\varepsilon(u_{xx}+u_x)+u_{yy}+u_{zz}+2u_{xy}+2u_{xz}+2u_{yz}+u_y+u_z+u\\
&2') \quad Lu=u_{xx}+u_x+\varepsilon(u_{yy}+u_{zz}+2u_{xy}+2u_{xz}+2u_{yz}+u_y+u_z+u\\
&3) \quad Lu=\varepsilon(u_{xx}+2u_{xy}+2u_{xz})+u_{yy}+u_{zz}+2u_{yz}+u_x+u_y+u_z+u\\
&3') \quad Lu=u_{xx}+2u_{xy}+2u_{xz}+\varepsilon(u_{yy}+u_{zz}+2u_{yz}+u_x+u_y+u_z+u)\\
&4) \quad Lu=\varepsilon(u_{xx}+2u_{xy}+2u_{xz}+u_x)+u_{yy}+u_{zz}+2u_{yz}+u_y+u_z+u\\
&4') \quad Lu=u_{xx}+2u_{xy}+2u_{xz}+u_x+\varepsilon(u_{yy}+u_{zz}+2u_{yz}+u_y+u_z+u)\\
&5) \quad Lu=\varepsilon(u_{xx}+u_{yy})+u_{zz}+2u_{xy}+2u_{yz}+u_x+u_y+u_z+u\\
&5') \quad Lu=u_{xx}+u_{yy}+\varepsilon(u_{zz}+2u_{xy}+2u_{xz}+2u_{yz}+u_x+u_y+u_z+u)\\
&6) \quad Lu=\varepsilon(u_{xx}+u_{yy}+2u_{xy})+u_{zz}+2u_{xz}+2u_{yz}+u_x+u_y+u_z+u\\
&6') \quad Lu=u_{xx}+u_{yy}+2u_{xy}+\varepsilon(u_{zz}+2u_{xz}+2u_{yz}+u_x+u_y+u_z+u)\\
&7) \quad Lu=\varepsilon(u_{xx}+u_{yy}+2u_{xy}+u_x+u_y)+u_{zz}+2u_{xz}+2u_{yz}+u_z+u\\
&7') \quad Lu=u_{xx}+u_{yy}+2u_{xy}+u_x+u_y+\varepsilon(u_{zz}+2u_{xz}+2u_{yz}+u)\\
&8) \quad Lu=\varepsilon(u_{xx}+u_{yy}+2u_{xy}+2u_{xz}+2u_{yz})+u_{zz}+u_x+u_y+u_z+u\\
&8') \quad Lu=u_{xx}+u_{yy}+2u_{xy}+2u_{xz}+2u_{yz}+\varepsilon(u_{zz}+u_x+u_y+u_z+u)\\
&9) \quad Lu=\varepsilon(u_{xx}+u_{yy}+2u_{xy})+2u_{xz}+2u_{yz}+u_x+u_y)+u_{zz}+u_z+u\\
&9') \quad Lu=u_{xx}+u_{yy}+2u_{xy}+2u_{xz}+2u_{yz}+u_x+u_y+\varepsilon(u_{zz}+u_z+u)\\
&10) \quad Lu=\varepsilon(u_{xx}+u_{yy}+u_{zz})+2u_{xy}+2u_{xz}+2u_{yz}+u_x+u_y)+u_z+u\\
&10') \quad Lu=u_{xx}+u_{yy}+u_{zz}+\varepsilon(2u_{xy}+2u_{xz}+2_{yz}+u_x+u_y+u_z+u)
\end{aligned}
$$

In the general anisotropic 3D case, even line relaxation is not sufficient if standard coarsening is maintained. Instead, *plane relaxation* is necessary in certain cases. In the numerical computation we use plane relaxation scheme with $N = 9$ because the difficulty of application on the available computer if $N = 17$, "Personal computer with an 80386 processor and 640 kilobytes of conventional memory plus 1024k of extended memory".

Table (5) summarizes the results of using the multigrid algorithm with plane relaxation scheme.

From studying the numerical results of the 3D case it is noted that the plane relaxation scheme is fail if

$$\frac{|a_{\alpha\alpha}|}{|a_{\alpha\beta}|} \ll 1 \quad \text{or} \quad \frac{|a_{\alpha\alpha}|}{|b_{\alpha}|} \ll 1.$$

Note.

1) If $a_{11} \ll a_{22}$, $a_{22} \ll a_{33}$ and $a_{22} \approx a_{33}$, then the use of $(y - z)$ plane (the plane that cuts the x-axis at right angle) relaxation scheme is considerably of better result.

2) If $a_{11} \ll a_{33}$, $a_{22} \ll a_{33}$ and $a_{11} \approx a_{22}$, then the use of $(x-z)$ plane or $(y-z)$ plane (the planes that cut the $(x-y)$ plane at right angle) relaxation scheme has the same convergence behaviour and are considerably of better result.

We investigate the following examples that satisfy the following conditions for the three types of PDE's.

$$\frac{|a_{11}|}{|a_{12}|} \not\ll 1, \quad \frac{|a_{11}|}{|a_{13}|} \not\ll 1, \quad \text{and} \quad \frac{|a_{11}|}{|b_1|} \not\ll 1$$

1) Examples for the elliptic partial differential equations:

P1) $Lu = \varepsilon(u_{xx}+4u_{xy}+8u_{xz}+u_x)+u_{yy}+u_{zz}+2u_{yz}+u_y+u_z+u,$
P2) $Lu = \varepsilon(u_{xx}+2u_{xy}+2u_{xz}+u_x)+u_{yy}+u_y+u_{zz}+u_z+u.$

2) Examples for the hyperbolic partial differential equations:

P1) $Lu = \varepsilon(u_{xx}+2u_{xy}+2u_{xz}+u_x)+u_{yy}-u_{zz}+2u_{yz}+u_y+u_z+u,$
P2) $Lu = \varepsilon(u_{xx}+u_x)+u_{yy}+u_y-u_{zz}+u_z+u.$

3) Examples for the parabolic partial differential equations:

P1) $Lu = \varepsilon u_{xx}+2u_{xy}+2u_{xz}+u_z)+u_{yy}+u_{zz}+2u_{yz}+u_y+u_z+u,$
P2) $Lu = \varepsilon(u_{xx}+2u_{xy}+2u_{xz})+u_{yy}+2u_{yz}+u_{zz}+u.$

The following analytical solutions are used for the 3D case.

1) $u_{ex} = x^2y^2z^2(1-x^2)(1-y^2)(1-z^2)$
2) $u_{ex} = \cos(x^2y^2+z)$
3) $u_{ex} = \sin(xyz)$
4) $u_{ex} = 3\sin(xyz)+e^{xyz}$
5) $u_{ex} = e^{xyz}$

Tables (6), (7) and (8) summarize the results of using the multigrid algorithm with plane relaxation scheme, for the first problem from each type.

Conclusion. In the numerical computation the multi-grid algorithm is used with line relaxation scheme for the 2D problems and plane relaxation scheme for 3D problems. It is noted that the line relaxation scheme and plane relaxation scheme are of acceptable results for the three types of PDE's if

the following conditions are satisfied:

condition 1: $\dfrac{|a_{\alpha\alpha}|}{|a_{\alpha\beta}|} \not\ll 1$

condition 2: $\dfrac{|a_{\alpha\alpha}|}{|b_{\alpha}|} \not\ll 1$

and specially for the hyperbolic type

condition 3: $\dfrac{|a_{\alpha\alpha}|}{|a_{\beta\beta}|} \not\ll 1$

where $\alpha \neq \beta$

Future proposed work. Further investigation has not been finished for the ultra hyperbolic type of time dependent 3D PDE.

Notation. Below we list those symbols which have special meaning throughout the paper:

D	:	dimension of Ω
f	:	right hand side of PDE
h	:	discretization parameter (grid size or step length)
I_h^{2h}	:	restriction operator
I_{2h}^h	:	interpolation operator
L	:	operator associated with the boundary value problem
N	:	number of the points in one direction
r	:	residual
R	:	injection
Ω	:	domain of boundary value problem
$\partial\Omega$	:	boundary of Ω
u_{exact}	:	exat solution
u_{apporx}	:	approximate solution
PDE	:	partial differential equation
λ	:	eigenvalue
$\ll$	:	very small compared with
$\not\ll$	:	negation of $\ll$

REFERENCES

1. BRANDT, A. – *Multi–level adaptive solution to boundary–value–problem*, Math. Comp., 31 (1977), 333–390.
2. BRANDT, A. – *A Guide to Multigrid Development.* Lecture Notes in Math. 960, Springer–Verlag, Berlin, (1982), 221–310.
3. THOLE, CLEMENS–AUGUST, TROTTENBERG, ULRICH – *Basic Smoothing Procedures for the Multigrid Treatment of Elliptic 3D–Operator*, March 1985.
4. FADDEEV, D.K. and FADDEEVA, V.N. – *Computational Methods of Linear Algebra* (translated by Robert C. Williams), W.H. Freeman and Company 1963.
5. GOLDSTEIN, C.I. – *Analysis and application of multigrid preconditioners for singularly perturbed boundary value problems*, SIAM J. Numer. Anal. vol. 26, no. 5, October 1989, 1090–1123.
6. GARTEL, UTE – *Parallel multigrid solver for 3D anisotropic elliptic problems*, Arbeitspapiere der GMD 390, June 1989.
7. GARTEL, UTE, KRECHEL, A., NIESTAGGE, A., PLUM, H.-J. – *Parallel multi–grid solution of 2D and 3D anisotropic elliptic equations: Standard and nonstandard smoothing*, International Series of Numerical Mathematics, vol.98, Birkhäuser Verlag, Basel, 1991.
8. HACHBUSCH, W., TROTTENBERG, U. – *Multigrid Methods: Special Topics and Applications,* Bonn, October 1–4, 1990.
9. HACHBUSCH, W. – *The frequency decomposition multi–grid method*, Numer. Math. 56, 1989, Springer–Verlag, 229–245.
10. HASSAN NASR, A., OSAMA ELGIAR, M., HODA, IBRAHIM – *Multigrid solu–tions for solving elliptic partial differential equations with a mixed derivative*, Term. 18^{th} INI Conf. for Stat., Comp. Sci., Scient. & Social Appl., Cairo 17–22 April, 19935, vol. 5, 257–265.
11. HASSAN NASR, A., OSAMA ELGIAR, M., FATEMA HASSAN, M. – *The relaxation schemes for the two dimensional anisotropic partial differential equations using multigrid method.* 20^{th} INI Conf. for Stat., Comp. Sci., Scient. & Social Appl., Cairo 3–9 April, 1995.
12. PADDON, D.J. and HOLSTEIN, H. – *Multigrid Methods for Integral and Differential Equations*, Clarendon Press. Oxford, 1985.
13. MCBRYAN, OLIVER A. – *Multigrid method on Parallel computers.* A Survey of recent developments. Impact of Computing in Science and Engineering 3, (1991) 1–75.

14.STUBEN, K., TROTTENBERG, U. – *Multigrid methods fundamental algorithms model problem analysis and applications*, GMD–Studien nr. 96 December 1984.

15.WESSELING, P. – *An introduction to multigrid methods*, Copyright ©1992 by John Wiley & Sons Ltd.

16.WESSELING, P. – *Liner multigrid methods,* Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania 1987.

17.WILLIAM, L.B. – *A multigrid tutorial.* Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania 1987.

18.HOLTER, WILLIAM, H. – *A vectorized multigrid solved for the three–dimensional Poisson equation*, In Supercomputer Application, A.H.L. Emmen (ed.), 1985.

Received: 16.VII.1997

* *Benha Higher Institute of Technology*
Benha
EGYPT

** *Faculty of Science*
Al–Azhar University
Cairo
EGYPT