Full Multigrid Method for Solving the General Three Dimensional Elliptic Partial Differential Equation with Variable Coefficients and with General Boundary Conditions

Osama El-Giar
Department of Basic Science, Modern Academy for Engineering and Technology in Maadi,
Cairo, Egypt.

Abstract— In this paper we apply a full multigrid method with a $W\left(\gamma_1,\gamma_2\right)$ cycle to obtain the numerical solution of the three dimensional general elliptic partial differential equation with variable coefficients and with general boundary conditions (Robbin's and Neumann's boundary conditions) in a unit cube. Red-black gauss-Seidel iteration is used for relaxation process, linear interpolation is used to transfer the correction values from coarse grids to fine grids and half weighting restriction process is used to restrict the residuals from fine grids to coarse grids. Numerical examples have been given to illustrate the efficiency of this method.

Keywords— Multigrid method, Numerical Analysis, Three Dimensional General Elliptic Partial Differential Equation, Robin's and Neumann's Boundary conditions.

1 Introduction

In this paper we consider the general linear elliptic partial differential equation in the three dimensional space:

$$-a(x,y,z)u_{xx} - b(x,y,z)u_{yy} + c(x,y,z)u_{zz} + d(x,y,z) = f(x,y,z) \in \Omega \subset R^3 \qquad (1.1a)$$

Subject to the general boundary conditions $u_n + \alpha(x,y,z)u = \beta(x,y,z) \in \partial\Omega \subset R^3$ (1.1b)

Where $\partial\Omega$ is a unit cube given by

$$\Omega = \{(x,y,z), x,y,z \in (0,1)\}\ \} \qquad (1.2)$$

Where the subscript n denotes the derivative in the outer normal direction, and the variable coefficients $a(x,y,z)$, $b(x,y,z)$ and $c(x,y,z)$ satisfy the ellipticity condition. Equation 1.1a with the boundary condition 1.1b is a well-posed problem provided that

$$d(x,y,z) \geq 0, \quad \alpha(x,y,z) \geq 0 \qquad (1.3a)$$

$$d(x,y,z) + \alpha(x,y,z) > 0 \qquad (1.3b)$$

If $\alpha > 0$ then we have the general mixed boundary conditions (Robin's boundary conditions), while for $\alpha = 0$, we have the Neumann boundary conditions.

2 Finite Difference Discretization

For each grid we replace each term in equation 1.1a by a second order finite difference approximation for two kinds of points as follows:

1. Interior points: These are interior points that are not laying on the boundary planes: In
This case equation 1.1a can be replaced by the 7 point difference scheme given by the following discretizaton:

$$-a_{i,j.k}(u_{i-1,j,k}-2u_{i,j,k}+u_{i+1,j,k})-b_{i,j,k}(u_{i,j-1,k}-2u_{i,j,k}+u_{i,j+1,k})$$
$$-c_{i,j.k}(u_{i,j,k-1}-2u_{i,j,k}+u_{i,j,k+1})+h^2d_{i,j,k}u_{i,j,k}=-h^2f_{i,j,k},\ i,j,k=2,...,N-1 \qquad (2.1)$$

Where h is the mesh size between the grid points in each of the three directions x, y and z and N is the number of grid points in each direction.

2. Boundary points: These are boundary points that are laying on the boundary planes, and consists of three kinds of points:
   (a) Corner points.
   (b) Edge points.
   (c) Plane points.

For these points finite difference schemes will contain values at points outside our domain, these values can be eliminated using the known values of the boundary conditions, so the finite difference equation will be 4 point scheme for corner points, 5 point scheme for the edge points and 6 point scheme for the plane points see Figure 1

3 Multigrid Method

A sequence of 4 grids Ω_{kh} have been used with mesh sizes $kh, k=2^n, n=0,1,...3,$ Where $h=\frac{1}{32}$. The elliptic equation have been solved using full multigrid $W(\gamma_1,\gamma_2)$- cycle method with the coarse grid correction (CGC), see Figure 2 And has the following components:

1. Three dimensional Red-black point Gauss-Seidel relaxation: An efficient smoothing iteration (relaxation) process is the red-black Gauss-Seidel point iterations (relaxation) given by splitting our domain: $\Omega_k=\{(ih,jh,kh),\ \ i,j,k\in Z, Z$ is the set of integer numbers $\}$ (3.1)
   Into red (even) points:
   $$\Omega_k^e=\{(ih,jh,kh),\ \ \in\Omega_{k,}: i+j+k \text{ is even }\} \qquad (3.2)$$

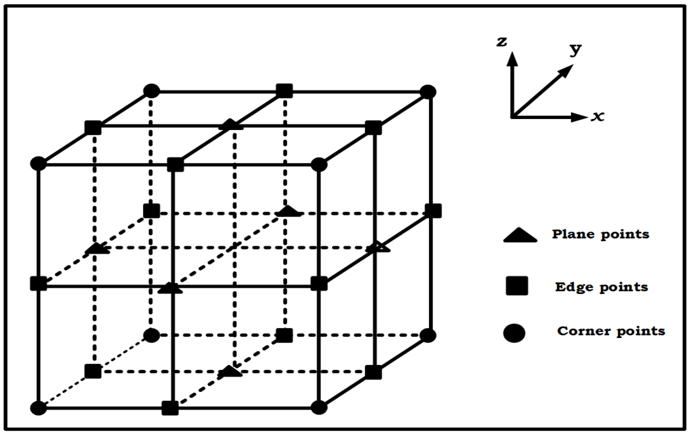

Figure 1: Corner, edge and plane points in 3D

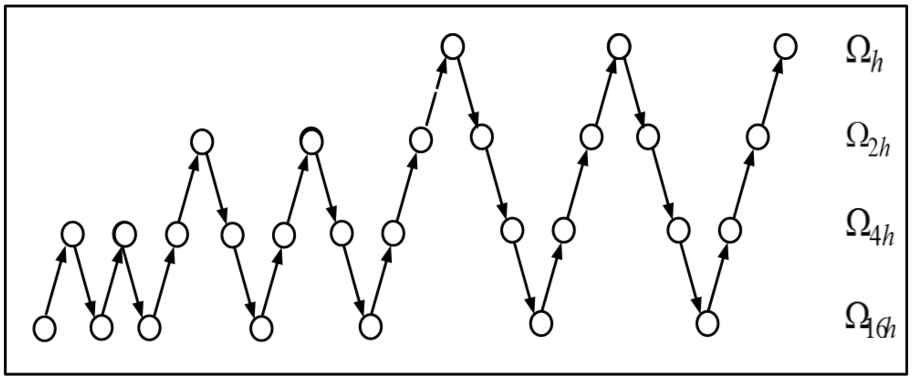

Figure 2: 3D-FMG W cycle

And the remaining points are black (odd):

$$\Omega_k^o = \{(ih, jh, kh) \in \Omega_{k}, : i + j + k \text{ is odd}\} \tag{3.3}$$

First the solutions are calculated at the even (odd) points using the solution at the odd (even) points and vise-versa. The relaxation step is given by:

$$u_{i,j.k} = \frac{h^2 f_{i,j,k} + a_{i,j,k}(u_{i-1\ ,j,k} + u_{i+1\ ,j,k}) + b_{\ i,j,k}(u_{i,j-1,k} + u_{i,j+1,k}) + c_{i,j,k}(u_{i,j,k-1} + u_{i,j,k+1})}{2a_{i,j,k} + 2b_{i,j,k} + 2c_{i,j,k} + h^2 d_{i,j,k}} \tag{3.4}$$

2. Coarse to fine interpolation: It takes coarse-grid vector (solution or correction) u^{2h} defined on the coarse grid Ω^{2h} and produces fine grid vector u^h defined on the fine grid Ω^h according to the rule $I_{2h}^h u^{2h} = u^h$. Since the interpolation is followed by Red-black point Gauss-Seidel relaxation, then we need to interpolate only at the red points, see [3].

We have two kind of red points:

(a) Common points: they consists of coarse grid points that coinciding with fine grid points: then

$$u^h = u^{2h} \tag{3.5}$$

(b) Intermediate points: they consists of coarse grid points that are not coinciding with fine grid points, then u^h = Average of u_{2h} at the 4 nearest neighbour coarse grid points at the plane of interest (3.6)

See Figure 3 for two grid scheme (fine and coarse grids) in the three dimensional space:

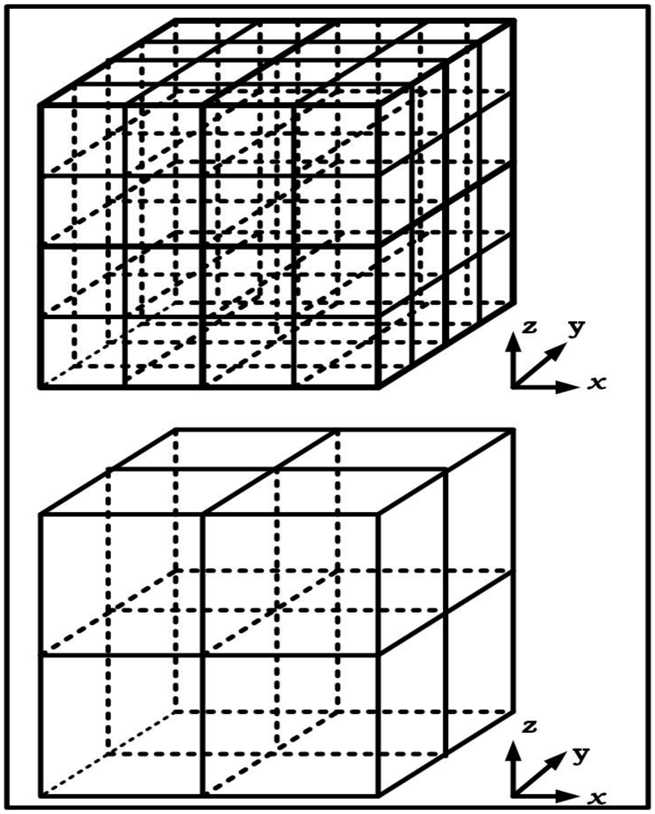

Figure 3: Two grids scheme in 3D

3. Fine to coarse restriction operator: The restriction operator denoted by $\boldsymbol{I_h^{2h}}$ takes the residual vector $\boldsymbol{R^h}$ computed on the fine-grid and transfer it to the coarse-grid according to the rule: $\boldsymbol{I_h^{2h} R^h = R^{2h}}$ where the components of $\boldsymbol{R_{i,j}^{2h}}$ of $\boldsymbol{R^{2h}}$ are given by two restriction operators: half-weight and full-weight restriction operator; since we use the point relaxation, then the half-weight operator is sufficient in our case: see [1] and [3]: Since we use red-black relaxation scheme, then the residuals at fine grid black points are zero, then we will use the half injection for the fine to coarse grid residual transfer, i. e.;

$$\boldsymbol{R_{2h} = \frac{1}{2} R_h} \tag{3.7}$$

4 Test problems, Results and Performance

In this section we report the results and performance of some test problems using FMG W- cycle with $\boldsymbol{\gamma_1 = 2 + \frac{1}{2}}$ relaxation steps (black-red-black-red-black) before coarse grid correction (CGC) and $\boldsymbol{\gamma_2 = 1 + \frac{1}{2}}$ relaxation steps (black-red-black) after coarse grid corrections.

All our implementations in Fortran were executed on a Pentium 4 PC workstation compiler. The accuracy of the method is measured by both L2 norm of both defect and error and the max norm of both defect and the error. The general form of the test problem is given by equation 1.1

1. Test Problem 1:

$$a(x,y,z) = b(x,y,z) = c(x,y,z) = 1,$$
$$d(x,y,z) = 0,$$
$$\alpha(x,y,z) = e^{x}(1+\sin y^{2})(1+z), \qquad (4.1)$$

With exact solution: $U_{exact} = e^{xyz}$

2. Test Problem 2:

$$a(x,y,z) = x^{2}+2y+z^{2}+1,$$
$$b(x,y,z) = \frac{1}{2}e^{x+y+z}$$
$$c(x,y,z) = 0 \qquad (4.2)$$
$$d(x,y,z) = \sin(2x+3y+4z+1),$$
$$\alpha(x,y,z) = 2(x+y+z+1),$$

With exact solution: $U_{exact} = e^{xyz}$

3. Test Problem 3:

$$a(x,y,z) = b(x,y,z) = c(x,y,z) = 1,$$
$$d(x,y,z) = \cos(2x+3y+4z+1),$$

$$c(x,y,z) = 0$$
$$a(x,y,z) = (x+y+z+1), \qquad (4.3)$$

With exact solution $U_{exact} = e^{xyz}$

4. Test Problem 4:

$$a(x,y,z) = b(x,y,z) = c(x,y,z) = 1,$$
$$d(x,y,z) = 5(x^{2}+2y+3z+1),$$
$$a(x,y,z) = 0 \qquad (4.4)$$

With exact solution $U_{exact} = e^{xyz}$

Test Problem1: Equation 4.1

Test problem	H	Max N(R)	L_2(R)	Max N(E)	L_2(E)
Equation 4.1	$\frac{1}{2}$	0.888178D-15	0.111022D-14	0.315789D-01	0.376639D-01
	$\frac{1}{4}$	0.160460D-01	0.213329D-01	0.827788D-02	0.725603D-02
	$\frac{1}{8}$	0.198637D-02	0.173205D-02	0.206911D-02	0.151670D-02
	$\frac{1}{16}$	0.169113D-03	0.107354D-03	0.518516D-03	0.277345D-03
	$\frac{1}{32}$	0.184572D-04	0.592582D-05	0.136210D-03	0.645742D-04
	$\frac{1}{64}$	0.653092D-05	0.897826D-06	0.356495D-04	0.194849D-04

Table1: Fortran implementation of test problem 1 using $FMW(2+\frac{1}{2},1+\frac{1}{2})$ on PC

Test Problem2: Equation 4.2

Test problem	H	MaxN(R)	L_2(R)	Max N(E)	L_2(E)
Equation 4.2	$\frac{1}{2}$	0.142109D-13	0.153339D-13	0.580755D-01	0.615730-01
	$\frac{1}{4}$	0.957244D-02	0.121291D-01	0.146367D-01	0.798582D-01
	$\frac{1}{8}$	0.141472D-01	0.103552D-01	0.325443D-01	0.210595D-02
	$\frac{1}{16}$	0.128269D-01	0.536659D-02	0.129795D-02	0.536659D-02
	$\frac{1}{32}$	0.573197D-02	0.148969D-02	0.829015D-03	0.782471D-03
	$\frac{1}{64}$	0.168145D-02	0.279195D-03	0.378353D-03	0.458523D-03

Table2: Fortran implementation of test problem 2 using $FMW(2+\frac{1}{2},1+\frac{1}{2})$ on PC

Test Problem3: Equation 4.3

Test problem	H	MaxN(R)	L_2(R)	Max N(E)	L_2(E)
Equation 4.3	$\frac{1}{2}$	0.444089D-15	0.666134D-15	0.562274D-01	0.594297D-01
	$\frac{1}{4}$	0.250139D-02	0.313833D-02	0.141399D-01	0.764961D-01
	$\frac{1}{8}$	0.244246D-02	0.245743D-02	0.322975D-02	0.205777D-02
	$\frac{1}{16}$	0.168671D-02	0.102369D-02	0.117414D-02	0.130537D-02
	$\frac{1}{32}$	0.660588D-03	0.253824D-03	0.671030D-03	0.689167D-03
	$\frac{1}{64}$	0.154286D-03	0.447228D-04	0.251633D-03	0.307112D-03

Table3: Fortran implementation of test problem 3 using $FMW(2+\frac{1}{2},1+\frac{1}{2})$ on PC

Test Problem4: Equation 4.4

Test problem	H	Max N(R)	L_2(R)	MaxN(E)	L_2(E)
Equation 4.4	$\frac{1}{2}$	0.355271D-14	0.435683D-14	0.509018D-01	0.547927D-01
	$\frac{1}{4}$	0.1822693D-00	0.242629D-00	0.410444D-01	0.471976D-01
	$\frac{1}{8}$	0.535787D-02	0.385232D-02	0.530110D-02	0.228335D-02
	$\frac{1}{16}$	0.1826211D-02	0.749076D-03	0.673002D-03	0.809307D-03
	$\frac{1}{32}$	0.414341D-03	0.160322D-03	0.584537D-03	0.853606D-03
	$\frac{1}{64}$	0.109012D-03	0.532519D-04	0.700447D-03	0.119115D-02

Table4: Fortran implementation of test problem 4 using $FMW(2+\frac{1}{2},1+\frac{1}{2})$ on PC

5 Conclusion

The implementation of a full multigrid method with a $W(2+\frac{1}{2}, 1+\frac{1}{2})$ cycle to obtain the numerical solution of the three dimensional general elliptic partial differential equation with variable coefficients and with general boundary conditions in the three space dimensions, has been shown to be an efficient method. W-cycle has been used instead of V -cycle to achieve a better convergence rate due to the complicated boundary conditions.

References

1) HackbucshW., "Multigrid Methods and applications," *Springer Berlin Heidelberg Berlin*,- 2003.

2) W. L. Briggs, V. E. Henson, and S. F.Mc Cormick, "A Multigrid Tutorial, Second Edition" SIAM Books, *Philadelphia* – 2000.

3) Briggs L., "A Multigrid Tutorial," *Society for Industrial and applied mathematics* – 2000.

4) W. L. Briggs, V. E. Henson, and S. F.McCormick, "A Multigrid Tutorial, Second Edition" *SIAM Books, Philadelphia* – 2000.

5) McCormick S. F., "Mulitigrid Methods ," *SIAM, Phildelphia-Pennsylvania* -1987.

6) Briggs W. L, "Mulitigrid tutorial ," SIAM, Phildelphia-Pennsylvania – 1987.

7) G. D. Smith, "Numerical Solution of Partial Differential Equations: Finite Difference Methods.", *Oxford Applied Mathematics & Computing Science Series*, - January 16, 1986.

8) Brandt A., "Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problem," *Springer, Berlin* - 1973.