International Journal of the Physical Sciences Vol. 5(8), pp. 1277-1283, 4 August 2010
Available online at http://www.academicjournals.org/IJPS
ISSN 1992 - 1950 ©2010 Academic Journals

Full Length Research Paper

A one-way dissection of high-order compact scheme for the solution of 2D Poisson equation

F. M. Okoro* and E. A. Owoloko

Department of Mathematics, Covenant University, Ota, Ogun State, Nigeria.

Accepted 21 June, 2010

We present a one-way dissection formulation of high-order compact scheme for the solution of 2D Poisson equation. One-way dissection is a type of matrix reordering, divide and conquers procedure. Efficient and concise compact schemes of 4th and 6th orders are derived using the truncation errors of the Taylors' series expansion of the governing equation. The system is split into sub-domains and each sub-domain is treated separately. Two test problems are solved to show the fourth order performance of the scheme. The direct method is used to achieve a quick solution to the problems.

Key words: Poisson equation, one-way dissection, high-order compact scheme, sparse matrix, direct method.

INTRODUCTION

The need for finite difference schemes to meet up with the rapid updates of the computer gave rise to the compact difference schemes. Compact difference schemes are high-order implicit methods. These schemes generally require smaller stencils than the traditional explicit finite difference counterparts. The use of compact finite difference has been known for sometime (Collatz, 1996). However, their implementation as difference schemes approximating partial differential equations began in the early 1970s for some fluid mechanics problems (Gamet et al., 1999). Since that time, several distinct classes of compact schemes have been developed. In the last ten years, much work has been done with compact schemes by many authors. For instance: (Ge and Zhang, 2000; Li and Tang, 2001; Liu and Wang, 2008; Sun et al., 2003; Timothy, 2006). Lele (1992) gave an extensive analysis of compact scheme and the work spurred much interest in compact schemes. Compact scheme has been used successfully on problems described by Navier Stokes equations and also in the scattering of electro-magnetic waves (LI et al., 1995). High-order compact scheme is one most effective numerical approach to solve Poisson's equations. It has received most attention due to the advantages in solution accuracy (Spotz and Carey, 1996). Accuracy achievable with central difference is easily obtainable on coarser grids using the high-order stencils. When dealing with boundary value problems, the complete compact scheme consists of two different types of formulas. The interior formula, which is the heart of the compact scheme, approximates derivative values at all but the boundary and near boundary points. To approximate derivatives values at boundary and near boundary points, one-sided difference scheme that mimic the implicit nature and the formal order of accuracy of the interior may be used. The schemes that employ non-uniform grids are more suitable for natural boundary than those with uniform grids (Shukla and Zhong, 2005). Discretization of Poisson's equation using any of the schemes mostly leads to a sparse matrix.

Nested dissection is one of the matrix compaction techniques for reordering the entries of a sparse matrix. Other techniques include the popular minimum-degree ordering algorithm. Nested dissection ordering methods were proposed in the early 1970s and have been known since then to be theoretically superior to minimum-degree methods for important classes of sparse symmetric definite matrices. Recently, nested-dissection methods have been shown experimentally to be more effective than minimum-degree methods (Ashcraft and Liu, 1996).

*Corresponding author. E-mail:moibi1@yahoo.com. Tel: 08034683821.

Even at that, most success has only been reported on theoretical study.

The experiment with full nested dissection is in our subsequent report. In this paper, we are concerned with using one-way dissection. The splitting of the system is done in only the horizontal or vertical lines. That is, it is a top-down or left-right scheme. It uses the notion of separator whose nodes are numbered last. One-way dissection was design with parallelism in mind or has been mostly experimented on parallel computers using iterative procedures. Here, we experiment the solution of the high-order discretization of the Poisson's equation using Direct Method with LU decomposition.

FORMULATION OF HIGH-ORDER COMPACT SCHEME

Consider the Poisson equation

$$\nabla^2 u = f, \quad (1)$$

for specified forcing function f in 2D domain Ω with appropriate boundary conditions on $\partial\Omega$. Here, Ω is assumed to be a rectangular solid. We wish to approximate Equation (1) to high-order by compact scheme on a structured grid of uniform mesh size h. First, let us introduce the following notations (Wang et al., 2008).

$$\delta_+ u_i = \frac{u_{i+1} - u_i}{h} = \delta x_+ \quad and \quad \delta_- u_i = \frac{u_i - u_{i-1}}{h} = \delta x_-$$

Denote the standard forward and backward difference schemes.

Also,

$$\delta_0 u_i = \frac{1}{2}(\delta_+ u_i + \delta_- u_i) = \frac{u_{i+1} - u_{i-1}}{2h}$$

for the first-order central finite difference scheme with respect to x.

$$\delta_+ \delta_- = \delta^2 x u_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} = \frac{\delta_+ - \delta_-}{h}$$

represents the standard second order central difference scheme for the approximation to the second partial derivative of u in the x direction where $u_i = u(x_i)$. Difference operators $\delta_0 y$ and $\delta^2 y$ are defined similarly. By using Taylors' series, the first and second derivatives of fourth order and sixth order accurate finite difference can be approximated by

$$\delta_0 u = \frac{du}{dx} + \frac{h^2}{3!}\frac{d^3 u}{dx^3} = \left(1 + \frac{h^2}{6}\delta^2\right)\frac{du}{dx} + 0(h^4).$$

$$\delta_0 u = \frac{du}{dx} + \frac{h^2}{3!}\frac{d^3 u}{dx^3} + \frac{h^4}{5!}\frac{d^5 u}{dx^5} = \left(1 + \frac{h^2}{6}\delta^2 + \frac{h^4}{120}\delta^4\right)\frac{du}{dx} + o(h^6) \quad (2)$$

We may rewrite Equation (2) for $\delta_0 u$ as

$$\frac{du}{dx} = \left(1 + \frac{h^2}{6}\delta^2\right)^{-1}\delta_0 u + o(h^4)$$

$$= \left(1 + \frac{h^2}{6}\delta^2 + \frac{h^4}{120}\delta^4\right)^{-1}\delta_0 u + o\left(h^6\right). \quad (3)$$

$$\delta^2 x u_i = \frac{d^2 u}{dx^2} + \frac{h^2}{12}\frac{d^4 u}{dx^4} = \left(1 + \frac{h^2}{12}\frac{d^2}{dx^2}\right)\frac{d^2 u}{dx^2} = \left(1 + \frac{h^2}{12}\delta^2\right)\delta^2 u + o(h^4)$$

$$= \left(1 + \frac{h^2}{12}\delta^2 + \frac{h^4}{360}\delta^4\right)\delta^2 u + o(h^6). \quad (4)$$

The central difference scheme for Eq. (1) in two dimensions can be written as

$$\delta^2 x u_{ij} + \delta^2 y u_{ij} + \tau_{ij} = f_{ij}, \quad (5)$$

$$where \quad u_{ij} = u(x_i, y_j), \quad f_{ij} = f(x_i, y_j) \quad and$$

$$\tau_{ij} = \frac{h^2}{12}\left[\frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4}\right]_{ij} + \frac{h^4}{360}\left[\frac{\partial^6 u}{\partial x^6} + \frac{\partial^6 u}{\partial y^6}\right]_{ij} + o(h^6). \quad (6)$$

For the fourth-order compact scheme, we need a compact $o(h^2)$ approximation of the terms in the first square bracket in Equation (6). Accordingly, we take the appropriate derivative in Equation (1):

$$\frac{\partial^4 u}{\partial x^4} = \frac{\partial^2 f}{\partial x^2} - \frac{\partial^4 u}{\partial x^2 \partial y^2} \quad and \quad \frac{\partial^4 u}{\partial y^4} = \frac{\partial^2 f}{\partial y^2} - \frac{\partial^4 u}{\partial y^2 \partial x^2}. \quad (7)$$

Substituting Equation (7) into the first square bracket of Equation (6), we get

$$\tau_{ij} = \frac{h^2}{12}\left[\nabla^2 f - 2\frac{\partial^4 u}{\partial x^2 \partial y^2}\right]_{ij} + o(h^4). \quad (8)$$

The fourth-order compact scheme follows from substituting the above τ_{ij} into Equation (5). That is

$$\nabla^2 U_{ij} + \frac{h^2}{6}\left[\partial^2 x \partial^2 y\right] U_{ij} = f_{ij} + \frac{h^2}{12}\nabla^2 f_{ij} + o(h^4), \qquad (9)$$

Where, $u_i = U_i + o(h^4)$.

The scheme may be written explicitly as

$$\frac{1}{6}(U_{i+1,j+1} + U_{i+1,j-1} + U_{i-1,j+1} + U_{i-1,j-1}) + \frac{2}{3}(U_{i+1,j} + U_{i-1,j} + U_{i,j+1} + U_{i,j-1})$$
$$-\frac{10}{3}U_{ij} = \frac{h^2}{12}(f_{i+1,j} + f_{i-1,j} + f_{i,j+1} + f_{i,j-1} + 8f_{ij}). \qquad (10)$$

This is the well known fourth-order accurate nine-point compact scheme.

For the derivation of a sixth-order compact scheme we proceed as follows: Equation (6) is given as

$$\tau_{ij} = \frac{h^2}{12}\left[\nabla^2 f - 2\frac{\partial^4 u}{\partial x^2 \partial y^2} + \frac{h^4}{30}\left(\frac{\partial^6 u}{\partial x^6} + \frac{\partial^6 u}{\partial y^6}\right)\right]_{ij} + o(h^6).$$

(11)

We need a fourth-order approximation of $\partial^4 u / \partial x^2 \partial y^2$ in Equation (11) which can be written as

$$\left[\frac{\partial^4 u}{\partial x^2 \partial y^2}\right]_{ij} = \delta^2 x \delta^2 y u_{ij} + \frac{h^2}{12}\left[\frac{\partial^6 u}{\partial x^4 \partial y^2} + \frac{\partial^6 u}{\partial x^2 \partial y^4}\right]_{ij} + o(h^6),$$

(12)

Substituting Equation (12) into Equation (11), we get

$$\tau_{ij} = \frac{h^2}{12}\left(-\nabla^2 f_{ij} + 2\delta^2 x \delta^2 y u_{ij}\right)$$
$$-\frac{h^4}{360}\left[\frac{\partial^6 u}{\partial x^6} + 5\frac{\partial^6 u}{\partial x^4 \partial y^2} + 5\frac{\partial^6 u}{\partial x^2 \partial y^4} + \frac{\partial^6 u}{\partial y^6}\right]_{ij} + o(h^6). \qquad (13)$$

To get a compact sixth-order approximation requires compact expressions for the four derivatives of order six in Equation (13). This can be done by further differentiating Equation (7) to get the following relations:

$$\frac{\partial^4 f}{\partial x^2 \partial y^2} = \frac{\partial^6 u}{\partial x^4 \partial y^2} + \frac{\partial^6 u}{\partial x^2 \partial y^4}$$

(14)

$$\frac{\partial^6 u}{\partial x^6} + \frac{\partial^6 u}{\partial y^6} = \nabla^4 f - \frac{\partial^4 f}{\partial x^2 \partial y^2}$$

(15)

Substituting Equation (14), Equation (15) into Equation (13), we get

$$\tau_{ij} = \frac{h^2}{12}\left(-\nabla^2 f_{ij} + 2\delta^2 x \delta^2 y u_{ij}\right) - \frac{h^4}{360}\left(\nabla^4 f + 4\frac{\partial^4 f}{\partial x^2 \partial y^2}\right)_{ij} + o(h^6).$$

(16)

The compact sixth-order approximation of the 2D Poisson equation can thus be written as

$$\nabla^2 u_{ij} + \frac{h^2}{6}\delta^2 x \delta^2 y u_{ij} = f_{ij} + \frac{h^2}{12}\nabla^2 f_{ij}$$
$$+ \frac{h^4}{360}\nabla^4 f_{ij} + \frac{h^4}{90}\left[\frac{\partial^4 f}{\partial x^2 \partial y^2}\right]_{ij} + o(h^6).$$

(17)

Where ∇^2 is the Laplacian operator and ∇^4 is the biharmonic operator.

In Equation (17), we assume the derivative of f can be determined analytically. In the case where f is not known analytically, we need only a fourth-order accurate approximation of $\nabla^2 f_{ij}$ and a second-order accurate approximation of $\nabla^4 f_{ij}$ and $\left[\partial^4 f / \partial x^2 \partial y^2\right]_{ij}$. (Liu and Wang, 2008). A fourth-order accurate approximation of $\nabla^2 f_{ij}$ can be obtained using $\left(1 - \frac{h^2}{12}\delta^2\right)\delta^2 f_{ij}$ to get

$$\frac{1}{12h^2}[-f_{i+2,j} - f_{i-2,j} + 16 f_{i+1,j} + 16 f_{i-1,j} - 60 f_{ij}$$
$$+ 16 f_{i,j+1} + 16 f_{i,j-1} - f_{i,j+2} - f_{i,j-2}].$$

A second-order accurate approximation of $\nabla^4 f_{ij}$ can also be obtained as

$$\nabla^4 f_{ij} = \left[\frac{\partial^4 f}{\partial x^4} + \frac{\partial^4 f}{\partial y^4}\right]_{ij} = \frac{1}{h^2}\left(f^{iv}_{i+1,j} + f^{iv}_{i-1,j} + f^{iv}_{i,j+1} + f^{iv}_{i,j-1} - 4f^{iv}_{ij}\right)$$

Also,

$$\left[\frac{\partial^4 f}{\partial x^2 \partial y^2}\right]_{ij} = \delta^2 x \delta^2 y f_{ij} = \frac{1}{h^2}\left(f''_{i+1,j} + f''_{i-1,j} + f''_{i,j+1} + f''_{i,j-1} - 4f''_{ij}\right).$$

A nine-point $o(h^6)$ scheme may be expressed as

$$\frac{1}{6}\left(U_{i+1,j+1} + U_{i+1,j-1} + U_{i-1,j+1} + U_{i-1,j-1}\right) + \frac{2}{3}\left(U_{i+1,j} + U_{i-1,j} + U_{i,j+1} + U_{i,j-1}\right) - \frac{10}{3}U_{ij}$$
$$= h^2\left[\frac{7}{12}f_{ij} + \frac{1}{9}\left(f_{i+1,j} + f_{i-1,j} + f_{i,j+1} + f_{i,j-1}\right) - \frac{1}{144}\left(f_{i+2,j} + f_{i-2,j} + f_{i,j+2} + f_{i,j-1}\right)\right]$$
$$+ h^4\left[\frac{-2}{45}f''_{ij} + \frac{1}{90}\left(f''_{i+1,j} + f''_{i-1,j} + f''_{i,j+1} + f''_{i,j-1}\right)\right] - \frac{1}{90}f^{iv}_{ij}$$
$$+ \frac{1}{360}\left[f^{iv}_{i+1,j} + f^{iv}_{i-1,j} + f^{iv}_{i,j+1} + f^{iv}_{i,j-1}\right] \qquad (18)$$

Using the usual natural (row) ordering on Equation (10) or Equation (18) leads to a system of equations of the form

$$Ax = b. \qquad (19)$$

where A is a sparse matrix.

A ONE-WAY DISSECTION FORMULATION

In the one-way dissection, the splitting of the coefficient matrix in the system (Equation 19) is done with only horizontal or vertical separator lines. The nested dissection uses both. The coefficient matrix A is reduced to interior grid points. We partition this grids into sub-domains D_1, D_2,. . ., as well as vertical lines of grid points called the separator set. We labeled this as S_i. This is the principle of one-way dissection. Such partitioning can be taken as an example of domain decomposition. Grid points in the first sub-domain are numbered using natural ordering followed by the points in the separator set.

We have divided the unknowns into 2_{q-1} sets, D_1, D_2, - - -,D_q and S_1,- - - ,S_{q-1} such that each D_i has q unknowns and the unknowns in the S_i, 'separate' the unknowns in the D_i. The unknowns are lined up and partitioned as

$$D_1,\ S_1,\ D_2,\ S_2,\ - - -,\ S_{q-1},\ D_q. \qquad (20)$$

For q = 4, we have 7 sets and the system becomes a 30 × 30 matrix with the D_i = 4 blocks of 6 × 6 matrices and S_i = 3 blocks of 2 × 2 matrices. Equation (10) is written in corresponding order to get a system in the block form:

$$\begin{bmatrix} D & & & & & & \\ & D & & & & & \\ & & D & & & & \\ & & & D & & & \\ & & & & S & & \\ & & & & & S & \\ & & & & & & S \end{bmatrix} \qquad (21)$$

Where $D_l = \begin{bmatrix} \frac{10}{3} & \frac{-2}{3} & 0 & \frac{-2}{3} & \frac{-1}{6} & 0 \\ \frac{-2}{3} & \frac{10}{3} & \frac{-2}{3} & \frac{-1}{6} & \frac{-2}{3} & \frac{-1}{6} \\ 0 & \frac{-2}{3} & \frac{10}{3} & 0 & \frac{-1}{6} & \frac{-2}{3} \\ \frac{-2}{3} & \frac{-1}{6} & 0 & \frac{10}{3} & \frac{-2}{3} & 0 \\ \frac{-1}{6} & \frac{-2}{3} & \frac{-1}{6} & \frac{-2}{3} & \frac{10}{3} & \frac{-2}{3} \\ 0 & \frac{-1}{6} & \frac{-2}{3} & 0 & \frac{-2}{3} & \frac{10}{3} \end{bmatrix}$,
l = 1 , 2 , 3 , 4,

and $S = \begin{bmatrix} \frac{10}{3} & \frac{-2}{3} \\ \frac{-2}{3} & \frac{10}{3} \end{bmatrix}$ for the fourth-order scheme.

NUMERICAL EXAMPLES

In this section, we perform two numerical tests to determine the accuracy and efficiency of the scheme to solve the resultant system (Equation 21). The first test case was taken from (Strikwerda, 2004) with the same source term distribution and vanishing boundary conditions. Also, the second test from (Nabavi et al., 2007), modified for k = 0 and 0 < x, y < 1. The one-way dissection of the fourth order compact difference scheme is compared with the analytical solution. The maximum error (E_{max}) at each step size and at each grid point with the root-mean square error (E_s) are analyzed to access the performance of the scheme. The scheme achieved a higher nodal accuracy at a small increase in computational cost compared with some iterative

Table 1. Computational results for example 1, step size h = 1/5. E_{max} = 5.0E-4 at (2/5, 1/5); E_s = 1.0E-4; E_s/h^4 = 1.6E-7.

Location(x , y)	Analytic solution	Approx. solution	Location (x , y)	Analytic solution	Approx. solution
(1/5 , 1/5)	0.1947	0.1946	(1/5 , 3/5)	0.5534	0.5534
(2/5 , 1/5)	0.1830	0.1825	(2/5 , 3/5)	0.5201	0.5200
(3/5 , 1/5)	0.1640	0.1638	(3/5 , 3/5)	0.4660	0.4660
(4/5 , 1/5)	0.1384	0.1384	(4/5 , 3/5)	0.3934	0.3934
(1/5 , 2/5)	0.3817	0.3815	(1/5 , 4/5)	0.7031	0.7031
(2/5 , 2/5)	0.3587	0.3585	(2/5 , 4/5)	0.6607	0.6607
(3/5 , 2/5)	0.3214	0.3213	(3/5 , 4/5)	0.5921	0.5921
(4/5 , 2/5)	0.2713	0.2713	(4/5 , 4/5)	0.4998	0.4998

Table 2. Computational results for example 1, step size h = 1/6. E_{max} = 1.0E-4 at (1/6, 2/6), (3/6, 2/6) and (3/6, 5/6); E_s = 0.

Location(x , y)	Analytic solution	Approx. solution	Location(x , y)	Analytic solution	Approx. solution
(1/6 , 1/6)	0.1636	0.1636	(3/6 , 3/6)	0.4207	0.4207
(2/6 , 1/6)	0.1568	0.1568	(4/6 , 3/6)	0.3768	0.3768
(3/6 , 1/6)	0.1456	0.1456	(5/6 , 3/6)	0.3224	0.3224
(4/6 , 1/6)	0.1304	0.1304	(1/6 , 4/6)	0.6098	0.6098
(5/6 , 1/6)	0.1116	0.1116	(2/6 , 4/6)	0.5843	0.5843
(1/6 , 2/6)	0.3227	0.3226	(3/6 , 4/6)	0.5427	0.5427
(2/6 , 2/6)	0.3092	0.3092	(4/6 , 4/6)	0.4860	0.4860
(3/6 , 2/6)	0.2871	0.2872	(5/6 , 4/6)	0.4158	0.4158
(4/6 , 2/6)	0.2571	0.2571	(1/6 , 5/6)	0.7299	0.7299
(5/6 , 2/6)	0.2200	0.2200	(2/6 , 5/6)	0.6994	0.6994
(1/6 , 3/6)	0.4728	0.4728	(3/6 , 5/6)	0.6496	0.6495
(2/6 , 3/6)	0.4530	0.4530	(4/6 , 5/6)	0.5817	0.5817
			(5/6 , 5/6)	0.4977	0.4977

schemes. The experiments were performed using MATLAB 7.0.

$$E_{max} = |U_{exact} - u_{approx.}|_{ij}$$

$$E_s = \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(U_{ij}-u_{ij})^2}{(N)^2}}$$

Example 1. Consider the following Poisson's problem (Tables 1 and 2).

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = -2\cos x \sin y \qquad 0 \le x, y \le 1$$

$$U(x,0) = 0, \; U(x,1) = \cos x \sin 1;$$

$$U(0,y) = \sin y, U(1,y) = \cos 1 \sin y$$

The exact solution is $U(x,y) = \cos x \sin y.$

Example 2 (Tables 3 and 4).

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = F(x,y)$$

$$U(x,0) = U(0,y) = 0; \qquad U(x,1) = \sin\frac{\pi}{2}x, \; U(1,y) = \sin\frac{\pi}{2}y.$$

$$F(x,y) = -2\left(\frac{\pi}{2}\right)^2 \sin\frac{\pi}{2}x\sin\frac{\pi}{2}y, \qquad 0 \le x, y \le 1.$$

The exact solution is $u(x,y) = \sin\frac{\pi}{2}x\sin\frac{\pi}{2}y.$

Conclusions

We have combined two methods for the solution of 2D Poisson equation - the compact difference scheme and the one-way dissection method. Each of the methods has been successfully used to solve problems in numerical partial differential equations. The need for effective data structure in many solution algorithms motivates the

Table 3. Computational results for example 2, step size h = 1/6. E_{max} = 9.0E-4 at (3/6, 3/6); E_s = 2.6E-4; E_s/h^4 = 1.0E-4.

Location (x , y)	Analytic solution	Approx. solution	Location (x , y)	Analytic solution	Approx. solution
(1/6 , 1/6)	0.0670	0.0669	(3/6 , 3/6)	0.5000	0.4991
(2/6 , 1/6)	0.1294	0.1293	(4/6 , 3/6)	0.6124	0.6120
(3/6 , 1/6)	0.1830	0.1829	(5/6 , 3/6)	0.6830	0.6829
(4/6 , 1/6)	0.2241	0.2240	(1/6 , 4/6)	0.2241	0.2240
(5/6 , 1/6)	0.2500	0.2500	(2/6 , 4/6)	0.4330	0.4328
(1/6 , 2/6)	0.1294	0.1293	(3/6 , 4/6)	0.6124	0.6120
(2/6 , 2/6)	0.2500	0.2500	(4/6 , 4/6)	0.7500	0.7497
(3/6 , 2/6)	0.3536	0.3532	(5/6 , 4/6)	0.8365	0.8364
(4/6 , 2/6)	0.4330	0.4328	(1/6 , 5/6)	0.2500	0.2500
(5/6 , 2/6)	0.4830	0.4839	(2/6 , 5/6)	0.4830	0.4829
(1/6 , 3/6)	0.1830	0.1829	(3/6 , 5/6)	0.6830	0.6829
(2/6 , 3/6)	0.3536	0.3532	(4/6 , 5/6)	0.8365	0.8364
			(5/6 , 5/6)	0.9330	0.9329

Table 4. Computational results for example 2, step size h = 1/8. E_{max} = 1.0E-4; E_s = 0.

Location (x , y)	Analytic solution	App. solution	Location(x , y)	Analytic solution	App. solution
(1/8 , 1/8)	0.0381	0.0380	(5/8 , 4/8)	0.5879	0.5879
(2/8 , 1/8)	0.0747	0.0746	(6/8 , 4/8)	0.6533	0.6532
(3/8 , 1/8)	0.1084	0.1084	(7/8 , 4/8)	0.6935	0.6935
(4/8 , 1/8)	0.1379	0.1379	(1/8 , 5/8)	0.1622	0.1622
(5/8 , 1/8)	0.1622	0.1622	(2/8 , 5/8)	0.3182	0.3182
(6/8 , 1/8)	0.1802	0.1802	(3/8 , 5/8)	0.4619	0.4619
(7/8 , 1/8)	0.1913	0.1913	(4/8 , 5/8)	0.5879	0.5879
(1/8 , 2/8)	0.0747	0.0746	(5/8 , 5/8)	0.6913	0.6913
(2/8 , 2/8)	0.1464	0.1464	(6/8 , 5/8)	0.7682	0.7682
(3/8 , 2/8)	0.2126	0.2126	(7/8 , 5/8)	0.8155	0.8155
(4/8 , 2/8)	0.2706	0.2706	(1/8 , 6/8)	0.1802	0.1802
(5/8 , 2/8)	0.3182	0.3182	(2/8 , 6/8)	0.3536	0.3535
(6/8 , 2/8)	0.3536	0.3535	(3/8 , 6/8)	0.5133	0.5132
(7/8 , 2/8)	0.3753	0.3753	(4/8 , 6/8)	0.6533	0.6532
(1/8 , 3/8)	0.1084	0.1084	(5/8 , 6/8)	0.7682	0.7682
(2/8 , 3/8)	0.2126	0.2126	(6/8 , 6/8)	0.8536	0.8536
(3/8 , 3/8)	0.3087	0.3086	(7/8 , 6/8)	0.9061	0.9061
(4/8 , 3/8)	0.3928	0.3928	(1/8 , 7/8)	0.1913	0.1913
(5/8 , 3/8)	0.4619	0.4619	(2/8 , 7/8)	0.3753	0.3753
(6/8 , 3/8)	0.5133	0.5132	(3/8 , 7/8)	0.5449	0.5449
(7/8 , 3/8)	0.5449	0.5449	(4/8 , 7/8)	0.6935	0.6935
(1/8 , 4/8)	0.1379	0.1379	(5/8 , 7/8)	0.8155	0.8155
(2/8 , 4/8)	0.2706	0.2706	(6/8 , 7/8)	0.9061	0.9061
(3/8 , 4/8)	0.3928	0.3928	(7/8 , 7/8)	0.9619	0.9619
(4/8 , 4/8)	0.5000	0.5000			

approach. We developed a fourth and sixth order compact schemes for the Poisson equation. Using the truncation errors of the Taylor series expansion of the governing equation, a concise, more straight forward and easier to implement scheme is derived. The one-way dissection is a simple divide and conquers strategy for reordering the entries of a sparse matrix. The system is partitioned into sub-domains and each sub-domain is treated separately. The partitions lead to a block diagonal system which is solved by a direct method of solution.

Direct methods are important because of their robustness. In many cases, direct methods are often the method of choice because finding and computing a good preconditioner for an iterative method can be computationally more expensive than using a direct method. Direct removal of fill-in during Gaussian elimination (which is the best direct method in many cases) has not been achieved. However, some heuristics for fill reducing ordering have been developed (Timothy, 2006). We have used the L and U decomposition in the solution. Our interest is the accuracy of solution which is achieved with step sizes $(\frac{1}{5}, \frac{1}{6}, \frac{1}{8})$. The numerical experiments show that the method works well on the test problems and our results verify that the method is $o(h^4)$ accurate. The experiment with $o(h^6)$ compact scheme discretization using Neumann boundary conditions and with 3D problem is in our next report. This direct solver approach of the compact scheme with full-nested dissection can be used to solve problems in Mathematical Physics.

REFERENCES

Ashcraft C, Liu JWH (1996). Robust ordering of sparse matrices using multi-section, Technical Report, ISSTECH-95-024, Boeing Computer Services,

Collatz L (1996). The Numerical Treatment of Differential Equations, Springer- Verlag,

Gamet L, Ducross F, Cond FN, Poinsot T (1999). Compact finite difference scheme on non-uniform meshes. Application to direct numerical simulations of compressible flows, Int. J. Numer. Meth. Fluids, 29: 159-191.

Ge L, Zhang J (2000). Symbolic computation of high order compact difference schemes for three dimensional linear elliptic partial difference equations with variable coefficients, Technical Report, Department of Computer Science, University of Kentucky, Lexington, KY. 299-300.

Lele SK (1992). Compact finite difference schemes with spectral-like resolution, J. Comp. Phys., 103: 16-42.

Li M, Tang T (2001). A compact fourth-order finite difference scheme for unsteady viscous incompressible flows, J. Sci. Comp., 16(1): 29-45.

Li M, Tang T, Fornberg B (1995). A compact fourth-order finite difference scheme for the steady incompressible Navier-Stokes equation, Int. J. Numer. Meth., 20: 1137-1151.

Liu JG, Wang C (2008). A fourth order numerical method for the Primitive Equations formulated in Mean Vorticity, Comm. Comp. Phys., 4(1): 26-55.

Nabavi M, Saddiqui MHK, Dargahi J (2007). A new 9-point sixth-order accurate compact finite difference method for the Helmholtz equation, J. Sound Vibration, 307: 972-982.

Shukla RK, Zhong X (2005). Derivation of high order compact finite difference schemes for non-uniform grid using polynomial interpolation, J. Comp. Phys., 204: 404-429.

Spotz WF, Carey GF (1996). A high-Order Compact formulation for the 3D Poisson equation, Numer. Methods Partial Diff. Equat., 12: 235-243.

Strikwerda JC (2004). Finite difference schemes and Partial Differential Equations, 2nd Edition, Siam, Philadelphia.

Sun H, Kang N, Zhang J, Calson ES (2003). A Fourth-Order Compact difference scheme on face centered cubic grids with multi-grid methods for solving 2D convection diffusion equation, Math. Comp. Simulation, 63: 651-661.

Timothy AD (2006). Direct Methods for Sparse Linear Systems, Siam, Philadelphia, 2006.

Wang X, Yang Z, Huang G, Chen B (2008). A high-order compact difference scheme for 2D Laplace and Poisson equations in non-uniform grid systems, Comm. Nonlinear Sci. Numer. Simulation, 14: 379-398.