A formally fourth-order accurate compact scheme for 3D Poisson equation in cylindrical and spherical coordinates

Ming-Chih Lai[*] Jui-Ming Tseng
Department of Applied Mathematics
National Chiao Tung University
Hsinchu 300, TAIWAN

Abstract

In this paper, we extend our previous work (M.-C. Lai, A simple Compact Fourth-Order Poisson Solver on Polar Geometry, J. Comput. Phys., 182, 337-345 (2002)) to three-dimensional cases. In particular, we present a spectral/finite difference scheme for Poisson equation in cylindrical and spherical coordinates. The scheme relies on the truncated Fourier series expansion, where the partial differential equations of Fourier coefficients are solved by a formally fourth-order accurate compact difference discretization. Here the formal fourth-order accuracy means that the scheme is exactly fourth-order accurate while the poles are excluded and is third-order accurate otherwise. Despite the degradation of one order of accuracy due to the presence of poles, the scheme handles the poles naturally; thus, no pole condition is needed. The resultant linear system is then solved by the Bi-CGSTAB method with the preconditioner arising from the second-order discretization which shows the scalability with the problem size.

Keywords: Poisson equation; cylindrical coordinates; spherical coordinates; symmetry constraint; Fast Fourier Transform; Bi-CGSTAB method

1 Introduction

In many physical problems, one often needs to solve the Poisson equation on a three-dimensional non-Cartesian domain, such as cylindrical or spherical domains.

[*] Corresponding author. Department of Applied Mathematics, National Chiao Tung University, 1001, Ta Hsueh Road, Hsinchu 300, TAIWAN. E-mail: mclai@math.nctu.edu.tw

For example, the projection method in the simulation of incompressible flow in a pipe requires solving the pressure Poisson equation. It is convenient to rewrite the equation in those coordinates. The first problem that must be dealt with is the coordinate singularities (or poles) caused by the transformation. It is important to note that the occurrence of those singularities is due to the representation of the governing equation in those coordinates and the solution itself is regular if the source function and the boundary conditions are smooth.

For the past few years, the first author and his collaborators have developed a class of FFT-based fast direct solvers for poisson equation on 2D [1] and 3D [2] cylindrical and spherical domains. The methods have three major features, namely, the coordinate singularities can be treated easily, the resulting linear equations can be solved efficiently by existing available fast algorithms, and the different boundary conditions can be handled without substantial differences. Besides, the method is easy to implement. Despite those aforementioned advantages of our algorithm, the numerical schemes in 3D domain [2] are only second-order accurate.

Recently, the first author has developed a simple compact fourth-order Poisson solver on 2D polar geometry [3]. In fact, the scheme in [3] is formally fourth-order meaning that it has fourth-order accuracy only for the problem excluding the polar origin but degrades to third-order accuracy when the origin is included. Despite the degradation of one order of accuracy due to the presence of pole, the scheme handles the pole naturally; thus, no pole condition is needed. There are a few papers in the literature that discuss fourth-order finite difference schemes for the Poisson equation in 2D polar [6, 4, 5] and 3D cylindrical coordinates [6, 7]. However, those papers need to derive some special equations at $r = 0$ (that is, pole condition). In this article, we shall extend the methodology presented in [3] to the three-dimensional cylindrical and also spherical geometries.

2 Poisson equation in 3D cylindrical coordinates

The Poisson equation in a finite circular cylinder $\Omega = \{0 < r \leq 1, 0 \leq \theta < 2\pi, 0 \leq z \leq 1\}$ can be conveniently represented in cylindrical coordinates as

$$\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} + \frac{\partial^2 u}{\partial z^2} = f(r, z, \theta), \tag{2.1}$$

$$u(r, 1, \theta) = u_T(r, \theta), \quad u(r, 0, \theta) = u_B(r, \theta), \quad u(1, z, \theta) = u_S(z, \theta). \tag{2.2}$$

Here, we restrict the Dirichlet boundary conditions on the top, bottom and the sidewall boundaries.

2.1 Fourier mode equations

Since the solution u is periodic in θ, we can approximate it by the truncated Fourier series as

$$u(r,z,\theta) = \sum_{n=-N/2}^{N/2-1} \hat{u}_n(r,z)\, e^{in\theta}, \tag{2.3}$$

where $\hat{u}_n(r,z)$ is the complex Fourier coefficient given by

$$\hat{u}_n(r,z) = \frac{1}{N} \sum_{k=0}^{N-1} u(r,z,\theta_k)\, e^{-in\theta_k}, \tag{2.4}$$

and $\theta_k = 2k\pi/N$ with N the number of grid points along a circle. The above transformation between the physical space and Fourier space can be efficiently performed by the Fast Fourier Transform (FFT) with $O(N \log_2 N)$ arithmetic operations.

Substituting the expansions of (2.3) into Eq.(2.1), and equating the Fourier coefficients, we derive $\hat{u}_n(r,z)$ satisfying the PDE

$$\frac{\partial^2 \hat{u}_n}{\partial r^2} + \frac{1}{r}\frac{\partial \hat{u}_n}{\partial r} + \frac{\partial^2 \hat{u}_n}{\partial z^2} - \frac{n^2}{r^2}\ \hat{u}_n = \hat{f}_n(r,z), 0 < r \leq 1, 0 \leq z \leq 1. \tag{2.5}$$

$$\hat{u}_n(r,0) = \hat{u}_B^n(r), \quad \hat{u}_n(r,1) = \hat{u}_T^n(r), \quad \hat{u}_n(1,z) = \hat{u}_S^n(z). \tag{2.6}$$

Here, the nth Fourier coefficient of the right-hand side function $\hat{f}_n(r,z)$ and the boundary values $\hat{u}_S^n(z)$, $\hat{u}_T^n(r)$, $\hat{u}_B^n(r)$ are defined in a similar fashion as (2.4). In the following subsection, we shall use the notations $U(r,z) = \hat{u}_n(r,z)$ and $F(r,z) = \hat{f}_n(r,z)$, respectively.

Using the truncated Fourier series expansion, the original 3D Poisson equation (2.1) now becomes a set (N) of 2D Fourier mode equations (2.5). In fact, we only need to solve half of Fourier modes, say $n = 0, 1, \ldots, N/2-1$ since u is a real valued function and we have $u_{-n}(r,z) = \overline{u_n(r,z)}$. Furthermore, since those Fourier mode equations are fully decoupled, they can be solved in parallel. After we solve those Fourier mode equations and obtain the values of $\hat{u}_n(r,z)$, the solution $u(r,z,\theta)$ can be obtained via the inverse FFT as (2.3). In [2], we have developed a second-order finite difference scheme to solve the Fourier mode equation (2.5). In this paper, our goal is to derive a formally fourth-order accurate compact scheme for the equation (2.5).

2.2 Formally fourth-order compact difference discretization

In order to derive a fourth-order finite difference approximation to Eq.(2.5), obviously, the first and second derivatives, U_r , U_{rr} and U_{zz}, must be approximated to fourth-order accurately. To proceed, let us write down some difference formulas for the first and second derivatives with the truncation errors $O(\Delta r^4)$ and $O(\Delta z^4)$ as follows.

$$U_r = \delta_r^1 U - \frac{\Delta r^2}{6} U_{rrr} + O(\Delta r^4), \tag{2.7}$$

$$U_{rr} = \delta_r^2 U - \frac{\Delta r^2}{12} U_{rrrr} + O(\Delta r^4), \tag{2.8}$$

$$U_{zz} = \delta_z^2 U - \frac{\Delta z^2}{12} U_{zzzz} + O(\Delta z^4). \tag{2.9}$$

Here δ_r^1, δ_r^2 and δ_z^2 are the centered difference operators for the first and second derivatives, defined as

$$\delta_r^1 U_{ij} = \frac{U_{i+1,j} - U_{i-1,j}}{2\,\Delta r}, \qquad \delta_r^2 U_{ij} = \frac{U_{i+1,j} - 2U_{i,j} + U_{i-1,j}}{\Delta r^2}, \tag{2.10}$$

$$\delta_z^2 U_{ij} = \frac{U_{i,j+1} - 2\,U_{i,j} + U_{i,j-1}}{\Delta z^2}, \tag{2.11}$$

where U_{ij} are the discrete values defined at the grid points (r_i, z_j). As in [2], we choose a shifted grid to avoid the polar singularity as

$$r_i = (i - 1/2)\,\Delta r, \qquad z_j = j\,\Delta z, \tag{2.12}$$

for $1 \le i \le L+1; 0 \le j \le M+1$, with $\Delta r = 2/(2L+1)$ and $\Delta z = 1/(M+1)$. Notice that, unlike the traditional mesh [9], we do not put the grid points on the polar axis directly, thus; no pole conditions are needed.

In order to have fourth-order approximations for U_r, U_{rr} and U_{zz}, we need to approximate the higher order partial derivatives U_{rrr}, U_{rrrr} and U_{zzzz} in Eqs.(2.7), (2.8) and (2.9) to be second-order accurate. In addition, those approximations should involve at most the neighboring nine-point stencil to meet the compact requirement. To accomplish this, we differentiate Eq.(2.5) with respect to r and z and obtain the higher order partial derivatives of U as

$$U_{rrr} = F_r - \frac{U_{rr}}{r} + \frac{1+n^2}{r^2} U_r - \frac{2n^2}{r^3} U - U_{zzr}, \tag{2.13}$$

$$U_{rrrr} = F_{rr} - \frac{F_r}{r} + \frac{3+n^2}{r^2} U_{rr} - \frac{3+5n^2}{r^3} U_r + \frac{8n^2}{r^4} U + \frac{U_{zzr}}{r} - U_{zzrr}, \tag{2.14}$$

$$U_{zzzz} = F_{zz} - U_{rrzz} - \frac{U_{rzz}}{r} + \frac{n^2}{r^2} U_{zz}. \tag{2.15}$$

Now the partial derivatives U_{rrr}, U_{rrrr} and U_{zzzz} are written in terms of lower order partial derivatives which are no higher than second-order in r and z. Using the standard centered difference approximations to those lower order partial derivatives

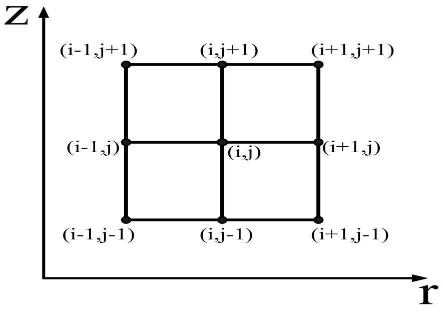

Figure 1: A nine-point compact stencil.

in Eqs.(2.13)-(2.15) and substituting those approximations into the equations (2.7)-(2.9) and (2.5), we obtain the following difference scheme

$$\begin{aligned}
&\delta_r^2 U_{i,j} - \frac{\Delta r^2}{12}[\delta_r^2 F_{i,j} - \frac{1}{r_i}\delta_r^1 F_{i,j} + \frac{3+n^2}{r_i^2}\delta_r^2 U_{i,j} - \frac{3+5n^2}{r_i^3}\delta_r^1 U_{i,j} \\
&+\frac{8n^2}{r_i^4}U_{i,j} + \frac{1}{r_i}\delta_r^1\delta_z^2 U_{i,j} - \delta_r^2\delta_z^2 U_{i,j}] + \frac{1}{r_i}\delta_r^1 U_{i,j} \\
&-\frac{\Delta r^2}{6r_i}[\delta_r^1 F_{i,j} - \frac{1}{r_i}\delta_r^2 U_{i,j} + \frac{1+n^2}{r_i^2}\delta_r^1 U_{i,j} - \frac{2n^2}{r_i^3}U_{i,j} - \delta_r^1\delta_z^2 U_{i,j}] - \frac{n^2}{r_i^2}U_{i,j} \\
&+\delta_z^2 U_{i,j} - \frac{\Delta z^2}{12}[\delta_z^2 F_{i,j} - \delta_z^2\delta_r^2 U_{i,j} - \frac{1}{r_i}\delta_z^2\delta_r^1 U_{i,j} + \frac{n^2}{r_i^2}\delta_z^2 U_{i,j}] = F_{i,j}, \qquad (2.16)
\end{aligned}$$

for $1 \leq i \leq L$, $1 \leq j \leq M$. Note that, the scheme involves centered difference approximations to first or second order partial derivatives in r and z so only a nine-point compact stencil is used, see the Figure 1 for illustration. One can also see that if the order terms of Δr^2 and Δz^2 are ignored in the equation (2.16), then the scheme recovers to the usual second-order accurate scheme as in [2].

In order to close the linear system, the numerical boundary values $U_{0,j}, U_{L+1,j}$ and $U_{i,0}, U_{i,M+1}$ should be supplied. The numerical boundary value $U_{0,j}$ can be given by $U_{0,j} = (-1)^n U_{1,j}$ due to the symmetry constraint of Fourier coefficients $(\hat{u}_n(-\Delta r/2, z_j) = (-1)^n \hat{u}_n(\Delta r/2, z_j))$ [3]. And the other numerical boundary can be easily obtained by the given Dirichlet boundary values $U_{L+1,j} = \hat{u}_S^n(z_j), U_{i,0} = \hat{u}_B^n(r_i)$

and $U_{i,M+1} = \hat{u}_T^n(r_i)$.

2.3 Numerical results

In this subsection, we perform some numerical tests on the accuracy and efficiency of our scheme. Since the matrix of the resultant linear system in (2.16) is nonsymmetric, we use the BiConjugate Gradient Stabilized method (Bi-CGSTAB) [10] to solve the linear systems. The stopping criterion of the convergence is based on the relative residual which the tolerance ranges from $10^{-9} - 10^{-13}$ depending on the different Fourier modes.

Table 1 shows the maximum relative errors for three different solutions of Poisson equation in cylindrical coordinates. In all our tests, we use L mesh points in the radial (r) and axial (z) directions, and $2L$ points in the azimuthal (θ) direction. The rate of convergence is computed by the formula $\log_2(\frac{E_{L/2}}{E_L})$, where E_L is the maximum relative error with mesh resolution L.

From Table 1, we can see that the errors of the solutions show third-order convergence for all examples in the case of solid cylinder ($0 < r \leq 1$). The loss of one order of accuracy seems to come from the discretization near the polar origin. This can be seen from the following truncation error analysis. In the Fourier mode equation (2.5), the $U_r (= \partial \hat{u}_n / \partial r)$ term is divided by r. So the second-order approximation of U_{rrr} in (2.7) is divided by an $O(\Delta r)$ term near the origin, which makes the approximation of U_{rrr}/r first-order accurate. This has the consequence that the overall truncation error of the U_r/r term in the vicinity of the origin is $O(\Delta r^3)$ and thus so is the Fourier mode equation (2.5). However, this loss of accuracy does not appear when solving the problem on a cylinder that excludes the polar singularity such as the case of $0 < a \leq r \leq 1$. Let us explain why is the case next.

As in the solid cylinder case, we need to solve Eq.(2.5) with the Dirichlet boundary condition at $r = 1$ and an additional boundary condition imposed at $r = a$. Instead of setting a grid as in (2.12), we choose a regular grid in the radial direction as

$$r_i = a + i\Delta r, \;\; i = 0, 1, ..., L, L + 1, \tag{2.17}$$

with the mesh width $\Delta r = (1 - a)/(L + 1)$. Now the second-order approximation of U_{rrr} in (2.7) is divided by an $O(a + \Delta r)$ term instead of an $O(\Delta r)$ term, so the truncation error of the U_{rrr}/r term is still $O(\Delta r^2)$ which makes the discretization error of U_r/r term is $O(\Delta r^4)$. Therefore, the overall truncation error of Eq.(2.5) is $O(\Delta r^4)$. One can see in Table 1 that the fourth-order convergence indeed can be achieved for all test examples for the case of $0.5 \leq r \leq 1$.

In order to speed up the convergence of Bi-CGSTAB iteration, we have applied different preconditioners which include Block Jacobi (BJ) [8], Incomplete LU factorization (ILU) [8], and the Fast Direct Solver (FDS) arising from the second-order discretization for the equation (2.5) which is developed in [2]. Here, we solve the

	$0 < r \leq 1$		$0.5 < r \leq 1$	
L	$\|E_L\|_\infty$	Rate	$\|E_L\|_\infty$	Rate
$u(r,z,\theta) = e^{r\cos\theta + r\sin\theta + z}$				
8	7.8137E-05		1.5465E-07	
16	9.8506E-06	2.99	1.0920E-08	3.82
32	1.2566E-06	2.97	7.3409E-10	3.90
64	1.5941E-07	2.98	4.7580E-11	3.95
$u(r,z,\theta) = r^3(\cos\theta + \sin\theta)z(1-z)$				
8	9.1438E-04		8.8994E-07	
16	1.0755E-04	3.09	6.4128E-08	3.80
32	1.3008E-05	3.05	4.3173E-09	3.89
64	1.5966E-06	3.03	2.8035E-10	3.95
$u(r,z,\theta) = \cos(\pi(r^2\cos^2\theta + r\sin\theta))\sin(\pi z^2)$				
8	7.4000E-03		7.5000E-03	
16	3.3150E-04	4.48	1.7101E-05	8.78
32	4.0782E-05	3.02	1.2221E-06	3.81
64	5.0424E-06	3.02	8.1011E-08	3.92

Table 1: The maximum relative errors for different solutions to Poisson equation in cylindrical coordinates.

difference equation (2.16) with the Fourier mode number $n = 1$. The tolerance for the relative residual is chosen as 10^{-9}.

Table 2 shows the number of iterations needed to be convergent for different preconditioners applied to Bi-CGSTAB iteration. The column of ”Bi-CGSTAB” is the one without any preconditioner which, as expected, has the largest number of iterations. The preconditioners BJ and ILU both need double iterations when the grid points are doubled. One can see that, the FDS preconditioner turns out to be the most efficient one since it has the least number of iterations, and the iterations are kept to be a constant when we double the grid points.

L	Bi-CGSTAB	BJ	ILU	FDS
8	25	16	6	7
16	52	34	9	7
32	97	65	19	7
64	182	134	38	7

Table 2: The performance comparison of different preconditioners for the cylindrical case.

3 Poisson equation in 3D spherical coordinates

In this section, we perform the similar derivation as the cylindrical case and develop a spectral/finite difference scheme for Poisson equation in 3D spherical domain. The Poisson equation with Dirichlet boundary in a spherical shell $\Omega = \{R_0 \le \rho \le 1, 0 \le \phi \le \pi, 0 \le \theta \le 2\pi\}$ can be written in spherical coordinates as

$$\frac{\partial^2 u}{\partial \rho^2} + \frac{2}{\rho}\frac{\partial u}{\partial \rho} + \frac{1}{\rho^2}\frac{\partial^2 u}{\partial \phi^2} + \frac{\cot\phi}{\rho^2}\frac{\partial u}{\partial \phi} + \frac{1}{\rho^2 \sin^2\phi}\frac{\partial^2 u}{\partial \theta^2} = f(\rho, \phi, \theta), \tag{3.18}$$

$$u(R_0, \phi, \theta) = u_I(\phi, \theta), \quad u(1, \phi, \theta) = u_S(\phi, \theta). \tag{3.19}$$

Here, the boundary condition should be imposed on the inner ($\rho = R_0 > 0$) and outer ($\rho = 1$) surfaces of the sphere.

As the cylindrical case, the main difficulty for solving Eq.(3.18) is to treat the coordinate singularities along the polar axis where north ($\phi = 0$) and south ($\phi = \pi$) poles are located. Most of numerical approaches including finite difference and spectral methods involve imposing additional pole conditions to capture the behavior of the solution in the vicinity of the poles. In [1, 2], we have developed a series of FFT-based second-order fast Poisson solver without pole conditions for 2D and 3D spherical domains. In the following, we will develop a formally fourth-order compact scheme for Eqs. (3.18)-(3.19). To the best of our knowledge, we have not seen any related work in the literature.

3.1 Fourier mode equations

As in the cylindrical coordinate case, we approximate u by the truncated Fourier series as

$$u(\rho, \phi, \theta) = \sum_{n=-N/2}^{N/2-1} \hat{u}_n(\rho, \phi)\, e^{in\theta}, \tag{3.20}$$

where $\hat{u}_n(\rho, \phi)$ is the complex Fourier coefficient given by

$$\hat{u}_n(\rho, \phi) = \frac{1}{N}\sum_{k=0}^{N-1} u(\rho, \phi, \theta_k)\, e^{-in\theta_k}, \tag{3.21}$$

and $\theta_k = 2k\pi/N$ and N is the number of grid points along a latitude circle. Once again, the above transformation between the physical space and Fourier space can be efficiently performed by the Fast Fourier Transform (FFT) with $O(N \log_2 N)$ arithmetic operations. The expansion for the function f can be written in the similar fashion.

Substituting those expansions into Eq.(3.18), and equating the Fourier coefficients, $\hat{u}_n(\rho, \phi)$ then satisfies the PDE

$$\frac{\partial^2 \hat{u}_n}{\partial \rho^2} + \frac{2}{\rho}\frac{\partial \hat{u}_n}{\partial \rho} + \frac{1}{\rho^2}\frac{\partial^2 \hat{u}_n}{\partial \phi^2} + \frac{\cot\phi}{\rho^2}\frac{\partial \hat{u}_n}{\partial \phi} - \frac{n^2}{\rho^2 \sin^2\phi}\hat{u}_n = \hat{f}_n(\rho, \phi), \tag{3.22}$$

$$\hat{u}_n(R_0, \phi) = \hat{u}_I^n(\phi), \quad \hat{u}_n(1, \phi) = \hat{u}_S^n(\phi). \tag{3.23}$$

Here, $\hat{u}_I^n(\phi)$ and $\hat{u}_S^n(\phi)$ are the nth Fourier coefficient of $u_I(\phi, \theta)$ and $u_S(\phi, \theta)$, respectively. Next, we need to derive a formally fourth-order compact scheme for the Fourier mode equations (3.22).

3.2 Formally fourth-order compact difference discretization

As in [2], we choose a grid in (ρ, ϕ) plane by

$$\rho_i = R_0 + i\,\Delta r, \qquad \phi_j = (j - 1/2)\,\Delta\phi, \tag{3.24}$$

for $0 \le i \le L+1, 0 \le j \le M+1$ with $\Delta\rho = (1 - R_0)/(L+1)$ and $\Delta\phi = \pi/M$. By the choice of those mesh points, we avoid placing points directly at north ($\phi = 0$) and south ($\phi = \pi$) poles. Again, let the discrete values be denoted by $U(\rho_i, \phi_j) \approx \hat{u}_n(\rho_i, \phi_j)$, and $F(\rho_i, \phi_j) \approx \hat{f}_n(\rho_i, \phi_j)$.

Our goal is to derive a fourth-order finite difference approximation to Eq.(3.22). Obviously, the first and second derivatives, U_ρ, $U_{\rho\rho}$, U_ϕ and $U_{\phi\phi}$ must be approximated to fourth-order accurately. In order to have fourth-order approximations for U_ρ, $U_{\rho\rho}$, U_ϕ and $U_{\phi\phi}$, we need to approximate the higher order derivatives $U_{\rho\rho\rho}$, $U_{\rho\rho\rho\rho}$, $U_{\phi\phi\phi}$ and $U_{\phi\phi\phi\phi}$ to be second-order accurate. We then reduce those higher order partial derivatives to lower order by differentiating the equation (3.22) and use the regular centered difference to approximate those lower order partial derivatives. The derivation is very similar to the cylindrical case, so we omit the detail here. After a tedious calculation, we obtain a finite difference scheme as follows.

For $1 \leq i \leq L, 1 \leq j \leq M$, we have

$$\begin{aligned}
&\delta_\rho^2 U_{i,j} - \frac{\Delta\rho^2}{12}[\delta_\rho^2 F_{i,j} - \frac{2}{\rho_i}\delta_\rho^1 F_{i,j} + (\frac{8 + n^2\csc^2\phi_j}{\rho_i^2})\delta_\rho^2 U_{i,j} + \frac{10n^2\csc^2\phi_j}{\rho_i^4}U_{i,j} \\
&+(\frac{-8-6n^2\csc^2\phi_j}{\rho_i^3})\delta_\rho^1 U_{i,j} - \frac{10}{\rho_i^4}\delta_\phi^2 U_{i,j} - \frac{10\cot\phi_j}{\rho_i^4}\delta_\phi^1 U_{i,j} + \frac{6\cot\phi_j}{\rho_i^3}\delta_\rho^1\delta_\phi^1 U_{i,j} \\
&+\frac{6}{\rho_i^3}\delta_\rho^1\delta_\phi^2 U_{i,j} - \frac{\cot\phi_j}{\rho_i^2}\delta_\rho^2\delta_\phi^1 U_{i,j} - \frac{1}{\rho_i^2}\delta_\rho^2\delta_\phi^2 U_{i,j}] + \frac{2}{\rho_i}\{\delta_\rho^1 U_{i,j} - \frac{\Delta r^2}{6}[\delta_\rho^1 F_{i,j} \\
&-\frac{2}{\rho_i}\delta_\rho^2 U_{i,j} + (\frac{2 + n^2\csc^2\phi_j}{\rho_i^2})\delta_\rho^1 U_{i,j} + \frac{2\cot\phi_j}{\rho_i^3}\delta_\phi^1 U_{i,j} + \frac{2}{\rho_i^3}\delta_\phi^2 U_{i,j} - \frac{\cot\phi_j}{\rho_i^2}\delta_\rho^1\delta_\phi^1 U_{i,j} \\
&-\frac{1}{\rho_i^2}\delta_\rho^1\delta_\phi^2 U_{i,j} - \frac{2n^2\csc^2\phi_j}{\rho_i^3}U_{i,j}]\} + \frac{1}{\rho_i^2}\{\delta_\phi^2 U_{i,j} - \frac{\Delta\phi^2}{12}[\rho_i^2\delta_\phi^2 F_{i,j} - \rho_i^2\cot\phi_j\delta_\phi^1 F_{i,j} \\
&+((3+n^2)\csc^2\phi_j - 1)\delta_\phi^2 U_{i,j} + (-3-5n^2)\csc^2\phi_j\cot\phi_j\delta_\phi^1 U_{i,j} \\
&+2n^2\csc^2\phi_j(4\csc^2\phi_j - 3)U_{i,j} + 2\rho_i\cot\phi_j\delta_\phi^1\delta_\rho^1 U_{i,j} + \rho_i^2\cot\phi_j\delta_\phi^1\delta_\rho^2 U_{i,j} \\
&-2\rho_i\delta_\phi^2\delta_\rho^1 U_{i,j} - \rho_i^2\delta_\phi^2\delta_\rho^2 U_{i,j}]\} + \frac{\cot\phi_j}{\rho_i^2}\{\delta_\phi^1 U_{i,j} - \frac{\Delta\phi^2}{6}[-\rho_i^2\delta_\phi^1\delta_\rho^2 U_{i,j} - 2\rho_i\delta_\phi^1\delta_\rho^1 U_{i,j} \\
&+(1+n^2)\csc^2\phi_j\delta_\phi^1 U_{i,j} - \cot\phi_j\delta_\phi^2 U_{i,j} - 2n^2\csc^2\phi_j\cot\phi_j U_{i,j} + \rho_i^2\delta_\phi^1 F_{i,j}]\} \\
&-\frac{n^2\csc^2\phi_j}{\rho_i^2}U_{i,j} = F_{i,j}. \qquad (3.25)
\end{aligned}$$

Note that, the scheme involves centered difference approximations to first or second order partial derivatives in ρ and ϕ so only a nine-point compact stencil is used. One can also see that if the order terms of $\Delta\rho^2$ and $\Delta\phi^2$ are ignored in the equation (3.25), then the scheme recovers to the usual second-order accurate scheme as in [2].

When $j = 1$ for Eq.(3.25), the numerical boundary value $U_{i,0}$ can be given by $U_{i,0} = (-1)^n U_{i,1}$. This is because the Fourier coefficient satisfies the symmetry constraint $\hat{u}_n(\rho_i, -\Delta\phi/2) = (-1)^n \hat{u}_n(\rho_i, \Delta\phi/2)$ [2]. Similarly, another numerical boundary value $U_{i,M+1}$ can also be obtained by $U_{i,M+1} = (-1)^n U_{i,M}$ for the same reason. So the numerical boundary values in the ϕ direction are provided and no pole condition is needed in our finite difference setting. The other numerical boundary $U_{0,j}, U_{L+1,j}$ are given by the boundary values $\hat{u}_I^n(\phi_j), \hat{u}_S^n(\phi_j)$.

3.3 Numerical results

In this subsection, we perform similar numerical tests on the accuracy and efficiency of our scheme for the spherical coordinates case. In all our tests, we use L mesh points in the radial (ρ) and colatitude (ϕ) directions, and $2L$ points in the longitude (θ) direction. The inner radius is chosen as $R_0 = 0.5$. The difference equation (3.25) is solved by the Bi-CGSTAB method for $n = 0, 1, \ldots L-1$ where the tolerance of stopping ranges from $10^{-9} - 10^{-13}$ depending on the different Fourier modes. Once we obtain the Fourier coefficients, the numerical solution can be computed by the inverse FFT as in (3.20). Table 3 shows the maximum errors of the method for three different solutions of Poisson equation in a spherical shell. One can see that the errors of the solutions show third-order convergence for all test examples. The reason of losing one order of accuracy is exactly the same as the cylindrical case so we omit the discussion here.

Table 4 shows the number of iterations needed for solving the difference equation (3.25) of $n = 1$ by the Bi-CGSTAB method with different preconditioners. The tolerance of stopping for the relative residual is chosen as 10^{-9}. Generally speaking, the performance of those preconditioners are almost the same as the cylindrical case. It is interesting to see that the ILU preconditioner seems to have the less number of iterations than the FDS when the number of grid points is small. However, the ILU does increase the iterations as the grid points increasing while the FDS preconditioner still keeps the number of iterations.

L	$\|E_L\|_\infty$	Rate
$u(\rho,\phi,\theta) = e^{\rho\sin\phi\cos\theta+\rho\sin\phi\sin\theta+\rho\cos\phi}$		
8	3.7000E-03	
16	5.4095E-04	2.77
32	7.4825E-05	2.85
64	1.0067E-05	2.89
$u(\rho,\phi,\theta) = \rho^3(\cos\theta+\sin\theta)\sin\phi(1-\rho\cos\phi)$		
8	5.2000E-03	
16	9.5143E-04	2.45
32	1.5249E-04	2.64
64	2.2549E-05	2.76
$u(\rho,\phi,\theta) = \cos(\pi(\rho^2\cos^2\theta\sin^2\phi+\rho\sin\theta\sin\phi))\sin(\pi\rho^2\cos^2\phi)$		
8	5.3700E-02	
16	7.1000E-03	2.92
32	8.3568E-04	3.09
64	9.7240E-05	3.10

Table 3: The maximum relative errors for different solutions to Poisson equation in spherical coordinates.

4 Conclusion and acknowledgement

In this paper, we present a formally fourth-order compact difference scheme for 3D Poisson equation in cylindrical and spherical coordinates. The solver relies on the

L	Bi-CGSTAB	BJ	ILU	FDS
8	30	27	3	11
16	56	54	5	13
32	99	106	7	13
64	196	191	14	13

Table 4: The performance comparison of different preconditioners for the spherical case.

truncated Fourier series expansion, where the partial differential equations of Fourier coefficients are solved by a formal fourth-order compact difference discretizations without pole conditions. The resultant linear system is then solved by the Bi-CGSTAB method with different preconditioners. The numerical results confirm the formal accuracy of our scheme. Meanwhile, the preconditioner arising from the second-order fast direct solver shows the scalability of Bi-CGSTAB with the problem size.

The authors are supported in part by the National Science council of Taiwan under research grant NSC-93-2115-M-009-008.

References

[1] M.-C. Lai and W.-C. Wang, Fast direct solvers for Poisson equation on 2D polar and spherical gepmetries. *Numer. Meth. Partial Diff. Eq.*, **18**, 56-68, (2002).

[2] M.-C. Lai, W.-W. Lin and W. Wang, A fast spectral/difference method without pole conditions for Poisson-type equations in cylindrical and spherical geometries. *IMA J. Numer. Anal.*, **22**, 537-548, (2002).

[3] M.-C. Lai, A simple compact fourth-order Poisson solver on polar geometry. *J. Comput. Phys.*, **182**, 337-345, (2002).

[4] R. C. Mittal and S. Gahlaut, High-order finite-difference schemes to solve Poisson's equation in polar coordinates, *IMA J. Numer. Anal.*, **11**, 261, (1991).

[5] M. K. Jain, R. K. Jain and M. Krishna, A fourth-order difference scheme for quasilinear Poisson equation in polar co-ordinates, *Communications in Numerical Methods in Engineering*, vol. **10**, 791-797 (1994).

[6] S. R. K. Iyengar and R. Manohar, High order difference methods for heat equation in polar cylindrical coordinates, *J. Comput. Phys.*, **77**, 425 (1988).

[7] S. R. K. Iyengar and A. Goyal, A note on multigrid for the three-dimensional Poisson equation in cylindrical coordinates, *J. Comput. Appl. Math.*, **33**, 163-169, (1990).

[8] B. Richard et al. *Templates for the solution of linear systems:Building blocks for iterative methods*, SIAM (1994).

[9] J. C. Strikwerda, *Finite Difference schemes and Partial Differential Equations*, Page 286-287, Wadsworth & Brooks/Cole, (1989).

[10] H. A. Van der Vorst, Bi-CGSTAB : A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, **13**, 631-644, (1992).