Journal of Computational Physics **160,** 453–469 (2000)
doi:10.1006/jcph.2000.6461, available online at http://www.idealibrary.com on IDEAL®

A Direct Spectral Collocation Poisson Solver in Polar and Cylindrical Coordinates

Heli Chen, Yuhong Su, and Bernie D. Shizgal[1]

Institute of Applied Mathematics, University of British Columbia, Vancouver, British Columbia, Canada
E-mail: hchen@math.ubc.ca, George@seagatesoftware.com, shizgal@theory.chem.ubc.ca

Received September 17, 1999; revised January 28, 2000

In this paper, we present a direct spectral collocation method for the solution of the Poisson equation in polar and cylindrical coordinates. The solver is applied to the Poisson equations for several different domains including a part of a disk, an annulus, a unit disk, and a cylinder. Unlike other Poisson solvers for geometries such as unit disks and cylinders, no pole condition is involved for the present solver. The method is easy to implement, fast, and gives spectral accuracy. We also use the weighted interpolation technique and nonclassical collocation points to improve the convergence.

Key Words: Poisson solver; spectral collocation; polar coordinates; cylindrical coordinates.

1. INTRODUCTION

There is considerable interest in the development of robust and efficient numerical methods of solution of the Poisson equation (PE). A fast and accurate Poisson solver would find immediate application in diverse fields. These include computer simulation of plasma physics [1] and industrial plasma engineering [2] as well as galactic dynamics [3]. Also, in computational fluid dynamics, the solution of the Navier–Stokes equation involves the solution of the Poisson equation for the pressure [4]. The present work was motivated by the need for a fast Poisson solver in gaseous electronics [5–7]. Most Poisson solvers are based on finite difference and finite element methods. Spectral methods based on Fourier, Chebyshev, and Legendre basis functions have also been used [8–17]. In the present paper, we develop a direct spectral collocation solver for the solution of the 2D Poisson equation

[1] Also with Department of Chemistry, University of British Columbia, Vancouver, British Columbia, Canada V6T 1Z1.

0021-9991/00 $35.00

in polar coordinates written as

$$\frac{\partial^2\phi}{\partial r^2}+\frac{1}{r}\frac{\partial\phi}{\partial r}+\frac{1}{r^2}\frac{\partial^2\phi}{\partial\theta^2}=\rho(r,\theta) \tag{1}$$

for several domains which include a part of a disk, a circular annulus, and a whole disk. The solution of the Poisson equation for these domains was recently reported [8, 12, 13, 15, 16]. We also consider a 3D Poisson equation

$$\frac{\partial^2\phi}{\partial r^2}+\frac{1}{r}\frac{\partial\phi}{\partial r}+\frac{1}{r^2}\frac{\partial^2\phi}{\partial\theta^2}+\frac{\partial^2\phi}{\partial z^2}=\rho(r,\theta,z) \tag{2}$$

in cylindrical geometry. We are interested in developing a spectral method of solution of the Poisson equation for these geometries subject to specific boundary conditions.

Most Poisson solvers are based on the Poisson equation written in Cartesian coordinates. For problems in 2D and 3D domains, these solvers often involve a transformation of the polar domain to a rectangular domain or the cylindrical domain to a cubic domain, and the Poisson equation is then solved in Cartesian coordinates [15, 16, 18, 19]. An important aspect of the choice of a numerical method of solution of the Poisson equation is the analytic behavior of the solution in the vicinity of the origin and the appropriateness of the numerical approach to approximate accurately this behavior. The terms in $1/r$ and $1/r^2$ in Eqs. (1) and (2) can lead to numerical singularities. These singularities are often responsible for the very slow convergence of numerical solutions. Often the convergence can be accelerated significantly by imposition of additional boundary conditions called pole conditions. This has been discussed by Gottlieb and Orszag [20] and by Huang and Sloan [13]. If the solution domain is a part of a disk or an annulus disk, the problem is well defined and no coordinate singularity is involved. If the solution domain is a whole disk or a cylinder, the main difficulty for solving these problems is to treat the coordinate singularity along the polar axis at the center, $r=0$. Most Poisson solvers for polar and cylindrical domains involve additional pole conditions to capture the behavior of the solution as $r\to 0$ obtained by an asymptotic analysis which varies from problem to problem [13, 15, 16]. Discussions about various pole conditions can be found in many references [11, 13, 15, 16, 20–22]. In this paper, we will present a direct, simple algorithm that does not involve specifying a pole condition. We use a set of collocation points excluding the center at $r=0$, and hence the singularity is avoided. Canuto *et al.* [4] and Shen [16] mentioned this idea in their work, but to the best of our knowledge, there are no results reported in the literature with such an approach. It is useful to mention that there should be situations for which particular singularities must be taken into account explicitly in terms of pole conditions [4, 20].

For the Poisson equation on a circular disk or a cylinder, the solution is periodic in θ. Therefore, we choose a Fourier collocation method for the discretization in θ. We use a spectral collocation method related to a set of Gauss–Radau points which excludes $r=0$ in the radial direction. The resulting discretized Poisson equation can be written in a form of a tensor product with the substitution of derivative matrices in each variable. The resulting algebraic equations can be solved with a two-step eigenvalue technique described elsewhere [18, 23, 24]. The method is direct, simple, and fast.

Most calculations using spectral methods in the literature are based on classical polynomials (especially Chebyshev and Legendre) for the discretization of the Poisson equation in the polar direction [8, 13, 15, 16, 19]. In this paper, we also introduce collocation points

based on nonclassical polynomials and demonstrate an improved convergence for some cases.

This paper is organized as follows. In the next section, we will present details of the numerical method and its implementation to the Poisson equation in several domains. The results and their discussions are presented in Section 3. All the results presented in this paper are calculated with MATLAB 5.3 on a HP700 computer. The MATLAB built-in functions are called whenever available to keep our m-program simple and easy to use.

2. NUMERICAL METHOD

2.1. *Spectral Collocation Method*

The numerical method presented in this paper is a direct spectral collocation method. The derivative operators in the Poisson equation, Eqs. (1) or (2), are substituted by the discretized derivative matrices. For periodic problems, we consider a Fourier approximation which is usually used for the discretization in the θ direction. The method is refered to as Fourier collocation by Canuto *et al.* [4] and as pseudospectral Fourier by Gottlieb *et al.* [25]. We adopt the Fourier collocation by Canuto *et al.* Let N be an even number; collocation points θ_j are evenly spaced and defined as follows

$$\theta_j = \frac{2\pi j}{N} \quad j = 0, 1, \ldots, N-1. \tag{3}$$

The explicit formulae for the first and second Fourier derivative matrices can be found in references [4, 25, 26].

For nonperiodic problems, we use a spectral collocation method based on weighted interpolants [27], which approximates the solution by

$$u(x) \approx \sum_{j=0}^{N} \frac{v(x)}{v(x_j)} l_j(x) u_j, \quad a \le x \le b, \tag{4}$$

where $\{x_j\}_{j=0}^{N}$ is a set of collocation points in $[a, b]$, $v(x)$ is some weight function, $u_j = u(x_j)$, and $\{l_j(x)\}_{j=0}^{N}$ is a set of Lagrange interpolation polynomials of degree N defined as

$$l_j(x) = \prod_{i=0, i\neq j}^{N} \frac{x - x_j}{x_i - x_j}, \tag{5}$$

which satisfies $l_j(x_k) = \delta_{ik}$.

The mth derivative matrix $\mathbf{D}^{(m)}$ is defined by taking the mth derivative of the interpolant $(v(x)/v(x_j))l_j(x)$ and evaluating it at the collocation points; that is

$$D_{ij}^{(m)} = \frac{d^m}{dx^m}\left[\frac{v(x)}{v(x_j)} l_j(x)\right]_{x=x_i}. \tag{6}$$

For a spectral collocation method, the nodes x_j are associated with a set of orthogonal polynomials with respect to a weight function $w(x)$, which is not necessarily equal to $v(x)$ in Eq. (4). One of the most popular polynomial sets is the Chebyshev polynomials

$T_n(x) = \cos(n \cos^{-1} x)$ in $[-1, 1]$. Two sets of associated collocation points used in this paper are the Chebyshev Gauss-Lobatto points defined by

$$x_j = \cos \frac{\pi j}{N}, \quad (j = 0, \ldots, N), \tag{7}$$

and the Chebyshev Gauss–Radau points defined by

$$y_j = \cos \frac{\pi j}{N+1}, \quad (j = 0, \ldots, N). \tag{8}$$

The corresponding derivative matrices can be calculated with Eq. (6). Their explicit forms with $v(x) = 1$ can be found in references [4, 25, 28, 29]. (Note: In Ref. [25, p. 15], the equation of the derivative matrix for Chebyshev Gauss–Radau should be $\tilde{D}_{ij}^{(1)} = ((1 + y_j)/(1 + y_i))D_{ij}^{(1)} - (\delta_{ij})/(1 + y_i)$. The second term is missing).

In spectral methods, most collocation points used are associated with the classical polynomials such as Chebyshev and Legendre polynomials. However, they are not necessarily the best choice that would give a rapid convergence of the solution. In this paper, we also use collocation points based on nonclassical polynomials introduced in our previous work [30–32]. This nonclassical spectral method, referred to as the quadrature discretization method (QDM), involves the calculation of the quadrature points and weights with Gautschi's Stieltjes procedure [33]. The details on how to generate QDM Gauss points can be found in our previous papers. Let $\{P_n(x)\}_{n=0}^N$ be a set of orthogonal polynomials with respect to an arbitrary weight function $w(x)$ on an interval [a, b]. The QDM Lobatto points are defined by zeros of P_N, x_i, $(i = 1, 2, \ldots, N)$, plus the two end points $x_0 = a$ and $x_{N+1} = b$. The QDM Radau points are defined similarly except only one of the end points $x_0 = a$ or $x_{N+1} = b$ is included.

The function $v(x)$ in Eq. (4) is an arbitrary function ($v(x) \neq 0$). For a "regular" spectral method, $v(x) = 1$. We sometimes can choose $v(x)$ such that the function $\frac{u(x)}{v(x)}$ can be accurately approximated by a polynomial basis set so that the solution $u(x)$ converges more rapidly. For $v(x) \neq 1$, we call the approach a *weighted* method. The function $v(x)$ is not necessarily the same as the weight function $w(x)$ related to the orthogonal polynomial basis set. However, sometimes it is useful to choose $v(x) = w(x)$ or $v(x)$ as a function associated with $w(x)$ [30, 32].

2.2. *Discretization of the Poisson Equation in Polar Coordinates*

To implement our solver, we first rewrite the Poisson equation, Eq. (1), in the form

$$r^2 \frac{\partial^2 \phi}{\partial r^2} + r \frac{\partial \phi}{\partial r} + \frac{\partial^2 \phi}{\partial \theta^2} = f(r, \theta), \tag{9}$$

where $f(r, \theta) = r^2 \rho(r, \theta)$. Substituting derivative matrices in both r and θ, we can write the discretized form of the above equation as the following,

$$\sum_{k=0}^{N_r} A_{ik} \phi_{kj} + \sum_{k=0}^{N_\theta} \phi_{ik} B_{kj} = f_{ij}, \tag{10}$$

where $f_{ij} = f(r_i, \theta_j)$ and $\phi_{ij} = \phi(r_i, \theta_j)$. The matrix $\mathbf{A}$, which involves derivatives in r, operates on $\phi(r_k, \theta_j)$ from the left and is defined as

$$A_{ik} = r_i^2 D_{r,ik}^{(2)} + r_i D_{r,ik}^{(1)}. \tag{11}$$

The matrix **B**, which involves derivatives in θ, operates on $\phi(r_k, \theta_j)$ from the right and is defined as

$$B_{kj} = D^{(2)}_{\theta,jk}. \tag{12}$$

$\mathbf{D}_r$ and $\mathbf{D}_\theta$ are the derivative matrices with respect to r and θ, respectively. To take account of boundary conditions, we can transfer the known quantities $f^{(BC)}_{ij}$ on the left-hand side of Eq. (10) to the right-hand side and redefine $F_{ij} = f_{ij} - f^{(BC)}_{ij}$ and Eq. (10) now reads

$$\sum_k A_{ik}\phi_{kj} + \sum_k \phi_{ik}B_{kj} = F_{ij}. \tag{13}$$

The dummy summations indices in Eq. (13) are defined later and depend on the specific domain and boundary conditions.

For a part of a disk with Dirichlet boundary conditions, we use Lobatto collocation in both directions and we have

$$\sum_{k=1}^{N_r-1} A_{ik}\phi_{kj} + \sum_{k=1}^{N_\theta-1} \phi_{ik}B_{kj} = F_{ij}, \tag{14}$$

where $F_{ij} = f_{ij} - f^{(BC)}_{ij}$ with $f^{(BC)}_{ij} = A_{i0}\phi_{0j} + A_{iN_r}\phi_{N_r j} + \phi_{i0}B_{0j} + \phi_{iN_\theta}B_{N_\theta j}$. For an annulus with Dirichlet boundary conditions, we use Lobatto collocation in r and Fourier collocation in θ and we have

$$\sum_{k=1}^{N_r-1} A_{ik}\phi_{kj} + \sum_{k=0}^{N_\theta} \phi_{ik}B_{kj} = F_{ij}, \tag{15}$$

and $f^{(BC)}_{ij} = A_{i0}\phi_{0j} + A_{iN_r}\phi_{N_r j}$. For a whole disk with a Dirichlet boundary condition at the edge $r = R$, we use Radau collocation in r and Fourier collocation in θ and we have

$$\sum_{k=0}^{N_r-1} A_{ik}\phi_{kj} + \sum_{k=0}^{N_\theta} \phi_{ik}B_{kj} = F_{ij}, \tag{16}$$

and $f^{(BC)}_{ij} = A_{iN_r}\phi_{N_r j}$.

2.3. *Two-Step Direct Solver*

The algebraic equations (14), (15), and (16) can be written in a more general form,

$$\sum_{k=1}^{M} \tilde{A}_{ik}\Phi_{kj} + \sum_{k=1}^{N} \Phi_{ik}\tilde{B}_{kj} = \tilde{F}_{ij}, \tag{17}$$

or

$$\tilde{\mathbf{A}} \cdot \mathbf{\Phi} + \mathbf{\Phi} \cdot \tilde{\mathbf{B}} = \tilde{\mathbf{F}}, \tag{18}$$

where $\tilde{\mathbf{A}}$ is a $M \times M$ matrix related to **A** and $\tilde{\mathbf{B}}$ is a $N \times N$ matrix related to **B**. $\mathbf{\Phi}$ is the $M \times N$ solution maxtrix. $\tilde{\mathbf{F}}$ is the $M \times N$ matrix of the right-hand side. The above equation

can be solved directly in terms of eigenvalues and eigenvectors of matrices $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$ [18, 23, 24]. The original method for the solution of Eqs. (17) or (18) is due to Lynch *et al.* [23]. The method has been modified by Haidvogel and Zang [24] and also by Zhao and Yedlin [18]. The method is summarized as follows. Let the eigenvalues of $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$ be λ and ε, respectively, and associated eigenvector matrices $\mathbf{P}$ and $\mathbf{Q}$; that is,

$$\sum_{k=1}^{M} \tilde{A}_{ik} P_{kn} = \lambda_n P_{in}, \tag{19}$$

and

$$\sum_{k=1}^{N} \tilde{B}_{jk} Q_{kn} = \varepsilon_n Q_{jn}. \tag{20}$$

If we multiply Eq. (17) with $\mathbf{Q}$ from the right and use Eq. (20), we get

$$\sum_{k=1}^{M}\sum_{j=1}^{N} \tilde{A}_{ik}\Phi_{kj} Q_{jn} + \varepsilon_n \sum_{k=1}^{N} \Phi_{ik} Q_{kn} = \sum_{j=1}^{N} \tilde{F}_{ij} Q_{jn}. \tag{21}$$

Now let the inverse of the matrices $\mathbf{P}$ and $\mathbf{Q}$ be denoted by $\mathbf{P}^{-1}$ and $\mathbf{Q}^{-1}$, respectively. Multipling Eq. (21) on the left with $\mathbf{P}^{-1}$ and using Eq. (19) in adjoint form, we get

$$\lambda_l \sum_{k=1}^{M}\sum_{j=N}^{M} P^{-1}_{lk}\Phi_{kj} Q_{jn} + \varepsilon_n \sum_{i=1}^{M}\sum_{k=1}^{N} P^{-1}_{li}\Phi_{ik} Q_{kn} = \sum_{i=1}^{M}\sum_{j=1}^{N} P^{-1}_{li}\tilde{F}_{ij} Q_{jn}. \tag{22}$$

We note that the two summations in Eq. (22) are the same except for different dummy summation indices, so we can get that

$$\sum_{i=1}^{M}\sum_{j=1}^{N} P^{-1}_{li}\Phi_{ij} Q_{jn} = \frac{1}{\lambda_l + \varepsilon_n} \sum_{i=1}^{M}\sum_{j=1}^{N} P^{-1}_{li}\tilde{F}_{ij} Q_{jn}. \tag{23}$$

We denote the right-hand side of Eq. (23) by G_{ln} and to recover the solution $\mathbf{\Phi}$ we simply multiply Eq. (23) on the right by $\mathbf{Q}^{-1}$ and on the left by $\mathbf{P}$ and find that

$$\Phi_{mp} = \sum_{l=1}^{M}\sum_{n=1}^{N} P_{ml} G_{ln} Q^{-1}_{np}. \tag{24}$$

In summary, the evaluation of the solution $\mathbf{\Phi}$ involves a two-step algorithm.
The first step is to calculate the elements of matrix $\mathbf{G}$,

$$G_{ln} = \frac{1}{\lambda_l + \varepsilon_n} \sum_{i=1}^{M}\sum_{j=1}^{N} P^{-1}_{li} F_{ij} Q_{jn}, \tag{25}$$

followed by the second step given by

$$\mathbf{\Phi} = \mathbf{P} \cdot \mathbf{G} \cdot \mathbf{Q}^{-1}. \tag{26}$$

The operation count for the preprocessing step Eq. (25) is $2MN(M+N)$. This is followed by Eq. (26) with an operation count of $2MN(M+N-1)$.

The eigenvalues and the eigenfunctions of $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$ and the inverse of the eigenfunctions need to be calculated only once. This is referred to as the preprocessing stage of the calculation. For a given problem, if the equation is to be solved many times, the computation time for the preprocessing stage (which is done only once) is then negligible compared with the total time involved. This can be important for time dependent computations involving an iteration.

It should be pointed out that this two-step technique only works for the separable equation of the form given by Eq. (17). This direct eigenvalue technique can also be used for 3D problems if the equation can be discretized as given by

$$\sum_m A_{im}U_{mjk} + \sum_m B_{jm}U_{imk} + \sum_m C_{km}U_{ijm} = F_{ijk}, \tag{27}$$

where **A**, **B**, and **C** are square matrices associated with differential operators in each of three dimensions and **U** is the 3D solution matrix. Equation (27) can be solved by the two-step method. Let λ, ϵ, and ω be the eigenvalues and **P**, **Q**, and **R** be the eigenvector matrices of matrices **A**, **B**, and **C**, respectively. First we calculate

$$W_{mnl} = \frac{1}{\lambda_m + \epsilon_n + \omega_l} \sum_i \sum_j \sum_k P^{-1}_{mi} Q^{-1}_{nj} R^{-1}_{lk} F_{ijk}, \tag{28}$$

and the solution **U** is given by

$$U_{ijk} = \sum_m \sum_n \sum_l P_{im} Q_{jn} R_{kl} W_{mnl}. \tag{29}$$

2.4. *Direct Solver for 3D Poisson Equation in Cylindrical Coordinates*

As was done for the 2D problem in polar coordinates, the 3D Poisson equation in cylindrical coordinates Eq. (2) can be rewritten as

$$r^2 \frac{\partial^2 \phi}{\partial r^2} + r \frac{\partial \phi}{\partial r} + \frac{\partial^2 \phi}{\partial \theta^2} + r^2 \frac{\partial^2 \phi}{\partial z^2} = f(r, \theta, z), \tag{30}$$

where $f(r, \theta, z) = r^2 \rho(r, \theta, z)$.

In this paper, we consider a problem in a cylinder with domain $0 \le r \le R$, $0 \le \theta \le 2\pi$, and $a \le z \le b$ and boundary conditions

$$\phi(R, \theta, z) = g(\theta, z); \quad \phi(r, \theta, a) = p(r, \theta); \quad \phi(r, \theta, b) = q(r, \theta), \tag{31}$$

where $g(\theta, z)$, $p(r, \theta)$, and $q(r, \theta)$ are arbitary functions to be specified. We choose Radau collocation in r, Fourier collocation in θ (since the problem is periodic in θ), and Lobatto collocation in z. In addition, the discretized boundary conditions are

$$\phi_{N_r jk} = g(\theta_j, z_k); \quad \phi_{ij0} = p(r_i, \theta_j); \quad \phi_{ijN_z} = q(r_i, \theta_j). \tag{32}$$

Hence, the discretization matrix of Eq. (30) can be written in the form

$$\sum_{l=0}^{N_r} A_{il}\phi_{ljk} + \sum_{l=0}^{N_\theta} B_{jl}\phi_{ilk} + \sum_{l=0}^{N_z} r_i^2 C_{kl}\phi_{ijl} = f_{ijk}, \tag{33}$$

where **A** and **B** are defined as in Eqs. (11) and (12) and **C** is the second derivative matrix in the z direction. To take account of the boundary conditions, we transfer the known quantities on the left-hand side of Eq. (33) to the right-hand side and redefine the right-hand side; that is,

$$F_{ijk} = f_{ijk} - A_{iN_r}\phi_{N_r jk} - r_i^2\left(C_{k0}\phi_{ij0} + C_{kN_z}\phi_{ijN_z}\right). \tag{34}$$

Equation (33) now becomes

$$\sum_{l=0}^{N_r-1} A_{il}\phi_{ljk} + \sum_{l=0}^{N_\theta} B_{jl}\phi_{ilk} + \sum_{l=1}^{N_z-1} r_i^2 C_{kl}\phi_{ijl} = F_{ijk}. \tag{35}$$

Unfortunately, the two-step direct method for the 3D system described in Section 2.3 does not work for this case since the third term in the left-hand side of Eq. (35) involves a three dimensional matrix $r_i^2 C_{kl}$. In this paper, we combine r and z dimensions together as one dimension and θ as the other dimension and use the 2D two-step method to solve the equation directly. As for the 2D case, Eqs. (25) and (26), the operation count for solving Eq. (35) is approximately $4(N_r + N_z)N_\theta(N_r + N_z + N_\theta)$.

3. NUMERICAL CALCULATIONS AND RESULTS

3.1. *Poisson Equation for a Part of a Disk*

The Poisson equation, Eq. (1), for a part of a disk is defined in the domain $0 \le r \le R$ and $0 \le \theta \le \theta_0$. In this case, we choose Chebyshev Gauss–Lobatto points in both r and θ directions. In order to apply the Chebyshev method, the domain for the part disk is transformed into a rectangular domain with the transformation defined by

$$\begin{aligned} r &= \frac{x+1}{2} \\ \theta &= \frac{(\Theta+1)\theta_0}{2}, \end{aligned} \tag{36}$$

where $-1 \le x \le 1$ and $-1 \le \Theta \le 1$.

The benchmark problem we choose is the one introduced by Bernardi and Karageorghis [19], which has a potential of the form

$$\phi(r,\theta) = r^{\frac{\pi}{\theta_0}}(1-r)\ln r \sin\left(\frac{\pi\theta}{\theta_0}\right), \tag{37}$$

where the radius of the disk is $R = 1$ and θ_0 is chosen as some multiple of π. Equation (37), when substituted in Eq. (1), defines the charge function $\rho(r,\theta)$. The Dirichlet, homogeneous boundary conditions are then given by Eq. (37) with $\phi(0,\theta) = \phi(R,\theta) = 0$ and $\phi(r,0) = \phi(r,\theta_0) = 0$. Bernardi and Karageorghis [19] use a Galerkin method based on

TABLE I

Numerical Error E_2 of the Numerical Solution of the PE on a Part of a Disk with $\phi(r, \theta) = r^{\pi/\theta_0}(1 - r) \ln r \sin(\pi\theta/\theta_0)$ and $\theta_0 = 3\pi/2$

N_θ/N_r	4	8	12	16
		Chebyshev		
4	7.5E-03	2.1E-03	1.1E-03	8.2E-04
8	7.6E-03	2.1E-03	1.0E-03	6.4E-04
12	7.6E-03	2.1E-03	1.0E-03	6.4E-04
		Weighted Chebyshev[a]		
4	3.2E-04	3.9E-04	4.1E-04	4.2E-04
8	1.2E-08	1.2E-08	1.2E-08	1.2E-08
12	1.1E-13	1.2E-13	1.2E-13	1.2E-13
16	4.1E-16	5.1E-16	9.2E-16	2.0E-16

[a] Interpolation weight function $v(r) = r^{(\pi/\theta_0)-1} \ln r$.

the Legendre–Lobatto discretization. We calculate $\phi(r, \theta)$ from the Poisson equation and define the error E_2 of the numerical solution by

$$E_2 = \sqrt{\frac{1}{N_r N_\theta} \sum_{i=0}^{N_r} \sum_{j=0}^{N_\theta} |\phi_{\text{numer}}(r_i, \theta_j) - \phi_{\text{exact}}(r_i, \theta_j)|^2}, \tag{38}$$

where ϕ_{exact} and ϕ_{numer} are the exact and numerical solutions, respectively. The number of grid points in the r and θ variables is $N_r + 1$ and $N_\theta + 1$, respectively.

The convergence in r and θ for the Chebyshev method with $\theta_0 = \frac{3\pi}{2}$ is shown in the top part of Table I versus N_r and N_θ. It is clear from Table I that the convergence is rapid in θ and slower in r. The main source of the slow convergence is the function $r^{2/3} \ln r$ in the solution, Eq. (37), which cannot be accurately expanded in polynomials. To improve the convergence in r, we applied the weighted Chebyshev method with $v(r) = r^{(\pi/\theta_0)-1} \ln(r)$ in Eq. (4). The errors calculated with this weighted method are given in the bottom part of Table I, and a significant improvement in the convergence is evident. For $N_r = 16$, the error calculated by the weighted Chebyshev method improves to $O(10^{-16})$ compared to $O(10^{-3})$ with the regular Chebyshev method. Similar behavior is also observed for other values of θ_0.

Table II compares errors obtained with the regular Chebyshev method and the weighted Chebyshev method in comparison with the results reported by Bernardi and Karageorghis [19] for several values of N, $N = N_r = N_\theta$, and θ_0. The regular Chebyshev method and the one by Bernardi and Karageorghis are comparable and converge relatively slowly. The convergence gets worse with increasing θ_0. The weighted Chebyshev method improves the convergence of the solution significantly as compared to the other two methods. The numerical error of the solution is within machine accuracy with only 16×16 meshes for all values of θ_0.

3.2. *Poisson Equation for an Annulus*

For an annulus, the solution domain is defined by $r_0 \le r \le r_1$ and $0 \le \theta \le 2\pi$. Since the solution is periodic in θ, Fourier discretization is the choice for the θ direction. We choose the

TABLE II

Comparison of the Error E_2 of the Numerical Solution of the PE on a Part of a Disk with $\phi(r, \theta) = r^{\pi/\theta_0}(1 - r) \ln r \sin(\pi\theta/\theta_0)$

Methods	$N = 4$	$N = 8$	$N = 12$	$N = 16$
	$\theta_0 = \pi/4$			
Chebyshev	4.6E-04	1.6E-07	3.0E-09	2.1E-10
Bernardi and Karageorghis [19]	2.2E-04	1.5E-08	4.6E-10	3.6E-11
Weighted Chebyshev[a]	2.4E-05	6.6E-10	6.1E-15	2.8E-16
	$\theta_0 = \pi/2$			
Chebyshev	2.9E-04	8.8E-06	1.2E-06	3.3E-07
Bernardi and Karageorghis [19]	6.0E-05	2.1E-06	2.6E-07	5.4E-08
Weighted Chebyshev[a]	2.4E-05	6.6E-10	6.1E-15	2.8E-16
	$\theta_0 = \pi$			
Chebyshev	2.5E-03	3.2E-04	1.1E-04	5.2E-05
Bernardi and Karageorghis [19]	1.1E-03	1.2E-04	2.8E-05	9.6E-06
Weighted Chebyshev[a]	2.4E-04	7.2E-09	6.7E-14	8.5E-16
	$\theta_0 = 3\pi/2$			
Chebyshev	7.5E-03	2.1E-03	1.0E-03	6.4E-04
Bernardi and Karageorghis [19]	1.7E-02	3.5E-03	1.3E-03	6.7E-04
Weighted Chebyshev[a]	3.2E-04	1.2E-08	1.2E-13	2.0E-16

[a] Interpolation weight function $v(r) = r^{(\pi/\theta_0)-1} \ln r$.

Chebyshev Gauss–Lobatto collocation method for the r direction. To apply the Chebyshev points, we transform the domain (r, θ) into (x, Θ) with the following transformation,

$$r = \frac{x(r_1 - r_0) + (r_1 + r_0)}{2}$$
$$\theta = \Theta, \tag{39}$$

where $-1 \le x \le 1$ and $0 \le \Theta \le 2\pi$.

The test problem we use is the one chosen by Christopher *et al.* [8] with the potential

$$\phi(r, \theta) = [r^4 - (r_0 + r_1)r^3 + r_0 r_1 r^2] \sin\theta + 0.5 \frac{\ln(r/r_0)}{\ln(r_1/r_0)} \tag{40}$$

with $r_0 = 0.37$ and $r_1 = 1$ and boundary conditions $\phi(r_0, \theta) = 0$ and $\phi(r_1, \theta) = 0.5$. They used a conformal mapping method which mapped the annulus domain into a complex domain and solved the problem on the complex domain by FFT.

Tables III and IV show our results with the Chebyshev–Fourier collocation method in comparison with the results by Christopher *et al.* [8]. The error E_{abs}, which is defined as the absolute value of the difference between the exact solution $\phi_{\text{exact}}(r, \theta)$ and the numerical solution $\phi_{\text{numer}}(r, \theta)$, is given at selected points for both 8×8 and 16×16 meshes. As shown in these tables, the numerical solution with the present method converges very rapidly and is more accurate than those reported in [8].

We also calculate the maximum error E_∞ defined by

$$E_\infty = \max\{|\phi_{\text{numer}}(r_i, \theta_j) - \phi_{\text{exact}}(r_i, \theta_j)|, \quad i = 0, \ldots, N_r, \quad j = 0, \ldots, N_\theta\}, \tag{41}$$

where ϕ_{numer} and ϕ_{exact} are the numerical and exact solution, respectively. The maximum

TABLE III
Absolute Maximum Error E_{abs} of the Numerical Solution of the PE for an Annulus with 8 × 8 Meshes

r/θ	0.79	2.36	3.93	5.50
		Present		
0.98	1.17E-07	1.17E-07	1.17E-07	1.17E-07
0.91	2.16E-08	2.16E-08	2.16E-08	2.16E-08
0.81	6.48E-08	6.48E-08	6.48E-08	6.48E-08
0.69	7.23E-08	7.23E-08	7.23E-08	7.23E-08
0.56	6.26E-08	6.26E-08	6.26E-08	6.26E-08
0.46	7.82E-08	7.82E-08	7.82E-08	7.82E-08
0.39	2.91E-07	2.91E-07	2.91E-07	2.91E-07
		Christopher *et al.* [8]		
0.88	1.19E-03	1.68E-03	1.19E-03	1.19E-03
0.78	7.23E-04	7.23E-04	7.23E-04	7.23E-04
0.69	6.77E-04	6.77E-04	6.77E-04	6.77E-04
0.61	4.90E-04	4.90E-04	4.90E-04	4.90E-04
0.54	3.80E-04	3.80E-04	3.80E-04	3.80E-04
0.47	2.41E-04	2.41E-04	2.41E-04	2.41E-04
0.42	1.24E-04	1.24E-04	1.24E-04	1.24E-04

error E_∞ of the solution calculated by the present method versus N_r and N_θ is plotted in Fig. 1. It is easy to see that the solution converges exponentially in r. The convergence in θ is extremely rapid since the solution in θ is a simple combination of a sine function and a constant, which can be approximated exactly by the Fourier method.

TABLE IV
Absolute Error E_{abs} of the Numerical Solution of the PE for an Annulus with 16 × 16 Meshes

r/θ	0.79	2.36	3.93	5.50
		Present		
0.91	6.29E-14	6.33E-14	6.35E-14	6.37E-14
0.75	1.34E-13	1.33E-13	1.33E-13	1.33E-13
0.69	2.11E-13	2.11E-13	2.12E-13	2.12E-13
0.62	5.92E-14	5.99E-14	6.06E-14	6.07E-14
0.56	2.55E-13	2.55E-13	2.56E-13	2.56E-13
0.46	2.13E-13	2.13E-13	2.13E-13	2.13E-13
0.42	6.94E-13	6.95E-13	6.95E-13	6.95E-13
		Christopher *et al.* [8]		
0.88	5.59E-05	5.59E-05	5.59E-05	5.59E-05
0.78	4.93E-05	4.93E-05	4.93E-05	4.93E-05
0.69	4.05E-05	4.05E-05	4.05E-05	4.05E-05
0.61	3.18E-05	3.18E-05	3.18E-05	3.18E-05
0.54	2.34E-05	2.34E-05	2.34E-05	2.34E-05
0.47	1.54E-05	1.54E-05	1.54E-05	1.54E-05
0.42	7.69E-06	7.69E-06	7.69E-06	7.69E-06

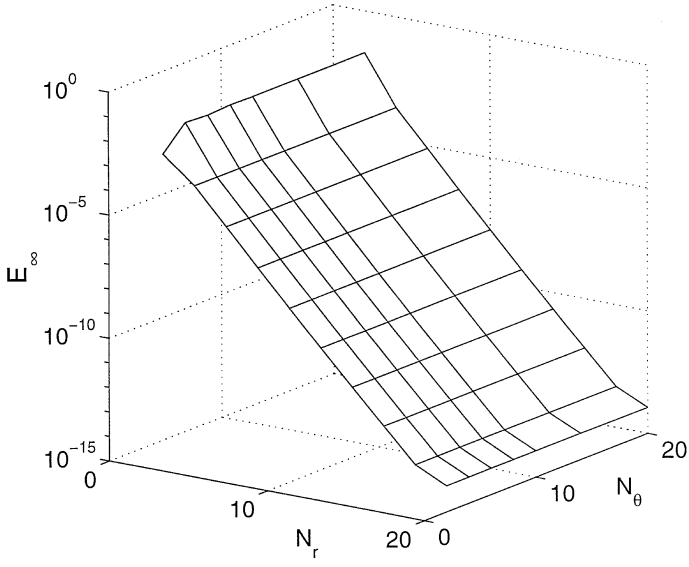

FIG. 1. Maximum error E_∞ of the numerical solution of the Poisson equation for an annulus with the potential given by Eq. (37).

3.3. *Poisson Equation for a Unit Disk*

The domain for the Poisson equation for a unit disk is defined by $0 \le r \le 1, 0 \le \theta \le 2\pi$. In contrast to the other two domains we discussed earlier, it has a coordinate singularity at the center $r = 0$. In this paper, we have developed an algorithm which uses a set of Radau collocation points in the r direction which excludes $r = 0$, so that the coordinate singularity is not involved. Since the problem is periodic in θ, we again choose Fourier discretization in the θ direction.

The first example we choose is given by

$$\phi(r, \theta) = \alpha e^{-ar^2}(C + r \sin b\theta), \tag{42}$$

with $a = 30$, $b = 5$, $C = 1$, $\alpha = 1$. We first solve this problem by using Chebyshev Radau points in the r direction. For this example, the exponential decay in the r direction causes a sharp increase (peak) in the value of the solution near the origin. An expansion of the solution based on nonclassical polynomials may be appropriate and improve the convergence in the r direction. Thus, we also test a set of QDM Radau points based on orthogonal polynomials with respect to a Gaussian weight function $w(r) = e^{-4r^2}$. Results for both methods expressed in terms of E_∞ are given in Table V for pairs of values of N_θ and N_r. The table shows that the QDM method converges faster than the Chebyshev method. In the θ direction, the solution converges slower than that in the r direction when the number of points N_θ is less than 8. However, for $N_\theta > 10$, the convergence of the solution basically depends on the convergence in the r direction. Figure 2 plots the maximum error E_∞ of the numerical solution versus N_r when $N_\theta = 16$. The dashed curve is for the Chebyshev method and the solid curve is for the QDM. It can be seen that the QDM provides a more accurate solution than with the Chebyshev method. Both appear to show exponential convergence.

We also compare the present approach with other methods such as the even parity method by Eisen *et al.* [15], the Chebyshev Gauss–Lobatto method by Huang and Sloan [13],

TABLE V
Maximum Error E_∞ of the Numerical Solution of the PE for a Unit Disk with $\phi(r, \theta) = e^{-30r^2}(1 + \sin 5\theta)$

$N_r \times N_\theta$	Chebyshev	QDM[a]
4×4	2.9053E+00	1.4553E+00
8×4	1.1396E+00	1.0271E+00
4×8	1.6095E+00	5.2168E-01
8×8	2.2849E-01	1.1273E-01
16×8	1.1829E-01	1.1895E-01
8×16	1.7453E-01	2.9162E-03
16×16	3.9074E-04	1.9774E-06
16×32	3.9074E-04	1.9774E-06
20×20	2.8271E-06	2.4878E-08
32×16	1.9307E-10	1.0199E-11
32×32	1.9354E-10	1.0771E-11

[a] Based on nonclassical points w.r.t. $w(r) = e^{-4r^2}$.

and the Chebyshev Galerkin method by Shen [16]. These other three methods all require the implemention of pole conditions. The present calculations use the Chebyshev Radau collocation in r and Fourier collocation in θ. In order to compare the present results with the cited references, we take

$$N_r = N, \quad N_\theta = 2N. \tag{43}$$

Table VI shows the present results in comparison with the results of the previous works for $N = 8$. The choice of potentials is given in the first column. The second column lists the present results by using the regular Chebyshev Gauss–Radau method. As shown in the

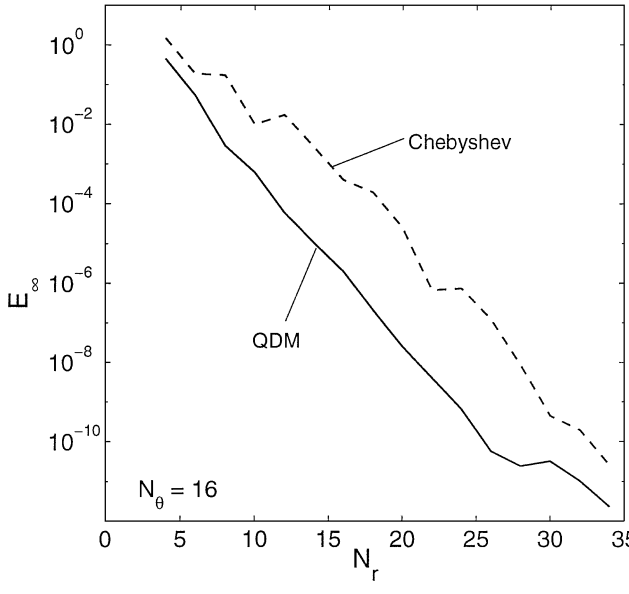

FIG. 2. Maximum numerical error E_∞ of the numerical solution of the Poisson equation for a unit disk with the potential given by Eq. (42). The dashed curve is for the Chebyshev method. The solid curve is for the QDM.

TABLE VI
Comparison of the Error E_∞ of the Numerical Solution of the PE for a Unit Disk with the Potential $\phi(r, \theta)$ and $N = 8$

$\phi(r, \theta)$	Present I	Present II	Huang and Sloan [13]	Eisen *et al.* [15]	Shen [16]
$e^{r\cos(\theta)+r\sin(\theta)}$	2.7153E-08		2.611E-08	3.272E-06	2.6E-08
r^3	6.3838E-16		3.553E-14	2.922E-02	3.8E-16
r^4	8.6288E-15		3.303E-14		
r^5	9.0826E-15		3.000E-14	1.225E-03	
$r^{2.5}$	2.1934E-04	1.8328E-15	2.274E-05	7.677E-02	1.3E-04
$r^{3.5}$	3.1194E-05	1.8703E-15	5.261E-06		
$r^{5.5}$	4.4314E-06	1.6297E-15	5.275E-07		
$\cos(7r\sin\theta + 8r\cos\theta + 0.7)$	3.9645E-01		4.11E-01	1.474E+00	

table, the results are comparable with those in Refs. [13] and [16] and better than those in Ref. [15]. To improve the convergence for the three examples with potentials $r^{2.5}$, $r^{3.5}$, and $r^{5.5}$, we also calculate the solution by using weighted Chebyshev with $v(r) = \sqrt{r}$ so that the weighted solution $\frac{\phi(r,\theta)}{v(r)}$ becomes a low order polynomial with respect to r and can be approximated almost exactly. The results with the weighted Chebyshev method are listed in the third column of Table VI denoted as Present II. As expected, the convergence is significantly improved and the solution is numerically exact. For the last example, the numerical solution for all the methods converges slowly, primarily because the Fourier approximation in θ converges slowly. In Table VII, the convergence for this case is shown for larger N_r and N_θ. The slow convergence in θ relative to r is clear.

As mentioned earlier, the previous works used $N_\theta = 2N_r$. However, for the 2D Poisson equation, the overall convergence of the solution depends on the convergence in both r and θ directions. As seen in the first test problem with the potential Eq. (40) for the Chebyshev method, the convergence in θ is more rapid than that in the r direction. So rather than choosing $N_\theta = 2N_r$, we can use a much smaller N_θ for the same accuracy.

In Table VIII, we list the CPU time of the 2D Poisson solver required for pairs of N_r and N_θ. The CPU time is calculated by the MATLAB function "cputime." Since the present method gives spectral accuracy, the solution usually converges very fast and only a small number of mesh points are required for excellent accuracy. So we only list the mesh points up to 64×64 in Table VIII. As seen from the table, less than a second CPU time is needed for a 64×64 grid.

TABLE VII
Maximum Error E_∞ of the Solution of the PE for a Unit Disk with the Exact Solution $\cos(7r\sin\theta + 8r\cos\theta + 0.7)$

N_r / N_θ	8	16	32	48	64
8	5.8300E+00	3.9645E-01	3.9645E-01	1.2412E-01	1.2411E-01
16	5.7924E+00	4.0401E-01	1.4001E-04	7.7584E-07	7.7654E-07
24	5.8209E+00	4.0814E-01	1.4085E-04	1.6953E-09	4.4409E-13
32	5.8187E+00	4.1103E-01	1.3902E-04	1.7012E-09	3.9457E-13

TABLE VIII
CPU Time of the 2D Poisson Solver for a Unit Disk

N_r	N_θ	CPU (s)
4	4	0.01
4	8	0.01
8	8	0.02
8	16	0.02
16	16	0.05
16	32	0.09
32	32	0.18
32	64	0.37
64	64	0.79

3.4. *Poisson Equation for a Cylinder*

To further test our solver, we solve a three-dimensional Poisson equation in cylindrical geometry with the potential given by

$$\phi(r, \theta, z) = r^2 \sin(5\theta) \sin(3.5z), \quad 0 \le r \le 1, \quad 0 \le \theta \le 2\pi, \quad -1 \le z \le 1. \tag{44}$$

Equation (2) is discretized by using Chebyshev Gauss–Radau collocation in r, Fourier collocation in θ, and Chebyshev Gauss–Lobatto collocation in z. Dirichlet boundary conditions are determined with Eq. (44). Table IX shows the maximum error E_∞ of the solution and CPU time. The CPU time includes the time for the preprocessing stage for the matrix diagonalization in the two-step algorithm. The numerical solution converges rapidly and the maximum error is $O(10^{-11})$ with $N_r = 4$, $N_\theta = N_z = 16$. The rate of convergence is very rapid in r and relatively slower in z and θ.

We also test the solver for an equation with an exact solution used by Tan [17, Eq. (4.2)]. Tan only solved this Poisson equation on domains of a part of cylinder and cylindrical annulus with Chebyshev and Fourier method and did not discuss the solution in a whole cylinder in which case a coordinate singularity occurs. Thus we extend the work in [17]

TABLE IX
Maximum Error E_∞ of the Numerical Solution of the PE for a Cylinder with the Potential $\phi(r, \theta, z) = r^2 \sin(5\theta) \sin(3.5z)$

N_r	N_θ	N_z	E_∞	CPU (s)
4	4	4	5.7281E-01	0.02
4	8	4	1.9332E-01	0.03
4	4	8	9.9017E-01	0.05
4	8	8	3.5171E-01	0.07
4	8	16	3.5177E-01	0.26
8	8	8	3.3899E-01	0.22
8	8	16	3.3904E-01	1.19
4	16	8	1.5211E-04	0.09
8	16	8	1.5922E-04	0.29
4	16	16	1.2206E-11	0.32
8	16	16	1.2059E-11	1.30
16	16	16	1.2960E-11	9.05

TABLE X
Maximum Error E_∞ of the Numerical Solution of the PE for a Cylinder with the Potential Given by Eq. (44)

N_r	N_θ	N_z	E_∞
4	4	4	1.5774E-02
4	8	4	1.5774E-02
4	4	8	1.5424E-02
4	8	8	1.5424E-02
8	4	4	3.0395E-03
8	8	8	4.0739E-06
8	16	8	4.1036E-06
8	8	16	4.0936E-06
16	8	8	5.8126E-07
16	16	16	1.6917E-11

and solve the equation in a cylinder. This potential in the cylindrical domain is given by

$$\phi(r,\theta,z) = [\cos(\pi(r-1)) + \sin(\pi(r-1))][\cos(\theta-1) + \sin(\theta-1)] \times \left[\cos\frac{\pi(z-1)}{2} + \sin\frac{\pi(z-1)}{2}\right]. \quad (45)$$

Table X shows the accuracy attained for the resolution up to $16 \times 16 \times 16$ for which case the maximum error E_∞ reduces to $O(10^{-11})$.

ACKNOWLEDGMENTS

This research is supported by a grant to BDS from the Natural Sciences and Engineering Council of Canada. One of us (BDS) is grateful to Professor Kenichi Nanbu for arranging a visit to the Institute of Fluid Science at Tohoku University, Sendai, Japan, where this work was initiated.

REFERENCES

1. C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation* (McGraw-Hill, New York, 1985).
2. J. R. Roth, *Industrial Plasma Engineering* (Inst. of Phys., London, 1995).
3. J. Birney and S. Tremaine, *Galactic Dynamics* (Princton University Press, Princeton, 1987).
4. C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics* (Springer-Verlag, Berlin, 1987).
5. V. V. Serikov and K. Nanbu, Spatio-temporal variations in a three-dimensional DC glow discharge via particle-in-cell simulation. *IEEE Trans. Plasma Sci.* **27**, 84 (1999).
6. P. A. Miller and M. E. Riley, Dynamics of collisionless plasma sheaths, *J. Appl. Phys.* **82**, 3689 (1997).
7. M. J. Grapperhuis and M. J. Kushner, A semiananlytic radio frequency sheath model integrated into two dimensional hybrid model for plasma processing reactors, *J. Appl. Phys.* **81**, 569 (1997).
8. I. Christopher, G. Knorr, M. Shoucri, and P. Bertrand, Solution of the Poisson equation in an annulus, *J. Comput. Phys.* **131**, 323 (1997).
9. E. Braverman and M. Israeli, A fast 3D Poisson solver of arbitrary order accuracy, *J. Comput. Phys.* **144**, 109 (1998).
10. A. Averbuch, M. Israeli, and I. Vozovoi, A fast Poisson solver of arbitrary order accuracy in rectangular regions, *SIAM J. Sci. Comput.* **19**, 933 (1998).

11. B. Bouaoudia and P. S. Marcus, Fast and accurate spectral treatment of coordinate singularities, *J. Comput. Phys.* **96**, 217 (1991).
12. A. Karageorghis, Conforming spectral methods for Poisson problems in cuboidal domains, *J. Sci. Comput.* **9**, 341 (1994).
13. W. Huang and D. M. Sloan, Pole condition for singular problems: The pseudospectral approximation, *J. Comput. Phys.* **107**, 254 (1993).
14. H. Dang-Vu and C. Delcarte, An accurate solution of the Poisson equation by the Chebyshev collocation method, *J. Comput. Phys.* **104**, 211 (1993).
15. H. Eisen, W. Heinrichs, and K. Witsch, Spectral methods and polar coordinate singularities, *J. Comput. Phys.* **96**, 241 (1991).
16. J. Shen, Efficient spectral-Galerkin methods III: Polar and cylindrical geometries, *SIAM J. Sci. Comput.* **18**, 1583 (1997).
17. C. S. Tan, Accurate solution of three-dimensional Poisson's equation in cylindrical coordinate by expansion in Chebyshev polynomials, *J. Comput. Phys.* **59**, 81 (1985).
18. S. Zhao and M. J. Yedlin, A new iterative Chebyshev spectral method for solving the elliptic equation, *J. Comput. Phys.* **113**, 215 (1994).
19. C. Bernardi and A. Karageorghis, Spectral method in a part of a disk, *Numer. Math.* **73**, 265 (1996).
20. D. Gottlieb and S. Orszag, *Numerical Analysis of Spectral Methods: Theory and Applications* (SIAM, Philadelphia, 1977).
21. S. A. Orszag, Fourier series on spheres, *Mon. Weather Rev.* **102**, 56 (1974).
22. T. Matsushima and P. S. Marcus, A spectral method for polar coordinates, *J. Comput. Phys.* **120**, 365 (1995).
23. R. E. Lynch, J. R. Rice, and D. H. Thomas, Direct solution of partial different equations by tensor product methods, *Numer. Math.* **6**, 185 (1964).
24. D. B. Haidvogel and T. Zang, The accurate solution of Poisson's equation by expansion in Chebyshev polynomials, *J. Comput. Phys.* **30**, 167 (1979).
25. D. Gottlieb, M. Y. Hussaini, and S. A. Orszag, Introduction, theory and applications of spectral methods, in *Spectral Methods for Partial Differential Equation*, edited by Voigt, Gottlieb, and Yousuff, (SIAM, Philadelphia, 1984).
26. T. A. Zang, Y. S. Wang, and M. Y. Hussaini, Spectral multigrid methods for elliptic equations, *J. Comput. Phys.* **48**, 485 (1982).
27. J. A. C. Weideman, Spectral methods based on non-classical orthogonal polynomials, in *Approximations and Computation of Orthogonal Polynomials*, edited by Gautschi, Golub, and Opfer, *Intern. Ser. Numer. Math.* **131**, 239 (1999).
28. Y. Su, *Collocation Spectral Methods in the Solution of Poisson Equation*, PhD Thesis (University of British Columbia, Canada, 1998).
29. R. Peyret, *Introduction to Spectral Methods*, Von Karman Institute Lecture Series 1986-04 (Rhode-Saint Genese, Belgium, 1986).
30. H. Chen and B. D. Shizgal, The quadrature discretization method (QDM) in the solution of the Schrödinger equation, *J. Math. Chem.* **24**, 321 (1998).
31. B. D. Shizgal and H. Chen, The quadrature discretization method in the solution of the Fokker-Planck equation with nonclassical basis functions, *J. Chem. Phys.* **107**, 8051 (1997).
32. B. D. Shizgal and H. Chen, The quadrature discretization method (QDM) in the solution of the Schrödinger equation with nonclassical basis functions, *J. Chem. Phys.* **104**, 4137 (1996).
33. W. Gautschi, Orthogonal polynomials-constructive theory and applications, *J. Comput. Appl. Math.* **12, 13**, 61 (1985).