TAIWANESE JOURNAL OF MATHEMATICS
Vol. 17, No. 4, pp. 1167-1181, August 2013
DOI: 10.11650/tjm.17.2013.2574
This paper is available online at http://journal.taiwanmathsoc.org.tw

A FAST POISSON SOLVER BY CHEBYSHEV PSEUDOSPECTRAL METHOD USING REFLEXIVE DECOMPOSITION

Teng-Yao Kuo, Hsin-Chu Chen and Tzyy-Leng Horng*

Abstract. Poisson equation is frequently encountered in mathematical modeling for scientific and engineering applications. Fast Poisson numerical solvers for 2D and 3D problems are, thus, highly requested. In this paper, we consider solving the Poisson equation $\nabla^2 u = f(x, y)$ in the Cartesian domain $\Omega = [-1, 1] \times [-1, 1]$, subject to all types of boundary conditions, discretized with the Chebyshev pseudospectral method. The main purpose of this paper is to propose a reflexive decomposition scheme for orthogonally decoupling the linear system obtained from the discretization into independent subsystems via the exploration of a special reflexive property inherent in the second-order Chebyshev collocation derivative matrix. The decomposition will introduce coarse-grain parallelism suitable for parallel computations. This approach can be applied to more general linear elliptic problems discretized with the Chebyshev pseudospectral method, so long as the discretized problems possess reflexive property. Numerical examples with error analysis are presented to demonstrate the validity and advantage of the proposed approach.

1. Introduction

There have been considerable studies in the development of robust and efficient solvers for Poisson equation. A fast and accurate Poisson solver has many scientific and engineering applications. These include computer simulations of plasma physics [3], industrial plasma engineering [17], and planetary dynamics [4]. The popular projection method for solving Navier-Stokes equations also involves solving a Poisson equation for the pressure field [7].

Received October 30, 2012, accepted January 3, 2013.
Communicated by Wen-Wei Lin.
2010 *Mathematics Subject Classification*: 35J05, 65F05, 65N35.
Key words and phrases: Poisson equation, Chebyshev pseudospectral method, Chebyshev collocation derivative matrix, Coarse-grain parallelism, Reflexive property.
*Corresponding author.

In this paper, we consider solving the Poisson equation $\nabla^2 u = f(x, y)$ in a Cartesian domain $\Omega = [-1, 1] \times [-1, 1]$ with all types of boundary conditions:

$$
\begin{cases}
\nabla^2 u = f(x,y), \ (x,y) \in (-1,1) \times (-1,1), \\
\alpha_1 u + \beta_1 \dfrac{\partial u}{\partial x} = g_1(y), \ \text{at } x = 1, \\
\alpha_2 u + \beta_2 \dfrac{\partial u}{\partial x} = g_2(y), \ \text{at } x = -1, \\
\alpha_3 u + \beta_3 \dfrac{\partial u}{\partial y} = g_3(x), \ \text{at } y = 1, \\
\alpha_4 u + \beta_4 \dfrac{\partial u}{\partial y} = g_4(x), \ \text{at } y = -1,
\end{cases}
\tag{1.1}
$$

where α_i and β_i, $\forall i$, are constants and can not be zeros at the same time. Also, at least one of the α_i's is not zero to ensure the uniqueness of solution. The current paper proposes a reflexive decomposition scheme to solve (1.1) more efficiently when (1.1) is discretized by the Chebyshev pseudospectral method.

Most Poisson solvers are based on finite difference or finite element methods. While the geometry of computational domain is rectangular, the Chebyshev pseudospectral method has advantage over traditional finite difference and finite element methods in numerical accuracy. Its convergence rate is of exponential order $O\left(c^N\right)$, $0 < c < 1$, for smooth solutions where N is the number of grid points [18].

There have been several literatures about solving Poisson equation over a rectangular domain by Chebyshev spectral/pseudospectral method, to name a few as follows. Haidvogel and Zang [14] applied both ADI and matrix diagonalization techniques to solve Poisson equation over a square domain, and compared their efficiencies. Their method can be only applied to Dirichlet boundary conditions. Dang-Vu and Delcarte [11] presented the solution of Poisson equation using the resolution of the mixed collocation τ equations into two quasi-tridiagonal systems to simplify the differential operators. Shizgal *et al.* [20, 21, 12] used eigenvalue/eigenvector technique to diagonalize the discretized system, which is a very efficient method but only subject to Dirichlet boundary conditions. So far, the eigenvalue/eigenvector matrix diagonalization method for solving Poisson equation over a rectangular domain is most efficient [9, 13]. However, it can be only applied with ease for Dirichlet boundary conditions. For non-Dirichlet boundary conditions like Neumann or Robin ones, this matrix diagonalization method requests messy row operations or costly iterations [22].

To apply Chebyshev pseudospectral method coping with Neumann or Robin boundary conditions, even more general second-order arbitrary-coefficient elliptic partial differential equations, we need to use Kronecker product during discretization for Poisson equation in 2D/3D rectangular domains. The resultant linear system is very costly to solve, especially for 3D problems, by direct Gaussian elimination or iteration methods.

To solve this problem, here we explore the reflexive property [10] of Chebyshev collocation derivative matrix for possible coarse-grain decomposition of the resultant huge matrix. To be more elaborate, we apply reflexive decomposition to orthogonally decompose the original linear system into two, or more, smaller decoupled subsystems so that we can save enormous computation time by solving several smaller linear systems instead. This coarse-grain system reduction can further foster coarse-grain parallelism, which can save more computation time by parallel computation.

The organization of this paper is as follows. In section 2, we begin with an observation that the second-order Chebyshev collocation derivative matrix is centrosymmetric and then show that the matrix associated with the 2D discretized Poisson equation using Chebyshev pseudospectral method with a tensor product is block-centrosymmetric and therefore reflexive. In section 3, we further decompose the resultant matrix into submatrices via orthogonal reflexive decomposition. Operations count and numerical experiments with error analysis showing exponential convergence are presented in section 4, and conclusion is given in section 5.

2. Reflexive Property of Chebyshev Collocation Derivative Matrix

2.1. Chebyshev pseudospectral method

Given Chebyshev-Gauss-Lobatto points, $x_j = \cos(j\pi/N)$, $j = 0, 1, ..., N$, satisfying $T_N'(x_j)\left(1 - x_j^2\right) = 0$, where $T_N(x)$ is the Chebyshev polynomial of degree N, the Lagrange interpolating polynomials based on Chebyshev-Gauss-Lobatto points can be obtained as follows

$$\mathcal{L}_{N,j}(x) = \frac{(-1)^{j+1}\left(1 - x^2\right) T_N'(x)}{\bar{c}_j N^2 (x - x_j)}, \ j = 0, 1, ..., N,$$

where $\bar{c}_j = \begin{cases} 2, & \text{for } j = 0, \ N, \\ 1, & \text{otherwise,} \end{cases}$ and $\mathcal{L}_{N,j}(x_l) = \delta_{jl}$ with δ_{jl} being Kronecker delta. A function $u(x)$ can be approximated by interpolating polynomials above,

$$u(x) \approx \sum_{j=0}^{N} \mathcal{L}_{N,j}(x)\, u(x_j),$$

and its derivative values at Chebyshev-Gauss-Lobatto points can be therefore approximated by

$$u'(x_i) \approx \sum_{j=0}^{N} \mathcal{L}_{N,j}'(x_i)\, u(x_j) = \sum_{j=0}^{N} (D_N)_{ij}\, u(x_j), \ i = 0, 1, ..., N,$$

where D_N is Chebyshev collocation derivative matrix with $(D_N)_{ij} = \mathcal{L}'_{N,j}(x_i)$. The entries of D_N have long been available in literatures [15, 16, 18] and are given as

$$\begin{aligned}
(D_N)_{00} &= \frac{2N^2+1}{6}, \qquad (D_N)_{NN} = -\frac{2N^2+1}{6}, \\
(D_N)_{jj} &= \frac{-x_j}{2\left(1-x_j^2\right)}, \qquad j = 1, 2, ..., N-1, \\
(D_N)_{ij} &= \frac{c_i(-1)^{i+j}}{c_j(x_i - x_j)}, \qquad i \neq j, \text{ and } i, j = 0, 1, ..., N,
\end{aligned}$$

where $c_i = \begin{cases} 2, & \text{for } i = 0,\ N, \\ 1, & \text{otherwise,} \end{cases}$ and note that D_N is a $(N+1) \times (N+1)$ matrix.

Likewise, the k^{th} derivative values of a function $u(x)$ at Chebyshev-Gauss-Lobatto points can be approximated by

$$\left(\frac{\partial^k u}{\partial x^k}\right)_{x=x_i} \approx \sum_{j=0}^{N} \left(D_N^k\right)_{ij} u(x_j), \ i = 0, 1, ..., N,$$

where D_N^k represents the k^{th} power of D_N. Note that D_N is an anti-centrosymmetric matrix, that satisfies

$$(D_N)_{ij} = -(D_N)_{N-i,N-j}, \ i, j = 0, 1, ..., N,$$

or

$$D_N = -J_{N+1} D_N J_{N+1}, \text{ where } J_{N+1} = \begin{bmatrix} & & 1 \\ & \cdots & \\ 1 & & \end{bmatrix}.$$

Accordingly, the second-order derivative matrix D_N^2, denoted by S, will be a centrosymmetric matrix [1, 2, 6, 19] satisfying

$$(S)_{ij} = (S)_{N-i,N-j}, \ i, j = 0, 1, ..., N,$$

or

$$S = J_{N+1} S J_{N+1}.$$

Deriving this will use the property $J_{N+1}^2 = I_{N+1}$, where I_{N+1} is the identity matrix of dimension $N+1$. Extending this idea, D_N^k will be anti-centrosymmetric for any odd k and centrosymmetric for any even k, i.e., $\left(D_N^k\right)_{ij} = (-1)^k \left(D_N^k\right)_{N-i,N-j}$, $i, j = 0, 1, ..., N$.

To solve 2D Poisson problem like (1.1), we apply tensor product to discretize the equation as

$$L_N u = f, \text{ with } L_N \in \Re^{(N+1)^2 \times (N+1)^2}, \ u, f \in \Re^{(N+1)^2}, \tag{2.1}$$

where L_N is the discretized Laplacian operator that can be expressed as

$$L_N = I_{N+1} \otimes S + S \otimes I_{N+1}, \tag{2.2}$$

with $\otimes$ representing the Kronecker product [18]. The Kronecker product of two matrices A and B, denoted by $A \otimes B$, with dimensions $m \times n$ and $p \times q$ respectively, can be expressed as an $m \times n$ block matrix with the i, j block being $a_{ij}B$. For example,

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \otimes \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \left[\begin{array}{cc|cc} 1a_{11} & 2a_{11} & 1a_{12} & 2a_{12} \\ 3a_{11} & 4a_{11} & 3a_{12} & 4a_{12} \\ \hline 1a_{21} & 2a_{21} & 1a_{22} & 2a_{22} \\ 3a_{21} & 4a_{21} & 3a_{22} & 4a_{22} \\ \hline 1a_{31} & 2a_{31} & 1a_{32} & 2a_{32} \\ 3a_{31} & 4a_{31} & 3a_{32} & 4a_{32} \end{array}\right].$$

2.2. Implementation of boundary conditions

To cope with coarse-grain decomposition later, here we allow all kinds of boundary conditions at $y = \pm 1$, but only consider Dirichlet boundary conditions at $x = \pm 1$. It should be noted that at least on one of the coordinate direction the boundary conditions at both end boundaries are of Dirichlet type. Otherwise, the reflexive property is lost and the method may fail. Hence, S should be modified to include boundary conditions mentioned above. Let $\tilde{S}$ and $\bar{S}$ denote the boundary-operator-included S matrices along x and y directions respectively. We then have

$$\tilde{S} = \left[\begin{array}{ccccc} 1 & 0 & \cdots & 0 & 0 \\ \hline & & & & \\ & & (S)_{ij} & & \\ & & & & \\ \hline 0 & 0 & \cdots & 0 & 1 \end{array}\right] \quad \text{and } \bar{S} = \left[\begin{array}{c} \alpha_3 (I_{N+1})_{1,j+1} + \beta_3 (D_N)_{0,j} \\ \hline \\ (S)_{ij} \\ \\ \hline \alpha_4 (I_{N+1})_{N+1,j+1} + \beta_4 (D_N)_{N,j} \end{array}\right]. \tag{2.3}$$

Therefore, the resultant matrix L_N for this case is

$$L_N = I_{N+1} \otimes \bar{S} + \tilde{S} \otimes I_{N+1} = K_1 + K_2 \tag{2.4}$$

with explicit expression of K_1 and K_2 being

$$K_1 = \begin{bmatrix} 0 & & & & \\ & \bar{S} & & & \\ & & \ddots & & \\ & & & \bar{S} & \\ & & & & 0 \end{bmatrix} \text{ and } K_2 = \begin{bmatrix} I & 0 & \cdots & & 0 \\ s_{10} I_{00} & s_{11} I_{00} & & & s_{1,N} I_{00} \\ & & \ddots & & \\ s_{N-1,0} I_{00} & s_{N-1,1} I_{00} & & & s_{N-1,N} I_{00} \\ 0 & & \cdots & 0 & I \end{bmatrix},$$

where

$$s_{ij} = (S)_{ij}, \ I = I_{N+1}, \ \text{and } I_{00} = \begin{bmatrix} 0 & & \\ & I_{N-1} & \\ & & 0 \end{bmatrix}_{(N+1)\times(N+1)}.$$

Actually, we can further express

$$K_1 = I_{00} \otimes \bar{S} \text{ and } K_2 = \tilde{S} \otimes I_{00} + (I - I_{00}) \otimes (I - I_{00})\,. \tag{2.5}$$

Note that L_N above is already further modified to include the boundary condition at $y = \pm 1$. To take care of the compatibility of different kinds of boundary conditions at the corner boundary points, extra modification of the boundary condition there is further needed.

2.3. Reflexive property of L_N

Let R be the reflection matrix with $R = J_{N+1} \otimes I_{N+1}$. Here we want to show that L_N in (2.4) is reflexive with respect to R. It suffices to say so if both K_1 and K_2 are reflexive with respect to R. In following, we drop the subscripts of I and J for notational brevity. First, we can easily observe that K_1 is reflexive with respect to R by

$$RK_1R = \begin{bmatrix} & & & I \\ & & I & \\ & \cdots & & \\ I & & & \\ I & & & \end{bmatrix} \begin{bmatrix} 0 & & & \\ & \bar{S} & & \\ & & \ddots & \\ & & & \bar{S} & \\ & & & & 0 \end{bmatrix} \begin{bmatrix} & & & I \\ & & I & \\ & \cdots & & \\ I & & & \\ I & & & \end{bmatrix}$$
$$= \begin{bmatrix} 0 & & & & \\ & \bar{S} & & & \\ & & \ddots & & \\ & & & \bar{S} & \\ & & & & 0 \end{bmatrix} = K_1.$$

Similarly, K_2 also satisfies the reflexive property by

$$RK_2R = \begin{bmatrix} & & & I \\ & & I & \\ & \cdots & & \\ & I & & \\ I & & & \end{bmatrix} \begin{bmatrix} I & 0 & \cdots & 0 \\ s_{10}I_{00} & s_{11}I_{00} & & s_{1,N}I_{00} \\ & & \ddots & \\ s_{N-1,0}I_{00} & s_{N-1,1}I_{00} & & s_{N-1,N}I_{00} \\ 0 & & \cdots 0 & I \end{bmatrix} \begin{bmatrix} & & & I \\ & & I & \\ & \cdots & & \\ & I & & \\ I & & & \end{bmatrix}$$

$$= \begin{bmatrix} I & 0 & \cdots & & 0 \\ s_{N-1,N}I_{00} & & & s_{N-1,1}I_{00} & s_{N-1,0}I_{00} \\ & & \ddots & & \\ s_{1,N}I_{00} & & & s_{11}I_{00} & s_{10}I_{00} \\ 0 & & \cdots & 0 & I \end{bmatrix}$$

$$= \begin{bmatrix} I & 0 & \cdots & & 0 \\ s_{10}I_{00} & s_{11}I_{00} & & & s_{1,N}I_{00} \\ & & \ddots & & \\ s_{N-1,0}I_{00} & s_{N-1,1}I_{00} & & & s_{N-1,N}I_{00} \\ 0 & & \cdots & 0 & I \end{bmatrix} = K_2.$$

Here we used the centrosymmetric property $s_{ij} = \left(D_N^2\right)_{ij} = \left(D_N^2\right)_{N-i,N-j} = s_{N-i,N-j}$ for the derivation above.

3. Reflexive Decomposition of L_N

As seen in the previous section, the matrix L_N is reflexive with respect to R. This reflexive property enables us to decompose L_N into smaller submatrices using orthogonal transformation. The decomposition is referred to as the reflexive decomposition because of the reflection matrix R.

3.1. General forms of the decomposition

The general forms of the decomposition of L_N are categorized to $N+1$ being even or odd. For notational brevity, let

$$L_N = K. \tag{3.1}$$

Decomposition 1: N+1 even. Let $N+1 = 2k$. We evenly partition R and K into 2×2 sub-blocks as

$$R = \begin{bmatrix} 0 & R_1 \\ R_1 & 0 \end{bmatrix} \text{ and } K = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}, \tag{3.2}$$

where $R_1 = J_k \otimes I_{N+1}$; $K_{ij} \in \Re^{k(N+1)\times k(N+1)}$, $\forall\ i,j = 1,2$. From the reflexive property $K = RKR$, we can derive the following relations easily,

$$K_{11} = R_1K_{22}R_1,\ K_{12} = R_1K_{21}R_1,\ K_{21} = R_1K_{12}R_1,\ K_{22} = R_1K_{11}R_1. \tag{3.3}$$

Let Q be the following orthogonal matrix,

$$Q = \frac{1}{\sqrt{2}} \begin{bmatrix} I & -R_1 \\ R_1 & I \end{bmatrix}. \tag{3.4}$$

Using (3.3) and the facts $R_1^2 = I$ and $R_1^T = R_1$, it can be easily shown that the transformation $Q^T K Q$ is block-diagonal and therefore decouples K into two independent submatrices as follows [5, 8, 10],

$$Q^T K Q = \frac{1}{2}\begin{bmatrix} I & R_1 \\ -R_1 & I \end{bmatrix}\begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}\begin{bmatrix} I & -R_1 \\ R_1 & I \end{bmatrix} = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix}, \tag{3.5}$$

where

$$D_1 = K_{11} + K_{12}R_1 \text{ and } D_2 = K_{22} - K_{21}R_1.$$

Decomposition 2: N+1 odd. Let $N+1 = 2k+1$. Let R be partitioned as

$$R = \begin{bmatrix} & & R_1 \\ & I_{N+1} & \\ R_1 & & \end{bmatrix}, \quad R_1 = J_k \otimes I_{N+1}. \tag{3.6}$$

The matrix K is then partitioned in accordance with R as

$$K = \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix}, \quad K_{11}, K_{13}, K_{31}, K_{33} \in \Re^{k(N+1)\times k(N+1)},$$

$$K_{12}, K_{32} \in \Re^{k(N+1)\times(N+1)}, \quad K_{21}, K_{23} \in \Re^{(N+1)\times k(N+1)}, \quad K_{22} \in \Re^{(N+1)\times(N+1)}.$$

From $K = RKR$, we can derive

$$K_{11} = R_1 K_{33} R_1, \quad K_{12} = R_1 K_{32}, \quad K_{13} = R_1 K_{31} R_1, \quad K_{21} = K_{23} R_1. \tag{3.7}$$

Let Q be the following orthogonal matrix,

$$Q = \frac{1}{\sqrt{2}}\begin{bmatrix} I & 0 & -R_1 \\ 0 & \sqrt{2} I_{N+1} & 0 \\ R_1 & 0 & I \end{bmatrix}. \tag{3.8}$$

By (3.7), the orthogonal transformation $Q^T K Q$ yields [10]

$$Q^T K Q = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} \tag{3.9}$$

with

$$D_1 = \begin{bmatrix} K_{11} + K_{13}R_1 & \sqrt{2}K_{12} \\ \sqrt{2}K_{21} & K_{22} \end{bmatrix}, \quad D_2 = K_{33} - K_{31}R_1.$$

Then, the process of solving linear system (2.1) by reflexive decomposition starts with

$$Ku = f,$$

and then

$$Q^T K Q Q^T u = Q^T f,$$

or expressed as

$$\hat{K}\hat{u} = \hat{f}, \tag{3.10}$$

with $\hat{K} = Q^T K Q$, $\hat{u} = Q^T u$, and $\hat{f} = Q^T f$. By D_1 and D_2 obtained above, (3.10) can be further expressed as

$$\begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \end{bmatrix} = \begin{bmatrix} \hat{f}_1 \\ \hat{f}_2 \end{bmatrix}, \tag{3.11}$$

or

$$\begin{cases} D_1 \hat{u}_1 = \hat{f}_1, \\ D_2 \hat{u}_2 = \hat{f}_2. \end{cases}$$

After solving $\hat{u}_1$ and $\hat{u}_2$, we can recover $\hat{u}$, and finally obtain u by $u = Q\hat{u}$.

3.2. Explicit forms of the decomposition

In this section, we want to derive the explicit forms of D_1 and D_2 mentioned above for the convenience of coding. Again, we categorize cases according to $N + 1$ being even or odd.

Explicit form 1: $N + 1 = 2k$**.** Let I_k and O_k denote the identity matrix and null matrix of dimension k. Let also

$$I_{01} = \begin{bmatrix} 0 & \\ & I_{k-1} \end{bmatrix}, \; I_{10} = \begin{bmatrix} I_{k-1} & \\ & 0 \end{bmatrix}, \; O_{10} = \begin{bmatrix} 1 & \\ & O_{k-1} \end{bmatrix}, \; O_{01} = \begin{bmatrix} O_{k-1} & \\ & 1 \end{bmatrix}, \tag{3.12}$$

and we can easily see

$$I_{00} = \begin{bmatrix} I_{01} & 0 \\ 0 & I_{10} \end{bmatrix}. \tag{3.13}$$

Following the strategy in previous section, we can evenly partition $\tilde{S}$ into 2×2 sub-blocks as

$$\tilde{S} = \begin{bmatrix} \tilde{S}_{11} & \tilde{S}_{12} \\ \tilde{S}_{21} & \tilde{S}_{22} \end{bmatrix}, \; \tilde{S}_{ij} \in \Re^{k \times k}. \tag{3.14}$$

By (2.5), (3.13), and (3.14), K_1 and K_2 can then be expressed in their partitioned forms as

$$K_1 = I_{00} \otimes \bar{S} = \begin{bmatrix} I_{01} \otimes \bar{S} & 0 \\ 0 & I_{10} \otimes \bar{S} \end{bmatrix}, \tag{3.15}$$

and

$$
\begin{aligned}
K_2 &= \tilde{S} \otimes I_{00} + (I - I_{00}) \otimes (I - I_{00}) \\
&= \begin{bmatrix} \tilde{S}_{11} \otimes I_{00} & \tilde{S}_{12} \otimes I_{00} \\ \tilde{S}_{21} \otimes I_{00} & \tilde{S}_{22} \otimes I_{00} \end{bmatrix} + \begin{bmatrix} O_{10} \otimes (I - I_{00}) & O_k \\ O_k & O_{01} \otimes (I - I_{00}) \end{bmatrix} \\
&= \begin{bmatrix} \tilde{S}_{11} \otimes I_{00} + O_{10} \otimes (I - I_{00}) & \tilde{S}_{12} \otimes I_{00} \\ \tilde{S}_{21} \otimes I_{00} & \tilde{S}_{22} \otimes I_{00} + O_{01} \otimes (I - I_{00}) \end{bmatrix}.
\end{aligned} \tag{3.16}
$$

Now, from (3.15), (3.16), and (3.2), we have

$$
\begin{aligned}
K_{11} &= I_{01} \otimes \bar{S} + \tilde{S}_{11} \otimes I_{00} + O_{10} \otimes (I - I_{00}), \\
K_{12} &= \tilde{S}_{12} \otimes I_{00}, \\
K_{21} &= \tilde{S}_{21} \otimes I_{00}, \\
K_{22} &= I_{10} \otimes \bar{S} + \tilde{S}_{22} \otimes I_{00} + O_{01} \otimes (I - I_{00}).
\end{aligned}
$$

Accordingly, we then obtain the explicit forms of D_1 and D_2,

$$
\begin{aligned}
D_1 &= \Big(I_{01} \otimes \bar{S} + \tilde{S}_{11} \otimes I_{00} + O_{10} \otimes (I - I_{00}) \Big) + \Big(\tilde{S}_{12} \otimes I_{00} \Big) R_1, \\
D_2 &= \Big(I_{10} \otimes \bar{S} + \tilde{S}_{22} \otimes I_{00} + O_{01} \otimes (I - I_{00}) \Big) - \Big(\tilde{S}_{21} \otimes I_{00} \Big) R_1.
\end{aligned} \tag{3.17}
$$

Explicit form 2: $N + 1 = 2k + 1$**.** In this case, we partition K_1 and K_2 by their tensor-product forms as

$$
K_1 = I_{00} \otimes \bar{S} = \begin{bmatrix} I_{01} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & I_{10} \end{bmatrix} \otimes \bar{S},
$$

and

$$
\begin{aligned}
K_2 &= \tilde{S} \otimes I_{00} + (I - I_{00}) \otimes (I - I_{00}) \\
&= \begin{bmatrix} \tilde{S}_{11} & \tilde{S}_{12} & \tilde{S}_{13} \\ \tilde{S}_{21} & \tilde{S}_{22} & \tilde{S}_{23} \\ \tilde{S}_{31} & \tilde{S}_{32} & \tilde{S}_{33} \end{bmatrix} \otimes I_{00} + \begin{bmatrix} O_{10} & & \\ & 0 & \\ & & O_{01} \end{bmatrix} \otimes (I - I_{00}),
\end{aligned}
$$

where $\tilde{S}_{11}, \tilde{S}_{13}, \tilde{S}_{31}, \tilde{S}_{33} \in \Re^{k \times k}$, $\tilde{S}_{12}, \tilde{S}_{32} \in \Re^{k \times 1}$, $\tilde{S}_{21}, \tilde{S}_{23} \in \Re^{1 \times k}$ and $\tilde{S}_{22} \in \Re$. Accordingly, we have

$$
\begin{aligned}
K &= K_1 + K_2 \\
&= \begin{bmatrix} I_{01} \otimes \bar{S} + \tilde{S}_{11} \otimes I_{00} + O_{10} \otimes (I - I_{00}) & \tilde{S}_{12} \otimes I_{00} & \tilde{S}_{13} \otimes I_{00} \\ \tilde{S}_{21} \otimes I_{00} & \bar{S} + \tilde{S}_{22} \otimes I_{00} & \tilde{S}_{23} \otimes I_{00} \\ \tilde{S}_{31} \otimes I_{00} & \tilde{S}_{32} \otimes I_{00} & I_{10} \otimes \bar{S} + \tilde{S}_{33} \otimes I_{00} + O_{01} \otimes (I - I_{00}) \end{bmatrix}.
\end{aligned}
$$

Therefore, the decomposed submatrices D_1 and D_2 are obtained explicitly as follows,

$$\begin{aligned} D_1 &= \begin{bmatrix} I_{01}\otimes\bar{S}+\tilde{S}_{11}\otimes I_{00}+O_{10}\otimes(I-I_{00})+(\tilde{S}_{13}\otimes I_{00})R_1 & \sqrt{2}(\tilde{S}_{12}\otimes I_{00}) \\ \sqrt{2}(\tilde{S}_{21}\otimes I_{00}) & \bar{S}+\tilde{S}_{22}\otimes I_{00} \end{bmatrix}, \\ D_2 &= I_{10}\otimes\bar{S}+\tilde{S}_{33}\otimes I_{00}+O_{01}\otimes(I-I_{00})-(\tilde{S}_{31}\otimes I_{00})R_1. \end{aligned} \tag{3.18}$$

4. Error Analysis and Operations Count

The spectral convergence is still preserved during decomposition, and this is demonstrated by the following numerical experiments.

Experiment 1. Considering the following 2D Poisson equation

$$\nabla^2 u = f(x,y),\ (x,y)\in[-1,1]\times[-1,1],$$

$$f(x,y) = \left[2-\pi^2\left(y^2+2y+1\right)\right]\sin\pi x,$$

subject to the following boundary conditions

$$u\left(\pm1,y\right)=0,\ u_y\left(x,-1\right)=0,\ \text{and } u\left(x,1\right)+u_y\left(x,1\right)=8\sin\pi x,$$

with the exact solution being $u\left(x,y\right)=\left(y^2+2y+1\right)\sin\pi x$.

Though the aforementioned decomposition is illustrated by 2D cases, it actually can be extended to 3D as well. A 3D Poisson problem below is also computed here for demonstration.

Experiment 2. Considering the following 3D Poisson equation

$$\nabla^2 u = f(x,y,z),\ (x,y,z)\in[-1,1]\times[-1,1]\times[-1,1],$$

$$f(x,y,z)=2\left[\left(y^2-1\right)(z-1)+\left(x^2-1\right)(z-1)+\left(x^2-1\right)\left(y^2-1\right)(z+3)/8\right]e^{z/2},$$

subject to the following boundary conditions

$$u\left(\pm1,y,z\right)=u\left(x,\pm1,z\right)=0,\ u_z\left(x,y,-1\right)=0,$$

$$2u\left(x,y,1\right)+10u_z\left(x,y,1\right)=10\sqrt{e}\left(x^2-1\right)\left(y^2-1\right),$$

with the exact solution being

$$u\left(x,y,z\right)=\left(x^2-1\right)\left(y^2-1\right)(z-1)\,e^{z/2}.$$

The errors of both experiments above are shown in Table 1. Basically, we can observe the exponential convergence, the feature of pseudospectral methods, in Table 1 as N increasing until contaminated by round-off errors (in 3D case).

Table 1. L_∞ error vs. N for experiment 1 and 2

N	L_∞ error	
	2D	3D
3	4.31×10^{-1}	8.4×10^{-3}
5	5.8×10^{-2}	4.06×10^{-5}
7	1.3×10^{-3}	8.17×10^{-8}
9	2.49×10^{-5}	8.98×10^{-11}
11	2.98×10^{-7}	2.2×10^{-13}
13	3.19×10^{-9}	1.99×10^{-13}
17	5.27×10^{-13}	1.24×10^{-12}

Taking 2D Poisson problem as example, the structure of $L_N = K$ will look like Figure 1(a). The operations count of solving (2.1) without reflexive decomposition by Gaussian elimination would be

$$C_M = \frac{1}{3}M^3 + M^2 - \frac{1}{3}M, \tag{4.1}$$

with $M = (N+1)^2$. With reflexive decomposition, K is transformed to $\hat{K}$, and the structure of $\hat{K}$ is shown in Figure 1(b).

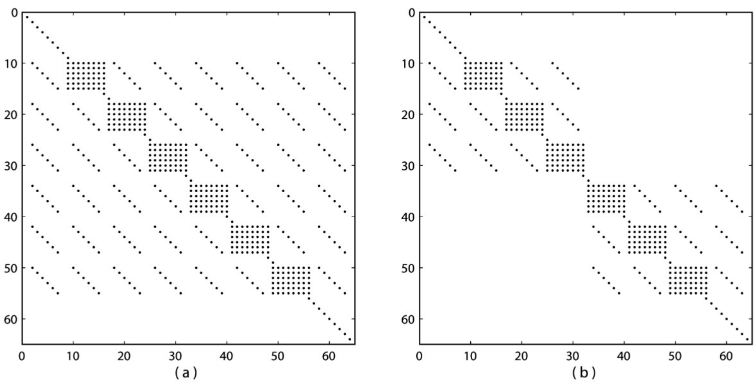

Fig. 1. The structure of K and $\hat{K}$ shown in (a) and (b) respectively.

The operations count of solving (2.1) with reflexive decomposition by Gaussian elimination, including overheads, would then be

$$C_M = \begin{cases} \frac{1}{12}M^3 + \frac{1}{2}M^2 + \frac{5}{3}M, & M \text{ is even}, \\ \frac{1}{12}M^3 + \frac{3}{4}M^2 + \frac{13}{6}M, & M \text{ is odd}. \end{cases} \tag{4.2}$$

In addition to the advantage of computing time reduction by the obvious comparison between (4.1) and (4.2), the decomposition also yields coarse-grain parallelism, which can be conducted by parallel computing. If higher computing efficiency is requested, especially when N is large, it should be noted that D_1 and D_2 turn out to be reflexive too, and can be decomposed by the same procedure stated above.

5. Conclusion

In this paper, we first explore the inherent reflexive property of second-order Chebyshev collocation derivative matrix subject to all kinds of boundary conditions. We then apply reflexive decomposition to decompose the matrix resultant from the discretization of Poisson equation by Chebyshev pseudospectral method into two smaller submatrices. Computing these two smaller linear systems definitely saves time from computing the original large system, not to mention its availability for coarse-grain parallel computing. This time saving effect is particularly notable for 3D cases. These two decomposed submatrices can be further decomposed to four even-smaller submatrices by repeating reflexive decomposition. Numerical examples are also presented to demonstrate the spectral accuracy. In addition, the current decomposition can be extended to solve $au_{xx} + bu_{yy} + cu_y + du = f\,(x, y)$ with a, b, c and d being the functions of x and y and symmetric about y axis subject to the constraint of reflexive property. Certainly, this includes the famous Helmholtz equation.

Acknowledgments

The second author Dr. Chen wishes to thank his home institution, Clark Atlanta University, U.S.A. for supporting this research area through a sabbatical leave and Feng Chia University and Tatung University, both in Taiwan for giving him enjoyable environments during his visit to the universities in 2009. The corresponding author Dr. Horng wants to thank the partial support of National Science Council of Taiwan under grant no. NSC-101-2115-M-035-001 and NSC-100-2632-E-035-001-MY3.

References

1. A. L. Andrew, Solution of equations involving centrosymmetric matrices, *Technometrics*, **15(2)** (1973), 405-407.
2. A. L. Andrew, Eigenvectors of certain matrices, *Linear Algebra Appl.*, **7** (1973), 151-162.
3. C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation*, McGraw-Hill, New York, 1985.
4. J. Birney and S. Tremaine, *Galactic Dynamics*, Princeton University Press, Princeton, 1987.

5. H. C. Chen, Increasing parallelism in the finite strip formulation: static analysis, *Neural Parallel Sci. Comput.*, **2** (1994), 273-298.

6. A. Cantoni and P. Butler, Eigenvalues and eigenvectors of symmetric centrosymmetric matrices, *Linear Algebra Appl.*, **13** (1976), 275-288.

7. C. Canuto, M. Y. Hussaini, A. Quarteroni and T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer-Verlag, Berlin, 1987.

8. H. C. Chen, T. L. Horng and Y. H. Yang, Reflexive decompositions for solving Poisson equation by Chebyshev pseudospectral method, *Proceedings of Neural, Parallel, and Scientific Computations*, **4** (2010), 98-103.

9. H. Chen, Y. Su and B. D. Shizgal, A direct spectral collocation Poisson solver in polar and cylindrical coordinates, *J. Comput. Phys.*, **160** (2000), 453-469.

10. H. C. Chen and A. Sameh, A matrix decomposition method for orthotropic elasticity problems, *SIAM J. Matrix Anal. Appl.*, **10(1)** (1989), 39-64.

11. H. Dang-Vu and C. Delcarte, An accurate solution of the Poisson equation by the Chebyshev collocation method, *J. Comput. Phys.*, **104** (1993), 211-220.

12. U. Ehrenstein and R. Peyret, A Chebyshev collocation method for Navier-Stokes equations with applications to double-diffusive convection, *Internat. J. Numer. Methods Fluids*, **9** (1989), 427-452.

13. T. L. Horng and C. H. Teng, An error minimized pseudospectral penalty direct Poisson solver, *J. Comput. Phys.*, **231(6)** (2012), 2498–2509.

14. D. B. Haidvogel and T. Zang, The accurate solution of Poisson's equation by expansion in Chebyshev polynomials, *J. Comput. Phys.*, **30** (1979), 167-180.

15. J. d. J. Martinez and P. d. T. T. Esperanca, A Chebyshev collocation spectral method for numerical simulation of incompressible flow problems, *J. Braz. Soc. Mech. Sci. & Eng.*, **XXIX(3)** (2007), 317-328.

16. R. Peyret, Spectral methods for incompressible viscous flow, *Applied Mathematical Sciences*, **148**, Springer-Verlag, 2002.

17. J. R. Roth, *Industrial Plasma Engineering*, Inst. of Phys., London, 1995.

18. L. N. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia, 2000.

19. J. R. Weaver, Centrosymmetric (cross-symmetric) matrices, their basic properties, eigenvalues, and eigenvectors, *The American Mathematical Monthly*, **92(10)** (1985), 711-717.

20. H. H. Yang, B. R. Seymour and B. D. Shizgal, A Chebyshev pseudospectral multi-domain method for steady flow past a cylinder up to Re = 150, *Comput. & Fluids*, **23** (1994), 829-851.

21. H. H. Yang and B. Shizgal, Chebyshev pseudospectral multi-domain technique for viscous flow calculation, *Comput. Methods Appl. Mech. Engrg.*, **118** (1994), 47-61.

22. S. Zhao and M. J. Yedlin, A new iterative Chebyshev spectral method for solving the elliptic equation $\bigtriangledown \cdot (\sigma \bigtriangledown u) = f$, *J. Comput. Phys.*, **113** (1994), 215-223.

Teng-Yao Kuo
Ph.D. Program in Mechanical and Aeronautical Engineering
Feng Chia University
Taichung 40724, Taiwan
E-mail: troyguo@yahoo.com.tw

Hsin-Chu Chen
Department of Computer and Information Science
Clark Atlanta University
Atlanta, GA 30314
U.S.A.
E-mail: hchen@cau.edu

Tzyy-Leng Horng
Department of Applied Mathematics
Feng Chia University
Taichung 40724, Taiwan
E-mail: tlhorng@math.fcu.edu.tw